Contributions to Statistics

Olga Valenzuela · Fernando Rojas ·
Luis Javier Herrera · Héctor Pomares ·
Ignacio Rojas *Editors*

Theory and Applications of Time Series Analysis

Selected Contributions from ITISE 2019

Springer

Contributions to Statistics

The series **Contributions to Statistics** contains publications in theoretical and applied statistics, including for example applications in medical statistics, biometrics, econometrics and computational statistics. These publications are primarily monographs and multiple author works containing new research results, but conference and congress reports are also considered.
Apart from the contribution to scientific progress presented, it is a notable characteristic of the series that publishing time is very short, permitting authors and editors to present their results without delay.

More information about this series at http://www.springer.com/series/2912

Olga Valenzuela · Fernando Rojas ·
Luis Javier Herrera · Héctor Pomares ·
Ignacio Rojas
Editors

Theory and Applications of Time Series Analysis

Selected Contributions from ITISE 2019

Springer

Editors

Olga Valenzuela
Faculty of Sciences
University of Granada
Granada, Spain

Fernando Rojas
ETSIIT, CITIC-UGR
University of Granada
Granada, Spain

Luis Javier Herrera
ETSIIT, CITIC-UGR
University of Granada
Granada, Spain

Héctor Pomares
ETSIIT, CITIC-UGR
University of Granada
Granada, Spain

Ignacio Rojas
ETSIIT, CITIC-UGR
University of Granada
Granada, Spain

ISSN 1431-1968
Contributions to Statistics
ISBN 978-3-030-56218-2 ISBN 978-3-030-56219-9 (eBook)
https://doi.org/10.1007/978-3-030-56219-9

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

This book gathers the extended versions of a selection of the best contributions presented in the sixth edition of the International Conference on Time Series and Forecasting, ITISE 2019, held in Granada (Spain) in September 2019. Since its first edition, the main objective of this conference is none other than to provide a friendly discussion forum for scientists, engineers, educators and students to debate about the latest ideas and realizations in the foundations, theory, models and applications in the field of time series analysis and forecasting. This is a strong multidisciplinary field since it is essentially about collecting information, analysing it to try to understand the reason for these collected values and, once this is achieved, to be able to anticipate its future evolution. And even in cases where this prediction is not very accurate, the fact of having at least one can be a substantial advantage, for example, in stock markets. As the brilliant Danish philosopher and poet Soren Kiergegaard put in a much more elegant way "Life can only be understood backwards; but it must be lived forwards". In our case, the way to understand life is through the collection of ordered sets of samples that we call time series.

The main topics in the last edition of the Conference were:

1. **Time series analysis and forecasting**

   - Nonparametric and functional methods
   - Vector processes
   - Probabilistic approaches to modelling macroeconomic uncertainties
   - Uncertainties in forecasting processes
   - Nonstationarity
   - Forecasting with Many Models. Model integration
   - Forecasting theory and adjustment
   - Ensemble forecasting
   - Forecasting performance evaluation
   - Interval forecasting
   - Data preprocessing methods: Data decomposition, seasonal adjustment, singular
   - Spectrum analysis, detrending methods, etc.

2. **Econometrics and forecasting**
   - Econometric models
   - Economic and econometric forecasting
   - Real macroeconomic monitoring and forecasting
   - Advanced econometric methods.
3. **Advanced methods and online learning in time series**
   - Adaptivity for stochastic models
   - Online machine learning for forecasting
   - Aggregation of predictors
   - Hierarchical forecasting
   - Forecasting with computational intelligence
   - Time series analysis with computational intelligence
   - Integration of system dynamics and forecasting models.
4. **High dimension and complex/big data**
   - Local versus global forecasts
   - Dimension reduction techniques
   - Multiscaling
   - Forecasting Complex/Big data.
5. **Forecasting in real problems**
   - Health forecasting
   - Atmospheric science forecasting
   - Telecommunication forecasting
   - Hydrological forecasting
   - Traffic forecasting
   - Tourism forecasting
   - Marketing forecasting
   - Modelling and forecasting in power markets
   - Energy forecasting
   - Climate forecasting
   - Financial forecasting and risk analysis
   - Forecasting electricity load and prices
   - Forecasting and planning systems
   - Applications in real problem (finance, transportation, networks, meteorology, e-health, environment, etc.).

High-quality candidate papers from the Conference ITISE 2019 (29 contributions) were invited to submit an extended version of their conference paper to be considered for this special publication in the book series of Springer: Contributions to Statistics. For the selection procedure, the information/evaluation of the chair of every session, in conjunction with the review comments and the summary of reviews, were taken into account.

So, now we are pleased to have reached the end of the whole process and present the readers with these final contributions that we hope will provide a clear overview of the thematic areas covered by the ITISE 2019 conference ranging from theoretical aspects to real-world applications.

For the sake of readability, the contributions presented in this book have been classified into different chapters according to their content. Some chapters of the book contain pure theoretical contributions. On the other hand, there are chapters with more practical contributions with the intention of providing the readers with a more real-world view of the field. As is common in these editions, a specific chapter of the book has been dedicated to a specific application field. In this case, a whole chapter is devoted to energy-related applications of time series analysis and forecasting.

In the following, we will make a short summary of what the reader may find in every chapter of the book:

- **Advanced statistical and Mathematical Methods for Time Series Analysis**. The main objective of this chapter is to present advanced statistical methodologies and theories that could be used to extract information from time series data. In the first contribution selected for this chapter, the authors propose the utilization of random forest techniques, commonly used in machine learning methods, for learning the subset of significant relationships for vector autoregressive models. The second contribution covers the description of the covariance function for general Laplacian autoregressive models in higher dimensions. In the third contribution, the authors study how a one-to-one correspondence between discrete-time autoregressive moving-average models and their continuous-time counterparts can be elicited. Finally, this chapter includes one last contribution that proposes a new statistical test for a random walk detection based on the arcsine law.
- **Econometric models and Forecasting**. One of the most prominent applications of time series modelling and forecasting lies within the field of Econometrics. This chapter aims at presenting some recent developments of time series research applied to financial and future data with the original idea of focusing on studies that develop and apply recent non-linear econometric models to reproduce financial market dynamics and to capture financial data properties with the hope of eventually predict the next economic bubble. Six contributions have been selected to that end. The first one deals with the problem of the automatic identification of unobserved component models, demonstrating the usefulness of information criteria to that end. The second contribution tries to investigate the temporal development of the spatial network of price relationships between the pig meat markets of 24 European countries and draws insights about the horizontal agricultural market integration process in the EU. The third contribution makes a comparative study of different models for forecasting Nigerian Stock Exchange Market Capitalization for both short and long horizons. The next paper in this chapter focuses on scoring methods to predict corporate default by specifically examining whether belonging to an industry branch influences the

results of the models. The fifth contribution builds and implements multifactor stochastic volatility models, where the main objective is the one-step ahead volatility prediction and to describe its relevance for the equity markets and, finally, this chapter concludes with a final paper aiming at investigating the validity of the Balassa–Samuelson effect in some selected African countries.

- **Energy time series forecasting**. This chapter makes particular emphasis on the application of time series analysis, modelling and forecasting applied to energy-related data. By energy, we refer to any kind of energy, such as electrical, solar, microwave, wind and so on. The first contribution proposes a new criterion for detecting the end of charging process of rechargeable batteries. Next, we present a paper which analyses the impact of adopting Daylight Saving Time on power consumption in the Spanish Electric System. The third contribution deals with 1-hour to 24-hour ahead wind speed forecasting using kernel ridge regression. The next paper presents how convolutional neural networks, commonly used in the first layers of deep neural networks when the input data are images, can be used in mid-term electricity load forecasting problems. And finally, the last paper deals with long- and short-term forecasting of power consumption using modified Long Short-Term Memory Neural Networks.
- **Forecasting Complex/Big data problems**. In the past few years, mechanisms to automate data extraction have proliferated so much that the amount of data that have to be analysed for a specific problem is starting to be one of the biggest difficulties. In addition to this, the type of data extracted can be of very different nature, thus making the integration of this information from these heterogeneous data sources another of the main difficulties to tackle. In this chapter, we give some insights into how to deal with such difficulties in the case of time series data. In the first contribution, an info-metrics approach based on normalized entropy is presented to measure the relationships between the dependent variable and each of the potential explanatory variables. In the next contribution, the authors show how the ability of text mining to convert large collections of text from unstructured to structured form can be used for in-depth quantitative analysis of online news data. In the third contribution, the author shows how the application of Big Data technologies can help us to extract knowledge about hotel tourism demand in Spain from data collected from the Google Data Mining tools. In the fourth and last contribution, several neural network models are trained using specific time series data created from traffic camera images with the final aim at optimizing traffic signal timing sequences in order to reduce congestion based on anticipated demand.
- **Time Series Analysis with Computational Intelligence**. Although time series analysis can be considered a discipline originated within the statistical area, in the past decades many computational intelligence methods or machine learning approaches have been proposed to solve time series-related problems. In fact, research in new computational intelligence approaches, their efficiency and their comparison to statistical methods and other fact-checked computational intelligence methods, is a significant topic in academic and professional projects. It is not uncommon for the existence of time series forecasting competitions which

try to elucidate which of the two main research streams is better. Within this topic, five contributions have been selected for this book. The first contribution makes a comparative study of how different machine learning techniques ranging from support vector machines to random forests perform in the forecasting of intense convective rainfall events. The next contribution deals with the prediction of transformer temperature in smart grids. To that end, the authors make use of long-term memory networks that use data from the previous 100 minutes to predict the transformer temperature for the next 100 minutes. The third contribution proposes the use of the KnoX method, developed to extract information from a neural network model, in the Gardon de Mialet flash floods modelling. In this way, it can be understood how the variables are handled by the neural network to approximate the modelled phenomenon. The next paper makes an experimental comparison of the performance of two well-known paradigms, namely artificial neural networks and fuzzy time series models. The authors select different types of models of every paradigm and compare them using short- and medium-term predictions of several time series. Finally, the last contribution is related to how to combine forecasts obtained from several different models using the Extreme Learning Machine method.

- **Time Series Analysis and Prediction in Other Real Problems**. The last chapter of the book is dedicated to other real applications of time series analysis, modelling and forecasting different from those especially mentioned before. The idea is to state explicitly that applications of time series analysis reach practically any scientific discipline imaginable. Five very different contributions were selected for this last chapter. The first presents a case study targeting the recent Brazilian load changes to illustrate how it is possible to combine data from three different distribution companies, creating a learning network and yielding reliable results where all other models failed. The second contribution presents a non-linear autoregressive neural network with exogenous inputs for time series forecasting and power transformers monitoring. As a non-linear model, the authors make use of a multilayer perceptron neural network. The next paper presents a method which takes into account calendar effects for short-term forecasting of the visits to the emergency department of a hospital. This method combines a calendar selection rule with a slightly modified version of the k-nearest neighbour classifier to predict the incoming visit volume for a tunable number of days ahead. The fourth contribution deals with the classification of textual messages from a log file to understand the type of messages being recorded. The method is based on both ARIMA and a hybrid ARIMA-GARCH model. The last contribution of this chapter and therefore of this book compares different time series models and some hybridizations of them to fit bivariate time series with a special aim at forecasting the unemployment rate in the USA.

Last but not least, we would like to point out that this edition of ITISE was organized by the University of Granada (UGR), Spain, together with the Spanish Chapter of the IEEE Computational Intelligence Society. The Guest Editors would also like to express their gratitude to all the people who supported them in the compilation of the book, and especially to the contributing authors for their submissions, the chairs of the different sessions and to the anonymous reviewers for their comments and useful suggestions in order to improve the quality of the papers.

We wish to thank our main sponsors as well: the Department of Computer Architecture and Technology of the UGR, the Faculty of Science of the UGR, the Research Center for Information and Communications Technologies (CITIC-UGR), the Spanish Network on Time Series (RESeT) and the Ministry of Science and Innovation for their support and grants. Finally, we wish also to thank Prof. Alfred Hofmann, Vice President Publishing—Computer Science, Springer-Verlag and Dr. Veronika Rosteck, Springer Editor, for their interest in editing a book series of Springer based on the best papers of ITISE 2019.

We hope the readers of this book find these contributions interesting and helpful.

Granada, Spain
January 2020

Olga Valenzuela
Fernando Rojas
Luis Javier Herrera
Héctor Pomares
Ignacio Rojas

Contents

Advanced Statistical and Mathematical Methods for Time Series Analysis

Econometric Models and Forecasting

Energy Time Series Forecasting

Forecasting Complex/Big Data Problems

Time Series Analysis with Computational Intelligence

Time Series Analysis and Prediction in Other Real Problems

Advanced Statistical and Mathematical Methods for Time Series Analysis

Random Forest Variable Selection for Sparse Vector Autoregressive Models

Dmitry Pavlyuk

Abstract Vector autoregressive (VAR) models are widely used for multivariate time series forecasting in many applied areas like transportation, finance, economics and brain sciences. The main advantages of a VAR model are its flexibility and ability to learn a structure of relationships from data, but the number of parameters is rapidly growing for the increasing dimensionality of the modelled time series. Thus, for high-dimensional time series estimation of VAR model's parameters becomes complicated or even infeasible. In this study, we propose the utilization of random forest (RF) techniques for learning the subset of significant relationships (feature filtering) for VAR models. The proposed approach allows determining a parsimonious specification of the VAR model, and as a result, obtains better forecasting performance. We present equation-wise and system-wise strategies for RF-based feature selection and discuss their advantages. We test properties of the proposed approach empirically by applying it to spatiotemporal urban traffic forecasting problem, which is emerging in the field of transportation and requires modelling of a large number of related time series, collected from thousands of sensors within a citywide road network. The RF-based approach is compared to the unrestricted model and to other popular variable selection methods for VAR models: the penalized VAR estimator and the refined VAR variable selection strategy. Obtained results demonstrate the advantages of the proposed RF-based approach: better forecasting accuracy, higher stability of estimates and good computational performance for high-dimensional time series.

Keywords Urban traffic forecasting · Spatiotemporal model · Feature selection · Big data · Multivariate time series

D. Pavlyuk (✉)
Transport and Telecommunication Institute, Riga, Latvia
e-mail: Dmitry.Pavlyuk@tsi.lv

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*, Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_1

1 Introduction

Number of real-time data sources is rapidly growing in all applied fields of science and technology. Thousands of sensors generate high-frequency time series in transportation, finances, brain sciences and other emerging scientific areas. The key feature of the resulting multivariate time series is extensive interdependencies of its components distributed over time. Urban traffic flows at distant road segments are related to a delay of several hours; shocks in stock markets propagate with a varying speed between industries; electroencephalography electrodes register sequences of brain areas' activation. Often the structure of these relationships is unknown and highly dynamic. Modern models of multivariate time series like vector autoregressive models have a flexible structure and allow discovering of these relationships from data. The cost of this flexibility is the large number of parameters that grow fast with increasing indimensionality of the time series. Such multi-parameter models suffer from overfitting and, as a consequence, often demonstrate bad out-of-sample forecasting accuracy. In addition, the problem is complicated by the dynamics of the relationship structure—some links could appear for a short period of time only. Thus, the strict definition of this structure is usually infeasible and researchers widely utilize data-driven feature selection methods for learning active relationships and excluding non-informative ones.

In this study, we contribute to the methodology of feature selection in multivariate time series models by application of the random forest (RF) technique. The methodology represents a special case of feature filtering and is applied to popular vector autoregressive (VAR) models.

The proposed methodology is approbated for the problem of spatiotemporal urban traffic forecasting, which is emerging in the transportation research area. Data sets of a traffic management system include information from thousands of sensors (inductive loop, cameras, etc.), deployed within a citywide road network, and perfectly demonstrate the problem of feature learning for high-dimensional multivariate time series. Spatiotemporal traffic flow relationships are caused both by physical reasons (movement of cars from one spatial location to another) and by latent reasons (simultaneous traffic flows to a city centre during morning rush hours). A structure of these relationships is highly dynamic and depends on traffic conditions, e.g. some relationships appear in a congested regime only for a short period of time. Using the large real-world data set, we demonstrate the utility of the proposed methodology for high-dimensional time series forecasting.

This paper is an extension of the work presented at the International Conference on Time Series and Forecasting (ITISE-2019) [1]. Comparing to the publication in conference proceedings, this paper has several crucial improvements: literature review was enhanced by the description of recent advances in RF-based feature selection algorithms; the research methodology was extended by the system-wise strategy of RF-based feature selection and presented in a more detailed manner; empirical results were obtained for the extended research methodology and additional hyperparameter tuning; finally, the conclusions were refactored.

2 State of the Art

VAR models [2], are the popular tool of multivariate time series forecasting. Originally developed for macroeconomic processes, nowadays VAR models are intensively applied in many areas like health research, video stream control, traffic engineering, among many others. Many real-world time series are high-dimensional in its nature and include data for hundreds or thousands of indicators. Although the application of VAR models for multivariate time series is straightforward and developed estimation algorithms are efficient, number of parameters explosively grows with increasing time series dimensionality. A large number of parameters lead to the lower forecasting performance of VAR models and creates difficulties for interpretation (the famous "curse of dimensionality" problem). Thus, the application of unrestricted VAR models for high-dimensional data is impractical. Several methodologies are suggested in the literature to deal with the curse of dimensionality. Following the terminology of feature engineering, we divide all methodologies into two classes: feature extraction and feature selection methods. Feature extraction corresponds to a reduction of the dimensionality by transforming of the high-dimensional data set into a derivative feature set of a smaller dimension. Dynamic factor models [3], which combine time series into linear factors, are the popular representatives of this class of methods. Feature selection techniques reduce the number of VAR model's parameters by setting restrictions on many model coefficients. Such limited specifications of VAR models are called sparse VAR. Comparatively, to feature extraction methodologies, sparse VAR models keep an initial set of features and discover relationships between them, which is advantageous for interpretation and further management of the analyzed process. This research is focused on the sparse VAR model specifications and corresponding feature selection methods.

2.1 Feature Selection in Vector Autoregressive Models

Feature selection techniques are conventionally subdivided [4] to filter methods, wrapper methods and embedded methods. Filter methods use preliminary feature ranking for selecting the most valuable features. In the context of VAR models, Davis et al. [5], applied a partial spectral coherence, based on conditional correlation, for feature selection in their two-step sparse model specification procedure. Other correlation-based VAR feature filtering approaches are proposed by Yang et al. [6, 7], Tanizawa et al. [8] and Yuen et al. [9]. Popular Bayesian VAR models also shrink complete VAR models towards a parsimonious specification by applying informative prior distributions of model parameters. Among several recent studies on Bayesian VAR [10, 11], Billio et al. [12] suggested Bayesian nonparametric prior distributions for VAR that combines clustering and shrinking restrictions.

The second class of feature selection techniques, wrapper methods, utilize information about VAR model performance in their iterative procedure of parsimonious specification search. Popular search strategies include stepwise-elimination of regressors and application of heuristic routines (genetic algorithms, particle swarm optimization). Classical wrapper strategies to model reduction are presented by Brüggemann [13]. Despite the good theoretical background and several promising evidences of wrapper technique application (e.g. PcGets algorithm and software [14]), this approach to a sparse VAR model specification is related to significant computational complexity and rarely used for high-dimensional time series.

The third class of feature selection techniques, embedded methods, incorporate feature selection into the model estimation process. Most popular embedded methods utilize different types of regularization penalties in VAR model estimators: L_1 (least absolute shrinkage and selection operator, LASSO) or elastic net (combination of L_1 and L_2 (Tikhonov) penalties). Penalties could be applied to all VAR parameters separately or by grouping parameters by a lag or by time series (to force the sparsity in the temporal dimension or in the indicator interrelationship structure, respectively). Regularization of high-dimensional VAR models is an emerging topic in literature: recently it was addressed by Basu and Michailidis [15], Barigozzi and Brownlees [16] and Nicholson et al. [17].

In addition to different classes of feature selection methods, discussed above, it should be mentioned that two general strategies are available: system-wise strategy and equation-wise strategy [13]. The system-wise strategy implements feature selection jointly for all VAR equations, while the equation-wise equation strategy deals with each equation independently. As VAR models are the special case of seemingly unrelated regressions and deleting features from one equation affects the estimates of others, the system strategy is more natural. At the same time, in case of the absence of instantaneous causality, equation-wise strategies also lead to optimal results [13], and could demonstrate better computational performance.

This study is focused on the development of a feature filtering method, while representatives of two other approaches (wrapper and embedded feature selection) are used for performance comparison. Regarding the strategy, we consider both system-wise and equation-wise strategies.

2.2 Random Forest for Feature Filtering

We propose to apply a random forest as a feature selection tool for controlling the sparsity of VAR models. The random forest [18], is a popular statistical learning approach, widely used for feature selection and forecasting [19]. Advantages of random forests include: ability to learn under the extremely large number of candidate features; the low computational complexity and easy parallelization of the learning algorithm; embedded estimation of feature importance; resistance to overfitting and data preprocessing problems (scaling, outliers, missing data).

A large variety of methods are developed for finding the optimal subset of features for random forest models [20]. Mainly these methods are based on backward elimination of unimportant features [21, 22], backward-forward stepwise selection [23] or permutation tests [24, 25]. Although these approaches demonstrate good forecasting performance, they are computationally intensive and does not work well for high-dimensional data. Degenhardt et al. [26] argued that although modern feature selection algorithms for random forests (like Boruta and Vita) are feasible for high-dimensional data sets, their computational complexity is high and detection power is questionable.

In contrast to optimal subset search algorithms, feature filtering for sparse VAR models is usually focused on finding potentially important features and exclusion of other ones. For example, cross-correlation-based approaches [6–8] and spectral coherence-based approaches [5, 9], use a predefined threshold for cross-correlation function or partial coherence to select candidate features. These approaches are not oriented to discovering the optimal subset of features under the assumption that estimates and forecasting power of VAR models will be adequate even with a limited number of insignificant features. Further, the VAR model specification can be refined by other methods (backward elimination or regularization). We follow this approach and use feature importance values, calculated using random forests, for the selection of potentially important features.

To the best of our knowledge, random forests are not previously applied to learning the sparsity structure of VAR models. Recently random forests were applied by Furqan and Siyal [27], Papagiannopoulou et al. [28] and Chikahara and Fujino [29], for efficient and stable learning of Granger causalities in multivariate time series, but without further application of discovered relationships. Tyralis and Papacharalampous [30] applied the random forest for feature selection in univariate autoregressive moving average models and demonstrate its preferable forecasting performance.

The point of our contribution to the hierarchy of feature selection methods for VAR models is presented in Fig. 1.

We apply the proposed random forest-based sparse VAR models to a spatiotemporal urban traffic forecasting problem and demonstrate its good computational complexity and forecasting performance. Thus, the study contributes both to the methodology of high-dimensional time series modelling and to the applied area of traffic forecasting.

3 Methodology and Data

This section presents the used notation, briefly summarizes vector autoregressive models and popular feature selection techniques, and introduces the proposed RF-based approach for feature filtering in VAR models.

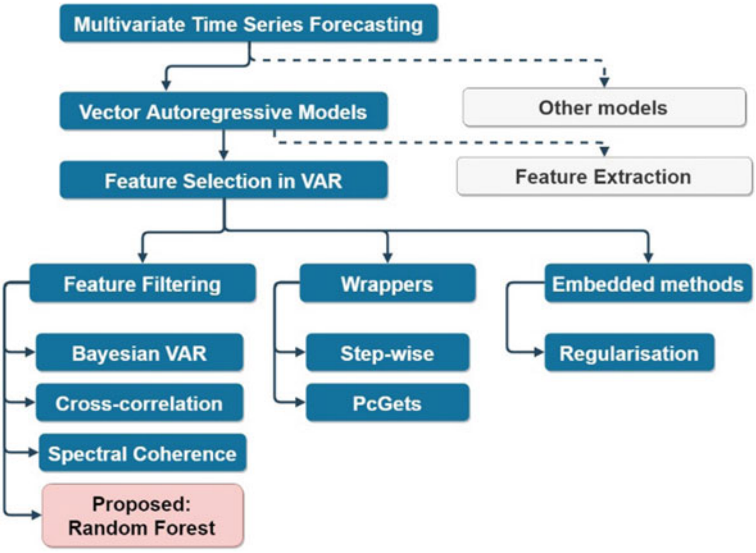


Fig. 1 Place of the study

3.1 Methods

A multivariate time series in discrete time is defined as a sequence of T observations of k-dimensional vector $Y_t = \left(y_{1,t}, y_{2,t}, \ldots, y_{k,t}\right)', t = 1, \ldots, T$. The complete (unrestricted) vector autoregressive model of order p, VAR(p), is conventionally written as:

$$Y_t = \mu + \sum_{l=1}^{p} \Phi^{(l)} Y_{t-l} + \varepsilon_t, \tag{1}$$

where $\Phi^{(l)} = \left\{\phi_{i,j}^{(l)}\right\}$ are $k \times k$ coefficient matrixes ($l = 1, \ldots, p; i, j = 1, \ldots, k$), $\mu = \{\mu_i\}$ is an optional $k \times 1$ vector of constant terms, $\varepsilon_t = \left\{\varepsilon_{i,t}\right\}$ is a $k \times 1$ vector of unobservable zero mean disturbances and non-singular covariance matrix $\sum_\varepsilon$.

The sparsity of VAR(p) models corresponds to setting elements of coefficient matrixes $\Phi^{(l)}$ to zero to reduce the number of model parameters. In this study, we consider the filter approach to controlling the model sparsity, which is based on the selection of non-zero coefficient before model estimation. Thus, we formulate the sparse VAR(p) model, introducing a set of binary matrixes $\mathrm{S}^{(l)} = \left\{s_{i,j}^{(l)}\right\}$ that represents relationships in VAR(p)

$$Y_t = \mu + \sum_{l=1}^{p} S^{(l)} \Phi^{(l)} Y_{t-l} + \varepsilon_t. \tag{2}$$

We will refer $S = \left[S^{(1)}, S^{(2)}, \ldots, S^{(p)}\right]$ and $\Phi = \left[\Phi^{(1)}, \Phi^{(2)}, \ldots, \Phi^{(p)}\right]$.

VAR(p) model can be fit by the ordinary least squares (OLS) estimator, but the number of estimated parameters equals to $\left(pk^2 + k\right)$ and becomes extremely large for high-dimensional time series. Regularization is a usual way to overcome the curse of dimensionality, which introduces a penalty function $\mathcal{P}(\Phi)$ with a regularization multiplier λ into the estimator objective function.

As an alternative approach, we propose to use random forests for feature filtering. Random forest is a popular machine learning technique, proposed by Breiman in 2001 [18]. This technique is widely used for feature selection and includes the following steps [19]:

1. Sample with the replacement of n training sets $\{Y_{ts}\}, ts \subseteq \{1, \ldots, T\}$
2. Training of a regression tree for Eq. (4), for every training set, randomly selecting features for every tree node
3. Estimating of the importance of each feature in every regression tree
4. Combining the obtained feature importance values (e.g. by averaging over training sets).

The key component for RF-based feature filtering is a selected metric of feature importance. In this study, we apply an increase of mean squared error (MSE) for these purposes. The metric is calculated as follows:

- Out-of-bag MSE is calculated for the random forest for VAR equation i ($MSE_{i,0}$);
- Values of the variable j are randomly permuted, the new random forest model is estimated, and its MSE is calculated as ($MSE_{i,j}$). If the variable j improves the forecasting performance, $MSE_{i,j}$ will be larger than $MSE_{i,0}$;
- Increase of MSE is calculated as ($MSE_{i,j}$—$MSE_{i,0}$)/$MSE_{i,0}$.

The parsimony of the resulting model specification plays an important role in high-dimensional models of tightly coupled time series. In practice, the model with simpler specification is frequently preferred over the model with slightly higher forecasting accuracy due to easier interpretation and further usage of the results for operational and strategic decision-making. Thus, a trade-off between the number of selected features and the model forecasting performance needs to be defined within the methodology. We control this trade-off their direct specification of the resulting sparsity of the VAR model. Similar to cross-correlation and partial coherence-based approaches, we are focused on finding the subset of potentially important features instead of the optimal subset. Thus, we do not utilize more advanced algorithms for RF feature selection (like Boruta [25]) and select features using the obtained increase of MSE values directly. For this selection, we utilize a threshold ($IncreaseMSE_{LB}$) that is a lower bound for the metric values, so only features with higher values are included in the resulting subset. Analyzed $IncreaseMSE_{LB}$ values are non-negative ($IncreaseMSE_{LB} = 0$ corresponds to exclusion of feature that have the negative effect of model forecasting performance) and case-specific, so the optimal value could be defined by cross-validation. Note that the distribution of the increase of MSE values is not normal (due to the correlation between errors in random trees), so classical tests for significance could be misleading.

Further, we have two strategies for the utilization of equation-wise feature importance values. The first strategy is the equation-wise, where the resulting feature set is determined for every equation independently. We directly utilize the $IncreaseMSE_{LB}$ threshold to implement this strategy. The second strategy is system-wise, where the resulting feature set is simultaneously determined for all equations. To implement this strategy, we aggregate feature importance metrics, obtained from the equation-wise RF models, and execute feature filtering of the joint results. Potentially, scales of dependent variables in different equations could be different, so a percentage increase of MSE is inappropriate for feature filtering (increase by 1% for one variable could be more beneficial than increase by 10% for another one). So, instead of the percentage increase of MSE, we use raw $MSE_{i,j}$ the system strategy. All other routines that presented above (including the $IncreaseMSE_{LB}$ threshold) are used without changes.

The resulting feature set is used for sparse VAR model specification and estimation. Summarizing the methodologies, stated above, we formulate 4 alternative model specifications:

- Unrestricted VAR model.
- Refined VAR (a model with excluded insignificant coefficients by backward elimination procedure, the system-wise strategy)—a representative of wrapper feature selection.
- Penalized VAR (LASSO penalties; the system-wise strategy)—a representative of the embedded feature selection.
- Random forest-based (RF-based) sparse VAR (separately for equation-wise and system-wise strategies)—the proposed representative of feature filtering.

The primary research question lies in a comparative forecasting performance of the candidate models. We applied the rolling analysis [31], with a constant window size (look-back interval) for tuning of hyperparameters and estimation of models' out-of-sample forecasting accuracy. Parameters of every model specification were tuned independently:

- Complete VAR model: look-back interval; maximal lag p.
- Refined VAR model: look-back interval; maximal lag p.
- Penalized VAR: look-back interval; regularization parameter λ; maximal lag p.
- RF-based sparse VAR: look-back interval; maximal lag p; strategy: system or equation-wise; threshold for feature importance, $IncreaseMSE_{LB}$.

The out-of-sample mean absolute error (MAE), averaged by time series, is used as the primary forecasting accuracy metric

$$MAE_t = \frac{1}{k}\sum_{i=1}^{k}\left|y_{i,t} - \hat{y}_{i,t}\right| \tag{3}$$

where $\hat{y}_{i,t}$ is a predicted value for a spatial location i and time point t.

In addition to MAE, forecasting accuracy of the candidate models is estimated by the mean absolute scaled error (MASE), which is a ratio of model MAE values and one-step ahead naïve forecast.

3.2 Data: Urban Traffic Forecasting

We applied the proposed methodology to a multivariate time series of urban traffic volume values, obtained from 103 stations on arterial roads in Minneapolis, USA. All stations are located within 6 min of travel time in uncongested traffic conditions from the city centre. We collected the data for 40 weeks (01 Jan 2017–07 Oct 2017) and temporally aggregated them in 1 min time frames. The first 30 weeks of data were used for learning of historical patterns and the last 10 weeks—for model validation. Historical patterns are learned independently for every univariate time series as median values, specific to a day of the week and time of the day. Data for 10 weeks, designed for model validation was detrended by subtracting historical patterns. In addition, we implemented standard data preprocessing procedures: removal of outliers (based on the physical capacity of roads); imputing missed data (by linear interpolation), and winsorization (by lower and upper bounds, identified by the interquartile range technique of outlier detection).

4 Results

Dimensionality of modelled time series is a key input for the sparse model specification. We tested all candidate models for two data sets:

- Random sample of 10 stations ($k = 10$) that referred to as the low-dimensional data set.
- Complete set of 103 sensors ($k = 103$) that are referred to as the high-dimensional data set.

We assume that the dimensionality of the first multivariate time series is small enough to keep stable estimates of the complete VAR model, while for the second data set sparse VAR specification will be beneficial.

Hyperparameter tuning was executed by a grid search, where every combination of hyperparameter values is tested by rolling window cross-validation. The rolling window was shifted over 69 days (10 weeks minus the first day that is used for a look-back window) every 4 h, which resulted in 414 model estimates per hyperparameter combination. Look-back interval is 16 h for all models ($T = 960$ min)—a minimal length of time series that ensure stabilized results on model forecasting performance.

The resulting hyperparameters values were selected as

- Optimal order of VAR models is 3 ($p = 3$), which was expected due to a limited spatial area of analysis (maximum travel time between sensors is 6 min in normal traffic conditions).
- Regularization parameter λ for the penalized VAR model was selected in a flexible manner for every cross-validation subsample. The time-specific optimal value is obtained by splitting the data set into two equal parts and using the second part for the local cross-validation of the regularization parameter λ [17].
- Sparsity ($IncreaseMSE_{LB}$ threshold) for the RF-based sparse VAR model is selected from a set of corresponding quartiles for non-negative values—0, 20, 50 and 100% (exclude only variables with negative effects on forecasting accuracy). Optimal sparsity for low-dimensional data set was selected as 50% (5 explanatory variables of 10 per time lag), for high-dimensional data set—as 20% (20 explanatory variables of 103 per time lag). Optimal values for system-wise and equation-wise strategies are discovered as identical.

The penalized VAR and RF-controlled sparse VAR model specifications allow tuning of parameters for a specific forecasting horizon. In this study, we arbitrary trained both models to optimize the one-step ahead MAE.

The resulting forecasting performance of the candidate models with optimally selected hyperparameters is presented in Table 1 (for low-dimensional data set) and Table 2 (for high-dimensional data set). The obtained one-step ahead forecast average MAE values are almost identical for all candidate models for the low-dimensional data set, while differ significantly for the high-dimensional one. In addition to average MAE and MASE values, we provide their 95th percentiles to explore spatial and temporal stability of obtained forecasts. Discussion of the presented results is provided in the next paper section.

Another point of our interest is the stability of model forecasting performance for longer forecasting horizons. We constructed h-step ahead forecasts for all models (h is a forecasting horizon, $h = 1, \ldots, 12$) using the iterative one-step ahead strategy (so forecasts for the next step were calculated using values, forecasted for the previous steps) and combined them into aggregated forecasts for longer intervals (from 1 to

Table 1 One-step ahead forecasting performance (low-dimensional data set)

Model	Average MAE	95th percentile of MAE	Average MASE	95th percentile of MASE
Complete VAR	4.14	6.37	0.521	0.613
Penalized VAR	4.15	6.33	0.516	0.619
Refined VAR	4.15	6.31	0.527	0.626
RF-based sparse VAR (equation-wise)	4.13	6.31	0.513	0.603
RF-based sparse VAR (system-wise)	4.14	6.34	0.520	0.622

Table 2 One-step ahead forecasting performance (high-dimensional data set)

Model	Average MAE	95th percentile of MAE	Average MASE	95th percentile of MASE	Complexity*, seconds
Complete VAR	4.53	8.47	0.592	0.878	0
Penalized VAR	4.69	6.89	0.539	0.682	682
Refined VAR	3.99	7.31	0.515	0.750	1169
RF-based sparse VAR (equation-wise)	4.06	6.57	0.503	0.675	871
RF-based sparse VAR (system-wise)	4.07	6.45	0.508	0.690	871

*Computation complexity, average seconds per model for feature selection

h min). Further, average MAE values were calculated for aggregated forecasts. A comparison of the obtained results is presented in Fig. 2 (for the low-dimensional data set) and Fig. 3 (for the high-dimensional data set).

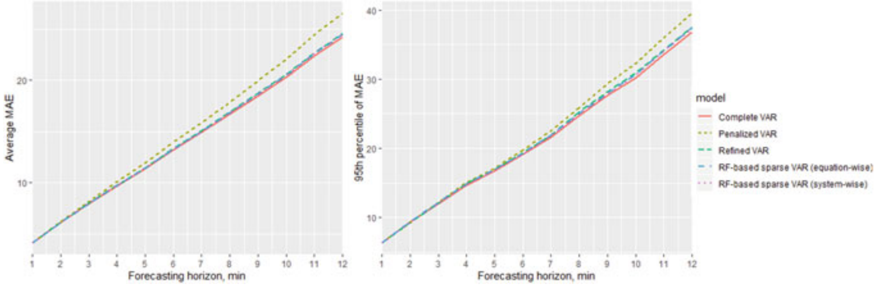


Fig. 2 Accuracy of the candidate models (mean and 95th percentile of MAE values) by forecasting horizon: low-dimensional data set

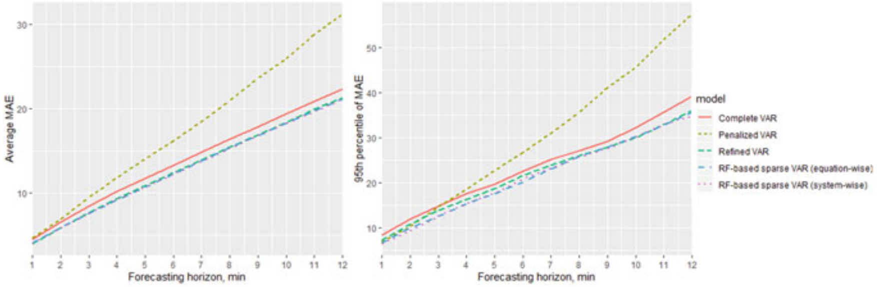


Fig. 3 Accuracy of the candidate models (mean and 95th percentile of MAE values) by forecasting horizon: high-dimensional data set

Note that the *h*-step ahead forecasting accuracy is almost identical for all models, except the penalized VAR, for low-dimensional data, but differ significantly for high-dimensional ones.

We provide open-source codes (R) for all developed procedures in the public repository http://bit.ly/ITISE2019 to ensure the reproducibility of the obtained results.

5 Discussion

The primary research interest is the comparison of the forecasting performance of the proposed RF-based sparse VAR model specification against other alternatives. For the low-dimensional data set, the RF-based sparse VAR model's forecasting performance is almost identical to the complete VAR model (average MAE is 4.14 for both models, Table 1). While the forecasting performance of models is similar, the parsimonious specification of the RF-based sparse VAR model could be considered as an advantage, because it provides an easier understanding of existing relationships and leads to more interpretable and manageable results. These results become more important under the observed stability of the RF-based sparse VAR model's forecasting performance for longer time intervals (Fig. 2). Performance of RF-based sparse VAR model is degrading almost with the same speed as the complete VAR model, while average MAE of the competitor penalized VAR is growing much faster (note that both RF-based VAR and penalized VAR are trained to optimize one-step ahead forecasts, so no model has a prespecified advantage). Thus, we conclude the good performance of the RF-based sparse VAR model for low-dimensional data sets, where the unrestricted VAR is widely considered as a primary model specification.

For larger dimensionality, the proposed RF-based sparse VAR model demonstrates a clear advantage in forecasting performance over complete and penalized VAR models. Its average one-step ahead forecast MAE value is 4.06 against 4.53 of the complete VAR model (Table 1), which is a statistically significant difference for the utilized number of cross-validation subsamples. This advantage keeps stable over longer forecasting intervals (Fig. 3), while forecasting accuracies of the complete and penalized VAR are degraded faster. Forecasting performance of the RF-based sparse VAR model is similar to the refined VAR, but its computational complexity is much lower (Table 2 contains average computation times for feature selection on the identical environment). Another comparative advantage of the RF-based sparse VAR model against all competitor specifications is demonstrated by the 95th percentile values of MAE—6.57 for the RF-based VAR model against 8.47 for the complete VAR (Table 2). This fact is considered as an evidence of improvement of the stability of the forecasting performance over space and time and supports our hypothesis about the general advantage of the proposed approach.

Regarding the equation-wise and system-wise RF-based strategies, our empirical results demonstrate almost the identical forecasting performance of the resulting VAR model's specifications. The absence of significant differences is also observed for the stability of forecasting accuracy over spatial and temporal dimensions (in terms of 95th percentile of MAE values), parsimony of the model specification (the sparsity level) and accuracy for longer forecasting horizons (Fig. 3). This fact can be case-specific—urban traffic flows at different road segments have the same nature and usually similar in terms of time series features. In this special case, equation-wise MAE values are comparable and their joint filtering does not overcome equation-wise selection. Another potential reason for the similarity of the equation-wise and system-wise results could be related to data preprocessing—traffic flow time series were detrended, so the model is used to forecast deviations from the historical averages. A comparison of the equation-wise and system-wise strategies has a case-specific character and requires additional empirical evidences.

In addition to the primary research interest, we should mention several observations from the obtained results:

1. Forecasting performance of the refined VAR model overcomes the complete VAR specification both for low-dimensional and high-dimensional data sets, but requires intensive computations;
2. Penalized VAR model's performance strictly depends on the prespecified target forecasting horizon (one-step ahead in our experiments), so this model should be separately tuned for every forecasting horizon;
3. Computational complexity of the proposed RF-based sparse VAR model is growing fast with increasing dimensionality of the time series, but keeps manageable (at least for several hundreds of dimensions) and the related algorithm is easily parallelized.

6 Conclusions

In this paper, we propose a new random forest-based approach to variable selection for vector autoregressive models. Within the proposed approach, we utilize the random forest for equation-wise feature selection and further apply the most important features for sparse VAR model specification.

The proposed approach was applied to the real-world urban traffic data set and tested against alternative model specifications: unrestricted VAR; refined VAR with excluded insignificant coefficients; and LASSO-penalized VAR. Obtained experimental results demonstrated the advantage of the proposed RF-based sparse VAR model in several aspects.

1. Forecasting performance of the RF-based sparse VAR model overcomes the performance of analyzed competitive models for high-dimensional data.
2. Parsimonious specification of the RF-based sparse VAR is also appropriate for low-dimensional data, which is an advantage in terms of model interpretability.
3. The proposed approach inherits advantages of random forests such as the ability to learn under an extremely large number of candidate features; low computational complexity, easy parallelization; resistance to overfitting, which makes it appropriate for high-dimensional modelling of big data.

Finally, we should mention a wide area for the future research in this direction. The system-wise strategy, described in this paper, should be extended to simultaneous learning of feature importance in all equations. This improvement requires the methodological development of multi-output random forests and their application for multivariate time series. In addition, obtained empirical results are case-specific, so deeper validation of the proposed approach for other data sets is highly required.

Acknowledgements The author was financially supported by the specific support objective activity 1.1.1.2. “Post-doctoral Research Aid” (Project id. N. 1.1.1.2/16/I/001) of the Republic of Latvia, funded by the European Regional Development Fund. Dmitry Pavlyuk’s research project No. 1.1.1.2/VIAA/1/16/112 “Spatiotemporal urban traffic modelling using big data”.

References

1. Pavlyuk, D.: Random Forest-controlled sparsity of high-dimensional vector autoregressive models. In: Valenzuela, O., Rojas, F., Pomares, H., Rojas, I. (eds.) ITISE 2019. International Conference on Time Series and Forecasting. Proceedings of Papers, pp. 343–354. Godel Impresiones Digitales S. L., Granada, Spain (2019)
2. Sims, C.A.: Macroeconomics and reality. Econometrica. **48**, 1 (1980). https://doi.org/10.2307/1912017
3. Forni, M., Lippi, M.: The generalized dynamic factor model: representation theory. Econom. Theory. **17**, 1113–1141 (2001)
4. Chandrashekar, G., Sahin, F.: A survey on feature selection methods. Comput. Electr. Eng. **40**, 16–28 (2014). https://doi.org/10.1016/j.compeleceng.2013.11.024
5. Davis, R.A., Zang, P., Zheng, T.: Sparse vector autoregressive modeling. J. Comput. Graph. Stat. **25**, 1077–1096 (2016). https://doi.org/10.1080/10618600.2015.1092978
6. Yang, K., Yoon, H., Shahabi, C.: CLe Ver: A feature subset selection technique for multivariate time series. In: Ho, T.B., Cheung, D., Liu, H. (eds.) Advances in Knowledge Discovery and Data Mining, pp. 516–522. Springer, Berlin (2005). https://doi.org/10.1007/11430919_60
7. Yang, K., Yoon, H., Shahabi, C.: A supervised feature subset selection technique for multivariate time series, vol. 10 (2005)
8. Tanizawa, T., Nakamura, T., Taya, F., Small, M.: Constructing directed networks from multivariate time series using linear modelling technique. Phys. Stat. Mech. Its Appl. **512**, 437–455 (2018). https://doi.org/10.1016/j.physa.2018.08.137
9. Yuen, T.P., Wong, H., Yiu, K.F.C.: On constrained estimation of graphical time series models. Comput. Stat. Data Anal. **124**, 27–52 (2018). https://doi.org/10.1016/j.csda.2018.01.019
10. Carriero, A., Clark, T.E., Marcellino, M.: Large Bayesian vector autoregressions with stochastic volatility and non-conjugate priors. J. Econom. (2019). https://doi.org/10.1016/j.jeconom.2019.04.024
11. Koop, G., Korobilis, D., Pettenuzzo, D.: Bayesian compressed vector autoregressions. J. Econom. **210**, 135–154 (2019). https://doi.org/10.1016/j.jeconom.2018.11.009
12. Billio, M., Casarin, R., Rossini, L.: Bayesian nonparametric sparse VAR models. J. Econom. (2019). https://doi.org/10.1016/j.jeconom.2019.04.022
13. Brüggemann, R.: Model Reduction Methods for Vector Autoregressive Processes. Springer, Berlin (2004)
14. Hendry, D., Krolzig, H.-M.: Automatic Econometric Model Selection Using PcGets. Timberlake Consultants Press, London (2001)
15. Basu, S., Michailidis, G.: Regularized estimation in sparse high-dimensional time series models. Ann. Stat. **43**, 1535–1567 (2015). https://doi.org/10.1214/15-AOS1315

16. Barigozzi, M., Brownlees, C.: NETS: network estimation for time series. J. Appl. Econom. **34**, 347–364 (2019). https://doi.org/10.1002/jae.2676
17. Nicholson, W.B., Matteson, D.S., Bien, J.: VARX-L: structured regularization for large vector autoregressions with exogenous variables. Int. J. Forecast. **33**, 627–651 (2017). https://doi.org/10.1016/j.ijforecast.2017.01.003
18. Breiman, L.: Random forests. Mach. Learn. **45**, 5–32 (2001). https://doi.org/10.1023/A:1010933404324
19. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Springer, New York (2009). https://doi.org/10.1007/978-0-387-84858-7
20. Speiser, J.L., Miller, M.E., Tooze, J., Ip, E.: A comparison of random forest variable selection methods for classification prediction modeling. Expert Syst. Appl. **134**, 93–101 (2019). https://doi.org/10.1016/j.eswa.2019.05.028
21. Svetnik, V., Liaw, A., Tong, C., Wang, T.: Application of Breiman's random forest to modeling structure-activity relationships of pharmaceutical molecules. In: Roli, F., Kittler, J., Windeatt, T. (eds.) Multiple Classifier Systems, pp. 334–343. Springer, Berlin (2004). https://doi.org/10.1007/978-3-540-25966-4_33
22. Díaz-Uriarte, R., Alvarez de Andrés, S.: Gene selection and classification of microarray data using random forest. Bioinformatics **7**, 3 (2006). https://doi.org/10.1186/1471-2105-7-3
23. Genuer, R., Poggi, J.M., Malot, C.T.: VSURF: an R package for variable selection using random forests. R J. **7**, 19–33 (2015). https://doi.org/10.32614/RJ-2015-018
24. Altmann, A., Toloşi, L., Sander, O., Lengauer, T.: Permutation importance: a corrected feature importance measure. Bioinformatics **26**, 1340–1347 (2010). https://doi.org/10.1093/bioinformatics/btq134
25. Kursa, M.B., Rudnicki, W.R.: Feature selection with the boruta package. J. Stat. Softw. **36**, (2010). https://doi.org/10.18637/jss.v036.i11
26. Degenhardt, F., Seifert, S., Szymczak, S.: Evaluation of variable selection methods for random forests and omics data sets. Brief. Bioinform. **20**, 492–503 (2019). https://doi.org/10.1093/bib/bbx124
27. Furqan, M.S., Siyal, M.Y.: Random forest granger causality for detection of effective brain connectivity using high-dimensional data. J. Integr. Neurosci. **15**, 55–66 (2016). https://doi.org/10.1142/S0219635216500035
28. Papagiannopoulou, C., Miralles, D.G., Decubber, S., Demuzere, M., Verhoest, N.E.C., Dorigo, W.A., Waegeman, W.: A non-linear granger-causality framework to investigate climate–vegetation dynamics. Geosci. Model Dev. **10**, 1945–1960 (2017). https://doi.org/10.5194/gmd-10-1945-2017
29. Chikahara, Y., Fujino, A.: Causal inference in time series via supervised learning. In: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, pp. 2042–2048. Stockholm, Sweden (2018). https://doi.org/10.24963/ijcai.2018/282
30. Tyralis, H., Papacharalampous, G.: Variable selection in time series forecasting using random forests. Algorithms **10**, 114 (2017). https://doi.org/10.3390/a10040114
31. Zivot, E., Wang, J.: Rolling analysis of time series. In: Modeling Financial Time Series with S-PLUS, pp. 313–360. Springer, New York (2006). https://doi.org/10.1007/978-0-387-32348-0_9

Covariance Functions for Gaussian Laplacian Fields in Higher Dimension

Gyorgy H. Terdik

Abstract In this paper, we describe the covariance function of a general Laplacian $AR(p)$ model in the higher dimension. The speed of decay is considered also showing that the exponential decay is also possible in higher dimensions at particular values of the order p, which is not necessarily an integer. Vecchia's method is applied for getting covariance functions corresponding rational spectra of stochastic Laplacian fields in three and higher dimensions.

Keywords Stochastic Laplacian fields · Stationary ARMA fields · Covariance function · Rational spectrum

1 Introduction

Modelling spatial data has become an important subject in diverse application areas including environmental and geophysical sciences, astrophysics, renewable energies, etc. [1–4]. Among these models, the dynamical processes are of utmost importance, since they describe the data through stochastic partial differential equations having a small number of parameters to be estimated. Assuming Gaussianity, the covariance function contains all the information necessary for modelling. Whittle [5] considered a model using first-order stochastic Laplacian equation on the plane, which corresponds to an $AR(1)$ model in time series analysis having a simple rational spectrum and a corresponding covariance function. More general rational spectra and covariance functions have been considered by Vecchia [6] and those have been used for model-identification Vecchia [7], and fitting continuous ARMA models to unequally

Supported by EFOP-3.6.2-16-2017-00015 supported by the European Union, co-financed by the European Social Fund.

G. H. Terdik (✉)
Faculty of Informatics, University of Debrecen, Kassai U. 26, 4028 Debrecen, Hungary
e-mail: terdik.gyorgy@inf.unideb.hu
URL: https://www.researchgate.net/profile/Gyorgy_Terdik

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_2

spaced spatial data, Jones [8, 9], on the plane. On this line, a particular covariance function has been used for modelling three-dimensional flows in hydrology by [10].

In this paper, we describe the covariance function for a general Laplacian $AR(p)$ model in higher dimensions. The speed of decay is considered also showing that the exponential decay is also possible in higher dimensions at some particular values of the order p, which is not necessarily a integer. Vecchia's method is applied for getting covariance functions corresponding rational spectra of stochastic Laplacian fields in three and four dimensions. These results allow further development of the covariance structure for spatio-temporal dynamical models given by Subba Rao and Terdik [11–13].

2 Frequency Domain Treatment of Stationary Fields in Higher Dimensions

As we shall see later, a stochastic Laplacian equation under some natural assumption provides a stationary solution, i.e., a stationary real-valued field on $\mathbb{R}^d$. We call a random field $X\left(\underline{x}\right), \underline{x} \in \mathbb{R}^d$, (second-order) stationary if it is homogeneous and isotropic, namely if the mean of $X\left(\underline{x}\right)$ is constant and the covariance function is invariant under translations, i.e., it is homogeneous, more over the covariance function is invariant under group of rotations, i.e., it is isotropic, [14–16].

We consider stationary fields $X\left(\underline{x}\right)$, with $\mathrm{E}\, X\left(\underline{x}\right) = 0$, the covariance function of a stationary field $X\left(\underline{x}\right)$ depends on the distance of locations only

$$\mathcal{C}\left(r\right) = \operatorname{Cov}\left(X\left(\underline{x}_1\right), X\left(\underline{x}_2\right)\right),$$

where $r = \left|\underline{x}_1 - \underline{x}_2\right|$. The spectral representation of a homogeneous field $X\left(\underline{x}\right)$ writes

$$X\left(\underline{x}\right) = \int_{\mathbb{R}^d} e^{i\underline{x}\cdot\underline{\lambda}} Z\left(d\underline{\lambda}\right), \quad \underline{\lambda}, \underline{x} \in \mathbb{R}^d,$$

where $\underline{x} \cdot \underline{\lambda} = \sum_{i=1}^{d} x_i \lambda_i$, and $Z\left(d\underline{\lambda}\right)$ is a stochastic spectral measure, [16].

From now on, let $\underline{x}, \underline{\lambda} \in \mathbb{R}^d$, and denote $r = \left|\underline{x}\right|, \rho = \left|\underline{\lambda}\right|$, such that $\underline{x} = r\widetilde{\underline{x}}, \underline{\lambda} = \rho\widetilde{\underline{\lambda}}$, where $\widetilde{\underline{x}}, \widetilde{\underline{\lambda}}$ are unit vectors from the unit sphere $\mathbb{S}_{d-1}$, in $\mathbb{R}^d$. Let us assume further that $d \geq 3$. Now, we rewrite the spectral representation in spherical polar coordinates

$$X\left(r, \widetilde{\underline{x}}\right) = \int_0^\infty \int_{\mathbb{S}_{d-1}} e^{i\rho r \widetilde{\underline{\lambda}}\cdot\widetilde{\underline{x}}} Z\left(\Omega\left(d\widetilde{\underline{\lambda}}\right)\rho^{d-1} d\rho\right), \tag{1}$$

where $\Omega\left(d\widetilde{\underline{\lambda}}\right)$ is the Lebesgue element of the surface area on $\mathbb{S}_{d-1}$. The Jacobi–Anger expansions of the exponent, see 14, provides the decomposition of $X\left(\underline{x}\right)$ in terms of polar coordinates $\left(r, \widetilde{\underline{x}}\right)$:

$$X\left(r,\widetilde{\underline{x}}\right)=s_{d-1}\sum_{\ell=0}^{\infty}\sum_{m=1}^{h(\ell,d)}i^{\ell}Y_{\ell}^{m}\left(\widetilde{\underline{x}}\right)\int_{0}^{\infty}j_{d,\ell}\left(r\rho\right)Z_{\ell}^{m}\left(\rho^{d-1}d\rho\right),\tag{2}$$

where $s_{d-1}=2\pi^{d/2}/\Gamma\left(d/2\right)$ is the surface area of the sphere $\mathbb{S}_{d-1}$, Y_{ℓ}^{m} and $j_{d,\ell}$ denote the Orthonormal Spherical Harmonics, see Appendix, item 1, and the Spherical Bessel function of the first kind, see (12), respectively.

Remark 1 We notice at this point that the case $d=2$ is a slightly different. The reason is that if $d=2$ the expansion (2) writes in terms of $\cos\ell\vartheta$, and $\sin\ell\vartheta$ traditionally, instead of the normed versions $Y_{\ell}^{1}\left(\widetilde{\underline{x}}\right)=\cos\ell\vartheta/\sqrt{2\pi}$, and $Y_{\ell}^{2}\left(\widetilde{\underline{x}}\right)=\sin\ell\vartheta/\sqrt{2\pi}$.

The decomposition (2) contains countable number of uncorrelated stationary processes since the stationarity of X implies and implied by that the spectral measure

$$F\left(d\underline{\lambda}\right)=E\left|Z\left(d\underline{\lambda}\right)\right|^{2}=E\left|Z\left(\Omega\left(d\widetilde{\underline{\lambda}}\right)\rho^{2}d\rho\right)\right|^{2}=F\left(\Omega\left(d\widetilde{\underline{\lambda}}\right)\rho^{2}d\rho\right),$$

is separated in space according to the direction $\widetilde{\underline{\lambda}}$ and wave number ρ, i.e., $F\left(\Omega\left(d\widetilde{\underline{\lambda}}\right)\rho^{2}d\rho\right)=\Omega\left(d\widetilde{\underline{\lambda}}\right)F\left(\rho^{2}d\rho\right)$, therefore the stochastic spectral measures

$$Z_{\ell}^{m}\left(\rho^{d-1}d\rho\right)=i^{\ell}\int_{\mathbb{S}_{d-1}}Y_{\ell}^{m}\left(\widetilde{\underline{\lambda}}\right)^{*}Z\left(\Omega\left(d\widetilde{\underline{\lambda}}\right)\rho^{d-1}d\rho\right)$$

in (2) are orthogonal

$$\begin{aligned}\operatorname{Cov}\left(Z_{\ell_1}^{m_1}\left(\rho_1 d\rho_1\right),Z_{\ell_2}^{m_2}\left(\rho_2 d\rho_2\right)^{*}\right)&=i^{(\ell_1-\ell_2)}\int_{\mathbb{S}_2}Y_{\ell_1}^{m_1}\left(\widetilde{\underline{\lambda}}\right)Y_{\ell_2}^{m_2}\left(\widetilde{\underline{\lambda}}\right)F\left(\Omega\left(d\widetilde{\underline{\lambda}}\right)\rho^{d-1}d\rho\right)\\&=i^{(\ell_1-\ell_2)}\delta_{\ell_1,\ell_2}\delta_{m_1,m_2}F\left(\rho^{d-1}d\rho\right).\end{aligned}$$

In case of absolute continuity of the spectral measure we have

$$F\left(\rho^{d-1}d\rho\right)=f\left(\rho\right)\rho^{d-1}d\rho,$$

where $f\left(\rho\right)$ denotes the spectral density, the spectrum for short.

2.1 Covariance Functions of Laplacian Fields

First we consider the connection between the spectrum and covariance function in general. One uses formulae (15) and (16) and arrives at the following form of the covariance function:

$$\begin{aligned}\mathcal{C}(r) &= \operatorname{Cov}\left(X\left(r_1, \underline{\widetilde{x}}_1\right), X\left(r_2, \underline{\widetilde{x}}_2\right)\right) \\ &= s_{d-1} \int_0^\infty j_{d,0}(\rho r) F\left(\rho^{d-1} d\rho\right),\end{aligned}$$

and in case of absolute continuity of F we replace the spectral measure by the spectrum

$$\mathcal{C}(r) = s_{d-1} \int_0^\infty j_{d,0}(\rho r) f(\rho)\, \rho^{d-1} d\rho.$$

The spectrum is also expressed by the inverse Fourier transform

$$f(\rho) = \frac{s_{d-1}}{(2\pi)^d} \int_0^\infty j_{d,0}(\rho r)\, \mathcal{C}(r)\, r^{d-1} dr,$$

as usual.

2.2 AR(p)

The most important model for a stationary field $X\left(\underline{x}\right)$ is a solution of stochastic Laplacian equation

$$\left(\triangle - c^2\right)^p X\left(\underline{x}\right) = \partial W\left(\underline{x}\right), \tag{3}$$

where $p > d/4$, c is a nonzero real constant and $\partial W\left(\underline{x}\right)$ denotes the Gaussian white noise, see Appendix, item 6, for the Laplacian operator $\triangle$. Here $X\left(\underline{x}\right)$ is a linear transform of the white noise. In general when $X\left(\underline{x}\right)$ is the result of a linear filter $A(\rho)$ on the white noise then the spectral density $f(\rho)$ has the form

$$f(\rho) = \frac{\sigma^2}{(2\pi)^d} \left|A(\rho)\right|^2, \tag{4}$$

in particular the spectral density of the solution of the equation (3) is given by

$$f(\rho) = \frac{1}{(2\pi)^d} \frac{\sigma^2}{\left(\rho^2 + c^2\right)^{2p}}, \tag{5}$$

which shows readily that $f(\rho)$ depends only on the wave number $\rho = \left|\underline{\lambda}\right|$, the distance between two points in frequency domain, and therefore the solution $X\left(\underline{x}\right)$ is stationary. The spectrum $f(\rho)$ is a rational function, hence it can be considered as an analogue of the spectrum of an autoregressive time series, so it looks reasonable calling $X\left(\underline{x}\right)$ to an AR(p) field. The corresponding covariance function follows from known integrals of Bessel functions, see [17], 11.4.44,

$$\begin{aligned}\mathcal{C}(r) &= \frac{s_{d-1}}{(2\pi)^d}\sigma^2 \int_0^\infty j_{d,0}(r\rho)\frac{\rho^{d-1}d\rho}{\left(\rho^2+c^2\right)^{2p}} \\ &= \sigma^2\frac{\Gamma(d/2)}{\Gamma(2p)}\left(\frac{r}{2c}\right)^{2p-d/2} K_{2p-d/2}(cr),\end{aligned} \tag{6}$$

where $K_{2p-d/2}$ denotes the modified Bessel function, see Appendix, item 7. Some broader assumption for the existence of the above integral is $d < 8p + 1$, nevertheless having a covariance function we need a finite positive limit at zero, and therefore we should assume $d/4 < p$, see (19).

Whittle [5] has experienced that for the case $d = 2, p = 1$, i.e. for an AR(1) field on the plane, the rate of decay of the correlation function is slower than exponential, which does not happen for an AR(p) model in time series analysis. Indeed we use (6), and obtain the correlation function

$$\mathcal{R}(r) = \frac{2}{\Gamma(2p-d/2)}\left(\frac{cr}{2}\right)^{2p-d/2} K_{2p-d/2}(cr),$$

with the rate of decay

$$\mathcal{R}(r) \sim \frac{\sqrt{\pi}}{\Gamma(2p-d/2)}\left(\frac{cr}{2}\right)^{2p-d/2-1/2} e^{-cr}, \quad z \to \infty, \tag{7}$$

see (18). In fact $\mathcal{R}(r) \sim \mathcal{O}\left(r^{2p-d/2-1/2}e^{-cr}\right), \quad z \to \infty$. We conclude that one can have a valid correlation function with exponential rate of decay in higher dimension easily, it happens if $d = 3$, $p = 1$, for instance, see row 3/1 in the Table 1. Let us generalize the order of the autoregression to a positive real number p, and assume $2p - d/2 - 1/2 = 0$, while $d/2 < 2p$, then by the asymptotic formula (7) the rate of decay of the correlation function is exponential. We give some more examples in Table 1.

Table 1 Correlation functions with rate of decays

d/p	$\mathcal{R}(r)$	$\mathcal{R}(r), r \to \infty$	$\mathcal{C}(0)$
$1/1$	e^{-cr}	e^{-cr}	1
$2/1$	$crK_1(cr)$	$\sqrt{\pi cr/2}e^{-cr}$	$1/2c^2$
$2/2$	$(cr)^3 K_3(cr)/2$	$\sqrt{\pi/2}(cr)^{5/2}e^{-cr}/2$	$1/24c^6$
$3/1$	e^{-cr}	e^{-cr}	$\sqrt{\pi}/2^{3/2}c$
$3/\frac{3}{2}$	$e^{-cr}(cr+1)$	cre^{-cr}	$\sqrt{\pi}/2^{7/2}c^3$
$3/2$	$e^{-cr}\left((cr)^2/3+cr+1\right)$	$(cr)^2 e^{-cr}/3$	$\sqrt{\pi/2}/2^4c^5$

2.3 ARMA Fields

A further possible generalization of the spectrum (5) is changing the AR spectrum to an ARMA one, namely into a rational function. Consider the following polynomials:

$$Q(z) = \sum_{k=0}^{q} q_k z^k, \quad P(z) = \sum_{k=0}^{p} p_k z^k,$$
$$Q(z) = \prod_{j=1}^{N} \left(z - b_j\right)^{n_j/2}, \quad P(z) = \prod_{j=1}^{M} \left(z - a_j\right)^{m_j/2} \tag{8}$$

and the stochastic equation

$$P\left(\nabla^2\right) X\left(\underline{x}\right) = Q\left(\nabla^2\right) W\left(\underline{x}\right), \tag{9}$$

by generalized sense. We have, see [15], p. 24, Theorem 11, that if $p - q > d/4$, and $P(z)$ is nonzero on $[-\infty, 0]$, then

$$f(\rho) = \frac{\sigma^2}{(2\pi)^d} \frac{Q^2\left(-\rho^2\right)}{P^2\left(-\rho^2\right)}, \tag{10}$$

is a valid spectral density where $\rho^2 = \left|\underline{\lambda}\right|^2$, and the solution $X\left(\underline{x}\right)$, is stationary.

Vecchia (1988), [7], has considered the case $d = 2$, and derived the covariance function according to the spectrum (10). Vecchia's method is based on the existence of the integral

$$\int_0^\infty j_{d,0}(r\rho) \frac{1}{\rho^2 + c^2} \rho^{d-1} d\rho = \Gamma(d/2) \left(\frac{r}{2c}\right)^{1-d/2} K_{d/2-1}(cr),$$

actually this integral is a particular case of (6), and it exists if $d < 5$, otherwise it is not necessarily a covariance function. The consequence of $d < 5$ is that Vecchia's method is valid for $d = 2, 3, 4$. We simplify the general result of Vecchia for the case of real roots of polynomials $P(z)$ and $Q(z)$.

Theorem 1 *Consider the stochastic Laplacian equation (9) for $d = 2, 3, 4$, with polynomials $P(z)$ and $Q(z)$ given by (8). Let $\left\{a_j, 1 \le j \le M\right\}$, satisfying $a_j > 0$, and $\left\{b_j, 1 \le j \le N\right\}$ be distinct real numbers, moreover m_j, n_j be positive integers. Then the covariance function of the ARMA filed (9) is given by*

$$\mathcal{C}_d(r) = (-1)^{2p-1} \sigma^2 \frac{S_{d-1}}{(2\pi)^d} \sum_{j=1}^{M} \frac{1}{\left(m_j - 1\right)!} \frac{\partial^{m_j-1}}{\partial a_j^{m_j-1}} W_j G_d\left(\sqrt{a_j} r\right), \tag{11}$$

where $2p=\sum_{j=1}^{M}m_j\geq\sum_{j=1}^{N}n_j+2,\quad r>0,\quad W_j=\prod_{k=1}^{N}\left(b_k-a_j\right)^{n_j}\diagup\prod_{k=1\neq j}^{M}\left(a_k-a_j\right)^{m_j}$, *and*

$$G\left(\sqrt{a}r\right)=\begin{cases}K_0\left(\sqrt{a}r\right), & d=2,\\ \pi\exp\left(-\sqrt{a}r\right)/2r, & d=3,\\ 2\sqrt{a}K_1\left(\sqrt{a}r\right)/r, & d=4.\end{cases}$$

The proof is analogue to the case of $d=2$, in [7], and therefore it is omitted. We give some examples.

Example 1 The covariance function for an AR(p) model in dimension $d=3$. There are three ways having the same results. One can use either formula (6)

$$\mathcal{C}_3\left(r\right)=\frac{\sigma^2}{2\pi^2}\frac{\sqrt{\pi}}{4}\left(\frac{r}{2c}\right)^{2p-d/2}K_{2p-3/2}\left(cr\right),$$

or formula (11) with $a=c^2$, $m_1=2p$, $M=1$,

$$\mathcal{C}_3\left(r\right)=\frac{\sigma^2}{2\pi^2}\frac{\pi}{2r}\frac{1}{(2p-1)!}\frac{\partial^{2p-1}}{\partial a^{2p-1}}\exp\left(-\sqrt{a}r\right),$$

or using the spatial form of $j_{3,0}\left(\rho\right)=\sin\rho/\rho$, and formula (21),

$$\mathcal{C}_3\left(r\right)=\frac{\sigma^2}{2\pi^2}\frac{\pi\exp\left(-\sqrt{a}r\right)}{2^{4p-2}\left(2p-1\right)!\left(\sqrt{a}\right)^{4p-3}}\sum_{k=0}^{2p-2}\frac{\left(4p-k-4\right)!\left(2r\sqrt{a}\right)^k}{k!\left(2p-k-2\right)!}.$$

Each of them gives the same result.
Let us consider an instant when $d=3, p=2$, and $M=1$, then we have the Laplacian model AR(2) by the equation

$$\left(\nabla^2-c^2\right)^2X\left(\underline{x}\right)=\partial W\left(\underline{x}\right),$$

with covariance function

$$\mathcal{C}_3\left(r\right)=\sigma^2\frac{1}{2^6\pi}\frac{1}{|c|^5}\left(1+r\left|c\right|+\frac{r^2\left|c\right|}{3}\right)\exp\left(-r\left|c\right|\right).$$

An other example is the following.

Example 2 The covariance function for an AR(2,1) when $d=3$. Consider the stochastic Laplacian equation

$$\left(\nabla^2-a_1\right)\left(\nabla^2-a_2\right)X\left(\underline{x}\right)=\partial W\left(\underline{x}\right)\left(\nabla^2-b_1\right)\partial W\left(\underline{x}\right),$$

the spectrum writes as

$$f\left(\rho\right)=\frac{\sigma^{2}}{\left(2\pi\right)^{3}}\frac{\left(\rho^{2}+b_{1}\right)^{2}}{\left(\rho^{2}+a_{1}\right)^{2}\left(\rho^{2}+a_{2}\right)^{2}},$$

Now $W_1=\left(b_1-a_1\right)\big/\left(a_2-a_1\right)$, $W_2=\left(b_1-a_2\right)\big/\left(a_1-a_2\right)$, and the covariance function is

$$\mathcal{C}_{3}\left(r\right)=\frac{\sigma^{2}}{8\pi}\left(\frac{b_{1}-a_{1}}{a_{2}-a_{1}}\frac{1}{\sqrt{a_{1}}}\exp\left(-\sqrt{a_{1}}r\right)+\frac{b_{1}-a_{2}}{a_{1}-a_{2}}\frac{1}{\sqrt{a_{2}}}\exp\left(-\sqrt{a_{2}}r\right)\right).$$

Conclusion Fitting a stochastic stationary model on the data makes necessary estimating the rate of decay of the covariance function first. We have shown that the rate of decay of covariance functions of ARMA fields is given clearly and depends on some parameters only, and therefore its estimation is possible.

3 Appendix

1. **Orthonormal Spherical Harmonics** with complex values are denoted by $Y_{\ell}^{m}\left(\widetilde{\underline{\lambda}}\right)$, $\widetilde{\underline{\lambda}}\in\mathbb{S}_{d-1}$, $\ell=0,1,2,\ldots$, $m=1,\ldots,h\left(\ell,d\right)$, where $h\left(\ell,d\right)=\left(2\ell+d-2\right)\left(\ell+d-3\right)!/\ell!\left(d-2\right)!$, $Y_{\ell}^{m}\left(\widetilde{\underline{\lambda}}\right)$ is of **degree** ℓ and **order** m (rank ℓ and projection m), see [18], [15, 19, 20].
2. **Spherical Bessel function** $j_{d,\ell}$, $d\geq 3$, of the first kind,

$$j_{d,\ell}\left(z\right)=\frac{\Gamma\left(d/2\right)}{\left(z/2\right)^{d_{2}}}J_{d_{2}+\ell}\left(z\right)=\frac{d_{2}\Gamma\left(d_{2}\right)}{\left(z/2\right)^{d_{2}}}J_{d_{2}+\ell}\left(z\right), \tag{12}$$

where $J_{\ell+1/2}$ is the Bessel function of the first kind, see DLMF, [21], 10.47.3, and $d_2=\left(d-2\right)/2$
3. **Jacobi–Anger expansions**:

$$\exp\left(i\underline{x}\cdot\underline{\lambda}\right)=\frac{\Gamma\left(\nu\right)}{\left(r\rho/2\right)^{\nu}}\sum_{k=0}^{\infty}\left(\nu+k\right)i^{k}J_{\nu+k}\left(r\rho\right)C_{k}^{\left(\nu\right)}\left(\widetilde{\underline{x}}\cdot\widetilde{\underline{\lambda}}\right),$$

$\nu\neq 0,-1,\ldots$, see DLMF, [21], 10.23.9, let $d_2=\left(d/2-2\right)/2$,

$$\exp\left(i\underline{x}\cdot\underline{\lambda}\right)=\frac{1}{d_{2}}\sum_{k=0}^{\infty}\left(\ell+d_{2}\right)i^{\ell}j_{d,\ell}\left(r\rho\right)C_{\ell}^{\left(d_{2}\right)}\left(\widetilde{\underline{x}}\cdot\widetilde{\underline{\lambda}}\right), \tag{13}$$

see DLMF, [21], 18.3.1.
Jacobi-Anger expansion, see [22], t. 2, 7.2.4 (27),

$$\exp\left(i\underline{x}\cdot\underline{\lambda}\right) = s_{d-1}\sum_{\ell=0}^{\infty}\sum_{m=1}^{h(\ell,d)} Y_\ell^m\left(\widetilde{\underline{x}}\right) Y_\ell^m\left(\widetilde{\underline{\lambda}}\right) i^\ell j_{d,\ell}\left(r\rho\right), \underline{\lambda}\in\mathbb{R}^d, \tag{14}$$

Note, if $d=2$, $h\left(\ell,d\right)=2$, and $Y_\ell^1\left(\widetilde{\underline{\lambda}}\right)=\cos\ell\vartheta/\sqrt{2\pi}$, and $Y_\ell^2\left(\widetilde{\underline{\lambda}}\right)=\sin\ell\vartheta/\sqrt{2\pi}$, see [23], Ch. IV., [19], Ch. 1.

4. **Addition Theorem**:

$$\begin{aligned}\sum_{m=1}^{h(\ell,d)} Y_\ell^m\left(\widetilde{\underline{x}}\right) Y_\ell^{m*}\left(\widetilde{\underline{y}}\right) &= \frac{h\left(\ell,d\right)}{s_{d-1}C_\ell^{d_2}\left(1\right)}C_\ell^{d_2}\left(\widetilde{\underline{x}}\cdot\widetilde{\underline{y}}\right)\\ &= \frac{\ell+d_2}{s_{d-1}d_2}C_\ell^{d_2}\left(\widetilde{\underline{x}}\cdot\widetilde{\underline{y}}\right),\end{aligned} \tag{15}$$

see [22], t. 2, 11.4 Theorem 4.

5. **Graf's and Gegenbauer's Addition Theorem**:

$$j_{d,0}\left(\rho r\right) = \frac{1}{d_2}\sum_{\ell=0}^{\infty}\left(\ell+d_2\right)C_\ell^{d_2}\left(\widetilde{\underline{x}}\cdot\widetilde{\underline{y}}\right)j_{d,\ell}\left(r_1\rho\right)j_{d,\ell}\left(\rho r_2\right), \tag{16}$$

see DLMF [21], 10.23.8.

6. The **Laplacian** ∇^2 in spherical coordinates is given by

$$\nabla^2 = \frac{1}{r^2}\triangle_{\mathbb{S}_{d-1}} + \frac{d-1}{r}\frac{\partial}{\partial r} + \frac{\partial^2}{\partial r^2},$$

where $\triangle_{\mathbb{S}_{d-1}}$ denotes the Laplace-Beltrami operator

$$\begin{aligned}\triangle_{\mathbb{S}_{d-1}} &= \frac{1}{\sin^{d-2}\vartheta_{d-1}}\frac{\partial}{\partial\vartheta_{d-1}}\left(\sin^{d-2}\vartheta_{d-1}\frac{\partial}{\partial\vartheta_{d-1}}\right)\\ &+ \sum_{j=1}^{d-2}\frac{1}{\sin^2\vartheta_{d-1}\cdots\sin^2\vartheta_{j+1}\sin^{j-1}\vartheta_j}\frac{\partial}{\partial\vartheta_j}\left(\sin^{j-1}\vartheta_j\frac{\partial}{\partial\vartheta_j}\right),\end{aligned} \tag{17}$$

on $\mathbb{S}_{d-1}$. The angles ϑ_j are defined by the spherical polar coordinates: $\widetilde{\underline{\lambda}}_d = \left[\sin\vartheta_{d-1}\widetilde{\underline{\lambda}}_{d-1}, \cos\vartheta_{d-1}\right]$, $\widetilde{\underline{\lambda}}_d \in \mathbb{S}_{d-1}$ $\widetilde{\underline{\lambda}}_{d-1}\in\mathbb{S}_{d-2}$, see [19].

7. The **Modified Bessel function** of the second kind K_ν is the solution of the Modified Bessel's Equation, see [21] 10.25.
   Limiting forms:

$$K_\nu\left(z\right)\sim\sqrt{\frac{\pi}{2z}}e^{-z}, \quad z\to\infty, \tag{18}$$

$$K_\nu\left(z\right)\sim\frac{1}{2}\Gamma\left(\nu\right)\left(\frac{1}{2}z\right)^{-\nu}, \quad z\to 0. \tag{19}$$

Special forms of Modified Bessel functions of the second kind:

$$K_{1/2}(z) = \sqrt{\frac{\pi}{2z}}e^{-z}, \tag{20}$$
$$K_{3/2}(z) = \sqrt{\frac{\pi}{2z}}e^{-z}\left(1+\frac{1}{z}\right),$$
$$K_{5/2}(z) = \sqrt{\frac{\pi}{2z}}e^{-z}\left(1+\frac{3}{z}+\frac{3}{z^2}\right),$$

see [17], 10.2.17.

8. Integral

$$\begin{aligned}
&\int_0^\infty j_{3,0}(\rho r)\frac{\rho^2 d\rho}{\left(\rho^2+a\right)^{2p}} \\
&= \int_0^\infty \frac{\sin(\rho r)\,\rho}{r\left(\rho^2+a\right)^{2p}}d\rho \\
&= \frac{\pi r\exp\left(-\sqrt{a}r\right)}{r2^{2(2p-1)}(2p-1)!\left(\sqrt{a}\right)^{2(2p-1)-1}}\sum_{k=0}^{2p-2}\frac{(2(2p-1)-k-2)!\left(2r\sqrt{a}\right)^k}{k!(2p-k-2)!} \\
&= \frac{\pi\exp\left(-\sqrt{a}r\right)}{2^{4p-2}(2p-1)!\left(\sqrt{a}\right)^{4p-3}}\sum_{k=0}^{2p-2}\frac{(4p-k-4)!\left(2r\sqrt{a}\right)^k}{k!(2p-k-2)!},
\end{aligned} \tag{21}$$

see [24], 3.737.2.

References

1. Bandyopadhyay, S., Jentsch, C., Subba Rao, S.: A spectral domain test for stationarity of spatio-temporal data. J. Time Ser. Anal. **38**(2), 326–351 (2017)
2. Frank, P., Steininger, T., Enßlin, T.A.: Field dynamics inference via spectral density estimation. Phys. Rev. E **96**(5), 052104 (2017)
3. Hama, K., Fujimoto, Y., Hayashi, Y.: Expected wind speed estimation considering spatio-temporal anisotropy for generating synthetic wind power profiles. Energy Procedia **155**, 309–319 (2018)
4. Hu, D.g., Shu, H.: Spatiotemporal interpolation of precipitation across Xinjiang, China using space-time CoKriging. J. Cent. South Univ. **26**(3), 684–694 (2019)
5. Whittle, P.: On stationary processes in the plane. Biometrika **41**(3/4), 434–449 (1954)
6. Vecchia, A.V.: A general class of models for stationary two-dimensional random processes. Biometrika **72**, 281–291 (1985)
7. Vecchia, A.V.: Estimation and model identification for continuous spatial processes. J. Roy. Stat. Soc. Series B (Methodological) **50**, 297–312 (1988)
8. Jones, R.H.: Fitting a stochastic partial differential equation to aquifer data. Stoch. Hydrol. Hydraulics **3**, 85–96 (1989). https://doi.org/10.1007/BF01544074

9. Jones, R.H., Vecchia, A.V.: Fitting continuous ARMA models to unequally spaced spatial data. J. Am. Stat. Assoc. **88**(423), 947–954 (1993)
10. Naff, R.L., Vecchia, A.V.: Stochastic analysis of three-dimensional flow in a bounded domain. Water Resour. Res. **22**(5), 695–704 (1986). https://doi.org/10.1029/wr022i005p00695
11. Subba Rao, T., Terdik, G.: On the frequency variogram and on frequency domain methods for the analysis of spatio-temporal data. J. Time Ser. Anal. **38**(2), 308–325 (2017). https://doi.org/10.1111/jtsa.12231
12. Subba Rao, T., Terdik, G.: A new covariance function and spatio-temporal prediction (kriging) for a stationary spatio-temporal random process. J. Time Ser. Anal. pp. n/a–n/a (2017). https://doi.org/10.1111/jtsa.12245
13. Terdik, G.: A covariance function for time dependent Laplacian fields in 3D. In: Valenzuela, O., Rojas, F., Pomares, H., Rojas, I. (eds.) ITISE 2019, International Conference on Time Series and Forecasting. Proceedings of Papers, 25-27 September 2019, vol. 1, p. 330 (2019)
14. Adler, R.J., Taylor, J.E.: Random Fields and Geometry. Springer Science & Business Media (2009)
15. Yadrenko, M.I.: Spectral Theory of Random Fields. Optimization Software Inc. Publications Division, New York (1983), translated from the Russian
16. Yaglom, A.M.: Correlation Theory of Stationary Related Random Functions, vol. I. Springer-Verlag, New York (1987)
17. Abramowitz, M., Stegun, I.A.: Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables. Dover Publications Inc., New York (1992), reprint of the 1972 edition
18. Müller, C.: Spherical harmonics. Lecture Notes in Mathematics, vol. 17. Springer-Verlag, Berlin, (1966)
19. Dai, F., Xu, Y.: Approximation Theory and Harmonic Analysis on Spheres and Balls. Springer (2013)
20. Varshalovich, D.A., Moskalev, A.N., Khersonskii, V.K.: Quantum Theory of Angular Momentum. World Scientific Press (1988)
21. Olver, F.W.J., Olde Daalhuis, A.B., Lozier, D.W., Schneider, B.I., Boisvert, R.F., Clark, C.W., Miller, B.R., Saunders, B.V. (eds.): NIST Digital Library of Mathematical Functions. http://dlmf.nist.gov/,Release1.0.17of2017-12-22, http://dlmf.nist.gov/
22. Erdélyi, A., Magnus, W., Oberhettinger, F., Tricomi, F.G.: Higher Transcendental Functions, vol. II. Robert E. Krieger Publishing Co. Inc., Melbourne, Fla. (1981), based on notes left by Harry Bateman, Reprint of the 1953 original
23. Stein, E.M., Weiss, G.: Introduction to Fourier analysis on Euclidean spaces. Princeton University Press, Princeton, N.J. (1971), princeton Mathematical Series, No. 32
24. Gradshteyn, I.S., Ryzhik, I.M.: Table of Integrals, Series, and Products. Academic Press Inc., San Diego, CA, 6th edn. (2000), translated from the Russian, Translation edited and with a preface by Alan Jeffrey and Daniel Zwillinger

The Correspondence Between Stochastic Linear Difference and Differential Equations

D. Stephen G. Pollock

Abstract An autoregressive moving-average model in discrete time is driven by a forcing function that is necessarily limited in frequency to the Nyquist value of π radians per sampling interval. The linear stochastic model that is commonly regarded as the counterpart in the continuous time of the autoregressive moving-average model is driven by a forcing function that consists of the increments of a Wiener process. This function is unbounded in frequency. The disparity in the frequency contents of the two forcing functions creates difficulties in defining a correspondence between the discrete-time and continuous-time models. These difficulties are alleviated when the continuous-time forcing function is limited in frequency by the Nyquist value. Then, there is an immediate one-to-one correspondence been the discrete-time autoregressive moving-average model and its continuous-time counterpart, of which the parameters can be readily inferred from those of the discrete-time model.

Keywords Stochastic differential equations · Frequency-limited stochastic processes · Oversampling

1 Introduction: The Discrete–Continuous Correspondence

Modern communications technology relies on the correspondence between continuous signals and the discrete sequences that come from sampling the signals rapidly at regular intervals. Familiar examples of the technology are the analog–digital conversions of digital radio, digital sound recordings and digital television; but the domain of this technology is much wider.

The basis of digital technology is the sampling theorem of Nyquist [6, 7] and of Shannon [9], which indicates that if a signal is sampled with sufficient rapidity, then it can be reconstituted with complete accuracy from the sampled sequence.

D. S. G. Pollock (✉)
University of Leicester, LE1 7RH Leicester, UK
e-mail: stephen_pollock@sigmapi.u-net.com
URL: https://www.le.ac.uk/users/dsgp1

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_3

The theorem is a commonplace amongst electrical engineers. It ought to be equally familiar to econometricians and statisticians and, in particular, to time-series analysts, but it has been widely ignored.

This discrete–continuous equivalence began to be widely recognised at the end of the nineteenth century with the advent of the cinema. The cinema creates moving pictures from a sequence of fixed images projected in rapid succession. In the early days of the cinema, the succession of images was insufficiently rapid to convey an impression of smooth motion. The pictures tended to flicker; and, in popular parlance, we still refer to visiting the cinema as ‘going to the flicks’.

There is a revealing picture by Marcel Duchamp, exhibited in the Paris Salon des Independents of 1912, which is titled *A Nude Descending a Staircase.* It exposes the paradox of the discrete–continuous correspondence; and it makes an allusion to the jerky motion of the early cinema.

Occasionally, the true nature of motion pictures is revealed by an odd quirk that occurs when the rate of sampling is insufficient to convey a convincing impression of a rapid motion. Those of a certain age will have seen a depiction of a stagecoach fleeing its pursuers. They will have noticed the blurred impression of the wagon wheels. At times, these appear to be rotating slowly in the direction of travel. At other times, they seem to be stationary, and they may even, on occasion, appear to be moving backwards.

These are instances of the so-called problem of aliasing, whereby a motion that is too rapid to be captured by the sampling process is proxied by a much slower motion.

The Shannon–Nyquist sampling theorem is an adjunct of a Fourier analysis, which depicts a temporal trajectory as a weighted combination of trigonometric functions. The theorem indicates that if the sampled sequence is fully to capture a continuous motion, then it is necessary that at least two observations should be made in the time that it takes for the trigonometric element of highest frequency to complete a single cycle. This rate of sampling, which corresponds to a signal frequency of π radians per sampling interval, is the so-called Nyquist relative frequency.

If the frequencies within the signal exceed the Nyquist value of π, then there will be an irremediable loss of information and it will not be possible fully to reconstitute the signal from the sampled data. Conversely, if the maximum frequency within the signal is less than the Nyquist value, then the sampling is over-rapid and other problems can arise; but these problems ought, in principle, to be remediable.

2 ARMA Estimation and the Effects of Over-Rapid Sampling

A problem can arise in the estimation of an ARMA model when the rate of sampling exceeds the maximum frequency within the signal. The problem can be illustrated with the deseasonalised quarterly data on the U.S. gross domestic product (GDP) from which a trend has been extracted with the filter of Leser [4] and of Hodrick

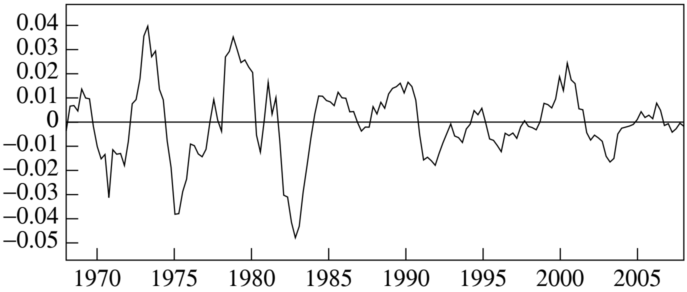


Fig. 1 The deviations of the logarithmic quarterly index of real US GDP from an interpolated trend. The observations are from 1968 to 2007. The trend is determined by a Hodrick–Prescott (Leser) filter with a smoothing parameter of 1600

and Prescott [2, 3]—see Fig. 1. The problem, which has been highlighted by Pollock [8], is revealed by examining the periodogram of the data, which is a product of its Fourier transform.

The Fourier analysis expresses the detrended data sequence $y(t) = \{y_t; t = 0, 1, \ldots, T-1\}$ as

$$y(t) = \sum_{j=0}^{[T/2]} \left\{\alpha_j \cos(\omega_j t) + \beta_j \sin(\omega_j t)\right\} = \sum_{j=0}^{T-1} \xi_j e^{\mathrm{i}\omega_j t}, \tag{1}$$

where $\omega_j = 2\pi j/T$; $j = 0, \ldots, [T/2]$ are the Fourier frequencies, which are placed at regular intervals running from zero up to the Nyquist frequency π, or just short of it by a half interval. Here, $[T/2]$ denotes the integer quotient of the division of T by 2.

The second expression, which employs complex exponential functions, arises from Euler's equations, whereby

$$\cos(\omega_j t) = \frac{e^{\mathrm{i}\omega_j t} + e^{-\mathrm{i}\omega_j t}}{2} \quad \text{and} \quad \sin(\omega_j t) = \frac{-\mathrm{i}}{2}(e^{\mathrm{i}\omega_j t} - e^{-\mathrm{i}\omega_j t}). \tag{2}$$

Conversely, there are

$$e^{\mathrm{i}\omega_j t} = \cos(\omega_j t) + \mathrm{i}\sin(\omega_j t) \quad \text{and} \quad e^{-\mathrm{i}\omega_j t} = \cos(\omega_j t) - \mathrm{i}\sin(\omega_j t), \tag{3}$$

and it follows that $\exp\{-\mathrm{i}\omega_j t\} = \exp\{\mathrm{i}\omega_{T-j} t\}$. Also, $\xi_j = (\alpha_j - \mathrm{i}\beta_j)/2$ and $\xi_{T-j} = (\alpha_j + \mathrm{i}\beta_j)/2$ for $j = 0, 1, \ldots, [T/2]$. These results enable the two expressions of (1) to be reconciled.

The coefficients α_j, β_j are obtained by regressing the data on the ordinates of the trigonometric functions $\cos(\omega_j t)$, $\sin(\omega_j t)$, where $t = 0, 1, \ldots, T-1$. It should be observed that if the maximum frequency in the signal is less than π, then some of these coefficients will be zero valued.

The periodogram is the plot of the squared amplitudes $\rho_j^2 = \alpha_j^2 + \beta_j^2$, and it conveys a frequency-specific analysis of variance. That is to say

$$V(y) = \frac{1}{T}\sum_t (y_t - \bar{y})^2 = \frac{1}{2}\sum_j \{\alpha_j^2 + \beta_j^2\} = \frac{1}{2}\sum_j \rho_j^2. \tag{4}$$

The periodogram of the detrended logarithmic quarterly index of real US GDP is depicted in Figs. 2 and 3.

An attempt can be made to capture the business cycle dynamics by fitting an AR(2) model to the detrended data. The expectation is that the poles of the model, i.e. its autoregressive roots, will be a conjugate complex pair. The modulus of the roots should represent the damping characteristics of the business cycle and their argument should represent an angular velocity, which would indicate the average duration of the business cycle. The parametric spectrum of the fitted ARMA model should mimic the shape of the periodogram, with its peak in roughly the same position as that of the periodogram.

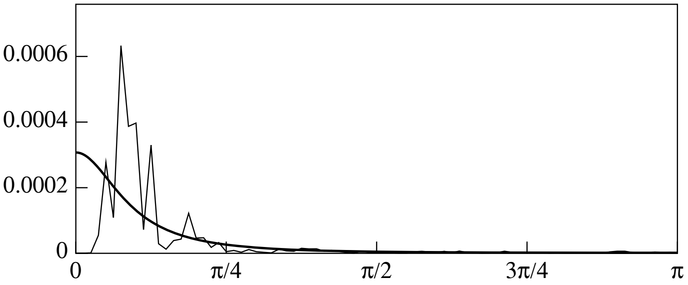


Fig. 2 The periodogram of the data points of Fig. 1 overlaid by the parametric spectral density function of an estimated regular AR(2) model

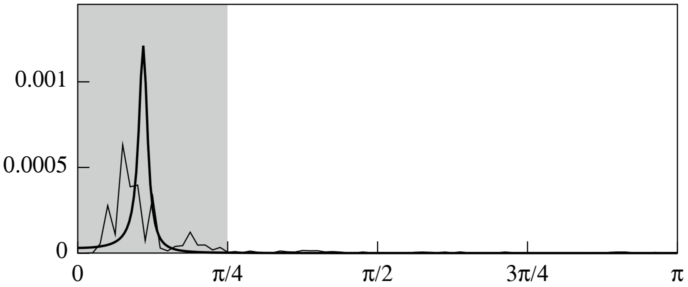


Fig. 3 The periodogram of the data points of Fig. 1 overlaid by the spectral density function of an AR(2) model estimated from de-noised frequency-limited data

When the parametric spectrum of the estimated AR(2) model is superimposed on the periodogram in Fig. 2, it becomes clear that, in place of the expected complex roots, there are two real-valued roots.

In diagnosing the problem, it is recognised that there are minor elements of noise affecting the data throughout the frequency interval running for the cut-off point of the spectral signature of the business cycle at $\omega_c = \pi/4$ up to the Nyquist frequency of π. This noise is making a significant contribution to the variance of the data without greatly affecting the autocovariance at positive lags. As a result, the initial values, which determine the estimates of the autoregressive parameters, show an exaggerated rate of decline, or damping, which gives rise to the real-valued poles.

The appropriate recourse would seem to be to remove the noise from the data by suppressing the associated periodogram ordinates in the interval $(\pi/4, \pi]$. When this is done, the estimation does deliver a pair of conjugate complex poles. However, in this case, the parametric spectrum in Fig. 3 misrepresents the periodogram in another way.

The poles are too close to the unit circle, i.e. their modulus is close to unity. The effect is to exaggerate the prominence of the spectral spike and to underestimate the rate of damping. Also, it can be seen that the peak is displaced to the right, implying that the argument is an overestimate, which exaggerates the frequency of the cycles. In consequence of the excessive rate of sampling, the initial autocovariances, which are too close to the origin, where the variance is to be found, are declining too slowly. Thus, the rate of damping is underestimated.

The spectral support of an ARMA process is the full Nyquist frequency interval $[0, \pi]$. Therefore, it is appropriate to dilate the spectral signature of the business cycle so that it fills the entire interval. This entails associating to each of the periodogram ordinates a higher frequency value. The frequencies are measured relative to the sampling interval. Therefore, they can be increased by increasing the length of the sampling interval.

In order to resample the data, it is usually necessary to reconstitute the underlying continuous trajectory. This can be achieved by a method of Fourier synthesis based on a version of equation (1) in which the coefficients associated with noisy elements, with frequencies in excess of the upper limit of the business cycle, have been set to zero.

The discrete temporal index, which is $t = 0, 1, \ldots, T - 1$, can be replaced within Eq. (1) by a continuous variable $t \in [0, T)$ to create the continuous trajectory. This can be resampled at intervals of π/ω_c units of time. In the present example, wherein $\omega_c = \pi/4$, the appropriate sampling interval is 4 units, which implies that only one in 4 of the points from the de-noised data is required; and there is no need to reconstitute the continuous trajectory in order to resample it. The effect of estimating an ARMA model with the de-noised and resampled data is shown in Fig. 4.

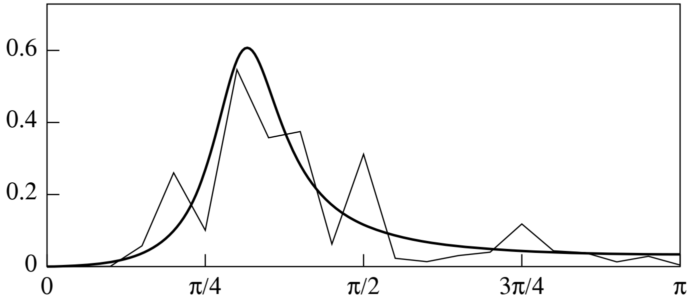


Fig. 4 The periodogram of the de-noised data that have been filtered and subsampled at the rate of 1 observation in 4, overlaid by the parametric spectrum of an estimated ARMA(2, 1) model

3 Sinc Function Interpolation and Fourier Interpolation

The procedure for resampling the data has implicitly defined a continuous ARMA process powered by a continuous frequency-limited white-noise process. The stochastic differential equations that are commonly supposed to be the continuous-time analogs of the ARMA models are driven by the increments of a Wiener process. The latter is an accumulation of a continuous steam of infinitesimal impulses. Such impulses are unbounded in frequency. The Wiener process has the characteristic that, whatever the rate of sampling, the accumulations that occur within the sampling intervals will constitute a discrete-time white-noise process.

In proposing a frequency-limited white noise, we resort to the sampling theorem. The theorem is commonly defined for square-integrable functions of time, defined of the real line $\mathcal{R} = (-\infty, \infty)$, and limited in frequency to the interval $[-\pi, \pi]$.

The Fourier integral transform has the following expression in the time domain and the frequency domain:

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \xi(\omega)e^{\mathrm{i}\omega t}d\omega \longleftrightarrow \xi(\omega) = \int_{-\infty}^{\infty} x(t)e^{-\mathrm{i}\omega t}dt. \tag{5}$$

However, with the frequency limitation, this becomes

$$x(t) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \xi_S(\omega)e^{\mathrm{i}\omega t}d\omega \longleftrightarrow \xi_S(\omega) = \sum_{k=-\infty}^{\infty} x_k e^{-\mathrm{i}k\omega}, \tag{6}$$

where $\{x_k; k = 0, \pm 1, \pm 2, \ldots\}$ is sampled at unit intervals from $x(t)$. Putting the RHS of (6) into the LHS and interchanging the order of integration and summation gives

$$x(t) = \frac{1}{2\pi}\sum_{k=-\infty}^{\infty} x_k \left\{\int_{-\pi}^{\pi} e^{\mathrm{i}\omega(t-k)}\right\} d\omega = \sum_{k=-\infty}^{\infty} x_k \varphi(t-k), \tag{7}$$

where

$$\varphi(t-k)=\frac{\sin\{\pi(t-k)\}}{\pi(t-k)} \tag{8}$$

is the so-call sinc function. The RHS of Eq. (7) defines a sinc function interpolation.

The sinc function centred on $k=0$, which is illustrated in Fig. 5, is formed by applying a bi-directional hyperbolic taper to an ordinary sine function. The succession of displaced sinc functions provides an orthonormal basis for the set of continuous functions that are limited in frequency to the Nyquist interval $[-\pi,\pi]$.

Equation (7) implies a simple prescription for converting a data sequence into a continuous function that is limited in frequency to the Nyquist interval. Sinc function kernels are attached to each of the discrete-time ordinates, and the sum is taken of the scaled kernels. The values at the integer points are those of their associated kernels; and these values are not affected by the kernels at the other integer points. This feature is illustrated by Fig. 6.

A continuous-time white-noise forcing function can be formed by replacing the impulses of a discrete-time white-noise process by sinc functions scaled by the values of those impulses. If $\varepsilon_t=\varepsilon(t)$ and $\varepsilon_s=\varepsilon(s)$ are elements sampled at arbitrary

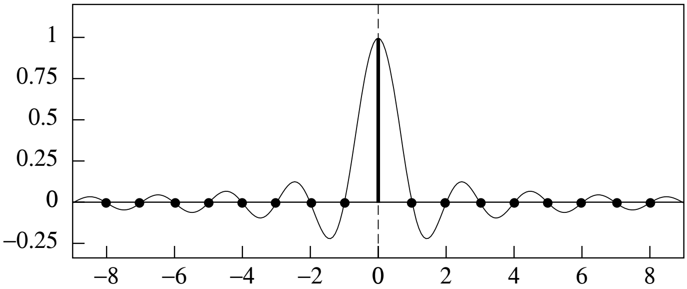


Fig. 5 The sinc function wave-packet $\varphi(t)=\sin(\pi t)/\pi t$ comprising frequencies in the interval $[0,\pi]$

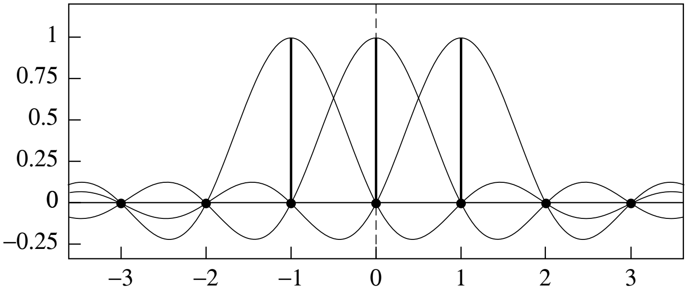


Fig. 6 The wave packets $\varphi(t-k)$, which are bounded in frequency by π, suffer no mutual interference when $k\in\{0,\pm1,\pm2,\pm3,\dots\}$

points from the continuous frequency-limited white-noise forcing function, then their covariance is the sinc function

$$C(\varepsilon_t, \varepsilon_s) = \sigma_\varepsilon^2 \varphi(t - s) = \sigma_\varepsilon^2 \varphi(\tau), \quad \tau = t - s, \tag{9}$$

where σ_ε^2 is the variance parameter. This result follows from recognising that $\varepsilon_s = \varepsilon(\tau)\varepsilon_t + \eta$, where η is uncorrelated with ε_t, and from the fact that $\varphi^2(\tau) = \varphi(\tau)$.

The practicality of a sinc function synthesis is prejudiced by the fact that the support of the kernel functions is the entire real line $\mathcal{R} = (-\infty, \infty)$. A practical synthesis replaces the sinc function by the so-called Dirichlet kernel, which is a periodic or circular function formed by wrapping the sinc function around a circle of a circumference T, equal to the number of data points, and by adding the overlying ordinates. In this context, the data points to which the kernels are to be fixed are also to be regarded as a periodic or circular sequence.

Consider the discrete Fourier transform expressed as follows:

$$x_t = \sum_{j=0}^{T-1} \xi_j e^{\mathrm{i}\omega_j t} \longleftrightarrow \xi_j = \frac{1}{T} \sum_{t=0}^{T-1} x_t e^{-\mathrm{i}\omega_j t} \quad \text{with} \quad \omega_j = \frac{2\pi j}{T}. \tag{10}$$

By putting the RHS of the latter into the LHS and commuting the two summations and allowing $t \in [0, T)$ to vary continuously, we get

$$x(t) = \frac{1}{T} \sum_{k=0}^{T-1} x_k \left\{ \sum_{j=0}^{T-1} e^{\mathrm{i}\omega_j (t-k)} \right\} = \sum_{k=0}^{T-1} x_k \varphi^\circ(t - k). \tag{11}$$

where

$$\varphi^\circ(t) = \frac{1}{T} \sum_{j=-n}^{n} e^{\mathrm{i}\omega_j t} = \frac{\sin(\{T/2\}\omega_1 t)}{T \sin(\omega_1 t/2)} \quad \text{with} \quad n = \frac{T-1}{2}. \tag{12}$$

is the periodic Dirichlet Kernel. An example is provided by Fig. 7.

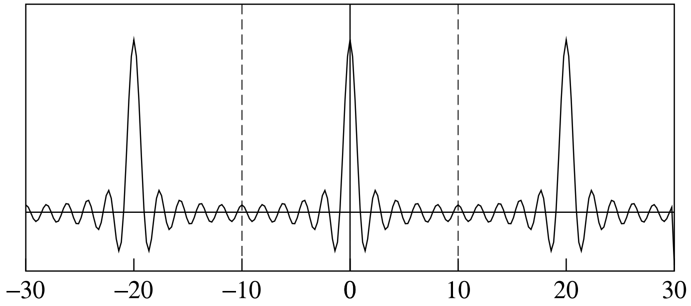


Fig. 7 The Dirichlet function $\sin(\pi t)/\sin(2\pi t/M)$ obtained from the inverse Fourier transform of a frequency-domain rectangle sampled at $M = 21$ points

Equation (11) implies that a sinc function interpolation of a finite data sequence that employs a sequence of Dirichelet kernels is equivalent to an interpolation based on a Fourier synthesis.

4 Discrete-Time and Continuous-Time Models

Whereas it is straightforward to derive a continuous version of an ARMA process (i.e. a CARMA process) by sinc function interpolation, we also require to represent it via a linear stochastic differential equation (an LSDE). The correspondence between difference equations and differential equations can be established by focusing, initially, on the first-order equations.

(In this paper, the acronym CARMA is reserved for the continuous-time linear stochastic differential equations that have the same frequency limitation as their corresponding discrete-time ARMA models. This is in spite of the common use of the acronym to denote continuous processes of unlimited frequency that are derived from ARMA models.)

The first-order autoregressive difference equation takes the form of

$$y(t) = \mu y(t-1) + \varepsilon(t) \quad \text{or} \quad y(t) = \frac{\varepsilon(t)}{1-\mu L} = \sum_{\tau=0}^{\infty} \mu^{\tau} \varepsilon(t-\tau). \tag{13}$$

Here, $y(t) = \{y_t; t = 0 \pm 1, \pm 2 \ldots\}$ denotes a sequence, and L is the lag operator such that $Ly(t) = y(t-1)$. (However, $y(t)$ will be used, equally, to denote a function of a continuous-time index.) Also, the forcing function $\varepsilon(t)$ is a white-noise sequence of independent and identically distributed random elements.

The corresponding first-order stochastic differential equation is denoted by

$$\begin{aligned} \frac{dy}{dt} &= \kappa y(t) + \zeta(t) \quad \text{or} \\ y(t) = \frac{\zeta(t)}{D-\kappa} &= \int_0^{\infty} e^{\kappa\tau}\zeta(t-\tau)d\tau = \int_{-\infty}^{t} e^{\kappa(t-\tau)}\zeta(\tau)d\tau, \end{aligned} \tag{14}$$

where D is the derivative operator such that $Dx(t) = dx/dt$. Here, the forcing function $\zeta(t)$ is a continuous white-noise process. It is either the derivative of a Wiener process or else it is a frequency-limited process formed by associating sinc functions to the elements of a discrete white-noise sequence. It can be seen that μ^{τ} and $e^{\kappa\tau}$ play the same role in the two equations, which is to diminish or to 'dampen' the effect of the impulses of the forcing functions as time elapses.

To convert the differential equation of (14) to the difference equation of (13), the integral on the interval $(-\infty, t]$ may be separated into two parts, which are the integrals over $(-\infty, t-1]$ and $(t-1, t]$:

$$\begin{aligned} y(t) &= e^{\kappa}\int_{-\infty}^{t-1} e^{\kappa(t-1-\tau)}\zeta(\tau)d\tau + \int_{t-1}^{t} e^{\kappa(t-\tau)}\zeta(\tau)d\tau \\ &= \mu y(t-1) + \varepsilon(t). \end{aligned} \tag{15}$$

We are interested, of course, in equations of higher orders. The ARMA(p, q) equation is denoted by

$$\begin{aligned} (1 + \alpha_1 L + \cdots + \alpha_p L^p)y(t) &= (\beta_0 + \beta_1 L + \cdots + \beta_p L^q)\varepsilon(t) \\ \text{or} \qquad \alpha(L)y(t) &= \beta(L)\varepsilon(t). \end{aligned} \tag{16}$$

Given that $p > q$ and that there are no repeated roots of $\alpha(z) = 0$, the rational function $\beta(z)/\alpha(z)$ is amenable to a partial-fraction decomposition, which gives rise to the equation

$$\begin{aligned} y(t) = \frac{\beta(L)}{\alpha(L)}\varepsilon(t) &= \left\{\frac{d_1}{1-\mu_1 L} + \frac{d_2}{1-\mu_2 L} + \cdots + \frac{d_p}{1-\mu_p L}\right\}\varepsilon(t) \\ &= \sum_{\tau=0}^{\infty}\left\{d_1\mu_1^{\tau} + d_2\mu_2^{\tau} + \cdots + d_p\mu_p^{\tau}\right\}\varepsilon(t-\tau). \end{aligned} \tag{17}$$

The linear stochastic differential equation of orders p and $q < p$, denoted by LSDE(p, q), is specified by the equation

$$\begin{aligned} (\phi_0 D^p + \phi_1 D^{p-1} + \cdots + \phi_p)y(t) &= (\theta_0 D^q + \theta_1 D^{q-1} + \cdots + \theta_q)\zeta(t) \\ \text{or} \qquad \phi(D)y(t) &= \theta(D)\zeta(t). \end{aligned} \tag{18}$$

On the assumption that there are no repeated roots, it has the following partial-fraction decomposition:

$$\begin{aligned} y(t) = \frac{\theta(D)}{\phi(D)}\zeta(t) &= \left\{\frac{c_1}{D-\kappa_1} + \frac{c_2}{D-\kappa_2} + \cdots + \frac{c_p}{D-\kappa_p}\right\}\zeta(t) \\ &= \int_0^{\infty}\left\{c_1 e^{\kappa_1\tau} + c_2 e^{\kappa_2\tau} + \cdots + c_p e^{\kappa_p\tau}\right\}\zeta(t-\tau)d\tau. \end{aligned} \tag{19}$$

5 ARMA Model and Its Continuous-Time CARMA Counterpart

A correspondence can be established between the discrete and continuous systems by invoking the principle of impulse invariance. This indicates that a sequence sampled at unit intervals from the impulse response function of the continuous system should be equal to the impulse response of the discrete-time system. This is possible only if the continuous system has the same frequency limitation as the discrete system, which is the present assumption.

Thus, at the integer values of τ, the functions

$$\psi(\tau) = c_1 e^{\kappa_1 \tau} + c_2 e^{\kappa_2 \tau} + \cdots + c_p e^{\kappa_p \tau} \tag{20}$$

and

$$\phi(\tau) = d_1 \mu_1^{\tau} + d_2 \mu_2^{\tau} + \cdots + d_p \mu_2^{\tau} \tag{21}$$

should be equal. The equality can be achieved by setting $e^{\kappa_j} = \mu_j$ and $c_j = d_j$, for all j. The discrete-time ARMA model is driven by a white-noise process that is limited in frequency by the Nyquist value of π radians per sample interval. Its direct continuous-time counterpart is a CARMA model, driven by a continuous frequency-limited white-noise process.

It is appropriate to adopt a CARMA model when there is clear evidence that the spectral density of the process is limited in frequency by the Nyquist value of π radians per sample interval, at which point the function should be zero valued. The evidence will be provided by the periodogram of the data. In cases where the limiting frequency of the process is less than the π, the resampling procedures outlined in Sect. 2 should be pursued before estimating the ARMA model.

Example 1 To illustrate the mapping from the discrete-time ARMA model to a continuous frequency-limited CARMA model, an ARMA(2, 1) model is chosen with conjugate complex poles $\alpha \pm \mathrm{i}\beta = \rho \exp\{\pm \mathrm{i}\theta\}$, where $\rho = \sqrt{\alpha^2 + \beta^2} = 0.9$ and $\theta = \tan^{-1}(\beta/\alpha) = \pi/4 = 45°$. The moving-average component has a zero of 0.5. The ARMA process generates prominent cycles of an average duration of roughly 8 periods.

The parameters of the resulting continuous-time CARMA model are displayed below, beside those of the ARMA model:

ARMA	CARMA
$\alpha_0 = 1.0$	$\phi_0 = 1.0$
$\alpha_1 = -1.272$	$\phi_1 = 0.2107$
$\alpha_2 = 0.8100$	$\phi_2 = 0.6280$
$\beta_0 = 1.0$	$\theta_0 = 1.0$
$\beta_1 = -0.5$	$\theta_1 = 0.2737$

The spectral density function of the ARMA process is illustrated in Fig. 8. Here, it will be observed that the function is virtually zero at the limiting Nyquist frequency of π. Therefore, it is reasonable to propose that the corresponding continuous-time model should be driven by a white-noise forcing function that is bounded by the Nyquist frequency.

The spectral density function of the CARMA process is the integral Fourier transform of the continuous autocovariance function, whereas the spectral density function of the ARMA process is the discrete Fourier transform of the autocovariance

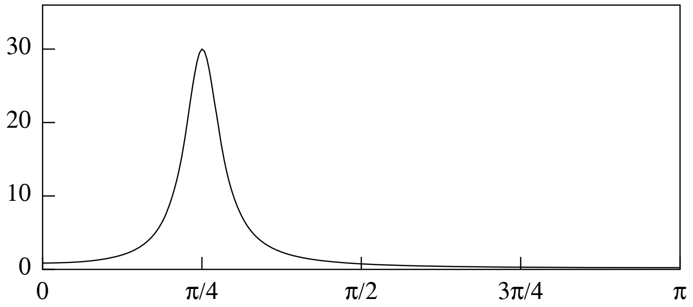


Fig. 8 The spectrum of the ARMA(2, 1) process $(1.0 - 1.273L + 0.81L^2)y(t) = (1 - 0.5L)e(t)$

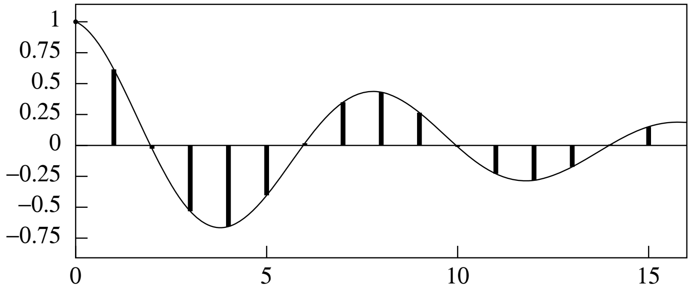


Fig. 9 The discrete autocovariance sequence of the ARMA(2, 1) process and the continuous autocovariance function of the corresponding CARMA(2, 1) process

sequence. The frequency limitation of the CARMA process means that there is no aliasing in the sampling process. Therefore, the two spectra are identical.

In Fig. 9, the discrete autocovariance function of the ARMA process is superimposed on the continuous autocovariance function of the CARMA process. The former has been generated by a recursive procedure. The latter has been generated by an analytic equation, to be presented below as Eq. (22), wherein the index τ of the lags varies continuously.

A principle of autocovariance equivalence is also satisfied, whereby the values sampled at the integer points from the continuous-time autocovariance function are equal to those of the discrete-time function.

6 Stochastic Differential Equations Driven by Wiener Processes

The white-noise forcing function of a conventional linear stochastic differential equation (LSDE) is the derivative of a Wiener process. The latter process consists of a continuous steam of infinitesimal impulses. Since a pure impulse is unbounded in frequency, so too is the forcing function.

The concept of a pure impulse is problematic from a physical point of view, since it implies a discrete and instantaneous change in momentum. The problem of unbounded frequencies can be mitigated, if not completely overcome, in the context of an LSDE, since its transfer function may impose a sufficient attenuation on the higher frequencies for the effect to be a virtual frequency limitation.

Whenever the spectral density function of an ARMA model has a significant value at the Nyquist frequency of π, there can be a reasonable supposition that the underlying continuous process has a frequency range that extends beyond the Nyquist limit. Therefore, it may be appropriate to adopt an LSDE with an unbounded forcing function as the continuous-time counterpart of the ARMA model.

In translating from an ARMA model to such an LSDE, it is no longer appropriate to invoke the principle of impulse invariance. Instead, the principle of autocovariance equivalence that was enunciated by Bartlett [1] must be adopted. The principle asserts that the parameters of the LSDE should be chosen so that its autocovariance function matches the autocovariance function of the ARMA model at the integer lags.

The autocovariance function of an ARMA model can be derived from its impulse response function, represented by Eq. (21). It takes the form of

$$\begin{aligned}\gamma^d(\tau) &= \sigma_\varepsilon^2 \sum_{j=0}^{\infty}\left(\sum_{k=1}^{p} d_k\mu_k^j\right)\left(\sum_{k=1}^{p} d_k\mu_k^{j+\tau}\right)\\ &= \sigma_\varepsilon^2 \sum_{i=1}^{p}\left\{\sum_{j=1}^{p}\frac{d_i d_j}{1-\mu_i\mu_j}\right\}\mu_i^\tau. \end{aligned}\tag{22}$$

The autocovariance function $\gamma^c(\tau)$ of the continuous-time LSDE process is also found via its impulse response function. It is assumed that the autocovariance of the white-noise forcing function at lag τ is

$$E\{\zeta(t)\zeta(t-\tau)\} = \delta(\tau)\sigma_\zeta^2,\tag{23}$$

where $\delta(\tau)$ is Dirac's delta function. Then,

$$\begin{aligned}\gamma^c(\tau) &= E\{y(t)y(t-\tau)\}\\ &= E\left\{\int_0^\infty \psi(u)\zeta(t-u)du\int_0^\infty \psi(v)\zeta(t-\tau-v)dv\right\}\\ &= \sigma_\zeta^2\int_0^\infty \psi(v)\psi(v+\tau)dv. \end{aligned}\tag{24}$$

Substituting the expression of (20) for the continuous-time impulse response function $\psi(t)$ into Eq. (24) gives

$$\gamma^c(\tau) = \sigma_\zeta^2 \int_0^\infty \psi(t)\psi(t+\tau)dt = \sigma_\zeta^2 \sum_i \sum_j \left\{ c_i c_j \int_0^\infty e^{(\kappa_i+\kappa_j)t+\kappa_i\tau} dt \right\}$$
$$= \sigma_\zeta^2 \sum_i \left\{ \sum_j c_i c_j \frac{-e^{\kappa_i\tau}}{\kappa_i + \kappa_j} \right\}. \tag{25}$$

This expression, which is liable to contain complex-valued terms, may be rendered in real terms by coupling the various conjugate complex terms.

In translating from the ARMA model to the LSDE, there continues to be a one-to-one correspondence between the poles of the two systems. If a complex pole of the ARMA model takes the form of

$$\mu = \alpha + \mathrm{i}\beta = \rho\{\cos(\omega) + \mathrm{i}\sin(\omega)\} = \rho e^{\mathrm{i}\omega}, \tag{26}$$

with

$$\rho = \sqrt{\alpha^2+\beta^2} \quad \text{and} \quad \omega = \tan^{-1}\left(\frac{\beta}{\alpha}\right), \tag{27}$$

then the corresponding pole of the LSDE and of the CARMA differential equation is

$$\kappa = \ln(\mu) = \ln(\rho) + \mathrm{i}\omega = \delta + \mathrm{i}\omega, \tag{28}$$

with $\delta \in (-\infty, 0)$, which puts it in the left half of the s-plane, as it is necessary for the stability of the system.

The principle of autocovariance equivalence can be expressed via the equation

$$\gamma_\tau^c\{\kappa(\mu), c\} = \gamma_\tau^d(\mu, d) \quad \text{for} \quad \tau \in \{0, \pm 1, \pm 2, \ldots\}. \tag{29}$$

Then, the parameters of the LSDE can be derived once a value of $c = [c_1, c_2, \ldots, c_p]$ of the vector of the numerator parameters of (19) has been found that satisfies this equation. The value of c can be found by using an optimisation procedure to find the zeros of the function

$$z(c) = \sum_{\tau=0}^{p} \{\gamma_\tau^c(c) - \gamma_\tau^d\}^2. \tag{30}$$

As Söderström [10, 11], and others have noted, there are ARMA models for which there are no corresponding LSDE's. The present procedure for translating from an ARMA model to an LSDE reveals such cases by its failure to find a zero-valued minimum of the criterion function. However, it can be relied upon to find the LSDE most closely related to the ARMA model.

The principle of autocovariance equivalence also indicates a way in which an ARMA model can be found to correspond to a given LSDE. The ARMA model is commonly described as the exact or equivalent discrete linear model (EDLM).

The autocovariance generating function of an ARMA model is

$$\gamma^d(z) = \sigma_\varepsilon^2 \frac{\beta(z)\beta(z^{-1})}{\alpha(z)\alpha(z^{-1})}, \tag{31}$$

whereas the z-transform of the elements γ_τ^c; $\tau \in \{0, \pm 1, \pm 2, \ldots\}$ sampled from the autocovariance function of the LSDE may be denoted by $\gamma^c(z)$. Putting the latter in place of $\gamma^d(z)$ and rearranging the equation gives

$$\sigma_\varepsilon^2 \beta(z)\beta(z^{-1}) = \alpha(z)\gamma^c(z)\alpha(z^{-1}). \tag{32}$$

Since the discrete-time autoregressive parameters within $\alpha(z)$ can be inferred from those of the LSDE, only the moving-average parameters within $\beta(z)$ and the variance σ_ε^2 needs to be derived from Eq. (32). They can be obtained via a Cramér–Wold factorisation of the LHS.

Example 2 The mapping from the discrete-time ARMA model to a continuous-time LSDE model can be illustrated, in the first instance, with the ARMA(2, 1) model of Example 1.

The parameters of the corresponding LSDE(2, 1) model are obtained by using the procedure of Nelder and Mead [5] to find the minimum of the criterion function of (30), where it is assumed that the variance of the forcing function is $\sigma_\zeta^2 = 1$. The minimands a, b of the criterion function are from the numerator coefficients $c, c^* = a \pm \mathrm{i}b$ of the partial-fraction decomposition of the LSDE(2, 1) transfer function.

There are four points that correspond to zero-valued minima, where the ordinates of the discrete and continuous autocovariance functions coincide at the integer lags. These points, together with the corresponding moving-average parameters, are as follows:

	a	b	θ_0	θ_1
(i)	−0.4544	0.2956	−0.9088	0.5601
(ii)	0.4544	0.4175	0.9088	0.5601
(iii)	−0.4544	−0.4174	−0.9088	−0.5601
(iv)	0.4544	−0.2956	0.9088	−0.5601

Here, the parameter values of (i) and (iv) are equivalent, as are those of (ii) and (iii). Their difference is a change of sign, which can be eliminated by normalising θ_0 at unity and by adjusting variance of the forcing function accordingly.

The miniphase condition, which corresponds to the invertibility condition of a discrete-time model, requires the zeros to be in the left half of the s-plane. Therefore, (ii) and (iii) on the *NE–SW* axes on the graphs of Fig. 11 are the chosen pair.

These estimates of the LSDE(2, 1) are juxtaposed below with those of the CARMA(2, 1) model derived from the same ARMA model:

CARMA	LSDE
$\phi_0 = 1.0$	$\phi_0 = 1.0$
$\phi_1 = 0.2107$	$\phi_1 = 0.2107$
$\phi_2 = 0.6280$	$\phi_2 = 0.6280$
$\theta_0 = 1.0$	$\theta_0 = 0.9088$
$\theta_1 = 0.2737$	$\theta_1 = 0.5601$

The autoregressive parameters of the CARMA model and of the LSDE model are, of course, identical. However, there is a surprising disparity between the two sets of moving-average parameters. Nevertheless, when they are superimposed on the same diagram—which is Fig. 10—the spectra of the two models are seen virtually to coincide. Moreover, the parameters of the ARMA model can be recovered exactly from those of the LSDE by an inverse transformation.

The explanation for this outcome is to be found in the remarkable flatness of the criterion function in the vicinity of the minimising points, which are marked on both sides of Fig. 11 by black dots. The flatness implies that a wide spectrum of the parameter values of the LSDE will give rise to almost identical autocovariance functions and spectra.

The left side of Fig. 11 shows some equally spaced contours of the z-surface of the criterion function, which are rising from an annulus that contains the minima. The minima resemble small indentations in the broad brim of a hat.

The right side of Fig. 11, which is intended to provide more evidence of the nature of the criterion function in the vicinity of the minima, shows the contours of the function $q = 1/(z + d)$, where d is a small positive number that prevents a division by zero. We set $d = (X - RM)/(R - 1)$, where $M = \min(z)$, $X = \max(z)$ and where $R = \max(q)/\min(q) = 60$. The extended lenticular contours surrounding

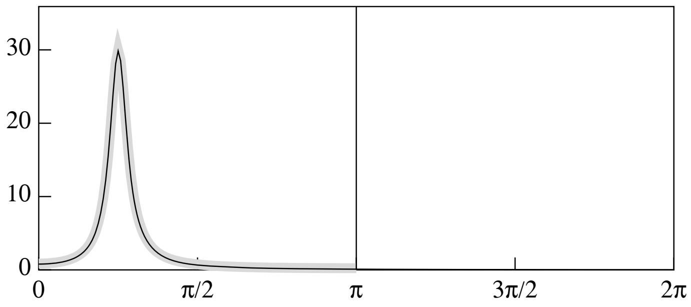


Fig. 10 The spectrum of the LSDE(2, 1) corresponding to the ARMA(2, 1) model of Example 1 plotted on top of the spectrum of that model represented by the thick grey line. The two spectra virtually coincide over the interval $[0, \pi]$

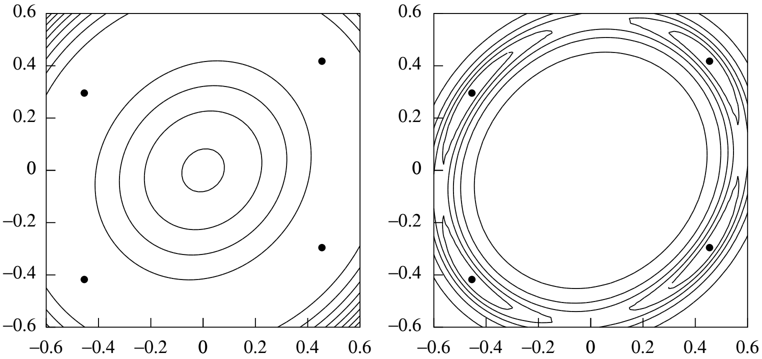


Fig. 11 *Left* The contours of the criterion function $z = z(a, b)$ together with the minimising values, marked by black dots. *Right* The contours of the function $q = 1/(z + d)$

the minimising points of the criterion function, which have become maxima in this diagram, are a testimony to the virtual equivalence of a wide spectrum of parameter values.

Example 3 A variant to the ARMA(2, 1) model is one that has a pair of complex conjugate poles $\rho \exp\{\pm i\theta\}$ with the same argument as before, which is $\theta = \pi/4 = 45°$, and with a modulus that has been reduced to $\rho = 0.5$. The model retains the zero of 0.5. The ARMA parameters and those of the corresponding LSDE are as follows:

ARMA	LSDE
$\alpha_0 = 1.0$	$\phi_0 = 1.0$
$\alpha_1 = -0.7071$	$\phi_1 = 1.3868$
$\alpha_2 = 0.2500$	$\phi_2 = 1.0973$
$\beta_0 = 1.0$	$\theta_0 = 1.5012$
$\beta_1 = -05$	$\theta_1 = 0.8905$

Figure 12 shows the spectral density functions of the LSDE and of the ARMA model superimposed on same diagram. The spectrum of the LSDE extends far beyond the Nyquist frequency of π, which is the limiting ARMA frequency.

The ARMA process, which is to be regarded as a sampled version of the LSDE, is seen to suffer from a high degree of aliasing, whereby the spectral power of the LSDE that lies beyond the Nyquist frequency is mapped into the Nyquist interval $[-\pi, \pi]$, with the effect that the profile of the ARMA spectrum is raised considerably. On this basis, it can be asserted that the ARMA model significantly misrepresents the underlying continuous-time process.

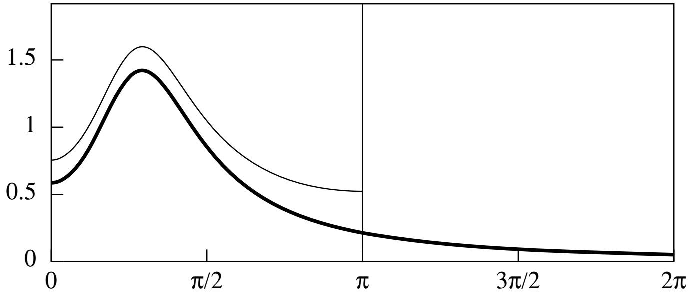


Fig. 12 The spectrum of the revised ARMA model superimposed on the spectrum of the derived LSDE, described by the heavier line

7 Summary and Conclusions

The intention of this paper has been to clarify the relationship between unconditional linear stochastic models in discrete and continuous time, and to provide secure means of computing the continuous models. The importance of an awareness of the frequency-domain characteristics of the forcing functions has been emphasised.

Example 1 has demonstrated a straightforward way of deriving a frequency-limited stochastic differential equation that corresponds to a discrete-time ARMA model. This has been described as a continuous-time CARMA model.

This model is a valid representation of the underlying process only if the maximum frequency of that process corresponds to the limiting frequency of the ARMA model, which is π radians per sampling interval. To ensure that this is the case, it may be necessary to reconstitute the continuous trajectory and to resample it at a reduced rate.

The forcing function of a conventional linear stochastic differential equation, or LSDE, which consists of the increments of a Wiener process, is unbounded in frequency. This seems to be inappropriate to a model of a frequency-limited process. Nevertheless, the transfer function of the LSDE may impose a radical attenuation on the higher frequencies that implies a virtual frequency limitation. Example 2 has illustrated such a case.

Example 3 has shown the aliasing effects that occur when the forcing function has no frequency limit and when the ARMA transfer function imposes only a weak attenuation on the high-frequency elements. This provides a ready justification for adopting an LSDE as the continuous-time counterpart of the ARMA model.

The spectral density function of the ARMA model will be formed by wrapping the spectrum of the LSDE around a circle of circumference 2π and by adding the overlying ordinates. In this way, the spectral component of frequencies in excess of the Nyquist value are mapped into the interval $[-\pi, \pi]$ to produce a discrete-time spectrum that may depart significantly from the continuous-time parent spectrum, as represented by the derived LSDE. This is seen in Fig. 12.

The methods for translating from an ARMA model to a continuous-time model, which may be a frequency-limited CARMA model or an LSDE model that is without an ostensible frequency restriction, have been realised in the computer program CONCRETE.PAS, which is available at the author's website, where both the compiled program and its code can be found.

An associated program CONTEXT.PAS, which plots the contour map of the surface of the criterion function that is employed in matching the autocovariance function of the LSDE(2, 1) to that of the ARMA(2, 1) model, is also available.

References

1. Bartlett, M.S.: On the theoretical specification and sampling properties of autocorrelated time series. Suppl. J. Roy. Stat. Soc. **8**, 27–41 (1946)
2. Hodrick, R.J., Prescott, E.C.: Postwar U.S. business cycles: an empirical investigation. Working Paper, Carnegie–Mellon University, Pittsburgh, Pennsylvania (1980)
3. Hodrick, R.J., Prescott, E.C.: Postwar U.S. Business cycles: an empirical investigation. J. Money, Credit Bank. **29**, 1–16 (1997)
4. Leser, C.E.V.: A simple method of trend construction. J. Roy. Stat. Soc. Ser. B **23**, 91–107 (1961)
5. Nelder, J.A., Mead, R.: A simplex method for function minimization. Comput. J. **7**, 308–313 (1965)
6. Nyquist, H.: Certain factors affecting telegraph speed. Bell Syst. Tech. J. **3**, 324–346 (1924)
7. Nyquist, H.: Certain topics in telegraph transmission theory, transactions of the AIEE, 47, 617–644. Proc. IEEE **90**, 280–305 (1928) (Reprinted in 2002)
8. Pollock D.S.G.: Stochastic processes of limited frequency and the effects of oversampling. Econom. Stat. **7**, 18–29 (2018)
9. Shannon, C.E.: Communication in the presence of noise. Proc. Inst. Radio Eng. **37**, 10–21 (Reprinted in 1998) Proc. IEEE **86**, 447–457 (1949)
10. Söderström, T.: On zero locations for sampled stochastic systems. IEEE Trans. Autom. Control **35**, 1249–1253 (1990)
11. Söderström, T.: Computing stochastic continuous-time models from ARMA models. Int. J. Control **53**, 1311–1326 (1991)

New Test for a Random Walk Detection Based on the Arcsine Law

Marcin Dudziński, Konrad Furmańczyk, and Arkadiusz Orłowski

Abstract In our work, we construct a new statistical test for a random walk detection, which is based on the arcsine law. Additionally, we consider a version of the unit root test for an autoregressive process of order 1, which is also related to the arcsine law. Furthermore, we conduct some simulation study in order to check the quality of the proposed test.

Keywords Random walk · Arcsine law · Test for a random walk detection

1 Introduction

Our objective is to introduce some proposal of a new test for a random walk detection. To the best of our knowledge, the main tools that have been applied in this context so far are the two celebrated tests—an Augmented Dickey–Fuller (ADF) test ([10]) and the Runs test ([13])—and through our work, we attempt to fill in a gap related to this field of investigations. The presented approach is a certain extension and a generalization of the research conducted in [2]. We also compare the quality of the proposed test with the efficiency and the power of the mentioned ADF and Runs test. The readers who are closely interested in the field of tests devoted to a random walk identification or to the existence of unit root are encouraged to refer to [6–9] and [11].

M. Dudziński · K. Furmańczyk (✉) · A. Orłowski
Institute of Information Technology, Warsaw University of Life Sciences, Warsaw, Poland
e-mail: konrad_furmanczyk@sggw.edu.pl

M. Dudziński
e-mail: marcin_dudzinski@sggw.edu.pl

A. Orłowski
e-mail: arkadiusz_orlowski@sggw.edu.pl

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_4

Our paper is organized as follows. In Sect. 1, we present a general idea leading to the construction of our test for a random walk identification, as well as we describe the construction of this test. In Sects. 2 and 3, we check the efficiency and the power of the introduced test, whereas we summarize our study in Sect. 4. The presented research and its results are an extension of the research and the results from [3].

1.1 Random Walk

Random walk theory states that the price of financial instrument in the subsequent time point is the sum of its price in the previous time point and some random variable with a finite variance, i.e. it is modelled with the use of a stochastic process called a random walk.

We say that a stochastic process $S_0, S_1, S_2, \dots, S_n$ is a random walk, if the following relations hold:

$$\begin{aligned} S_0 &= s_0, \\ S_1 &= s_0 + Y_1, \\ S_2 &= s_0 + Y_1 + Y_2, \\ &\vdots \\ S_n &= s_0 + Y_1 + Y_2 + \cdots + Y_n, \end{aligned}$$

where $Y_1, Y_2, \dots, Y_n$ form an iid sequence of symmetric r.v.'s.

In our considerations, we assume that $s_0 = 0$. Then,

$$S_t = \sum_{i=1}^{t} Y_i, \ t = 1, 2, \dots, n.$$

1.2 Ordinary Random Walk Test

Let

$$\Pi_n = |1 \le i \le n : \ S_i > 0|.$$

Then, obviously: Π_n—the number of those among the sums $S_1, \dots, S_n$, which are positive, Π_n/n—its frequency.

From the first arcsine law ([4, 10]), we have

$$\lim_{n\to\infty} P(\Pi_n < nx) = \int_0^x \frac{1}{\pi\sqrt{t(1-t)}} dt = \frac{2}{\pi} \arcsin(\sqrt{x}),$$

for all $x \in (0; 1)$.

Conclusion above may practically be used for $n \geq 20$, which means that

$$P\left(\frac{\Pi_n}{n} < x) \approx \frac{2}{\pi}\arcsin(\sqrt{x}\right) \text{ for } n \geq 20,$$

where obviously

$$\frac{\Pi_n}{n} = \frac{|1 \leq i \leq n : S_i > 0|}{n}.$$

From Fig. 1–depicting the density of the arcsine distribution—we observe that the values of Π_n/n in the close neighbourhood of 0.5 are the least probable and the most probable values for Π_n/n are close to 0 or 1 ([4]).

Thus, if we denote by α the significance level of the test H_0: S_n is a random walk process, we look for a critical area (a set of rejections) of the form

$$K_{c(\alpha)} = (0.5 - c(\alpha); 0.5 + c(\alpha)),$$

where $0 < c(\alpha) < 0.5$ satisfies the condition

$$\int_{0.5-c(\alpha)}^{0.5+c(\alpha)} \frac{1}{\pi\sqrt{x(1-x)}}dx = \frac{2}{\pi}\arcsin(\sqrt{x})\Big|_{0.5-c(\alpha)}^{0.5+c(\alpha)} = \alpha.$$

Hence, for $0 < c(\alpha) < 0.5$,

Fig. 1 Density of the arcsine distribution

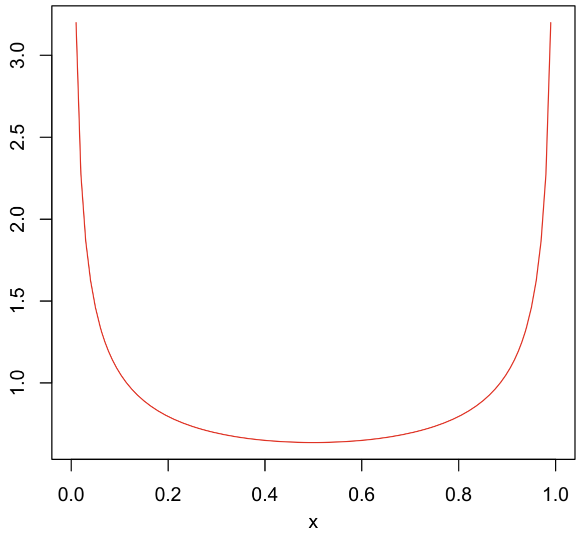


Table 1 Values of $c(\alpha)$

α	0.01	0.05	0.1
$c(\alpha)$	0.008	0.039	0.078

$$\arcsin(\sqrt{0.5 + c(\alpha)}) - \arcsin(\sqrt{0.5 - c(\alpha)}) = \frac{\pi\alpha}{2}.$$

The values of $c(\alpha)$, calculated numerically for the chosen significance levels according to the last formula, are collected in Table 1.

For $\alpha = 0.05$, we obtain $c(\alpha) = c(0.05) = 0.039$ and hence, the corresponding critical area is $K_{c(0.05)} = (0.5 - 0.039; 0.5 + 0.039) = (0.461; 0.539)$.

1.3 Random Walk Test for an AR(1) Process

Recall that an AR(1) process is defined as follows: $X_n = \rho X_{n-1} + \varepsilon_n$, where (ε_n) stands for the white noise with a mean zero and the variance σ^2. Observe that assuming the starting point $x_0 = 0$, we have:

$$\begin{aligned} X_1 &= \varepsilon_1, \\ X_2 &= \rho X_1 + \varepsilon_2 = \rho\varepsilon_1 + \varepsilon_2, \\ X_3 &= \rho X_2 + \varepsilon_3 = \rho^2\varepsilon_1 + \rho\varepsilon_2 + \varepsilon_3, \\ &\vdots \\ X_n &= \rho X_{n-1} + \varepsilon_n = \rho^{n-1}\varepsilon_1 + \rho^{n-2}\varepsilon_2 + \rho^{n-3}\varepsilon_3 + \cdots + \rho\varepsilon_{n-1} + \varepsilon_n. \end{aligned}$$

Therefore, the hypothesis H_0: (X_n) forms a random walk, is equivalent to the hypothesis that $\rho = 1$ (in this case (X_n) is a RW process, since then $X_n = \varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \ldots + \varepsilon_{n-1} + \varepsilon_n$).

2 Efficiency Evaluation of the Proposed Test

2.1 Gaussian Random Walk

The efficiency of our test has firstly been checked for a Gaussian random walk, i.e. in the case when $Y_i \sim N(0; 1)$, for $M = 1000$ generations of samples of the size $n = 1000$ or $n = 2000$:

$$
\begin{aligned}
&(y_1^{(1)}, y_2^{(1)}, \dots, y_n^{(1)}),\\
&(y_1^{(2)}, y_2^{(2)}, \dots, y_n^{(2)}),\\
&\vdots\\
&(y_1^{(1000)}, y_2^{(1000)}, \dots, y_n^{(1000)}).
\end{aligned}
$$

For every sample above, we calculated the values of the test statistic Π_n/n:

$$
\begin{aligned}
(\Pi_n/n)_{\text{emp}}^{(1)} &= \frac{|1 \le i \le n : y_1^{(1)} + y_2^{(1)} + \cdots + y_i^{(1)} > 0|}{n},\\
(\Pi_n/n)_{\text{emp}}^{(2)} &= \frac{|1 \le i \le n : y_1^{(2)} + y_2^{(2)} + \cdots + y_i^{(2)} > 0|}{n},\\
&\vdots\\
(\Pi_n/n)_{\text{emp}}^{(1000)} &= \frac{|1 \le i \le n : y_1^{(1000)} + y_2^{(1000)} + \cdots + y_i^{(1000)} > 0|}{n}.
\end{aligned}
$$

We calculated the number of those among Π_n/n, which belonged to the critical area $K_{c(0.05)} = (0.461; 0.539)$, i.e. we computed the number of rejections of H_0. We repeated this procedure 6 times and obtained the following numbers of rejections (out of 1000 possible rejections): 53, 44, 54, 33, 55, 50 (if $n = 1000$) or 44, 48, 43, 45, 50, 56 (if $n = 2000$). The small numbers of rejections may give an evidence about a good efficiency of the proposed test.

2.2 *Gaussian Mixture Model*

Secondly, the efficiency of our test has been checked for the case when (Y_i) is a Gaussian mixture model of the form

$$
Y_i \sim 1/8N(0; \sigma_1) + 1/8N(0; \sigma_2) + 1/4N(0; \sigma_3) + 1/2N(0; \sigma_4),
$$

where: $\sigma_1 = 0.2, \sigma_2 = 1, \sigma_3 = 0.5, \sigma_4 = 2$.
In this case, we generated $M = 1000$ samples from the distribution od Y_i, of the sizes: $n = 20, 30, 50, 100, 1000, 2000$. As previously, we calculated the number of those among Π_n/n, which belonged to the critical area $K_{c(0.05)} = (0.461; 0.539)$, i.e. we computed the number of rejections of H_0. We have collected the obtained results in Table 2, where, additionally, the corresponding results received by application of an Augmented Dickey–Fuller (ADF) test have also been included.

From Table 2, we may observe that the numbers of rejections obtained with the use of our test are small, which indicates that the efficiency of the proposed test is quite acceptable with comparison to the efficiency of the ADF test, for which the numbers of rejections are too large.

Table 2 Numbers of rejections

n	20	30	50	100	1000	2000
ADF_test	134	173	566	950	1000	1000
Proposed_test	41	57	38	46	40	42

3 Power Evaluation of the Proposed Test

3.1 An AR(1) Process with the Gaussian Innovations

We have checked here the power of our test by generating $M = 1000$ samples of the size $n = 30$, $n = 1000$ or $n = 2000$, of the corresponding AR(1) process with the Gaussian innovations and $\sigma = 3$:

$$
\begin{aligned}
&(x_1^{(1)}, x_2^{(1)}, \ldots, x_n^{(1)}),\\
&(x_1^{(2)}, x_2^{(2)}, \ldots, x_n^{(2)}),\\
&\vdots\\
&(x_1^{(1000)}, x_2^{(1000)}, \ldots, x_n^{(1000)}).
\end{aligned}
$$

We calculated the values of the test statistic Π_n/n:

$$
\begin{aligned}
(\Pi_n/n)_{\text{emp}}^{(1)} &= \frac{|1 \le i \le n : x_1^{(1)} + x_2^{(1)} + \cdots + x_i^{(1)} > 0|}{n},\\
(\Pi_n/n)_{\text{emp}}^{(2)} &= \frac{|1 \le i \le n : x_1^{(2)} + x_2^{(2)} + \cdots + x_i^{(2)} > 0|}{n},\\
&\vdots\\
(\Pi_n/n)_{\text{emp}}^{(1000)} &= \frac{|1 \le i \le n : x_1^{(1000)} + x_2^{(1000)} + \cdots + x_i^{(1000)} > 0|}{n}.
\end{aligned}
$$

As previously, we calculated the numbers of those among Π_n/n, which belonged to the critical area $K_{c(0.05)} = (0.461; 0.539)$. We obtained the following results (the numbers of rejections of H_0 among 1000 realizations) for the chosen values of ρ, after the 3 repetitions of the described procedure (see Table 3).

For the positive values of ρ, the results are quite promising—the larger ρ is, the smaller number of rejections of H_0 is obtained. Additionally, in case of the negative values of ρ, we get that the number of rejections is large and that these numbers increase if the sample size increases.

Next, we compare the test procedure from this subsection with the ADF and Runs tests for randomness of binary data series (we obtain them by putting $+1$, if the first differences are positive and -1, if otherwise).

3.1.1 Comparison with the ADF Test

For the chosen values of ρ, we obtained the following numbers of rejections of H_0 among 1000 realizations, after the 3 repetitions of the described procedure (see Table 4).

From Tables 3–4, we observe that—for the positive values of ρ—our test has a lower power than the ADF test for $\rho = 0.8$ and $\rho = 0.6$, but for the remaining cases, the powers of our test and the ADF test are comparable. Furthermore, in case of the negative values of ρ, we get that the numbers of rejections are large and that these numbers increase if the sample size increases. It may also be seen from the given tables that in the case of small samples ($n = 30$), the power of our proposed test is greater than the power of the ADF test.

Table 3 Numbers of rejections ($n = 30, n = 1000, n = 2000$, 3 replications). The proposed test for an AR(1) process with the Gaussian innovations

ρ	$n = 30$			$n = 1000$			$n = 2000$		
0.99	28	35	36	172	170	154	232	246	246
0.8	165	174	175	682	659	667	845	845	813
0.6	235	242	265	844	841	839	951	954	958
0.4	296	313	323	934	935	925	988	985	985
0.2	360	385	355	968	953	954	997	997	999
−0.2	470	505	480	996	996	996	1000	1000	1000
−0.4	499	508	483	999	997	999	1000	1000	1000
−0.6	570	563	553	999	1000	1000	1000	1000	1000
−0.8	631	648	691	1000	1000	1000	1000	1000	1000
−0.99	923	926	936	1000	1000	1000	1000	1000	1000

Table 4 Numbers of rejections ($n = 30, n = 1000, n = 2000$, 3 replications). The ADF test for an AR(1) process with the Gaussian innovations

ρ	$n = 30$			$n = 1000$			$n = 2000$		
0.99	54	46	43	179	148	156	546	533	552
0.8	73	59	63	1000	1000	1000	1000	1000	1000
0.6	81	107	88	1000	1000	1000	1000	1000	1000
0.4	138	121	139	1000	1000	1000	1000	1000	1000
0.2	143	146	148	1000	1000	1000	1000	1000	1000
−0.2	204	239	222	1000	1000	1000	1000	1000	1000
−0.4	245	265	261	1000	1000	1000	1000	1000	1000
−0.6	243	239	258	1000	1000	1000	1000	1000	1000
−0.8	287	274	262	1000	1000	1000	1000	1000	1000
−0.99	279	261	255	1000	1000	1000	1000	1000	1000

Table 5 Numbers of rejections ($n = 30, n = 1000, n = 2000$, 3 replications). The Runs test for an AR(1) process with the Gaussian innovations

ρ	$n = 30$			$n = 1000$			$n = 2000$		
0.99	71	64	52	50	55	53	58	61	44
0.8	47	58	55	503	482	502	798	840	827
0.6	73	63	74	991	989	992	1000	1000	1000
0.4	122	127	137	1000	1000	1000	1000	1000	1000
0.2	217	219	250	1000	1000	1000	1000	1000	1000
−0.2	569	571	563	1000	1000	1000	1000	1000	1000
−0.4	745	765	775	1000	1000	1000	1000	1000	1000
−0.6	901	908	902	1000	1000	1000	1000	1000	1000
−0.8	980	991	971	1000	1000	1000	1000	1000	1000
−0.99	1000	1000	999	1000	1000	1000	1000	1000	1000

3.1.2 Comparison with the Runs Test

For the chosen values of ρ, we obtained the following numbers of rejections of H_0 among 1000 realizations, after the 3 repetitions of the described procedure (see Table 5).

From Tables 3 and 5, we may see that—for the positive values of ρ—our test has a lower power than the Runs test for $\rho = 0.6$. However, for $\rho = 0.99$ and $\rho = 0.8$ our test has a better power than the Runs test. For the remaining cases, the powers of our test and the Runs test are comparable. Additionally, in case of the negative values of ρ, we get that the number of rejections is large and that these numbers increase if the sample size increases.

3.2 An AR(1) Process with the Student-T Innovations

In this subsection, we have checked the power of our test by generating $M = 1000$ samples, of the size $n = 30$, $n = 1000$ or $n = 2000$, of the AR(1) process with the Student-t innovations (for degrees of freedom 4) and $\sigma = 3$.

We proceeded as in the earlier subsections and calculated the numbers of those among Π_n/n, which belonged to the critical area $K_{c(0.05)} = (0.461; 0.539)$. We obtained the following results (the numbers of rejections of H_0 among 1000 realizations) for the chosen values of ρ, after the 3 repetitions of the described procedure (see Table 6).

From Tables 3 and 6, we can observe that our test works similarly in the case of an AR(1) process with the Gaussian innovations and in the case of an AR(1) process with the Student-t innovations.

3.2.1 Comparison with the ADF Test

For the chosen values of ρ, we obtained the following number of rejections of H_0 among 1000 realizations, after the 3 repetitions of the described procedure (see Table 7).

From Tables 6 and 7, we observe that—for the positive values of ρ—our test has a lower power than the ADF test for $\rho = 0.8$ and $\rho = 0.6$, but for the remaining cases of positive ρ, the powers of our test and the ADF test are comparable. Conclusions concerning the negative values of ρ are identical as in the previous examples.

Table 6 Numbers of rejections ($n = 30, n = 1000, n = 2000$, 3 replications). The proposed test for an AR(1) process with the Student-t innovations

ρ	$n = 30$			$n = 1000$			$n = 2000$		
0.99	38	36	35	165	166	155	225	242	218
0.8	152	155	164	696	638	649	822	828	838
0.6	236	231	267	843	845	832	965	955	944
0.4	315	299	302	929	930	941	983	989	989
0.2	367	363	343	966	958	970	998	996	997
−0.2	452	465	465	996	990	993	1000	1000	1000
−0.4	499	514	539	998	994	1000	1000	1000	1000
−0.6	573	575	557	1000	1000	1000	1000	1000	1000
−0.8	681	683	694	1000	1000	1000	1000	1000	1000
−0.99	921	918	930	1000	1000	1000	1000	1000	1000

Table 7 Numbers of rejections ($n = 30, n = 1000, n = 2000$, 3 replications). The ADF test for an AR(1) process with the Student-t innovations

ρ	$n = 30$			$n = 1000$			$n = 2000$		
0.99	47	47	42	149	133	157	560	545	542
0.8	67	61	69	1000	1000	1000	1000	1000	1000
0.6	84	98	97	1000	1000	1000	1000	1000	1000
0.4	118	116	130	1000	1000	1000	1000	1000	1000
0.2	152	165	153	1000	1000	1000	1000	1000	1000
−0.2	209	220	193	1000	1000	1000	1000	1000	1000
−0.4	249	252	234	1000	1000	1000	1000	1000	1000
−0.6	227	260	270	1000	1000	1000	1000	1000	1000
−0.8	267	264	251	1000	1000	1000	1000	1000	1000
−0.99	265	257	263	1000	1000	1000	1000	1000	1000

Table 8 Numbers of rejections ($n = 30, n = 1000, n = 2000$, 3 replications). The Runs test for an AR(1) process with the Student-t innovations

ρ	$n = 30$			$n = 1000$			$n = 2000$		
0.99	58	51	47	52	43	40	53	62	51
0.8	42	39	53	512	535	517	797	814	803
0.6	68	62	70	988	985	982	1000	1000	1000
0.4	138	124	148	1000	1000	1000	1000	1000	1000
0.2	218	234	219	1000	1000	1000	1000	1000	1000
−0.2	548	542	560	1000	1000	1000	1000	1000	1000
−0.4	764	756	731	1000	1000	1000	1000	1000	1000
−0.6	918	910	913	1000	1000	1000	1000	1000	1000
−0.8	984	984	979	1000	1000	1000	1000	1000	1000
−0.99	1000	1000	999	1000	1000	1000	1000	1000	1000

3.2.2 Comparison with the Runs Test

For the chosen values of ρ, we obtained the following number of rejections of H_0 among 1000 realizations, after the 3 repetitions of the described procedure (see Table 8).

From Tables 6 and 8, we may see that—for the positive values of ρ—our test has a lower power than the Runs test for $\rho = 0.6$, but for $\rho = 0.99$ and $\rho = 0.8$, our test has a better power than the Runs test and for the remaining cases of positive ρ, the powers of our test and the Runs test are comparable. Conclusions concerning the negative values of ρ are very similar as in the previous examples.

4 Conclusions

The principal goal of our study was to construct a new test for a random walk detection. The main idea of our approach was based on the first arcsine law. Apart from the construction of a new test, we examined its efficiency and power by using 1000 replications of the Monte Carlo simulation and computing the numbers of rejections of the null hypothesis that the given process forms a random walk. The corresponding qualities of the proposed test have been checked for the following types of processes: (i) the Gaussian random walk, (ii) the Gaussian mixture model, (iii) an AR(1) process with the Gaussian innovations, (iv) an AR(1) processes with the Student-t innovations. Moreover, the powers of our test have been compared with the powers of the two well-known tests used for the random walk identification, i.e. with the powers of the ADF and Runs tests. In particular, it is also worth to mention that: (1) the efficiency of our test for the Gaussian mixture model is reasonably good with comparison to the ADF test for this model, (2) in the case of small samples

($n = 30$), the power of our new test is greater than the power of the ADF test, (3) the powers of all the considered tests increase if the sample sizes are larger, (4) for the negative values of the correlation coefficient ρ, the powers of all the investigated tests are nearly the same, (5) the proposed test gives similar results both in the case of an AR(1) process with the Gaussian innovations and in the case of an AR(1) process with the Student-t innovations. The obtained results and comparisons indicate that the introduced test provides quite effective and relatively powerful tool leading to a random walk identification.

References

1. Dickey, D.A., Fuller, W.A.: Distributions of the estimators for autoregressive time series with a unit root. Jour. Amer. Stat. Assoc. **74**, 427–431 (1979). https://doi.org/10.2307/2286348
2. Dudziński, M., Furmańczyk, K., Orłowski, A.: Some proposal of the test for a random walk detection and its application in the stock market data analysis. Quant. Methods Econ. **19**, 339–346 (2018). https://doi.org/10.22630/MIBE.2018.19.4.32
3. Dudziński, M., Furmańczyk, K., Orłowski, A.: New test for a random walk detection based on the arcsine law. In: ITISE 2019 International Conference on Time Series and Forecasting, Proceedings of Papers 25–27 September 2019, vol. 1, pp. 236–243. Godel Impresiones Digitales S.L., Granada (Spain) (2019)
4. Feller, W.: An Introduction to Probability Theory and its Applications. Wiley, New York (1968)
5. Maddala, G.S.: Introduction to Econometrics. Morgan Kaufmann, Wiley, New York (2001)
6. Mankiw, N.G., Shapiro, M.D.: Trends, random walks, and tests of the permanent income hypothesis. J. Monet. Econ. **16**(2), 165–174 (1985). https://doi.org/10.1016/0304-3932(85)90028-5
7. Pantula, S.G., Farias-Gonzales, G., Fuller, W.A.: A comparison of unit-root test criteria. J. Bus. Econ. Stat. **12**, 449–459 (1994). https://doi.org/10.1080/07350015.1994.10524567
8. Phillips, P.C.B.: Time series regression with a unit root. Econometrica **55**, 277–301 (1987). https://doi.org/10.2307/1913237
9. Phillips, P.C.B., Perron, P.: Testing for a unit root in time series regression. Biometrika **75**, 335–346 (1988). https://doi.org/10.1093/biomet/75.2.335
10. Qiang, L., Jiajin, L.: Arcsine laws and its simulation and application. http://individual.utoronto.ca/normand/Documents/MATH5501/Project-3/Arcsine_laws_and_simu.pdf
11. Said, E.S., Dickey, D.A.: Testing for unit roots in autoregressive-moving average models of unknown order. Biometrika **71**, 599–607 (1984). https://doi.org/10.1093/biomet/71.3.599
12. Shiller, R.J., Perron, P.: Testing the random walk hypothesis: power versus frequency of observation. Econ. Lett. **18**, 381–386 (1985). https://doi.org/10.1016/0165-1765(85)90058-8
13. Siegel, S., Castellan, N.J.: Nonparametric Statistics for the Behavioural Sciences. McGraw-Hill, New York (1988)

Econometric Models and Forecasting

On the Automatic Identification of Unobserved Components Models

Diego J. Pedregal and Juan R. Trapero

Abstract Automatic identification of time series models is a necessity once the big data era has come and is staying among us. This has become obvious for many companies and public entities that have passed from a crafted analysis of each individual problem to handle a tsunami of information that has to be processed efficiently, online and in record time. Automatic identification tools have never been tried out on Unobserved Components models (UC). This chapter shows how information criteria, such as Akaike's or Schwarz's, are rather useful for model selection within the UC family. The difficulty lies, however, on choosing an appropriate and as general as possible set of models to search in. A set too narrow would render poor forecast accuracy, while a set too wide would be highly time consuming. The forecasting results suggest that UC models are powerful potential forecasting competitors to other well-known methods. Though there are several pieces of software available for UC modeling, this is the first implementation of an automatic algorithm for this class of models, to the best of the author's knowledge.

Keywords Unobserved components models · State-space systems · Kalman filter · Smoother algorithm · Maximum likelihood

1 Introduction

The era of big data is provoking a revolution in many research areas. Indeed, it can be said that in the area of time series forecasting the effect is particularly dramatic. Nowadays, big masses of time series ought to be forecast in short periods of time.

D. J. Pedregal (✉)
Industrial Engineering Politecnic, University of Castilla-La Mancha, 13071 Ciudad Real, Spain
e-mail: diego.pedregal@uclm.es

J. R. Trapero
Faculty of Chemical Sciences and Technologies, University of Castilla-La Mancha,
13071 Ciudad Real, Spain
e-mail: juanramon.trapero@uclm.es

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_5

Take as an example Walmart with 5,000 stores throughout the US whose forecasting needs amounts to 10 millions per second! ([24], p. 828). Therefore, at least in such contexts, the traditional crafted approach to identification one time series at a time must be replaced by automatic identification alternatives.

Automatic selection of models has received a great deal of attention in the time series literature. This interest extends from classical modeling techniques such as regression analysis, exponential smoothing, ARIMA, transfer functions, etc. ([6, 13, 15, 27]), to modern Big Data techniques such as Artificial Neural Networks, Support Vector Machines, etc. ([11, 12, 29]).

Though it is almost impossible to make an exhaustive list of all the proposed forecasting methods in the literature, a good guidance to this variety may be found in the results of predictive competitions [21]. The most common forecasting methods to today are Exponential Smoothing (ETS) and ARIMA methods.

– Exponential Smoothing methods remain the most widely used modeling technique in day-to-day business and industry since the 50s [5]. Given the success and the fact that it was proposed initially as a heuristic method, a major revision has taken place in the past 20 years that has dramatically changed the vision of these techniques [3, 16].
– The second method mostly used is ARIMA. ARIMA models have expanded since the 70s after the publication of the influential book by Box and Jenkins [2]. Various methods have been proposed for automatic identification [6, 16], TRAMO (together with SEATS) being probably the ARIMA automatic identification procedure most used worldwide in official statistical agencies.

In all this scientific landscape, there is a family of models with applications in many branches of science with rather good results, which has been conspicuously ignored, namely, the Unobserved Components models (UC, [4, 10, 23, 28]). UC models aim explicitly at decomposing a vector of time series on components with economic meaning, normal trend, seasonal and irregular, although it may also include other components, typically cycles and exogenous variables modeled as linear regressions, transfer functions, or nonlinear relationships.

The UC models have not been tested yet in automatic modeling settings for many reasons. First, UC models were developed by engineers and brought into economics by academics, with little interest in disseminating them among practitioners [22]. Second, UC methods are rarely taught at the undergraduate level, limiting access to the wide public. Third, there is a widespread intuition that UC models have nothing to add to other methods (especially Exponential Smoothing, [7]). Fourth, UC models are generally identified by hand, without any attempt to develop any automatic identification procedure. Finally, software packages are scarcer than packages for other more standard techniques. Some complete alternatives are, for example, STAMP [18], SSfpack [19], and SSpace [26].

The methods described in this chapter fill this gap by introducing a procedure to automatically select optimal UC models among a wide range of possibilities. The methods are useful in forecasting terms, but other byproducts are the estimated optimal components (trend, seasonal, irregular) that may be useful for other common and useful operations in time series analysis such as smoothing, signal extraction, seasonal adjustment, detrending, etc.

The chapter is organized as follows. Section 2 presents briefly the UC models in general and the range of possibilities for each component. Section 3 shows how the UCs are inserted in the general State-Space framework. The automatic identification procedure is presented in Sect. 4. Section 5 shows the method of working in practice on three real-life case studies and compared to other alternatives. Finally, Sect. 6 concludes.

2 Unobserved Components Models

The UC models aims at decomposing a time series into meaningful components. The most common decomposition is shown in Eq. (1), where T_t, S_t, and I_t stand for the trend, seasonal and irregular components, respectively.

$$z_t = T_t + S_t + I_t \tag{1}$$

There have been many approaches to deal with this decomposition, from which the *structural* approach set up in a State-Space (SS) framework is the most widespread.

Structural methods specify directly the particular dynamical models for each component involved, for which an ample range of possibilities exists. In general, all components are assumed stochastic, trends should be nonstationary by definition, seasonal components should show some sinusoidal behavior, and irregular components are usually considered either white or colored noise. The particular models chosen in this chapter for each component steam from a long tradition (see, among others, [4, 9, 10, 28]).

2.1 *Trend Components*

All trends considered are particular cases of the Generalized Random Walk (or Damped Trend, DT) model shown in Eq. (2), where T_t^* is referred to as the trend 'slope', $0 < \alpha \leq 1$, $\eta_{T,t}$, and $\eta_{T,t}^*$ are independent white noise sequences with variances $\sigma_{\eta_T}^2$ and $\sigma_{\eta_T^*}^2$, respectively.

$$\begin{bmatrix} T_{t+1} \\ T_{t+1}^* \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & \alpha \end{bmatrix} \begin{bmatrix} T_t \\ T_t^* \end{bmatrix} + \begin{bmatrix} \eta_{T,t} \\ \eta_{T,t}^* \end{bmatrix} \tag{2}$$

This model subsumes the following particular cases: (i) Random Walk (RW), by eliminating the second equation (i.e., $T_{t+1} = T_t + \eta_{T,t}$ or setting $\alpha = 0$, $\sigma_{\eta^*}^2 = 0$ and $T_1^* = 0$); (ii) Integrated Random Walk (IRW) with $\alpha = 1$ and $\sigma_{\eta_T}^2 = 0$, (it is equivalent to the well-known Hodrick–Prescott filter, [14, 28]); (iii) Local Linear Trend (LLT) with $\alpha = 1$, see e.g., [4, 10, 25]. All these trends are stochastic and have at least one unit root ensuring they are not stationary.

2.2 Seasonal Components

From all possibilities available in the literature (e.g., [9, 10, 28]), seasonal components used in this chapter take a stochastic trigonometric form. The seasonal component is built as the sum of individual sinusoidal terms for the fundamental period s and its harmonics. The number of harmonics in general is $[s/2] = s/2$ for even s numbers, and $[s/2] = (s-1)/2$ for uneven s numbers.

The overall seasonal component is the sum of all the sinusoidal harmonics $S_{j,t}$ in Eq. (3), where $\omega_j = 2\pi j/s$ is the frequency of each harmonic, $S^*_{j,t}$ is an additional state necessary for the specification, and $\eta_{j,t}$ and $\eta^*_{j,t}$ are independent white noises with common variance σ_j^2.

$$\begin{gathered} S_t = \sum_{j=1}^{[s/2]} S_{j,t} \\ \begin{bmatrix} S_{j,t+1} \\ S^*_{j,t+1} \end{bmatrix} = \begin{bmatrix} \cos\omega_j & \sin\omega_j \\ -\sin\omega_j & \cos\omega_j \end{bmatrix} \begin{bmatrix} S_{j,t} \\ S^*_{j,t} \end{bmatrix} + \begin{bmatrix} \eta_{j,t} \\ \eta^*_{j,t} \end{bmatrix} \end{gathered} \tag{3}$$

An usual assumption regarding the seasonal component is to make all the variance noises equal to each other, i.e., $\sigma_j^2 = \sigma^2, j = 1, 2, \ldots, [s/2]$. This is indeed the case of the popular Basic Structural Model (BSM) of [10], but also of exponential smoothing models [15]. A much more flexible assumption is allowing all variances to be different (strictly as they are specified in Eq. (3)), an option that, though increasing the number of parameters, it renders models that are still feasible for most time series.

2.3 Irregular Components

The empirical evidence in many cases is that, after taking into account trends and seasonal components, the remainder is just white noise. Therefore, the standard irregular component is just a Gaussian white noise with zero mean and constant variance σ_I^2. However, for the cases where a serial correlation problem still remains, colored irregular components may be considered in the form of stationary ARMA(p, q) models, in general of low orders.

An ARMA(p, q) model is of the form

$$I_t = \phi_1 I_{t-1} + \phi_2 I_{t-2} + \cdots + \phi_p I_{t-p} + \eta_{I,t} + \theta_1 \eta_{I,t-1} + \theta_2 \eta_{I,t-2} + \theta_q \eta_{I,t-q}$$

where $\eta_{I,t}$ is a Gaussian white noise with constant variance σ_I^2, and ϕ_i $(i = 1, 2, \ldots, p)$ and θ_k $(k = 1, 2, \ldots, q)$ are unknown parameters that ought to be estimated from the data.

3 State-Space Systems

Once the model for all the components is specified, the *structural* UC approach proceeds by assembling all of them in a single linear Gaussian SS system by block concatenation of the individual models, in which Eq. (1) plays the role of the observation equation. Then, all the statistical theory applicable to SS systems apply to the UC models straight away.

The minimum linear Gaussian SS system to deal with the whole set of models implemented in this chapter is shown in Eq. (4).

$$\begin{array}{rl} \text{Transition equation:} & \alpha_{t+1} = \Phi\alpha_t + R\eta_t, \;\; \eta_t \sim N(0, Q) \\ \text{Observation equation:} & \quad z_t = Z\alpha_t + \epsilon_t, \quad\; \epsilon_t \sim N(0, H) \end{array} \tag{4}$$

In these equations z_t is a univariate time series; α_t is a non-observable state vector of length n; η_t and ϵ_t are the state and observational independent noises with zero-mean Gaussian noises, with dimensions $r \times 1$ and 1×1, respectively; the initial state vector is assumed to be stochastic with Gaussian distribution, i.e., $\alpha_1 \sim N(a_1, P_1)$, and independent of all data and noises involved in the system; the remaining elements in (4) are the so-called system matrices with appropriate dimensions.

Much more complicated systems are possible, i.e., multivariate systems with time-varying system matrices, nonlinear, non-Gaussian, etc., but are not necessary for the present chapter, and therefore are not considered here. For another toolbox with many of such capabilities, see [26].

The main objective of SS systems is to obtain optimal estimations of the state vector and their covariances, in the sense of minimizing the mean square error, conditional on the particular model specified and all information available. Two sorts of estimates are most common in practice:

- Filtered output by the well-known Kalman Filter. It provides the optimal state vector estimation using all the information available up to any point in time.
- Smoothed output by Fixed Interval Smoother algorithms that renders the optimal estimates of the state vector based on the whole sample (past and future values), in a similar way to moving averages.

There are many issues related to state and parameter estimation in SS systems. The main ones concerning this chapter are

- Missing data: they are naturally interpolated by the Kalman Filter and Smoother algorithms, because of their inherent recursive nature. Forecasts are also naturally produced by signaling the future values as missing data.

- The typical problem of initial conditions common to all dynamic systems is solved by using the exact initialization proposed by [4], known as diffuse filtering and smoothing.
- Model parameters scattered along the system matrices are estimated by maximizing the diffuse log-likelihood [4].
- Maximization of the log-likelihood function requires optimization algorithms, which are usually Quasi-Newton type. Such algorithms take advantage of gradients of the likelihood surface, which may be computed either numerically or analytically [4]. Analytical gradients are possible for models that depend only on variance parameters in matrices Q and H in Eq. (4).

4 Automatic Forecasting Algorithm for UC

The automatic forecasting algorithm proposed below is based on information criteria (similar to [6, 16]) and performs remarkably well in practice, as will be shown in later worked examples. This is the first time that an algorithm of this nature is proposed in the literature about UCs.

The algorithm proceeds along the following steps:

- Step 1: Variance transformation. Decide whether to use the Box-Cox transformation or not [1]. This step is left to the user discretion because its benefits in terms in forecasting accuracy are not clear [15]. The approach by [8] is preferred here because it is not model dependent.
- Step 2: Model selection. A battery of models are estimated and the best is chosen according to the minimization of any information criterion, either the Akaike's (AIC) or Schwarz's (SBC), i.e.,

  $$AIC = -2\ln(L^*) + 2k$$

  $$SBC = -2\ln(L^*) + \ln(T)k$$

  where L^* is the likelihood value at the optimum, T is the length of the time series, and k the number of parameters in the model.

  The set of models to search for are 23 and are all the possible combinations of trends (none, RW, LLT, DT), seasonal components (none, all harmonics with equal variance, all harmonics with different variances), and irregulars (none or Gaussian noise). The none trend/none seasonal/none irregular is excluded from the models set.
- Step 3: ARMA model selection. A low-order ARMA model is then identified by AIC or SBC, on the innovations of the previous model estimated in step 2. The algorithm used is a simplified version of [16] for full nonstationary and seasonal

ARIMA models. The simplification consists of searching exclusively on stationary and nonseasonal models, since both non-stationarity and seasonality are already captured by the trend and seasonal component.

- Step 4: Joint final estimation. If an ARMA model is detected in the previous step, then the full UC model with the ARMA irregular component embedded should be estimated.
- Step 5: Forecasting step. Final forecasts are produced with the best of models in steps 2 or 4, depending on which one exhibits the smallest information criterion value.

5 Case Studies

The case studies considered below show how the UC automatic methods described in previous sections perform with respect to ARIMA and exponential smoothing (ETS) as implemented in the package `forecast` in `R` (functions `auto.arima` and `ets` were used, respectively) [16]. This package has gained the role of a standard to which any new method may be confronted. Two further methods are added to make comparisons more comprehensive, namely, a seasonal naïve method as a benchmark and the mean of the UC, ETS and ARIMA (see e.g., [20] about the importance of forecast combinations). Another dimension added to the case studies is checking whether the variance Box-Cox transformation improves forecast accuracy [1].

The case studies have been selected to be as varied as possible, they comprise a weather time series, another from macroeconomics and a demand database of a retail business typical of supply chain applications. The sampling intervals of the time series are also varied, ranging from quarterly to daily.

Comparisons are carried out on the basis of two error metrics, the symmetric Mean Absolute Percentage Error (sMAPE) and the Mean Absolute Scaled Error (MASE), see Eqs. (5), (6) and [17, 20]. z_t and $\hat{z}_t$ are the actual and forecast values at time t, respectively; T is the forecast origin; h is the forecast horizon; and n is the number of observations in the fitting sample

$$\text{sMAPE}_h = h^{-1} \sum_{i=1}^{h} \frac{2 \mid z_{T+i} - \hat{z}_{T+i} \mid}{\mid z_{T+i} \mid + \mid \hat{z}_{T+i} \mid} \times 100 \tag{5}$$

$$\text{MASE}_h = h^{-1} \sum_{i=1}^{h} \frac{\mid z_{T+i} - \hat{z}_{T+i} \mid}{(n-1)^{-1} \sum_{r=2}^{n} \mid z_r - z_{r-1} \mid} \tag{6}$$

5.1 Monthly Average Temperatures in Madrid at El Retiro Weather Station

Monthly average temperatures in Madrid from 1988 are shown in Fig. 1. Data is compiled from the El Retiro weather station and are publicly available at www-2.munimadrid.es/CSE6/control/seleccionDatos?numSerie=14020000020.

The series is dominated by the seasonal pattern. Maybe some noise is also present, but the trend, if any at all, is rather mild. All the methods are applied in a forecasting exercise consisting of a rolling out experiment in which the initial forecasting origin is set in December 2002 and the forecasting horizon is fixed at 12 months ahead. Then, one monthly observation is added at each iteration and the whole process is repeated until the end of the sample is reached. Therefore, 181 total rounds of 12 months-ahead forecasts from all models are produced and averaged along all the rounds to make final comparisons.

The models selected by the automatic identification of UCs confirm the initial intuitions based on Fig. 1: trends are always either nonexistent or damped with very small damping factors that effectively are very close to a nonexistent trend (i.e., the α parameter in Eq. (2) always estimated smaller than 0.3); seasonal components are very strong, about half of the runs with common variance for all harmonics and a half with different variances; the irregulars are identified either as nonexistent (74% of all runs) or white noise (26%). The λ parameter of the Box-Cox variance transformation is in general close to 1, implying that the series does not exhibit heteroskedasticity problems.

The average forecasting performance of all models used is shown in Table 1, sMAPE at the left-hand side and MASE at the right. Several conclusions may be extracted from this table: (i) forecasts deteriorate with the horizon for all models, as expected; (ii) all models show significant performance improvements over the Naïve, implying that they are really capturing the structure of the data beyond a naïve seasonal pattern; (iii) the ordering of models from best to worst according to

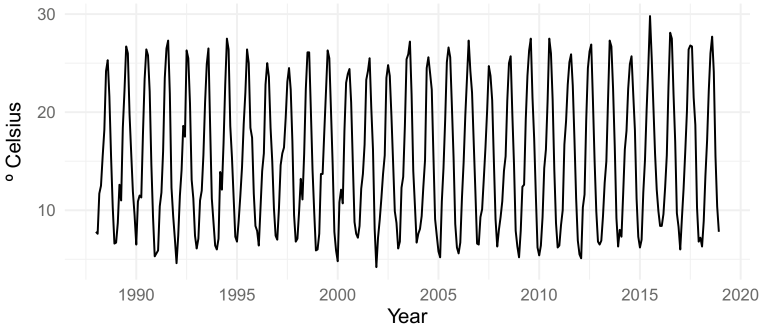


Fig. 1 Average temperatures in Madrid central

Table 1 Error metrics for Madrid average temperatures error forecasts for several models and forecasting horizons. sMAPE is on the left part of the table and MASE on the right part. Minimum of each row is emphasized both for sMAPE and MASE

sMAPE						MASE				
h	Naïve	ETS	ARIMA	UC	Mean	Naïve	ETS	ARIMA	UC	Mean
1	12.649	9.171	8.688	**8.560**	8.736	1.083	0.799	**0.740**	0.742	0.753
2	12.706	9.247	9.009	**8.876**	8.976	1.086	0.803	0.770	**0.768**	0.774
3	12.732	9.321	9.104	**9.004**	9.084	1.088	0.809	**0.779**	**0.779**	0.784
4	12.755	9.357	9.153	**9.073**	9.140	1.091	0.812	**0.784**	0.785	0.789
5	12.784	9.384	9.192	**9.111**	9.177	1.094	0.814	**0.788**	0.789	0.792
6	12.804	9.393	9.213	**9.146**	9.200	1.097	0.815	**0.790**	0.792	0.794
7	12.814	9.382	9.224	**9.161**	9.208	1.098	0.814	**0.791**	0.793	0.795
8	12.821	9.376	9.233	**9.171**	9.216	1.100	0.813	**0.792**	0.794	0.795
9	12.825	9.387	9.233	**9.176**	9.224	1.100	0.814	**0.792**	0.794	0.796
10	12.833	9.387	9.239	**9.180**	9.229	1.101	0.813	**0.792**	0.794	0.796
11	12.837	9.387	9.249	**9.187**	9.237	1.101	0.813	**0.792**	0.795	0.796
12	12.844	9.384	9.250	**9.197**	9.240	1.102	0.813	**0.792**	0.795	0.796

sMAPE is UC-Mean-ARIMA-ETS; (iv) almost the same classification is produced with the MASE, except that ARIMA is the best and UC is relegated to the second position, even though ARIMA, UC and Mean look actually very close to each other.

5.2 *Spanish Gross Domestic Product (GDP)*

Figure 2 shows the Spanish quarterly GDP between the first quarter of 1995 and the third quarter of 2019 in real terms (chain-linked volume index). It follows a pattern similar to many Western economies with a strong trend and seasonality and a big drop due to the 2008 recession followed by a final recovery.

The rolling out exercise in this case starts in the last quarter of 2009 and the forecasting horizon is fixed at 8 quarters ahead, i.e., the total number of 8 quarters-ahead forecast rounds is 32.

The UC model selected for most of the forecasting origins consists of a damped trend (with a damping parameter oscillating between 0.89 and 0.95), a seasonal component with equal variance for all harmonics, and no irregular component. The estimated components for the whole sample may be seen in Fig. 3.

The average forecasting performance of all models used is shown in Table 2. The table is divided into four quadrants reporting the average SMAPE and MASE metrics for each model with and without the variance Box-Cox transformation for horizons ranging from 1 to 12 months.

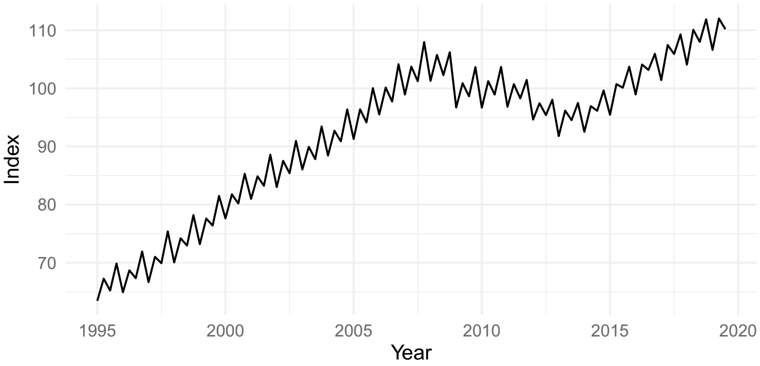


Fig. 2 Spanish real quarterly GDP between 1995 and 2019

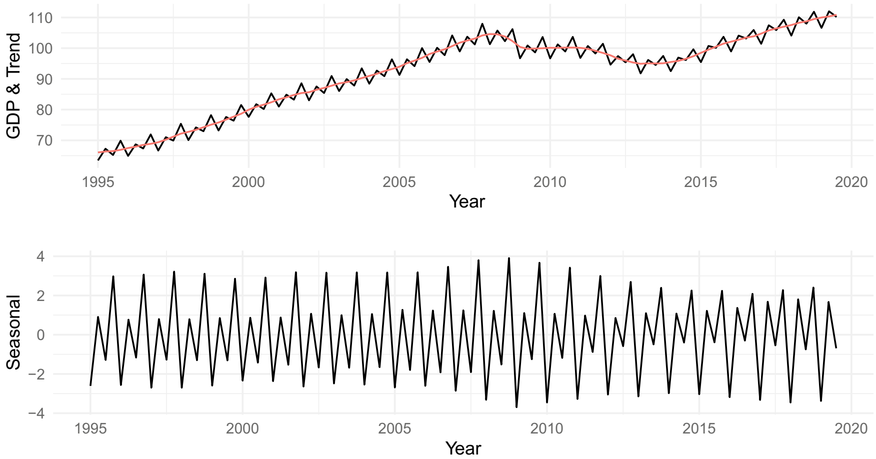


Fig. 3 Spanish real quarterly GDP between 1995 and 2019

Table 2 offers some interesting insights into this forecasting exercise, some in common with the previous case study. Firstly, forecasts deteriorate with the horizon for all models. Secondly, all models show significant performance improvements over the Naïve. Thirdly, the absolute winner in this case regardless of the error metric is the UC method, followed by Mean, ETS, and ARIMA. Finally, this classification, with just a few exceptions, is independent of whether the Box-Cox transformation is used or not, the error metric, and the forecasting horizon.

Table 2 Error measurements on Spanish GDP forecasts for several models and forecasting horizons. sMAPE is on top half and MASE at bottom. Results with Box-Cox variance transformation at the right side. Minimum of each row are emphasized

sMAPE						sMAPE Box-Cox			
h	Naïve	ETS	ARIMA	UC	Mean	ETS	ARIMA	UC	Mean
1	2.059	0.666	0.677	**0.576**	0.584	0.658	0.664	0.585	0.600
2	2.099	0.781	0.785	0.666	0.688	0.754	0.788	**0.665**	0.699
3	2.134	0.937	0.968	**0.798**	0.836	0.904	0.957	0.806	0.847
4	2.164	1.112	1.134	**0.943**	0.995	1.066	1.128	0.954	1.000
5	2.560	1.339	1.378	**1.148**	1.209	1.288	1.359	1.166	1.216
6	2.848	1.562	1.613	**1.353**	1.414	1.499	1.593	1.381	1.417
7	3.072	1.777	1.865	**1.572**	1.640	1.707	1.844	1.591	1.638
8	3.255	1.990	2.086	**1.781**	1.842	1.915	2.075	1.803	1.844
MASE						MASE Box-Cox			
h	Naïve	ETS	ARIMA	UC	Mean	ETS	ARIMA	UC	Mean
1	0.773	0.245	0.246	**0.210**	0.214	0.241	0.241	0.213	0.219
2	0.789	0.288	0.285	0.244	0.252	0.276	0.286	**0.243**	0.256
3	0.803	0.344	0.352	**0.293**	0.307	0.331	0.349	0.296	0.310
4	0.815	0.408	0.412	**0.346**	0.364	0.390	0.411	0.349	0.366
5	0.965	0.491	0.499	**0.422**	0.442	0.471	0.494	0.427	0.445
6	1.075	0.573	0.585	**0.497**	0.517	0.547	0.579	0.505	0.517
7	1.160	0.651	0.675	**0.578**	0.598	0.622	0.670	0.582	0.597
8	1.229	0.729	0.755	**0.655**	0.671	0.698	0.754	0.659	0.671

5.3 Demand Database

The last case study is more complex than the previous ones, because it consists of all the daily demand time series, 142 in total, collected from a Spanish fresh food franchise. The series are available for the last 200 days and have a variety of properties in terms of predominance of components, volatility, etc. Two typical examples are shown in Fig. 4. The bottom panel shows a time series dominated by the weekly pattern with a more or less stable mean, while the series at the top exhibits both a seasonal component and a decreasing trend much less stable.

The rolling experiment in this example consists of 48 runs for each of the 142 time series starting at day 140 and choosing a forecasting horizon of 14 days.

The heterogeneity of this bunch of time series is reflected in the variety of UC models automatically identified: trends are either damped (with damping factor varying between values close to 0 and 0.83) or Random Walks in equal parts (with a few of them nonexistent); 11% of seasonal components are identified with different variances for each of the three harmonics, 23% of the series are estimated without any seasonal component and the rest are chosen as seasonal components with common

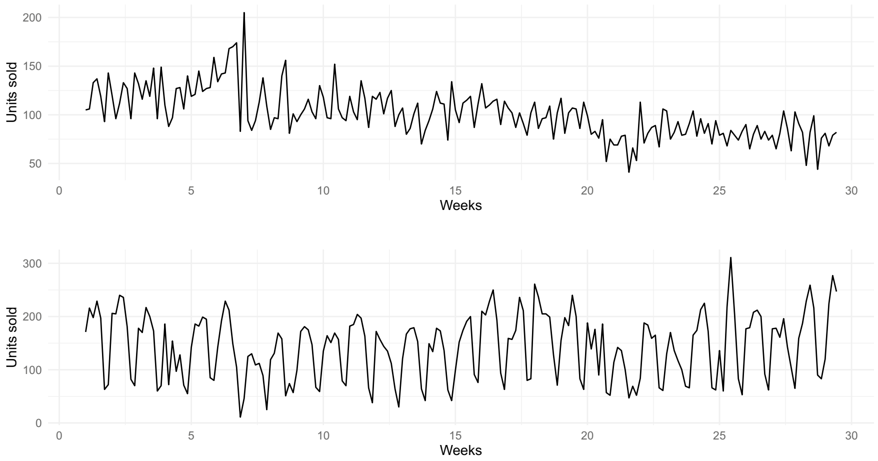


Fig. 4 Two examples of daily sales of a retailer in Spain

Table 3 Error measurements on demand time series for several models and selected forecasting horizons. sMAPE is on top half and MASE at bottom. Results with Box-Cox variance transformation at the right side. Minimum of each row are emphasized

sMAPE						sMAPE Box-Cox			
h	Naïve	ETS	ARIMA	UC	Mean	ETS	ARIMA	UC	Mean
1	6.016	4.717	4.914	4.710	4.634	4.684	4.865	4.696	**4.619**
3	5.927	4.865	5.049	4.857	4.779	4.844	4.992	4.841	**4.752**
5	5.911	4.947	5.084	4.931	4.841	4.942	5.041	4.926	**4.825**
7	5.857	4.927	5.054	4.900	4.819	4.937	5.022	4.911	**4.815**
9	5.833	4.927	5.051	4.887	4.813	4.940	5.022	4.909	**4.812**
11	5.817	4.935	5.053	4.888	**4.813**	4.949	5.030	4.913	4.817
13	5.814	4.940	5.059	4.891	**4.818**	4.959	5.038	4.920	4.826
14	5.813	4.939	5.060	4.886	**4.817**	4.958	5.043	4.919	4.829
MASE						MASE Box-Cox			
h	Naïve	ETS	ARIMA	UC	Mean	ETS	ARIMA	UC	Mean
1	1.136	0.838	0.893	0.840	0.828	**0.827**	0.897	0.835	0.828
3	1.115	0.871	0.924	0.873	0.860	0.858	0.922	0.866	**0.856**
5	1.109	0.890	0.938	0.891	0.877	0.881	0.936	0.887	**0.875**
7	1.097	0.894	0.939	0.894	**0.880**	0.890	0.939	0.894	0.881
9	1.098	0.902	0.948	0.900	**0.887**	0.900	0.948	0.903	0.890
11	1.094	0.906	0.951	0.902	**0.890**	0.910	0.952	0.909	0.896
13	1.092	0.910	0.954	0.905	**0.893**	0.960	0.955	0.911	0.914
14	1.092	0.912	0.956	0.906	**0.894**	0.962	0.958	0.913	0.916

variances; the irregular is nonexistent in 22% of cases, while the rest are just white noise. Heterogeneity is also detected on the Box-Cox transformation that oscillates between −0.35 and 1, with only 9 cases above 0.88.

Table 3 summarizes the values of the error metrics in a similar format to previous tables. In this case, only some selected forecasting horizons are shown to make the table shorter and the averages are calculated along time series and forecast origins, i.e., each value on the table is the average of $48 \times 142 = 6816$ forecast errors.

Table 3 shows that all methods outperform the Naïve method. Forecasts roughly worsen for longer forecasting horizons (not so clear as in previous case studies). The winner method is unambiguously the combination of methods (Mean), followed by UC, ETS, and ARIMA. There is an interesting distinct behavior of error depending on the horizon, because for horizons up to 7 days ahead (there are some variations depending on the method) the Box-Cox transformation gives lower errors that forecast with no transformation. There is only one exception to the previous rule, the sMAPE metric for the ARIMA method, for which forecasts are always better with Box-Cox transformation.

6 Conclusions

This chapter presents a novel automatic identification procedure for UC models, consisting of estimating a wide range of possible models and selecting the best according to any information criterion, like Akaike's or Schwarz's. This sort of algorithm is pretty useful in Big Data contexts, where many time series ought to be processed reliably in rather fast times.

The most important point is choosing an appropriate set of UC models, wide enough to be able to represent efficiently as many time series as possible. In that regard, the trend components available are either none, Random Walk, Local Linear Trend, or Damped Trend (see Eq. (2)). The seasonal component is either none, seasonal harmonics with equal variances, or with different variances. Finally, irregulars are allowed to select among none, white noise or ARMA processes. As far as the authors are concerned, this is the widest set of UC models available in the literature.

The previous algorithm is assessed on three case studies in comparison with other well-known methods, namely ARIMA and ETS as implemented in the `forecast` package in `R`.

The results show that the proposed identification algorithm is strongly competitive with the rest, being the best very often. Apart from this general conclusion that is the most important, there are other findings that were not specifically pursued, and therefore should be considered only partial to the particular case studies included. Firstly, there is little disagreement between both error metrics (sMAPE and MASE) when ordering the forecasting methods. Secondly, there are not clear improvements in forecasting accuracy when the Box-Cox variance transformation is used. Finally, combination of forecasts (at least the mean used in this chapter) does not imply better forecasts, only in the last case study the combination outperformed the rest.

To sum up, UC models automatically identified provides a nice tool that may enter the forecaster's toolbox, with some nice byproducts consisting of the optimal decomposition of time series in trend, seasonal component, and irregular, that often are required for detrending, signal extraction, seasonal adjustment, etc.

Acknowledgements This work was supported by the Spanish Government (MINECO/FEDER, UE) under the project with reference DPI2015-64133-R and by the Vicerrectorado de Investigación y Política Científica from UCLM through the research group fund program (PREDILAB; DOCM 24/01/2019 [2019/536]) and by the European Regional Development Fund.

References

1. Box, G., Cox, D.: An analysis of transformations. J. R. Stat. Soc. Ser. B **26**, 211–252 (1964)
2. Box, G.E.P., Jenkins, G.M., Reinsel, G.C., Ljung, G.M.: Time Series Analysis: Forecasting and Control. Wiley (2015)
3. De Livera, A., Hyndman, R., Snyder, R.: Forecasting time series with complex seasonal patterns using exponential smoothing. J. Am. Stat. Assoc. **106**, 1513–1527 (2011)
4. Durbin, J., Koopman, S.: Time Series Analysis by State Space Methods. Oxford University Press (2012)
5. Gardner, E.: Exponential smoothing: the state of the art—part ii. Int. J. Forecast. **22**, 637–666 (2006)
6. Gómez, V., Maravall, A.: Automatic Modeling Methods for Univariate Series. In: A Course in Time Series, pp. 171–201. Wiley (2001)
7. Gooijer, J., Hyndman, R.: 25 years of time series forecasting. Int. J. Forecast. **22**, 443–473 (2006)
8. Guerrero, V.M.: Time-series analysis supported by power transformations. J. Forecast. **12**(1), 37–48 (1993)
9. Harrison, P., Stevens, C.: Bayesian forecasting. J. R. Stat. Society. Ser. B (Methodological) **38**, 205–247 (1976)
10. Harvey, A.: Forecasting, Structural Time Series Models and the Kalman Filter. Cambridge University Press (1989)
11. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer Publishing, New York (2009)
12. Haykin, S.: Neural Networks and Learning Machines. Prentice Hall, New Jersey (2008)
13. Hocking, R.: The analysis and selection of variables in linear regression. Biometrics **32**, 1–49 (1976)
14. Hodrick, R., Prescott, E.: Postwar U.S. business cycles: an empirical investigation. J. Money Credit Bank. **29**, 1–16 (1997)
15. Hyndman, R., Koehler, A., Ord, J., Snyder, R.: Forecasting with Exponential Smoothing: The State Space Approach. Springer Science & Business Media (2008)
16. Hyndman, R., Khandakar, Y.: Automatic time series forecasting: the forecast package for R. J. Stat. Softw. **27**, 1–22 (2008)
17. Hyndman, R., Koehler, A.: Another look at measures of forecast accuracy. Int. J. Forecast. **22**(4), 679–688 (2006)
18. Koopman, S., Harvey, A., Doornik, J., Shephard, N.: Structural Time Series Analysis, Modelling, and Prediction Using STAMP. Timberlake Consultants Press, London (1999)
19. Koopman, S., Shephard, N., Doornik, J.: Statistical Algorithms for Models in State Space Form: SsfPack 3.0. Timberlake Consultants Press (2008)
20. Makridakis, S., Hibon, M.: The M3-competition: results, conclusions and implications. Int. J. Forecast. **16**, 451–476 (2000)

21. Makridakis, S., Spiliotis, E., Assimakopoulos, V.: The m4 competition: results, findings, conclusion and way forward. Int. J. Forecast. **34**, 802–808 (2018)
22. Pedregal, D.: State space modeling for practitioners. Foresight **54**, 21–25 (2019)
23. Pedregal, D., Young, P.: Statistical approaches to modeling and forecasting time series. In: A Companion to Economic Forecasting, pp. 69–104. Blackwell Publishing Ltd (2002)
24. Seaman, B.: Considerations of a retail forecasting practitioner. Int. J. Forecast. **34**, 822–829 (2018)
25. Taylor, C., Pedregal, D., Young, P., Tych, W.: Environmental time series analysis and forecasting with the `Captain` toolbox. Environ. Model. Softw. **22**(6), 797–814 (2007)
26. Villegas, M., Pedregal, D.: Sspace: a toolbox for state space modelling. J. Stat. Softw. **87–5**, 1–26 (2018)
27. Young, P.: Recursive Estimation and Time-Series Analysis. Springer (2011)
28. Young, P., Pedregal, D., Tych, W.: Dynamic harmonic regression. J. Forecast. **18**(6), 369–394 (1999)
29. Zhang, G.: Times series forecasting using a hybrid arima and neural network model. Neurocomputing **50**, 159–175 (2003)

Spatial Integration of Pig Meat Markets in the EU: Complex Network Analysis of Non-linear Price Relationships

Christos J. Emmanouilides and Alexej Proskynitopoulos

Abstract We analyze the spatial price causality structure between the pig meat markets of 24 European countries using weekly time-series data from 2007 to 2018 and non-linear Granger causality. The EU pig meat market is studied as a dynamic complex network of linkages between prices in member states. We investigate the temporal development of the spatial network of price relationships, and through the dynamics of its major structural characteristics we draw insights about the horizontal agricultural market integration process in the EU. Of interest is the evolution of the degree of market inter-connectedness, the strength and reciprocity of price relationships, the development of influential markets (hubs) and of market clusters with strongly interacting components.

Keywords Price relationships · Non-linear Granger causality · Complex networks · Spatial market integration · Generalized additive model

1 Introduction

Spatial price relationships are commonly studied in economics to provide empirical insights about the integration of geographically separated markets. In efficient, well-integrated markets, price dependencies tend to be strong, reciprocal, diffuse, and more homogeneous. On the other hand, in more segregated markets, price relationships may be clustered and exhibit a high degree of heterogeneity. Since its foundation, a major goal of EU's economic policy has been the establishment of a frictionless, more homogeneous common market of commodities and services.

C. J. Emmanouilides (✉)
College of Engineering and Technology, American University of the Middle East, Kuwait City, Kuwait
e-mail: christos.emmans@aum.edu.kw

A. Proskynitopoulos
Department of Operations, Kellogg School of Management, Northwestern University, Evanston, USA
e-mail: alexej.proskynitopoulos@kellogg.northwestern.edu

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*, Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_6

Several authors have studied empirically the horizontal integration of national EU agricultural markets; e.g., Serra et al. [25], Emmanouilides and Fousekis [7, 9] employed tests for long-run price convergence (Law of One Price), while Emmanouilides et al. [8] and Grigoriadis et al. [12] used copulas to study price co-movements. All these works have considered small subsets of national markets. Studies of long-run price convergence did not assess the causal structure of price dependence between markets. On the other hand, the copula-based dependence measures employed in the latter works do not reveal any information about the origin of price shocks that give rise to the observed price dependencies and their dynamics. As such, they treat each market in a pair as equi-important in determining the relationship.

To provide a more thorough look into the integration of EU primary commodity markets we include in our study 24 out of the 28 EU member states, excluding three countries having very small size (Malta, Cyprus, and Luxemburg) and Croatia that joined EU very recently (in 2013). In this paper, that extents the work of Emmanouilides and Proskynitopoulos [10], we gain insights into the dynamics of price relationships by employing a non-linear Granger causality framework and analyze the whole EU pig meat market as a complex network of bipartite price linkages. These causal linkages are directional and generally asymmetric, in contrast to copula-based measures that are agnostic about price shocks' origins and directional asymmetries.

Causal networks have been used recently to study linkages between financial markets (e.g., [2, 5, 26]) but not, to the best of our knowledge, to agricultural markets. Our network construction is based on the use of a non-parametric non-linear generalized additive modeling (GAM) framework for testing non-causality. Simulation results show that for finite samples our test performs better not only than the linear causality test that is commonly adopted in studies of causal network, but also than the widely used Hiemstra-Jones [16] and Diks-Panchenko [6] tests of non-linear causality. Below we describe briefly our work, together with some of our findings; Sect. 2 discusses the data and methods, Sect. 3 presents the test performance in finite samples, Sect. 4 provides the empirical analysis and results, Sect. 5 provides conclusions.

2 Data and Methods

2.1 Data

The data are complete series of weekly wholesale prices for pig animals (euro/100 kg) in 24 EU member states from January 1, 2007 to October 29, 2018, obtained by the European Commission. The series are positively correlated, mostly to a high degree (Pearson correlation coefficients range from 0.28 to 0.98, with a mean of 0.74), indicating that price changes are transmitted between market pairs. As is common empirical practice in studies of market integration, we analyze logarithmic price returns $r_{i,t} = d\ln p_{i,t}$ that de-trend the series from deterministic and stochastic components. $p_{i,t}$ denotes price at country i, $i = 1, \ldots 24$, in week t, $t = 1, \ldots 618$.

2.2 Filtering

Inference on causality can be sensitive to autocorrelation and ARCH effects that are typically present in price returns series; Autocorrelation might spuriously result in significant Granger causality between markets and distort the direction of causality (e.g., [26]). Neglected non-stationarities, such as ARCH dependence, any associated volatility clustering or other structural changes, may be manifested as spurious non-linearities in the series of returns (e.g., [1, 16, 18, 19]), and consequently may bias inference.

To deal with these potential problems we filtered each series with an ARMA(m, n)-GARCH(p, q) model, using several alternative error distributions that allow for a variety of shape and skewness specifications. With the BIC criterion, we selected parsimonious models with orders $m, n, p, q \in \{1, 2, 3, 4, 5\}$. Optimal models with any AR and ARCH effects removed from the residuals were retained.[1] Several conditional error distributions were tested and selected on grounds of parameter significance and parsimony, again via BIC.

2.3 Non-linear Granger Causality Networks

Denote with $s_{i,t}$ the standardized innovations of the ARMA-GARCH filtered price returns of market i. Causal price linkages between two markets i and j are then established through testing for Granger non-causality in conditional means in the following form

$$H_0 : E\left(s_{j,t+1}|s_{j,t}^{L_j}, s_{i,t}^{L_i}\right) = E\left(s_{j,t+1}|s_{j,t}^{L_j}\right),\ H_1 : E\left(s_{j,t+1}|s_{j,t}^{L_j}, s_{i,t}^{L_i}\right) \neq E\left(s_{j,t+1}|s_{j,t}^{L_j}\right) \quad (1)$$

where L_i, L_j indicate finite lags of the series of the two markets, respectively. Then, two equations for the conditional expectations are involved in testing non-causality, one for each hypothesis in (1)

$$H_0 : E\left(s_{j,t+1}|s_{j,t}^{L_j}, s_{i,t}^{L_i}\right) = f_j\left(s_{j,t}^{L_j}\right),\ H_1 : E\left(s_{j,t+1}|s_{j,t}^{L_j}, s_{i,t}^{L_i}\right) = f_{ji}\left(s_{j,t}^{L_j}, s_{i,t}^{L_i}\right) \quad (2)$$

[1] Residuals were tested for the presence of AR and ARCH effects with the Ljung–Box test [20] and the ARCH test of Engle [11].

$f_j(.)$ and $f_{ji}(.)$ can be arbitrary, smooth functions of their arguments. In linear non-causality testing, they assume the standard additive linear form. Péguin-Feissolle and Teräsvirta [23] suggested a linear form including a potentially large number of cross-lag interaction terms as Taylor approximations of $f_j(.)$ and $f_{ji}(.)$.

Here, we implement a more flexible non-linear specification of functions $f_j(.)$ and $f_{ji}(.)$ introduced by Hastie and Tibshirani [15] and further developed by others (e.g., [28]) in the context of generalized additive models (GAMs). Under this specification, and assuming a gaussian link function relating the conditional mean with the lagged series, (2) becomes

$$E\left(s_{j,t+1}|s_{j,t}^{L_j}\right) = a_{0j} + \sum_{r=1}^{L_j} f_r(s_{j,t-r}) + u_{j,t},\ u_{j,t} \sim iid\,N\left(0, \sigma_{u,j}^2\right) \tag{3a}$$

$$\begin{aligned} E\left(s_{j,t+1}|s_{j,t}^{L_j}, s_{i,t}^{L_i}\right) = a_{0ji} + \sum_{p=1}^{L_j} f_p(s_{j,t-p}) + \sum_{q=1}^{L_i} f_q(s_{i,t-q}) \\ + \eta_{j,t},\ \eta_{j,t} \sim iid\,N\left(0, \sigma_{\eta,j}^2\right) \end{aligned} \tag{3b}$$

Functions $\{f_p, f_q, f_r\}$ are usually specified as non-parametric smooth functions of a single lagged variable. Typical choices are local scatter smoothers (loess), smoothing splines or, as more recently developed, smooth expansions of basis functions chosen from a range of alternative families. Note that Eqs. (3a, 3b) can be readily amended to include smooth cross-lag interaction terms to capture more delicate non-linear dependencies, if needed. Expansion coefficients are estimated together with a set of penalty parameters that regulate over-fitting using a penalized maximum likelihood iterative estimation method such as IRLS with the smoothing parameters determined at each iteration step via cross-validation. The estimation algorithm minimizes the penalized deviance

$$D(\mathbf{a}) + \sum_{m\in\{p,q\},l=i,j} \lambda_m \int f_m''(s_{l,t-m})^2 ds_{l,t-m} = D(\mathbf{a}) + \sum_{m\in\{p,q\},l=i,j} \lambda_m \mathbf{a}^T \mathbf{S}_m \mathbf{a} \tag{4}$$

where $\{p, q\}$ indicates the full set of lags ($p = 1, \ldots, L_j$ and $q = 1, \ldots, L_i$), $\mathbf{a}$ is the vector of coefficients to be estimated, D is the deviance, λ_m are the penalties and $\mathbf{S}_m$ is a matrix of known parameters calculated by the basis functions and the penalties (for details see [27, 28]). Selection of optimal lags $L_i, L_j \in \{1, 2, \ldots, 10\}$ is done with the use of some information criterion, such a BIC. If the computational burden is too high, penalties can be set to a fixed value, but at the possible cost of not fully explaining non-linearity in dependence. However, to safeguard against this possibility, tests for neglected non-linearity (e.g., the BDS test [4]) can be applied on the residuals of (3a, 3b) and accordingly re-adjust the degree of smoothing. Alternatively, the simpler and faster "backfitting" estimation method of Hastie and Tibshirani [14, 15] may be preferred.

In the GAM framework, testing for Granger non-causality can be performed with a generalized likelihood ratio test on the estimated models (3a, 3b); Denote with $L(\hat{\mathbf{a}}_{\mathrm{H}_0})$ and $L(\hat{\mathbf{a}}_{\mathrm{H}_1})$ the likelihoods of the models (3a) and (3b), respectively, and with $\hat{\mathbf{a}}_{\mathrm{H}_0}$, $\hat{\mathbf{a}}_{\mathrm{H}_1}$ the corresponding sets of estimated parameters. Then, under the null of non-causality and the usual regularity conditions, the log-likelihood ratio follows asymptotically an approximate chi-square distribution, $2(\log L(\hat{\mathbf{a}}_{\mathrm{H}_1}) - \log L(\hat{\mathbf{a}}_{\mathrm{H}_0})) \sim \chi^2_{\nu}$, with $\nu = df_{\mathrm{H}_1} - df_{\mathrm{H}_0}$

After performing the non-causality test and estimating the error variances of (3a, 3b), the Granger Causality Index (GCI) is computed as

$$GCI_{i\to j} = \left(1 - \hat{\sigma}^2_{\eta,j} \Big/ \hat{\sigma}^2_{u,j}\right) \tag{5}$$

This index is based on the Granger–Wald test (e.g., [13, 17]), and quantifies the strength of causal influence market i exerts on market j. A significant test result indicates the presence of a *directional* link $\{i \to j\}$ between the two markets with a *weight* $GCI_{i\to j}$. Price relationships can be bi-directional, if $\{j \to i\}$ is statistically significant, or not (otherwise), and generally asymmetric as it is expected that $GCI_{i\to j} \neq GCI_{j\to i}$.

The causal network at any time t is defined as a graph $\mathbf{G}_t = (\mathbf{V}, \mathbf{E}_t)$, consisting of a set $\mathbf{V}$ of vertices (nodes/markets) and a set $\mathbf{E}_t$ of directed weighted edges (links). Set $\mathbf{E}_t$ contains all directional weighted links $\{i \to j\}$ between markets (i, j) for which the causality test gives a significant result.

2.4 Network Measures

We consider two kinds of measures of network characteristics: measures that characterize (a) the connectivity of individual nodes, and (b) the cohesiveness of the global network.

Individual Node Connectivity

For a directed weighted network, the *in-strength* (or in-degree), $\mathrm{d}^{in}(i)$, and the *out-strength* (or out-degree), $\mathrm{d}^{out}(i)$, of a node i are defined correspondingly as

$$\mathrm{d}^{in}(i) = \sum_{j\in\mathbf{V};\{j\to i\}\in\mathbf{E}_t} GCI_{j\to i} \tag{6a}$$

$$\mathrm{d}^{out}(i) = \sum_{j\in\mathbf{V};\{i\to j\}\in\mathbf{E}_t} GCI_{i\to j} \tag{6b}$$

A node's in-strength is the sum of the weights of all incoming links to the node; In our context, it represents the total causal influence exerted to market i from all markets with a statistically significant causal influence on it. A node's out-strength is the sum of the weights of all links originating from the node; It quantifies the overall magnitude of a market's causal influence on the whole market system, and as such it may be viewed as a measure of a market's importance in driving other markets' price dynamics.

Another aspect of individual node connectivity refers to the "importance" of a node with respect to other nodes in the network. A common measure is *closeness centrality*, that is, determined using the shortest path (sequence of edges) or geodesic distance d(*i, j*) between nodes (*i, j*). Closeness centrality quantifies how "close" a node is to the other nodes in the network. It is defined as the inverse of the total distance of the node from the other nodes,

$$c_{CL}(i) = 1 \Big/ \sum_{j \in \mathbf{V}} d(i, j) \tag{7}$$

It can be readily normalized to range in [0, 1] by multiplying with |**V**| − 1, where |.| indicates set cardinality. In our context, a high value of closeness centrality would indicate a market that has a high degree of causal relationships (i.e., is "close") with each of several other markets in the system. These relationships can be direct (one-to-one) or indirect (through short causal flow paths). Markets with high closeness centrality are typically more connected than markets with low closeness centrality.

Global Network Cohesiveness

We employ four measures of global network structure; network density, average strength, average shortest path length, and reciprocity. Network *density* is the frequency of realized edges (causal links, **E**) relative to the total number of possible edges. For a directed graph it is calculated as

$$D = |\mathbf{E}| \big/ (|\mathbf{V}| - 1)|\mathbf{V}| \tag{8}$$

The higher its value, the more densely inter-connected the market system is. The *average strength*, i.e., the mean total strength (both "in" and "out") of all network nodes, calculated as

$$\bar{d} = \sum_{i \in \mathbf{V}} \left(\mathrm{d}^{in}(i) + \mathrm{d}^{out}(i)\right) \Big/ |\mathbf{V}| \tag{9}$$

reflects the average strength of price linkages at the system level. The higher its value, the stronger on average the price relationships between the markets are. The *average shortest path length* is the mean of the shortest path lengths between all market pairs,

$$\bar{l} = \sum\nolimits_{i,j \in \mathbf{V}} d(i, j) \big/ (|\mathbf{V}| - 1)|\mathbf{V}| \tag{10}$$

an indicator of the system's price transmission efficiency; the smaller its value, the faster is the diffusion of price shocks in the system. *Reciprocity* is a measure of bidirectionality in causal price relationships. We calculate it as the ratio of the number of bi-directional edges over the total number of edges in the directed graph,

$$r = |[\{i \rightarrow j\} \wedge \{j \rightarrow i\}] \in \mathbf{E}_t| \big/ |[\{i \rightarrow j\} \vee \{j \rightarrow i\}] \in \mathbf{E}_t| \tag{11}$$

Higher values correspond to a higher degree of mutual interactions between markets, indicating higher efficiency in the flows of price shocks and a higher level of market integration.

2.5 Temporal Network Evolution

The dynamics of the network of price relationships are assessed by estimating causal networks and network measures for 359 consecutive rolling windows of 5-years width ($5 \times 52 = 260$ observations per window) in order to maintain sufficient sample sizes for the causality tests. Other plausible width options were also explored without noticing important qualitative differences in the results.

To empirically test for possible significant structural changes in the series of estimated global network measures we employed generalized M-fluctuation tests (e.g., [29]). Sequences of such breaks, if present, may indicate the onset of different stages/regimes in the market integration process, characterized by distinct network market structures.

3 Finite Sample Properties of the GAM-Test

In this section, we present Monte-Carlo simulation results on the comparative performance of the proposed GAM-based test of Granger non-causality in the conditional mean (Eq. (3a, 3b); henceforth GAMG) against the standard linear OLS-based test (LG) and the correlation integral-based non-linear Hiemstra-Jones ([16]; HJ) and Diks-Panchenko ([6]; DP) tests. Due to space restrictions, we incorporate only a brief part of an extensive comparison study that will be presented elsewhere.

To assess the performance of the tests we use simulated data from three alternative bivariate data-generating processes (DGPs) of time series $\{X_t, Y_t\}$, $t = 1, \ldots, T$, between which a one-directional non-linear Granger causality exists, from Y_t to X_t, both in the mean and the variance. DGPs exhibit non-linear AR(1) or AR(2) dependence in the first moment and ARCH(1) dependence in the second moment, as follows:

$$\text{DGP1}: X_t = 0.3X_{t-1} + 0.1Y_{t-1}^2 + \varepsilon_{X,t}, Y_t = 0.3Y_{t-1} + \varepsilon_{Y,t}$$

$$\text{DGP2}: X_t = 0.2Y_{t-2}^2 + \varepsilon_{X,t}, Y_t = 0.1Y_{t-1} + 0.1Y_{t-2}^2 + \varepsilon_{Y,t}$$
$$\text{DGP3}: X_t = -1.2Y_{t-1}\exp\left(-Y_{t-1}^2\right) + \varepsilon_{X,t}, Y_t = -1.2Y_{t-1}\exp\left(-Y_{t-1}^2\right) + \varepsilon_{Y,t}$$

where $\varepsilon_{X,t,}\ \varepsilon_{Y,t} \sim iid\, N(0, \sigma_t^2 = 1 + 0.1Y_t^2)$. Such an ARCH specification was used by [6]. Inclusion of causal dependence in the variance gives some advantage to the HJ and DP tests over the LG and GAMG tests, as in their standard form (Sect. 2.3) the latter may detect causality only in the mean.

Size and power are estimated by the empirical rejection rates of the true null that X_t does not cause Y_t and of the false null that Y_t does not cause X_t, respectively. Calculations are performed over 1000 simulated bivariate series of length $T \in$ [500, 1000] per DGP, for two nominal sizes $\alpha \in$ [0.05, 0.10]. After ARMA-GARCH filtering, the series are subjected to the tests for non-causality. To mimic real testing, in which actual DGPs are unknown, we select lag lengths for the LG and GAMG tests via the BIC criterion. For the GAMG test, they typically coincide with the actual lag lengths of the numerous different DGPs studied, indicating that GAM models are capable of correctly identifying the lag structure of non-linear time series processes.[2] For the HJ and DP tests, literature is non-conclusive on optimal lag length choice. Extensive experimentation provided strong evidence that the use of GAM-selected lag lengths results in improved performance of the HJ and DP tests, an additional benefit that the proposed GAM approach can offer in non-linear non-causality testing. Table 1 presents empirical performance results.

As expected, due to misspecification, the LG test has low power against non-linear alternatives (except when a strong linear causal component is present, as in DGP3), and its size is sensitive to the degree of non-linearity in the data (e.g., is markedly oversized for DGP3). All non-linear tests suffer from size distortions that diminish as the sample size increases; a result that is indicative of the consistency of the associated test statistics. The GAMG test is oversized for smaller samples, while the HJ and DP tests are undersized. For the GAMG test, empirical size converges to the nominal value when a stationary bootstrap is used to overcome finite sample deviations from asymptotic theory assumptions.[3] GAMG has clearly better power than the other tests in all cases (nearly 1). However, a fair power comparison should adjust for the different actual sizes of the tests in finite samples. For this purpose, we employ the receiver operating characteristic curve (ROC) of the estimated power against test size. As also suggested by Lloyd [21], we estimate the partial area under the ROC curve (pAUC) for plausible test sizes that range between 0.001 and 0.1. The values of pAUC for each test provide size-adjusted power estimates and are reported in Table 1. Bootstrap tests for correlated ROCs provide strong evidence that GAMG has indeed better size-adjusted power (i.e., higher pAUC) than the other non-causality tests included in the comparison, for all studied DGPs and sample sizes.

[2] Results are available upon request.

[3] For the HJ and DP tests, bootstrap does not appear to improve finite sample performance.

Table 1 Finite sample performance of non-causality tests

			GAMG				LG			HJ			DP		
DGP	T	α	Size	B.S.[a]	Power	pAUC	Size	Power	pAUC	Size	Power	pAUC	Size	Power	pAUC
1	500	0.05	0.088	0.068	0.998	0.996	0.062	0.178	0.551	0.016	0.782	0.895	0.020	0.810	0.914
		0.10	0.164	0.128	1.000		0.110	0.270		0.066	0.860		0.052	0.876	
	1000	0.05	0.064	0.045	1.000	1.000	0.068	0.180	0.553	0.060	0.970	0.963	0.066	0.976	0.975
		0.10	0.118	0.080	1.000		0.110	0.278		0.096	0.986		0.092	0.992	
2	500	0.05	0.118	0.090	0.990	0.996	0.058	0.288	0.599	0.028	0.854	0.939	0.010	0.852	0.951
		0.10	0.198	0.156	0.996		0.118	0.372		0.066	0.938		0.050	0.930	
	1000	0.05	0.082	0.056	1.000	1.000	0.062	0.344	0.630	0.029	0.996	0.997	0.026	0.994	0.997
		0.10	0.154	0.104	1.000		0.126	0.434		0.078	1.000		0.066	1.000	
3	500	0.05	0.074	0.060	1.000	1.000	0.074	0.980	0.973	0.020	0.582	0.785	0.028	0.652	0.837
		0.10	0.136	0.090	1.000		0.152	0.984		0.100	0.700		0.080	0.760	
	1000	0.05	0.066	0.044	1.000	1.000	0.072	1.000	1.000	0.026	0.842	0.928	0.029	0.912	0.961
		0.10	0.126	0.094	1.000		0.152	1.000		0.100	0.914		0.084	0.948	

[a] Empirical size from 500 stationary bootstrap replications (e.g., see [24])

4 Empirical Analysis and Results

First, logarithmic price returns were tested for unit roots on the unconditional mean with standard ADF and KPSS tests, along with the spectral wavelet test of Nason [22] that is shown to have good size properties for heavy-tailed series such as price returns. In all cases, the tests did not provide evidence against weak stationarity.

Then, we applied an ARMA-GARCH filter to all series of returns. Models were estimated by maximizing the joint log-likelihood of the system of equations involved. In most cases, a skewed t-Student distribution was adequate for the conditional error. Ljung-Box and Engle's ARCH tests did not indicate any residual AR or ARCH effects.

In the next step, we used the filtered series to (a) estimate for each rolling window and for each market pair the GAM models in Eq. (3a, 3b), (b) conduct the Granger non-causality tests, (c) construct the networks, and (d) calculate the network measures. As we perform a large number ($24 \times 23 = 552$) of simultaneous tests to construct a single rolling window network of statistically significant causal links, we apply a Benjamini–Hochberg [3] adjustment to the p-values from the likelihood ratio tests by controlling the false discovery rate (FDR) for our chosen significance level (we use $a = 0.05$).

4.1 Network Measures of Individual Node Connectivity

Statistics for the rolling windows estimates of the individual market connectivity measures are shown in Table 2. Values of closeness are normalized. To summarize coarsely the temporal evolution of connectivity measures for each market we estimated a linear trend and calculated the coefficient of determination R^2. Insignificant trends ($\alpha = 0.05$, HAC corrected) are marked as "ns". Overall, the most influential market appears to be Germany, followed by Austria, Netherlands, Belgium, and Poland with average out-strength values exceeding 2.00. Most linear trends are positive; Poland, Portugal, Lithuania, and France have the highest average annual (linear) growth rate, ranging from 0.17 to 0.39, while Slovenia and Belgium had a marked decline in out-strength in the period of study. Figure 1 shows the out-strength evolution for a subset of markets.

Slovakia, Estonia, Czech Republic, and Denmark appear to have the highest average in-strength values (2.02 or more), indicating that price shocks in these markets were rather driven externally. As might be expected, high out-strength markets tend to have small in-strength and vice versa, indicating a grouping into markets with high power ("hubs"; Germany, Austria, Netherlands, Poland, Belgium) driving price dynamics of lower power markets (Estonia, Slovakia, Czech Republic, Latvia, Denmark, Greece, Ireland, Bulgaria), while the remaining markets appear to interact less strongly as they have low average values of both in- and out-strength. In-strength trends are mostly positive, but smaller in magnitude than the out-strength

Table 2 Measures of individual node connectivity

Market	Out-strength				In-strength				Closeness			
	Mean	SD	Trend	R^2	Mean	SD	Trend	R^2	Mean	SD	Trend	R^2
BE	2.38	0.41	−0.15	0.55	1.07	0.25	0.11	0.76	0.83	0.05	−0.02	0.43
CZ	0.74	0.16	0.06	0.56	2.24	0.15	0.01^{ns}	0.02	0.65	0.05	0.02	0.60
DK	0.77	0.10	0.01^{ns}	0.02	2.02	0.33	0.15	0.79	0.63	0.03	0.00^{ns}	0.00
DE	5.45	0.35	0.12	0.50	0.85	0.16	−0.04	0.22	0.84	0.04	−0.01	0.50
EE	0.16	0.08	-0.01^{ns}	0.08	2.25	0.40	0.03^{ns}	0.03	0.50	0.07	0.01	0.15
GR	0.30	0.19	-0.05^{ns}	0.27	0.83	0.37	0.14	0.57	0.56	0.05	0.00^{ns}	0.02
ES	0.90	0.22	0.08	0.52	1.06	0.56	0.26	0.88	0.65	0.05	0.01^{ns}	0.07
FR	0.96	0.35	0.17	0.94	0.89	0.20	0.07	0.49	0.70	0.08	0.04	0.91
IE	0.39	0.34	−0.14	0.70	0.91	0.31	0.14	0.75	0.55	0.13	−0.05	0.60
IT	0.30	0.14	0.03	0.21	0.36	0.15	0.05	0.36	0.55	0.04	0.01	0.42
LV	0.51	0.13	0.03^{ns}	0.21	1.55	0.28	−0.04	0.09	0.61	0.06	0.03	0.78
LT	1.08	0.44	0.20	0.85	1.33	0.15	-0.03^{ns}	0.17	0.70	0.05	0.02	0.72
HU	1.30	0.30	0.14	0.84	1.45	0.24	0.03^{ns}	0.05	0.73	0.08	0.03	0.84
NL	2.47	0.24	−0.08	0.42	0.92	0.10	0.04	0.49	0.87	0.04	−0.01	0.31
AT	2.71	0.30	0.11	0.60	0.97	0.16	−0.04	0.27	0.82	0.03	0.00^{ns}	0.00
PL	2.04	0.82	0.39	0.91	0.86	0.35	−0.13	0.55	0.80	0.08	0.02	0.33
PT	0.91	0.41	0.20	0.93	1.09	0.30	0.12	0.59	0.64	0.08	0.04	0.84
SI	1.29	0.34	−0.15	0.79	1.37	0.16	0.05	0.41	0.70	0.05	−0.01	0.32
SK	0.68	0.10	0.00^{ns}	0.00	2.42	0.22	0.03^{ns}	0.07	0.64	0.03	0.01	0.12
FI	0.28	0.21	0.08	0.60	0.34	0.20	0.08	0.55	0.46	0.09	0.02	0.17
SE	0.58	0.15	-0.01^{ns}	0.01	0.29	0.14	0.05	0.60	0.57	0.04	0.00^{ns}	0.04
UK	0.24	0.17	-0.04^{ns}	0.20	0.31	0.12	−0.03	0.24	0.52	0.06	−0.02	0.22
BG	0.38	0.14	−0.05	0.62	1.16	0.14	-0.01^{ns}	0.03	0.55	0.09	−0.04	0.77
RO	0.55	0.22	0.09	0.72	0.82	0.17	0.03^{ns}	0.10	0.62	0.07	0.03	0.74

trends. Table 3 summarizes key characteristics of the causal interactions between the three market groups that are related to market power; Using the temporally averaged network of price relationships, for every pair (*k, l*) of groups, *k, l* = 1, 2, 3, we compute the density, D_{kl}^{out}, of causal out-links directed from group *k* to *l,* the average out-strength from *k* to *l* per link, $<d_{kl}^{out}>_L$, and the average out-strength from *k* to *l* per *k*-group market (*k*-market*)*, $<d_{kl}^{out}>_M$, as follows:

$$D_{kl}^{out} = |\mathbf{E}_{kl}| \Big/ \big(|\mathbf{V}_k| - \mathbf{1}_{\{k=l\}}(k,l)\big)|\mathbf{V}_l| \tag{12}$$

$$\left\langle d_{kl}^{out}\right\rangle_L = \sum_{i\in\mathbf{V}_k;\, j\in\mathbf{V}_l} GCI_{i\to j} \Big/ \Big(\big(|\mathbf{V}_k| - \mathbf{1}_{\{k=l\}}(k,l)\big)|\mathbf{V}_l|\Big) \tag{13}$$

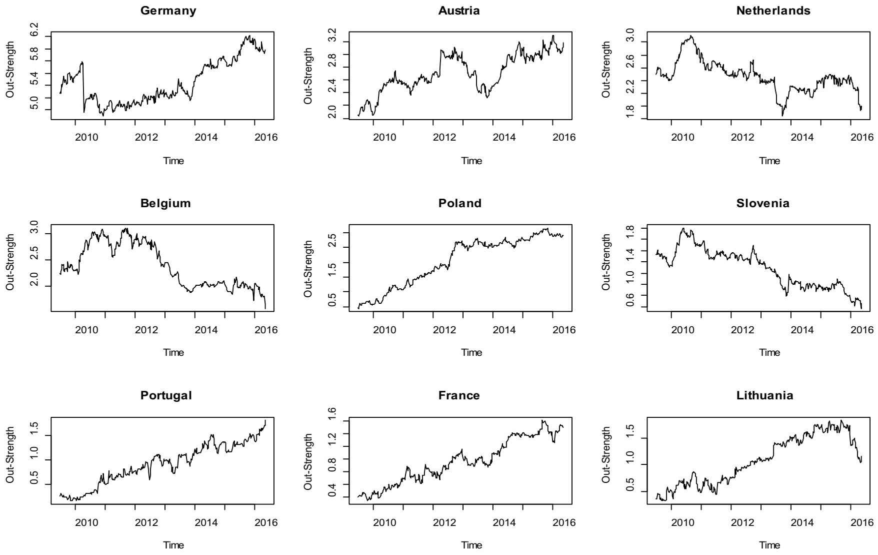


Fig. 1 Out-strength temporal evolution for selected markets

Table 3 Summary of interactions between market groups of varying market power

Market group (k)	Density, D_{kl}^{out}	Average out-strength per k-link, $<d_{kl}^{out}>_L$	Average out-strength per k-market, $<d_{kl}^{out}>_M$
$k = 1$ BE, DE, NL, AT, PL	$\mathbf{D}^{out} = \begin{bmatrix} 0.95 & 0.80 & 0.82 \\ 0.18 & 0.52 & 0.25 \\ 0.22 & 0.59 & 0.38 \end{bmatrix}$	$\langle \mathbf{d}^{out} \rangle_L = \begin{bmatrix} 0.18 & 0.15 & 0.10 \\ 0.03 & 0.07 & 0.04 \\ 0.05 & 0.09 & 0.05 \end{bmatrix}$	$\langle \mathbf{d}^{out} \rangle_M = \begin{bmatrix} 0.67 & 1.20 & 1.07 \\ 0.03 & 0.24 & 0.10 \\ 0.05 & 0.42 & 0.18 \end{bmatrix}$
$k = 2$ CZ, DK, EE, LV, SK, GR, IE, BG			
$k = 3$ ES, FR, IT, LT, HU, PT, SI, FI, SE, UK, RO			

Average strength per link	$k = 1$	$k = 2$	$k = 3$	Average strength per k-market	$k = 1$	$k = 2$	$k = 3$
Within group	0.18	0.07	0.05	Within group	0.67	0.24	0.18
Out-strength	0.15	0.03	0.08	Out-strength	2.27	0.13	0.47
In-strength	0.04	0.13	0.09	In-strength	0.16	1.33	0.56

$$\left\langle \mathrm{d}_{kl}^{out} \right\rangle_M = \sum_{i \in \mathbf{V}_k;\, j \in \mathbf{V}_l} GCI_{i \to j} \Big/ |\mathbf{V}_k| \tag{14}$$

$\mathbf{E}_{kl}$ denotes the set of links directed from k to l, $\mathbf{V}_k$ the set of k-group markets, $\mathbf{1}_{\{k=l\}}(k, l)$ an indicator function assuming value 1 if $k = l$ and 0 otherwise. D_{kl}^{out} quantifies the degree of "outward-directed" connectivity from market group k to l; values closer to 1 indicate a high degree of causal flows from k to l. $<\mathrm{d}_{kl}^{out}>_L$ and $<\mathrm{d}_{kl}^{out}>_M$ are complementary strength measures of the average influence market groups exert to each other, normalized by group sizes. The diagonal elements of the matrices correspond to within group connectivity and average interaction strengths. The lower part of Table 3 provides for each market group the average strengths (both per link and per k-market) of outward-directed (out-strength) and inward-directed (in-strength) causal price linkages; Group 1 markets are the most strongly connected, both with each other ($\mathrm{D}_{11}^{out} = 0.95$) and with other groups ($\mathrm{D}_{12}^{out} = 0.80$, $\mathrm{D}_{13}^{out} = 0.82$). They interact strongly with each other ($<\mathrm{d}_{11}^{out}>_L = 0.18$, $<\mathrm{d}_{11}^{out}>_M = 0.67$), exert the highest average influence on other groups (average out-strength per link is 0.15 and per 1-market is 2.27) and are not significantly

influenced by other groups (average in-strength per link is 0.04 and per k-market is 0.16). Prices in group 2 markets are driven by 1-markets ($<\mathrm{d}_{12}^{out}>_L = 0.15$, $<\mathrm{d}_{12}^{out}>_M = 1.20$), and to a lesser degree by 3-markets ($<\mathrm{d}_{32}^{out}>_L = 0.09$, $<\mathrm{d}_{32}^{out}>_M = 0.42$). 2-markets have on average the least causal influence on other groups. Group 3 market prices are strongly influenced by 1-markets and influence 2-markets. On average, inwards and outwards directed price interactions for 3-markets tend to balance out as their average magnitudes are 0.08 (0.47) and 0.09 (0.56) per link (per market), respectively.

It is also worth noting the evolution of interaction strengths in some markets; Over the observation period, Belgium and Ireland appear to have lost power (negative out- and positive in-strength trends), while Denmark and Spain show an increasing exposure to external price shocks (both have dominant-negative out-strength trends). On the other hand, the power of Germany and Poland has increased over time (the latter experienced a strong positive out-strength trend with a sizeable negative in-strength trend).

The markets with the most central role in the network of price causality flows, those with the highest closeness centrality, include markets with high values of interaction strengths and appear to be spatially located in central Europe and to be contiguous; Netherlands, Germany, Belgium, Austria, Poland, Hungary, France, Slovenia, but also Lithuania (average values from 0.7 to 0.87). Markets spatially located to the periphery of Europe have smaller closeness centralities (below 0.60, e.g., Finland, Estonia, UK, Bulgaria, Italy, Ireland, Greece, and Sweden). Linear time trends in centrality appear mixed in sign and small in magnitude.

4.2 Measures of Global Network Cohesiveness and Their Evolution

Summaries about the estimated distributions of network cohesiveness measures over all 5-years rolling windows are given in Table 4. Again, as a rough indicator for their temporal evolution, we estimated a linear trend and the corresponding R^2. All trends are significant ($\alpha = 0.05$, HAC corrected). HAC consistent generalized M-fluctuation tests (e.g., see [29]) detected the presence of breaks in all four series. The time series of the rolling windows estimates are plotted in Fig. 2. The estimated break points are shown with dashed lines.

The estimated values of the cohesiveness measures and their temporal evolution indicate that the total market inter-connectedness increases over time as the density (proportion of connected market pairs over the total) increases from around 0.40 to a plateau near 0.51. At the same time, the interaction strengths between markets also increase on average, by about 40 % from 1.79 to 2.61. The average shortest path length, as might be expected, exhibits reversed time trends relative to the density and average strength, and its value shows an overall decrease (from about 1.68 to a plateau around 1.55), consistent with a shortening of the distance between markets and a more efficient, faster system-wise spread of price shocks. Reciprocity, the proportion of price links that are mutual (two-way causal), is fluctuating initially (mid-2009 to mid-2013) from 0.20 to 0.30, with a local average of about 0.25, and then increases rapidly to reach 0.39 at its maximum. Overall, the evidence points toward higher levels of market integration.

The number of identified breaks is five for average strength and four for the other measures. The first break occurs almost simultaneously for all measures in June/July 2010. It coincides with the leveling of a rapid increase phase for network density, average strength, and reciprocity that occurs together with a rapid decline of average shortest path length. The second, third, and fourth breaks for density, average strength, and average shortest path length occur very close to one another around the start of 2012, 2013, and 2014, respectively. The third break for reciprocity occurs also around the start of 2014, and the last one in the beginning of 2015, shortly before the last

Table 4 Measures of network cohesiveness

Statistics	Density	Average strength	Average shortest path length	Reciprocity
Min	0.40	1.79	1.50	0.20
Mean	0.47	2.28	1.59	0.28
Median	0.47	2.28	1.58	0.26
Max	0.51	2.61	1.72	0.39
Std. dev.	0.03	0.20	0.04	0.05
Skewness	−0.69	−0.51	0.84	0.61
Trend	0.01	0.09	−0.01	0.02
R^2	0.54	0.78	0.22	0.56

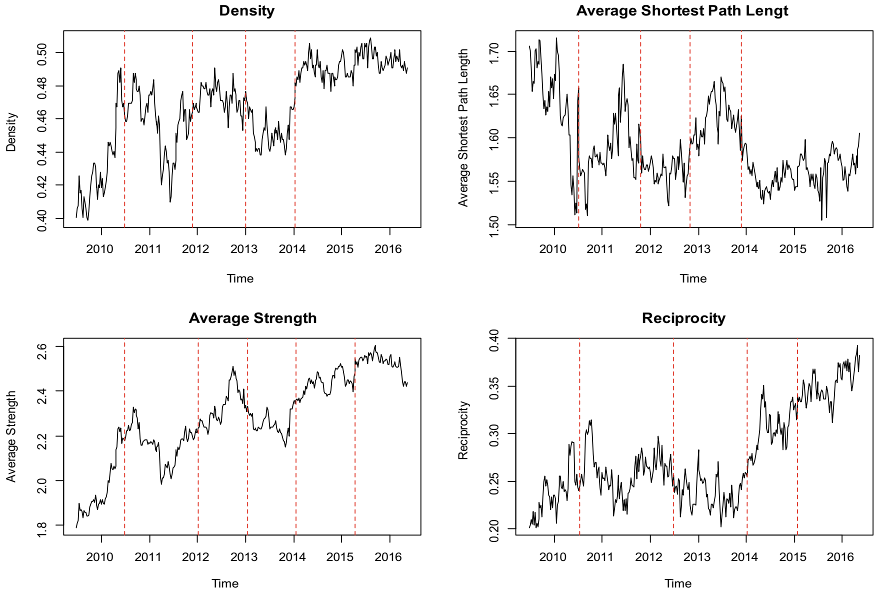


Fig. 2 Temporal evolution of estimated network cohesiveness measures

estimated break for average strength. It seems that there is a considerable degree of consistency in the appearance of breaks in the four network cohesiveness measures. These empirical findings provide evidence that the integration process in the EU pig meat market during the study period is rather a staged process, characterized by regimes within which the network of price relationships has distinct structural characteristics.

5 Conclusion

We present a part of an on-going research project on the study of price linkages between spatially separated primary commodity markets in Europe. Our analysis focuses on some structural aspects of temporally evolving complex networks of causal relationships in wholesale pig meat prices. The networks are constructed with the use of GAM-based non-linear models for testing and quantifying the strength and directionality of causal price relationships in the Granger sense. Simulation results indicate that the GAM-test has clearly better finite sample properties than the linear OLS test and the most widely used non-linear non-causality tests of Hiemstra-Jones and Diks-Panchenko.

The application of network analysis methods provides insights about key characteristics of the complex system of price interactions in the common EU market; Measures of individual connectedness are used to identify groups of markets with high power which have been leading the price transmission process during the studied period. Temporal analysis of these measures also offers insights on the changing role of individual markets in the price transmission process. The data provide evidence not only for a large degree of heterogeneity in market power between countries, but also for the existence of market segregation into high and low power groups (clubs) that are strongly connected to each other. The presence of such groups is an inefficiency of the market system in the European Union.

The analysis of system-level measures of cohesiveness sheds some light into the aggregate market integration process; Results are suggestive of temporal increase in (a) system inter-connectedness, (b) overall strength of price interactions, and (c) prevalence of bi-directional price relationships. Also, the length of price transmission paths connecting markets together had been decreasing over time. However slow these changes may be, they all point to an increasing degree of market integration in the EU pig meat market. The presence and timings of breaks in the structural network connectivity measures are indicative of a staged integration process that deserves to be studied further in order to identify possible driving market mechanisms.

References

1. Anagnostidis, P., Emmanouilides, C.J.: Nonlinearity in high-frequency stock returns: evidence from the Athens stock exchange. Phys. A **421**, 473–487 (2015)
2. Baumöhl, E., Kočenda, E., Lyócsa, S., Výrost, T.: Networks of volatility spillovers among stock markets. Phys. A **490**, 1555–1574 (2018)
3. Benjamini, Y., Hochberg, Y.: Controlling the false discovery rate: a practical and powerful approach to multiple testing. J. Roy. Stat. Soc. B **57**(1), 289–300 (1995)
4. Brock, W.A., Dechert, W.D., Scheinkman, J.A.: A test for independence based on the correlation dimension. SSRI Working Paper 8702, University of Wisconsin-Madison (1987)
5. Bu, H., Tang, W., Wu, J.: Time-varying co-movement and changes of co-movement structure in the Chinese stock market: a causal network method. Econ. Mod. **81**, 181–204 (2019)
6. Diks, C., Panchenko, V.: A new statistic and practical guidelines for nonparametric Granger causality testing. J. Econ. Dyn. Control **30**(9), 1647–1669 (2006)
7. Emmanouilides, C.J., Fousekis, P.: Testing for the LOP under non-linearity: an application to four major EU pork markets. Agric. Econ. **43**(6), 715–723 (2012)
8. Emmanouilides, C.J., Fousekis, P., Grigoriadis, V.: Price dependence in the olive oil markets of the Mediterranean. Span. J. Agricu. Res. **12**(1), 3–14 (2014)
9. Emmanouilides, C.J., Fousekis, P.: Assessing the validity of the LOP in the EU broiler markets. Agribus. Int. J. **31**(1), 33–46 (2015)
10. Emmanouilides, C.J., Proskynitopoulos A.: Spatial integration of agricultural markets in the EU: complex network analysis of non-linear price relationships in hog markets. In: Valenzuela, O., Rojas, F., Pomares, H., Rojas, I. (eds.) ITISE 2019, International Conference on Time Series and Forecasting, Proceedings of Papers, vol. 1, pp. 634–645 (2019)
11. Engle, R.F.: Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. Econometrica **50**(4), 987–1007 (1982)
12. Grigoriadis, V., Emmanouilides, C.J., Fousekis, P.: The integration of pigmeat markets in the EU. Evidence from a regular mixed Vine Copula. Rev. Agricul. Appl. Econ. **19**(1), 3–12 (2016)
13. Geweke, J.: Measurement of linear dependence and feedback between multiple time series. J. Am. Stat. Assoc. **77**(378), 304–313 (1982)
14. Hastie, T., Tibshirani, R.J.: Generalized additive models. Stat. Sci. **1**(3), 297–310 (1986)

15. Hastie, T., Tibshirani, R.J.: Generalized Additive Models. Monographs on Statistics and Applied Probability, vol. 43. CRC Press, London (1990)
16. Hiemstra, C., Jones, J.D.: Testing for linear and nonlinear Granger causality in the stock price-volume relation. J. Fin. **49**(5), 1639–1664 (1994)
17. Hlavăckova-Schindler, K., Paluš, M., Vejmelka, M., Bhattacharya, D.: Causality detection based on information-theoretic approaches in time series analysis. Phys. Rep. **441**, 1–46 (2007)
18. Hsieh, D.: Chaos and nonlinear dynamics: application to financial markets. J. Fin. **46**(5), 1839–1877 (1991)
19. Lee, T.H., White, H., Granger, C.W.J.: Testing for neglected nonlinearity in time series models: a comparison of neural network methods and alternative tests. J. Economet. **56**(3), 269–290 (1993)
20. Ljung, G.M., Box, G.E.P.: On a measure of a lack of fit in time series models. Biometrika **65**(2), 297–303 (1978)
21. Lloyd, C.J.: Estimating test power adjusted for size. J. Stat. Comput. Simul. **75**(11), 921–933 (2005)
22. Nason, G.P.: A test for second-order stationarity and approximate confidence intervals for localized autocovariances for locally stationary time series. J. Roy. Stat. Soc. B **75**, 879–904 (2013)
23. Péguin-Feissolle, A., Teräsvirta, T.: A general framework for testing the Granger noncausality hypothesis. SSE/EFI Working Paper Series in Economics and Finance 343, Stockholm School of Economics (1999)
24. Politis, D.N., Romano, J.P.: The stationary bootstrap. J. Am. Stat. Assoc. **89**(428), 1303–1313 (1994)
25. Serra, T., Gil, J., Goodwin, B.: Local polynomial fitting and spatial price relationship: price transmission in EU pork markets. Eur. Rev. Agric. Econ. **33**, 415–436 (2006)
26. Výrost, T., Lyócsa, S., Baumöhl, E.: Granger causality stock market networks: temporal proximity and preferential attachment. Phys. A **427**, 262–276 (2015)
27. Wood, S.N., Pya, N., Saefken, B.: Smoothing parameter and model selection for general smooth models (with discussion). J. Am. Stat. Assoc. **111**, 1548–1575 (2016)
28. Wood, S.N.: Generalized Additive Models: An Introduction with R. Chapman-Hall, Boca Raton (2017)
29. Zeileis, A., Hornik, K.: Generalized M-fluctuation tests for parameter instability. Stat. Neerl. **61**, 488–508 (2007)

Comparative Study of Models for Forecasting Nigerian Stock Exchange Market Capitalization

Isah Nura, Sani I. S. Doguwa, and Yusuf Basiru

Abstract This paper proposes two forecasting models for the Nigerian Stock Exchange Market Capitalization using the Autoregressive Integrated Moving Average (ARIMA) process and an Autoregressive Distributed Lag (ARDL) process. A better model was selected by comparing the forecast evaluation for the estimated models using pseudo out-of-sample forecasting procedure over 2016q2 to 2019q1. The statistical loss functions MAE_t, $RMSE_t$ and $MAPE_t$ for the t forecast horizon (t = 1, 2, …, 12) are used to compare the forecast performance of the two estimated models. The results show that ARIMA model outperforms ARDL model in three to four quarters forecast horizon. On the other hand, ARDL model outperforms ARIMA in one to two quarters, five to seven quarters as well as nine to twelve quarters forecast horizon. Therefore, in forecasting Nigerian Stock Exchange Market Capitalization in both short and long horizons, it can be concluded that ARDL is better model to be used.

Keywords Nigerian stock exchange market capitalization · ARIMA · ARDL · Statistical loss functions

1 Introduction

Time series modeling is a dynamic research area which has attracted attention of researcher's community over the last few decades. Time series forecasting thus can be termed as the act of predicting the future by understanding the past. It is applicable

I. Nura (✉) · Y. Basiru
Department of Statistics, College of Science and Technology, Jigawa State Polytechnic, Dutse, Nigeria
e-mail: nuraisagm@gmail.com

Y. Basiru
e-mail: byusuf@jigpoly.edu.ng

S. I. S. Doguwa
Department of Statistics, Ahmadu Bello University, Zaria, Nigeria
e-mail: sidoguwa@gmail.com

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_7

to the fields such as business, economic, finance, sciences, engineering, and so on. It is obvious that a successful time series forecasting depends on an appropriate model fitting. A lot of efforts have been made by researchers over the years for the development of efficient models to improve the forecasting accuracy. As a result, various important time series forecasting models have evolved in the literature. Some of the most popular and frequently used stochastic time series models are ARIMA, Vector Autoregressive (VAR) models, and ARDL.

Kapil and Hanuman [7] define "Market Capitalization" of a company as a current market price per stock multiplied by the number of outstanding shares. Market Capitalization is the universal benchmark for quantifying the value of a giving firm. Capital structure is an essential part of economic growth and development, and it plays a significant role in the economic premise of manufacturing and distribution. This implies that capital growth may facilitate faster rate of economic development. The growth of any stock market is measured by its total market capitalization. An Investor invests in financial securities for competitive and satisfactory returns. Before making any investment decision the generally considers the ex-post and ex-ante returns of the securities. This is because the investment in financial assets is always associated with different types of risks which are associated with different factors. The objective of this paper is to construct models of the Nigerian Stock Exchange Market Capitalization (NSEC) using the ARIMA process and ARDL process and evaluate the pseudo out-of-sample forecast performance of these models using some statistical loss functions to choose the best model for forecasting NSEC. The contribution of this paper are (a) construction/section of the best model for forecasting NSEC between ARIMA and ARDL (b) analyzing the NSEC involves four important variables real GDP (RGDP), official exchange rate (Naira to Dollar), Money Supply and Liquidity ratio (c) analyzing the prediction accuracy based on pseudo out-of-sample forecast technique on MAE, RMSE, and MAPE methods.

This paper is sectioned as follows: the introduction and overview of the study are presented in the first section, Sect. 2 provides the related works (literature review) that highlights empirical studies of forecast performance. Section 3 presents the methodology and material used in the study, while empirical results, model fitting, and performance evaluation of NSC are presented in Sect. 4, and finally, Sect. 5 concludes on the findings.

2 Literature Review

From the empirical studies, a number of studies have applied different methodologies to assess the forecast performance of different statistical modes among others: Jansen and Wang [6] investigate the forecasting performance of the error correction model using co-integration test and non-linear error correction model using equity yield on the 500 indices; the result shows that non-linear vector error correction model outperforms its linear version, based on ten years forecast horizon. Adebiyi et al. [1] examine the different types of inflation forecasting models using ARIMA, VAR, and Vector Error Correction Models (VECM). The empirical results show that ARIMA models modestly explain inflation dynamics in Nigeria. Iqbal and Uddin [10] compare the forecast performance of ARIMA, VAR, and VEC model, using some macroeconomic variables of USA, UK, and the G-7 Countries. The results in short-term forecast performance show that ARIMA and VAR models are superior, while in long term forecast, ECM outperformed other techniques. For the co-integration technique, ARDL is superior in forecasting performance. Taiwo and Olatayo [9] investigate the relationship between some Nigerian economic variables (Government Revenue and Expenditure, Inflation Rates, and Investment) and examine the forecast performance of VAR model and Time series regression model. The model and forecast performance is measured using Root Mean Square Forecast Error (*RMSFE*) and Mean Absolute Percentage Forecast Error (*MAPFE*). The result indicates that VAR model is better than time series regression model. Prapanna et al. [11] studied the effectiveness of autoregressive integrated moving average (ARIMA) model in forecasting stock prices of fifty-six Indian stocks; the research concluded to have found ARIMA 85% accurate for the prediction of all the stocks for sectors taken at a time. The research, however, uncovered that for specific sector independently, ARIMA model for fast-moving consumer goods (FMCG) outperform other sectors while that of banking and automobile sector had a lower prediction accuracy to other sectors. The forecast values from the *VAR* model are more realistic and closely reflect the current economic reality in Nigeria. Doguwa and Alade [4], employed the three short-term forecasting models for the adjusted external reserves using the seasonal autoregressive integrated moving average (SARIMA), seasonal autoregressive integrated moving average with an exogenous input (SARIMA-X), and an autoregressive distributed lag (ARDL) processes, using the pseudo out-of-sample forecasting procedure.

3 Methodology/Material

In this research two different statistical models are considered, namely ARIMA model and ARDL. Both ARIMA and ARDL models are estimated with data on Nigerian Stock Market Capitalization (NSEC), and other exogenous variables such as Real Gross Domestic Product (RGDP), official Exchange Rate of Naira to Dollar (EXR), Money Supply (MS), and liquidity Ratio (LR) in the case of ARDL model. The sample for the estimation and forecast evaluation spans the period from 1985q1 to 2019q1 and is divided into two parts. The first part is the training sample from 1985q1 to 2016q1 and the second part is the forecasting sample from 2016q2 to 2019q1 for forecast evaluation.

3.1 ARIMA Process

One of the methods of analysis adopted in this study is the Box-Jenkins (ARIMA) which is an econometric model used to forecast without considering other independent variables in making forecast.

An ARIMA (p, d, q) can be defined as follows:

$$\emptyset(L)(1-L)^{d} y_t = \theta(L) e_t \tag{1}$$

where L is the standard backward shift operator, function Ø, and θ are the standard autoregressive (AR) and moving average (MA) polynomial of order p and q.

$$\emptyset(L) = 1 - \emptyset_1 L - \emptyset_2 L^2 - \ldots - \phi_p L^p$$

$$\theta(L) = 1 + \theta_1 L + \theta_2 L^2 + \ldots + \theta_q L^q.$$

And d is the degree of differencing. d = 1 => the time series differencing once, d = 2 => the time series difference twice.

Also, as an illustration, the ARIMA (1, 1, 0) is of the form:

$$(1 - \emptyset_1 L)\Delta y_t = e_t \tag{2}$$

And the ARIMA (1, 1, 1) model is of the form:

$$(1 - \emptyset_1 L)\Delta y_t = (1 + \theta_1 L) e_t \tag{3}$$

Since d = 1

$$(1 - L) y_t = y_t - y_{t-1} = \Delta y_t$$

Using the properties of operator L, it follows that (3) are be expressed as follows:

$$\Delta y_t = \emptyset_1 \Delta y_{t-1} + \theta_1 e_{t-1} + e_t \tag{4}$$

where Δ is the difference operator. Also, d is order of integration and e_t is a Gausian white noise with zero mean and constant variance. The details of ARIMA modeling procedure are contained in Box and Jenkins [2]. For the NSEC series under study, the estimates of the parameters which meet the stationarity conditions are obtained using the Eviews software.

3.2 The ARDL Process

Pesaran et al. [8] proposed Autoregressive Distributed Lag (ARDL) approach to identify whether a long-run relationship exist, irrespective of whether the variables are I(0), I(1) or a combination of both. In such situation, the application of ARDL technique to co-integration will provide realistic and efficient estimates. Using bound test approach, the ARDL (p, q_1, q_2, q_3, q_4) representation of the statistical model of LNSEC is specified as

$$\begin{aligned} \mathrm{d}(LNSEC_t) &= \beta_1 + \beta_2\big(LRGDP_{t-1}\big) + \beta_3\big(EXR_{t-1}\big) + \beta_4\big(LR_{t-1}\big) + \beta_5(LMS)_{t-1} \\ &\quad + \sum\nolimits_{i=1}^{p} \gamma_{1i}\Delta LNSEC_{t-j} + \sum\nolimits_{j=0}^{q_1-1} \gamma_{2j}\Delta RGDP_{t-j} + \sum\nolimits_{j=0}^{q_2-1} \gamma_{3j}\Delta EXR_{t-j} \\ &\quad + \sum\nolimits_{j=1}^{q_3-1} \gamma_{4j}\Delta LR_{t-j} + \sum\nolimits_{j=1}^{q_4-1} \gamma_{5j}\Delta LMS_{t-j} + \gamma EC_{t-1} + e_t \end{aligned} \tag{5}$$

where β_1 is a constant, $\beta_2, \beta_3 \ldots \beta_7 \ldots$ are the long-run parameters of the model, and $\gamma_{1j}, \gamma_{2j} \ldots \gamma_{7j}$ are the short-run co-efficients. The error term is expected to be white noise. The letters p, $q_1, q_2, \ldots q_4$ are the optimum lag lengths that define the ARDL (p, $q_1, q_2, \ldots q_4$) model. The ARDL bound test for no co-integration among the variables against the presence of co-integration involves testing the null hypotheses of the absence of co-integration:

$$\mathrm{Ho} : \beta_2 = \beta_3 = \ldots = \beta_5 = \ 0\mathrm{VSH}_1 : \beta_2 \neq \beta_3 \neq \ldots \neq \beta_5 \neq 0$$

The hypothesis is tested by using F-statistic (Wald test). The distribution of this F-statistics is non-standard, irrespective of whether the variables in the system are I(0) or I(1) or both. If the computed F-statistic falls outside the bound, a conclusive decision can be made, without the need to know whether the variables are I(0) or I(1). That is, when the computed F-statistic is greater than the upper critical value, the H_0 is rejected (the variables are co-integrated). If the F-statistic is below the lower critical value, then H_0 cannot be rejected (i.e., there is no co-integration between the variables).

The long-run model is represented as

$$(LNSEC_t) = \beta_1 + \beta_2(LGDP_{t-1}) + \beta_3(EXR_{t-1}) + \beta_4(LR_{t-1}) + \beta_5(LMS)_{t-1} + e_t \tag{6}$$

Once the presence of co-integration exists, an appropriate distributed lag error correction model of equation is specified as follows:

$$\begin{aligned} d(LNSEC_t) &= \sum\nolimits_{i=1}^{p} \gamma_{1i}\Delta LNSEC_{t-j} + \sum\nolimits_{j=0}^{q_{1-1}} \gamma_{2j}\Delta RGDP_{t-j} + \sum\nolimits_{j=0}^{q_2-1} \gamma_{3j}\Delta EXR_{t-j} \\ &\quad + \sum\nolimits_{j=1}^{q_3-1} \gamma_{4j}\Delta LR_{t-j} + \sum\nolimits_{j=1}^{q_4-1} \gamma_{5j}\Delta LMS_{t-j} + \gamma EC_{t-1} + e_t \end{aligned} \tag{7}$$

where Δ the first is difference operator and $\gamma_{1j}, \gamma_{2j} \ldots \gamma_{7j}$ are the short-run co-efficient of the dynamic model, and γEC_{t-1} measures the speed of adjustment.

3.3 Test of Adequacy of Fitted Model

To test the adequacy of any selected model, residual analysis is employed. Jacque-Bera normality test, Breusch-Godfrey serial correlation LM test (BG LM F-statistic), and autoregressive conditional heteroskedasticity (ARCH LM) test would be adopted.

To test the normality of the residual, Jarque-Bera normality test procedure is applied; which sets the null hypothesis that the residuals are normally distributed. The null hypothesis would be rejected if the p-value is less than the significant level and can be concluded that the residuals are not normally distributed.

Evidence of serial correlation is past investigated on the residuals before using the parsimonious model for statistical inference. The Breusch-Godfrey serial correlation LM test (BG LM F-statistic) is used to test the null hypothesis of no serial correlation in the residuals up to a specific order.

Autoregressive conditional heteroskedasticity effects in the residuals are investigated. ARCH LM test is employed and tests the null hypothesis that there is no autoregressive conditional heteroskedasticity effect in the residuals. Accepting the null hypothesis will indicate that there is no ARCH effect in the residuals.

3.4 Performance Evaluation

The precision of the various forecasts model will be determined by selecting the most appropriate model by the use of some measures of performance, Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE), defined as

$$MAE = \frac{1}{h}\sum\nolimits_{t=1}^{h} \left|\hat{x}_t - x_t\right| \tag{8}$$

$$RMSE = \sqrt{\frac{1}{h}\sum\nolimits_{t=1}^{h} (\hat{X}_t - X_t)^2} \tag{9}$$

$$MAPE = \frac{1}{h}\sum\nolimits_{t=1}^{h} \left|\frac{\widehat{100}(X_t - X_t)}{\hat{X}_t}\right| \tag{10}$$

where $\hat{X}_t$ is the predicted value at time t and X_t is the actual value at time t, respectively.

The smaller the value of *RMSE*, *MAE,* and *MAPE*, the better is the forecasting performance of the model.

4 Results

4.1 Exploratory Data Analysis

The quarterly trend for the actual data of NSEC at end period was illustrated in Fig. 1, the line graph indicates that there is no stationarity in the time series data.

The line graph for LOG(NSEC) indicates that the nature of the trends was improved after taking the log transformation of the variable compare to Fig. 2.

4.2 Unit Roots Test

A stationary series must be obtained before it can be used to specify and estimate a model. The unit roots test will help us to determine the stationary of a series. The Augmented Dickey–Fuller test was used to test the stationary of the Log(NSEC), RGDP, EXR, Log(MS), and LR series. The unit root test results are presented in Table 1. The results indicate that Log(NSEC), EXR, and log(LMS) are non-stationary at level but stationary after taking a first difference, while RGDP and LR are stationary at level. The decision rule states that if p-value is greater than the significant level leading to non-rejection of the null hypothesis, otherwise rejects the null hypothesis,

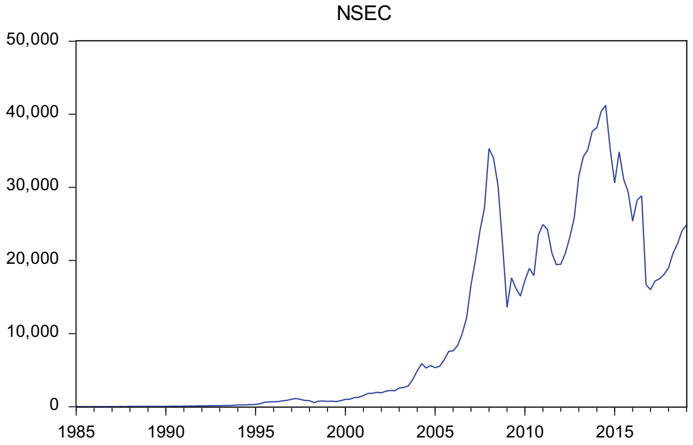


Fig. 1 Line graph for NSEC

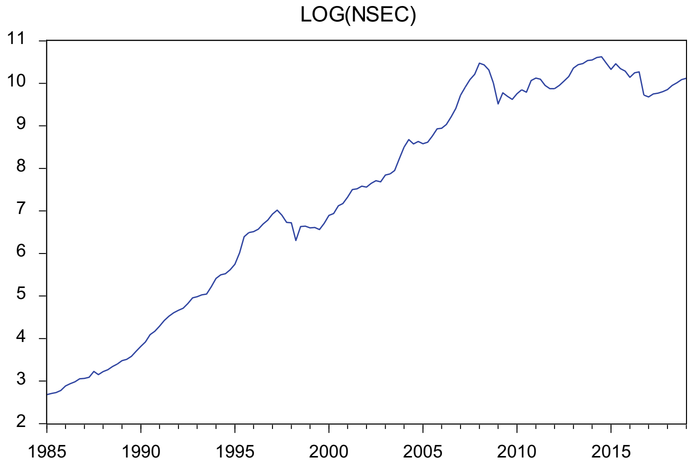


Fig. 2 Line graph for LOG(NSEC)

Table 1 Augmented Dickey–Fuller unit root test

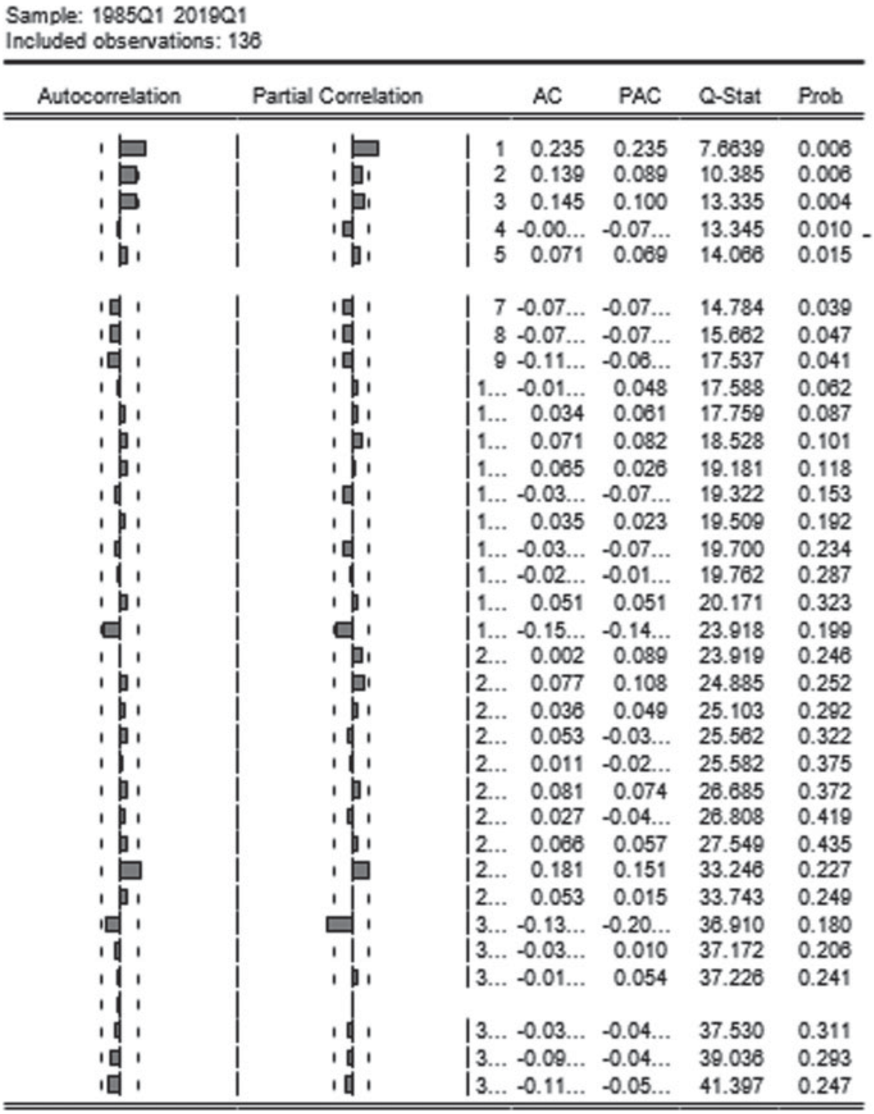
Variables	At level		At first differencing		Order of integration
	ADF	*p*-value	ADF	*p*-value	
Log(NSEC)	−1.8935	0.3346	−9.0786	0.0000***	1(1)
RGDP	−2.9342	0.0442**	−7.8845	0.0000***	I(0)
EXR	−0.9404	0.6894	−11.5909	0.0012***	I(1)
LR	−11.5903	0.0000***	−7.8532	0.0000***	I(0)
Log(MS)	−0.9012	0.7853	−8.5173	0.0000***	I(1)

Note ***and ** indicate the variables are significant at 1% and 5% significant level, at automatic maximum lags of 12

hence all the three variables are integrated at order one 1(1) and remaining two variables are stationary at I(0).

4.3 The ARIMA Process Result

Firstly, by plotting the correlogram of the stationary Log(NSEC), that is $\Delta Log(NSEC)$, the patterns of the ACF and PACF could be observed, and the value of parameter *p* and *q* for ARIMA model would be determined. From Fig. 3, the graph indicates that the ACF(MA) and PACF(AR) all died out after lag 1. Thus, the *p* and *q* values for the ARIMA (*p*, 1, *q*) model were set at 2, respectively. From different possible ARIMA combinations, the Akaike Information Criterion (AIC) is used to select the most desirable ARIMA model for Δ Log(NSEC).



Sample: 1985Q1 2019Q1
Included observations: 136

Autocorrelation	Partial Correlation		AC	PAC	Q-Stat	Prob
		1	0.235	0.235	7.6639	0.006
		2	0.139	0.089	10.385	0.006
		3	0.145	0.100	13.335	0.004
		4	-0.00...	-0.07...	13.345	0.010
		5	0.071	0.069	14.066	0.015
		7	-0.07...	-0.07...	14.784	0.039
		8	-0.07...	-0.07...	15.662	0.047
		9	-0.11...	-0.06...	17.537	0.041
		1...	-0.01...	0.048	17.588	0.062
		1...	0.034	0.061	17.759	0.087
		1...	0.071	0.082	18.528	0.101
		1...	0.065	0.026	19.181	0.118
		1...	-0.03...	-0.07...	19.322	0.153
		1...	0.035	0.023	19.509	0.192
		1...	-0.03...	-0.07...	19.700	0.234
		1...	-0.02...	-0.01...	19.762	0.287
		1...	0.051	0.051	20.171	0.323
		1...	-0.15...	-0.14...	23.918	0.199
		2...	0.002	0.089	23.919	0.246
		2...	0.077	0.108	24.885	0.252
		2...	0.036	0.049	25.103	0.292
		2...	0.053	-0.03...	25.562	0.322
		2...	0.011	-0.02...	25.582	0.375
		2...	0.081	0.074	26.685	0.372
		2...	0.027	-0.04...	26.808	0.419
		2...	0.066	0.057	27.549	0.435
		2...	0.181	0.151	33.246	0.227
		2...	0.053	0.015	33.743	0.249
		3...	-0.13...	-0.20...	36.910	0.180
		3...	-0.03...	0.010	37.172	0.206
		3...	-0.01...	0.054	37.226	0.241
		3...	-0.03...	-0.04...	37.530	0.311
		3...	-0.09...	-0.04...	39.036	0.293
		3...	-0.11...	-0.05...	41.397	0.247

Fig. 3 ACF and PACF plot of dlog(NSEC)

The results from Table 2 show that ARIMA (1, 1, 0) is preferred to other models since it has the lowest values of AIC.

After identifying the ARIMA model, the next step was to estimate the parameter co-efficients. The parameter estimation of the model was conducted using classical least square method.

Table 3 shows the estimated result. Of ARIMA (1, 1, 0), the results indicate that the co-efficients of AR (1) were significant at 5% levels. Again, based on parsimonious

Table 2 Postulated model and performance evaluation

Model	AIC	Adj. R^2	No. of sig. terms
1, 1, 1	−1.2978	0.0539	2 terms are significant
1, 1, 0	−1.2992	0.0611	All 2 terms are significant
0, 1, 1	−1.2931	0.0402	All 2 term are significant
2, 1, 1	−1.2766	0.0542	Only 1 term is significant
1, 1, 2	−1.2886	0.0581	Only 1 term is significant
0, 1, 2	−1.2961	0.0594	2 terms are significant
2, 1, 2	−1.2913	0.0753	3 terms are significant

Table 3 Estimation of parameter of ARIMA (1, 1, 0)

Dependent variable: D(LNSEC)			
Estimated models			
Parameters	Co-efficients	Stand error	Prob.
C	0.059535	0.017020	0.0007
$Ø_1$	0.247911	0.049846	0.0000
Adjusted R-squared	0.0611		
AIC	−1.2992		
Jarque-Bera normality test	2.4737		0.2903
ARCH LM test			
LM test	1.1911		0.3075

ARIMA (1, 1, 0) was selected for forecasting NSEC. From the *t*-statistics for the co-efficient variables AR (1) in Table 3, the null hypothesis said that the co-efficients are equal to zero is rejected. Thus, the model equation was presented as

$$\Delta(LNSEC_t) = 0.059535 + 0.247911\Delta(LNSEC_{t-1}) \tag{11}$$

To verify the suitability of the model, we plot the histogram and the ACF and PACF of the residuals. From the graph, it indicates that there is no spike at any lag indicating that all the residual autocorrelations are not significantly different from zero, at various lags points. Moreover, the test normality using Jarque-Bera shows that the NSEC is normally distributed at 10% significant level with Jaque-Bera statistics of 2.4737 an asymptotic probability of 0.2903.

4.4 The ARDL Model Result

The unit root test conducted on the variables suggests the use of the ARDL approach to estimate the parameters of the model. Using quarterly data from 1985q1 to 2013q3, the estimated order of an ARDL ($p, \boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{q}_3, \boldsymbol{q}_4$) model in the seven variables (LNSEC,

Table 4 Estimated of long-run co-efficient

Long-run co-efficient			
Variables	Co-efficient	Std. error	*P*-value
RGDP	−0.005115	0.020114	0.7997
EXR	0.009927	0.005330	0.0653
LR	−0.009100	0.008666	0.2960
LOG(MS)	0.880536	0.152576	0.0000
C	0.608767	0.753805	0.4211

Table 5 Estimated error correction model

Dependent variable: D(LNSEC)			
Estimated model	ARDL 40203		
Variables	Co-efficients	Stand error	Prob.
DLOG(NSEC(-1))	0.1262	0.0908	0.1626
DLOG(NSEC(-2))	0.1795	0.0883	0.0453
DLOG(NSEC(-3))	0.2078	0.0928	0.0255
D(RGDP)	−0.0017	0.0033	0.8014
D(EXR)	−0.0056	0.0028	0.0002
D(EXR(-1))	−0.004	0.0021	0.0198
D(LR)	−0.0023	0.0013	0.2645
DLOG(MS)	0.3991	0.2515	0.1158
DLOG(MS(-1))	−0.0541	0.358	0.8814
DLOG(MS(-2))	0.3944	0.252	0.1203
CointEq(-1)	−0.1524	0.0407	0.0003
Adjusted R-squared	0.3325		
AIC	−1.426		
F-statistics 3.9525			

LRGDP, INF, LEXR, MLRC, LR, and LMS) were selected using automatic searching across the 62,500 ARDL models, this result in the choice of ARDL (4, 0, 2, 0, 3) specification for D(LNSEC) with estimates of the levels relationships given by

$$(LNSEC_t) = -7.83 + 2.4(RGDP) + 0.52(EXR) - 0.04(MLRC) \\ (2.352) \qquad (3.261) \qquad (-2.299) \tag{12}$$

From Table 4, indicates that, EXR and Log(MS) have a significant impact to NSEC, and RGDP and LR are insignificant to NSEC. In other words, EXR is positively related to NSEC. Any 1-unit increase of EXR will induce 0.009927 unit of NSEC. Moreover, MS is positively related to NSEC, where by 1-unit increase in MS will induce 0.880536 unit of the NSEC.

Table 5, the result of bound test indicates that the value of F-statistics is 3.952451

Table 6 Breusch-Godfrey serial correlation LM test

F-statistic	0.023584	Prob. F(2, 89)	0.9767
Obs * R-squared	0.058797	Prob. Chi-square(2)	0.971

which is greater than upper critical value of 3.52 at 5% significant level, the H_0 is rejected (the variables are co-integrated).

According to Engle and Granger [5], when variables are co-integrated there must be an Error Correction Mode (ECM) which will describe the short-run relationship of co-integrated variables toward their equilibrium values. The result of the ECM was presented in Table 5, the one lagged of error term is negative and highly significant at 1% significant level, the co-efficient of −0.151984 indicates that about 15.20% disequilibrium is corrected on quarterly bases by change in NSEC.

The short-run equation (VECM) for ARDL model is represented as

$$\Delta(LNSEC_t) = \sum\nolimits_{i=1}^{3} \gamma_{1i} \Delta LNSEC_{t-j} + \sum\nolimits_{j=0}^{1} \gamma_{2j} \Delta RGDP_{t-j} + \sum\nolimits_{j=0}^{1} \gamma_{3j} \Delta ELXR_{t-j} + \sum\nolimits_{j=0}^{0} \gamma_{4j} \Delta LR_{t-j} + \sum\nolimits_{j=0}^{0} \gamma_{5j} \Delta LMS_{t-j} - 0.15198 EC_{t-1} \tag{13}$$

Serial Correlation Test

Result in Table 6 indicates that there is no presence of serial correlation in ARDL Error Correction Mode (ECM) using Breusch-Godfrey test.

4.5 *Performance Evaluations of the Fitted Models*

A pseudo out-of-sample forecast technique is used to evaluate the forecasting performance of a proposed model. The paper uses the training sample to estimate the parameters of the forecasting models and as a first step in our forecasting practice obtains one to 12 a head forecasts horizon starting from 2013q4 up to 2016q3 from these models. The actual data for 2013q3 is added to the training sample after which the parameters of the models are re-estimated.

Using the re-estimated models, we forecast the values from 2013q4 up to 2016q3. We then stored these forecasts by putting the first forecast (2013q4) as the second entry in the series 1 step ahead, the second forecast (2014q1) as the second entry in the series 2 steps ahead, and so on to the 12th forecast (2016q3) as the second entry in the series 12 steps ahead. We then test the quality of the obtained forecasts using three classical statistical loss functions: Mean Absolute Error (MAE), Mean Absolute Percent Error (MAPE), and Root Mean Squared Error (RMSE). The statistical loss functions MAEt and RMSEt and MAPEt for the t forecast horizon (t = 1, 2 …, 12) are used to compare the forecast performances of the estimated short-term forecasting models. We computed MAE, RMSE, and MAPE defined in Eqs. (8), (9), and (10)

Table 7 Forecast evaluation for ARIMA and ARDL model

Forecast horizon	RMSE		MAE		MAPE	
	ARIMA	ARDL	ARIMA	ARDL	ARIMA	ARDL
1	0.0319	0.0270	0.0271	0.0194	0.2587	0.1855
2	0.0418	0.0339	0.0370	0.0315	0.3518	0.2992
3	0.0362	0.0416	0.0280	0.0358	0.2666	0.3398
4	0.0383	0.0461	0.0316	0.0408	0.2993	0.3864
5	0.0915	0.0627	0.0632	0.0530	0.6040	0.5031
6	0.1058	0.0728	0.0759	0.0620	0.7259	0.5919
7	0.1063	0.0848	0.0801	0.0722	0.7664	0.6890
8	0.1198	0.1555	0.0931	0.1346	0.8928	1.2883
9	0.1167	0.0826	0.0923	0.0724	0.8853	0.6930
10	0.1246	0.0935	0.1007	0.0809	0.9717	0.7794
11	0.1220	0.0921	0.997	0.0805	0.9626	0.7764
12	0.1184	0.0885	0.0966	0.0747	0.9324	0.7200

for NSEC by choosing models (11) and (13). The results of the computations are presented in Table 7.

Table 7 indicates that ARDL model would be better and outperform ARIMA model in forecasting one to two quarters, five to seven quarters, and nine to twelve quarters for forecasting NSEC and ARIMA models should be better in forecasting three to four quarters for NSEC.

5 Conclusion

This paper proposes two-time series models for NSEC data using ARDL and ARIMA models and evaluates the pseudo out-of-sample forecast performance of the models using three statistical loss functions: mean absolute error and root mean squared error. The result indicates that ARDL has the least forecast error in one to two, five to seven, and nine to twelve quarters for forecasting NSEC and ARIMA model have the least forecast error only in three to four and five to six-quarters forecast horizons. In conclusion, this suggests that, choosing ARDL would be better and could outperform ARIMA model in forecasting one to two quarters, five to seven quarters, and nine to twelve quarters for forecasting NSEC and ARIMA model would be better in forecasting three to four quarters for forecasting NSEC, hence ARDL model is the best model in short and long term forecasting of NSEC.

References

1. Adebiyi, M.A., Adenuga, A.O., Abeng, M.O., Omanukwe, P.N., Ononugbo, M.C.: Inflation forecasting models for Nigeria. Central Bank of Nigeria. Occasional Paper No. 36. Research and Statistics Department, Abuja (2010)
2. Box, G.E.P., Jenkins, G.M.: Time Series Analysis: Forecasting and Control. Holden, San Francisco (1976)
3. Central Bank of Nigeria: Statistical Bulletin, vol. 24 (2013) www.cbn.org
4. Doguwa, S.I., Alade, S.O.: Time series modeling of Nigerian's external reserve. CBN J. Appl. Stat. **6**(1) (2015)
5. Engle, R.F., Granger, C.W.J.: Co-integration and error correction: representation, estimation and resting. Econimetrica **55**(2), 251–276 (1987)
6. Jansen, D.W., Wang, Z.: Evaluating the fed model of stock price valuation: an out-of-sample forecasting perspective. Economet. Anal. Fin. Econ. Time Series/Part B Adv. Econometr. **20**, 179–204 (2006)
7. Kapil, Hanuman: Prediction of market capitalization trend through selection of best ARIMA model with reference to India infrastructure companies. Int. J. Appl. Sci. Manage. **1**(2), 91–104 (2016)
8. Pesaran, M.H., Shin, Y., Smith, R.J.: Bounds testing approaches to the analysis of level relationships. J. Appl. Econometr. **16**, 289–326 (2001)
9. Taiwo, Olayato: Measuring forecast performance of vector auto-regression model and time series regression model. Am. J. Sci. Indus. Res. **4**(1), 49–58 (2013)
10. Iqbal, J., Uddin, M.N.: Forecast accuracy of error correction model: Interventional evidence for monitory aggregative. Int. J. Sci. Glob. Econ. Stud. **6**(1), 14–32 (2013)
11. Prapanna, M., Labani, S., Saptarsi, G.: Study of effectiveness of time series modeling (Arima) in forecasting stock prices. Int. J. Comput. Sci. Eng. Appl. (IJCSEA) **4**(2):13–29 (2014)

Industry Specifics of Models Predicting Financial Distress

Dagmar Camska

Abstract This paper focuses on scoring methods predicting corporate default. There exist many tools for the estimation of future distress or bankruptcy. Traditional research usually analyzes the accuracy of the methods and recommends which tools should be used. Although this paper examines existing scoring models, the research idea is different. The aim of the paper is to examine whether belonging to an industry branch influences the results of the models. The models are mainly used generally, without respect to the industry branches. However, companies belonging to different industries can reach different performance. Scoring approaches are based on performance expressed by financial ratios such as profitability, activity, liquidity, and leverage. Therefore, it could be assumed that the final values of the models would be affected. The paper focuses on three industry branches—construction, manufacture of fabricated metal products, and manufacture of machinery. The research idea is tested on four data subsamples consisting of healthy and insolvent companies and describing different time periods. The final values of the models for the individual companies are summed up by descriptive statistics. The gained results show that in specific economic circumstances there are significant differences for the different industries.

Keywords Corporate financial health · Possibilities of prediction · Czech Republic

1 Introduction

Scoring methods predicting corporate default have become a serious research area as well as a topic for practitioners since the 1960s. Altman [2] and Beaver [5] can be identified as pioneers in this area. The users of these prediction models appreciate ate quickness, low cost, transparency, and interpretability when they analyze the financial situation of one particular company. The users' aim is to mitigate business

D. Camska (✉)
MIAS School of Business, Czech Technical University in Prague, Kolejni 2637/2a, 16000 Prague 6, Czech Republic
e-mail: dagmar.camska@email.cz

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_8

risk and avoid making business with potentially defaulted entity. The users' positions in a business transaction can differ. They can be suppliers, customers, governmental bodies, or financial institutions. It does not matter in which position they are because potential default of their business partner can affect their own activities. Rational persons forecast possible future development and the models predicting corporate financial distress or default enable these forecasts.

In the research area of the prediction models there have been solved several issues. First of all, the primary researches have constructed these prediction tools which should be able to separate healthy and unhealthy business entities. The secondary researches focus on the explanatory power of already existing models. They conclude if the previous models are still sufficient or if there is a need of new tools providing higher explanatory power for current conditions. The primary contributions will be discussed in the section dedicated to literature review. There can be mentioned following examples [8, 18, 20, 25] or [31] of secondary papers focusing on explanatory power testing in the Czech Republic. The secondary research works do not include only accuracy testing but also discussions of economic conditions influence as [4, 21] or [24]. The question which could seem underdeveloped is an industry sector influence.

The models predicting financial distress are used in general. On one hand, it is their general advantage, on the other hand, it serves as their limitation too. It seems that enterprises belonging to different industries do not achieve comparable performance. Examples can be found in relevant literature. According Vlachý [33] industry sector affects value and structure of working capital and therefore corporate liquidity. It was proved by [13] or [29] that corporate leverage is determined by the industry branch. Financial forecasting performance depends on the belonging to the industry sector [12] or [23]. Industry performance is an important determinant of a company's profitability [15]. These findings lead to a statement financial ratios influenced by industry specifics entering the prediction models can cause significant differences in the final values. This research work highlights the importance of the industry sector belonging. The carried out analysis will prove or disprove if industry specifics should be respected in the case of the models predicting financial distress usage.

2 Literature Review in the Area of the Prediction Models

Ever since 1960s, numerous tools focusing on corporate financial situation forecasting have been created. Although some of them still maintain high explanatory power, many lack sufficient accuracy necessary for making important decisions. This explains why already existing prediction models are often so reexamine. As a rule, researchers subject one or several prediction tools to examination. However, the studies showing a broader perspective are exceptional. The exception providing the rule is Čámská [7, 8], who examined almost 4 dozens of the bankruptcy models, or precedent study of this article [9]. Conclusions of [7, 8] prove that although some of the tools maintain high accuracy, some others show a high error rate.

The prediction models used for this analysis have high explanatory power exploit Czech data. It seems, from what was discussed previously [25, 31], that the Czech prediction models are most appropriate for the Czech corporate financial data. Nevertheless the conclusions of [8] or [7] do not correspond with just mentioned statement. The prediction models have high accuracy relevant to the Czech data and it does not matter if they are originally Czech or they come from post-transitive European economies as well as from developed economies. The same applies the tools which have already lost their accuracy. The analysis is based on the models maintaining high explanatory power, with no regard where they were built.

Since the Czech data is analyzed, the emphasis is placed on Czech approaches as IN01 [28], IN05 [27] and Balance Analysis System by Rudolf Doucha [11]. Predecessors of these models are mainly considered to be Altman Z-Score [3], Bonita Index (in the German original Bonitätsanalyse) [34], Kralicek [22] or Taffler [1]. Their approaches were created in the developed economies facing different historical, political, and economic circumstances. They were transmitted into the Czech environment during the transition period and, surprisingly, they still maintain their popularity and accuracy [8]. The results of Kralicek [22] and Taffler [1] will not be displayed graphically, the reasons will be provided later.

Different historical, political, and economic conditions of the Czech Republic from those of the developed economies support testing of models created in previous transition and current post-transition economies. Therefore the prediction models designed in Poland, Hungary, Lithuania, and Latvia will also be used for the purpose of this study analysis. The models include Prusak, PAN-E, PAN-F, PAN-G, D2, D3 (all previously discussed in [19]) originated in Poland, Hungarian tool was designed by Hajdu and Virág [14], and Baltic prediction tools are attributed to Šorins and Voronova [16], Merkevicius [26], and R model [10]. Models' formulas can be found in relevant literature. This paper, according to cited references, is based on original models' versions.

3 Research Idea and Data

This chapter is mainly focused on the paper's idea explanation and used data description. The first subpart introduces the solved research issue and explains methods applied. The second part describes the data sample, its source, extraction, and size.

3.1 Paper's Idea and Used Methods

The idea of the research is based on the general usage of the prediction models. The models predicting financial distress usually do not reflect any specifics, such as belonging to the industry branch, companies' size, ownership structure and other factors. The enterprise belonging to the various industry sectors may perform differently. The achieved performance influences items included in financial statements and therefore values of financial indicators may be affected. The financial indicators can be represented by financial ratios describing profitability, liquidity, activity, or leverage. It stays an unsolved question if this influence is significant or not. Tradeoffs between the indicators can be observed. The value of the first ratio is worse, contrary, the value of the second ratio is better. At the end the final values are similar without a respect to industry branch.

The prediction models introduced in the preceding chapter will be applied to financial data. The results for individual business entities are summed up by main descriptive statistics, such as mean, median, quartiles etc. The comparison of these statistical characteristics for different industries will be conducted. This comparison can prove or disprove significant differences of the final scores of the bankruptcy formulas. It could lead to serious consequences that prediction models' users should be aware of industry specifics and the models predicting financial distress should not be used in general.

3.2 Data Sample

The data sample has to consist of different industry sectors, otherwise it is impossible to verify differences among particular industry branches. Being the largest production sectors in the Czech Republic, three industries create a backbone of this research. The analysis is based on the following sectors: Construction (CZ-NACE F), Manufacture of fabricated metal products, except machinery and equipment (CZ-NACE 25), and Manufacture of machinery and equipment (CZ-NACE 28). These branches were also focused on in the papers [7–9]. The advantage of these sectors is the largest data samples in respect of the healthy as well as the insolvent enterprises. The common feature of all these industries is their aim to manufacture final tangible products. In spite of the similarity in their aims, they can differ in terms of financial ratios' values mentioned in the introduction.

The sample contains data describing not only various industries but also different kinds of entities and different time periods. The first two parts describe the insolvent and healthy companies of the year 2012, as analyzed in previous papers [7] or [8]. The insolvent companies declared their insolvency according to the Czech Insolvency Act. The criterion used for the selection of the healthy entities was their performance of positive economic value added (respecting [17]). The third and fourth parts of the data sample are the largest in size. These subsamples are considered general, since no restriction has been applied to them. Each subsample respects belonging to the selected industry branches.

The third and fourth parts describe the latest data available for the years 2016 and 2017. The financial data have not been reported yet for the year 2019 and those for the year 2018 are still not fully publicly available. Despite the obligatory rule to report the financial statements, not all the Czech companies fulfill this duty or

Table 1 Size of the analyzed sample

Industry branch	Healthy 2012	Insolvent 2012	General 2016	General 2017
CZ-NACE 25	383	36	1,981	1,525
CZ-NACE 28	33	10	1,017	789
CZ-NACE F	229	38	4,499	3,564

publish their results on time [32]. This particularly concerns insolvent companies which are not willing to publish their statements [6]. It should also be emphasized that some managers do not consider accounting data to be relevant and transparent [30] and therefore they tend not to report.

Table 1 demonstrates the size of each subsample in respect to the selected industry sectors. The accounting data for the individual enterprises were extracted from the prepaid corporate database Albertina. The issue relating to financial statement reporting has been already explained. The comparison of the data samples for 2016 and 2017 proves that the companies do not publish on time. Some observations (enterprises) were excluded from this research, since their financial statements were incomplete or some values were zero; these shortcomings made the calculation of financial ratios and the evaluation by the prediction models impossible.

It should, however be emphasized that the preceding and current data samples are not compatible. The samples from 2016 and 2017 describe just general situation and as such cannot be polarized as the preceding data from 2012. The oldest data show the most ailing companies and on also companies with the strongest performance resulting from their positive economic value added.

4 Results

Section 2 introduced selected models predicting financial distress for which final values relating to each particular enterprise were calculated. The results of the individual models can be presented by descriptive statistics. The example of this approach is shown in Table 2 demonstrating the results for Altman Z-Score. Altman Z-Score was selected due to its international coverage. The main descriptive statistics are at disposal for each analyzed industry branch (CZ-NACE F, CZ-NACE 25, and CZ-NACE 28) in each data subsample.

Table 2 demonstrates significant differences between final Altman Z-Score values in different industry branches. The highest values were measured in the field of Construction and the lowest values in the field of Manufacture of machinery and equipment. The mean difference is 1.55 (also the median difference over 1.00). This difference almost equals the range of the grey zone [3]. The evidence provided even shows that many enterprises listed under the category of Construction (CZ-NACE F) were, according to Altman Z-Score, classified in the grey zone according to Altman Z-Score although their bankruptcy has already been declared. The results observed

Table 2 Altman Z-Score and its descriptive statistics for insolvent companies 2012

Healthy companies 2012	Manufacture of machinery and equipment	Manufacture of fabricated metal products	Construction
Mean	0.15	1.17	1.70
Median	0.92	1.17	1.98
Minimum	−8.63	−2.66	−8.79
Maximum	2.68	6.85	4.77
1st quartile	0.21	−0.003	1.15
3rd quartile	1.64	2.15	2.61
St. deviation	3.04	1.84	2.22
Trim mean	0.15	1.12	1.90

provide the evidence that placement within a particular industry plays a significant role in the area of bankruptcy prediction.

Table 2 presents the results obtained for only one model and one of four subsamples. The question suggesting itself is how to present the results obtained in a more complex way. Here the idea of visualization seems to serve this purpose. Figures 1, 2, 3 and 4 display one of the data subsamples with the results in all tested models. The models predicting financial distress are sorted by numbers as follows: 1—Altman, 2—IN01, 3—IN05, 4—Doucha, 5—Bonita, 6—Prusak 1, 7—Prusak 2, 8—PAN-E, 9—PAN-F, 10—PAN_G, 11—D2, 12—D3, 13—Hajdu and Virág, 14—Šorins and Voronova, 15—Merkevicus, 16—Rmodel. The same approach is applied in paper [9]. The Kralicek's [22] and Taffler's [1] results will not be displayed graphically. The reasons are not the same for each model. Kralicek Quick Test is based on the different metrics, since higher values mean worse situation. For the other models the

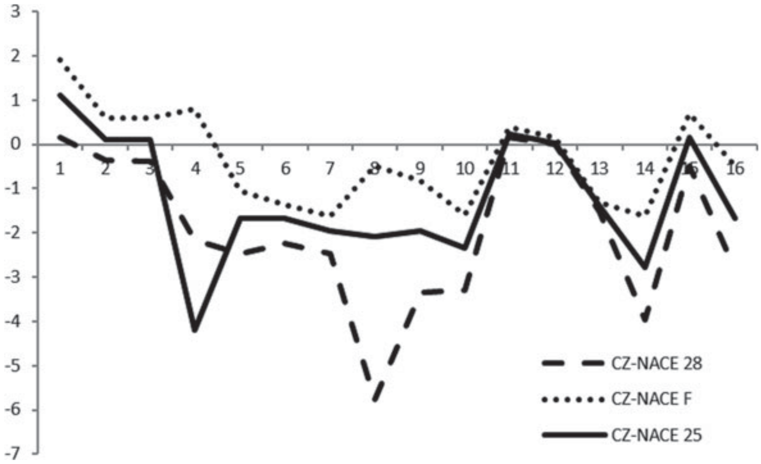


Fig. 1 Prediction models and their final values for insolvent companies 2012. *Source* [9]

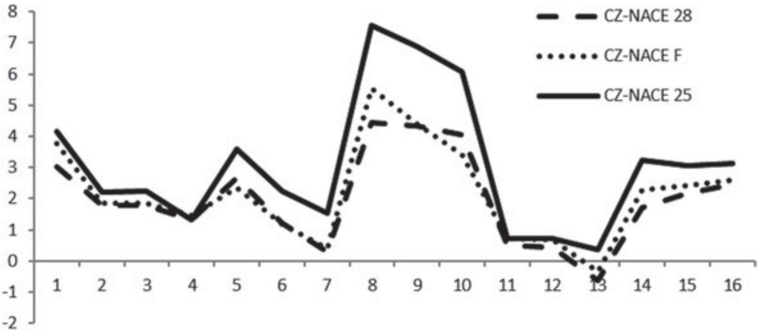


Fig. 2 Prediction models and their final values for healthy companies 2012. *Source* [9]

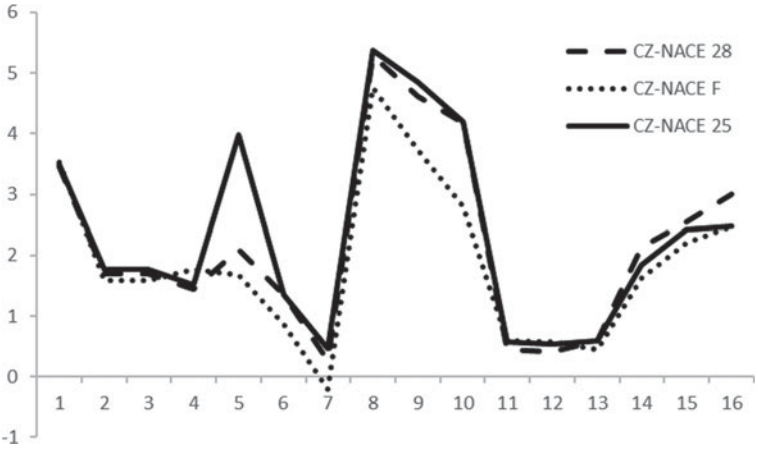


Fig. 3 Prediction models and their final values for general sample 2016

values desired are the highest because they present a strong performance. Taffler has not been visualized because its final scores are twice as high as those for the other models. Taffler's inclusion would cause a figure distortion.

The results are visualized in 4 figures. Each of them displays final scores of the tested prediction models for one subpart of the aforementioned described data sample (healthy entities 2012, insolvent entities 2012, general entities of the years 2016 and 2017). Since it is necessary to summarize the results for all analysed companies subjected to analysis, figure lines show trim mean for a particular group of businesses. The advantage of the trim mean is the possibility of outliers' separation. Dashed line represents the industry sector Manufacture of machinery and equipment (CZ-NACE 28), dotted line shows the industry sector Construction (CZ-NACE F) and finally solid one displays the results for Manufacture of fabricated metal products, except machinery and equipment (CZ-NACE 25).

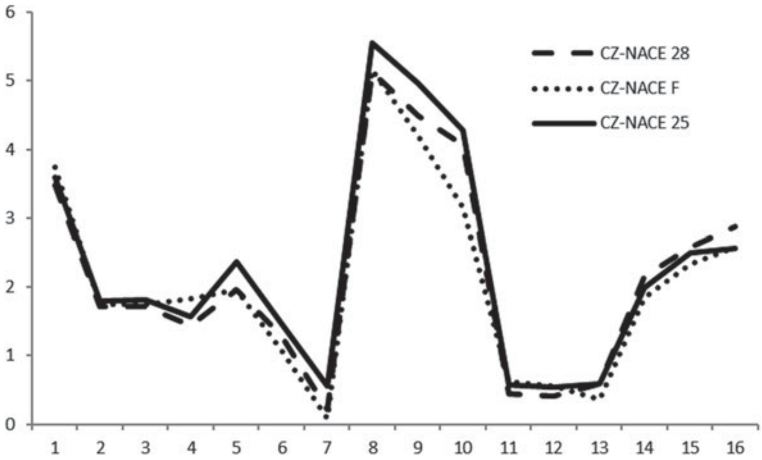


Fig. 4 Prediction models and their final values for general sample 2017. *Source* [9]

Figure 1 provides evidence corresponding with Table 2. It confirms existence of the differences among particular industries in the case of the insolvent enterprises. The construction industry (CZ-NACE F) reports the highest values, followed by Manufacture of fabricated metal products, except machinery and equipment (CZ-NACE 25) and the lowest values are reached in Manufacture of machinery and equipment (CZ-NACE 28). It should be said that not all prediction approaches show the same results because figure curves sometimes intersect. It has a consequence that the final ranking could not be generalized.

Figure 2 displays the results for the healthy companies from the year 2012. Differences are again visible. The highest values are measured in the field of CZ-NACE 25, than CZ-NACE F and the lowest valued are measured in the field of CZ-NACE 28. All not verified models provide the same results because of the curves intersections or curves closeness. Divergent conclusions arise from the indicators included in the individual prediction models. The prediction tools are based on different financial ratios and tradeoffs between their values cause difference in the final values of the models predicting financial distress.

It should be noted that the year 2012 cannot be considered stable. In that time the Czech economy still coped with the consequences of the last global economic crisis. The third and fourth parts of the data sample consists of the financial statements describing the stable time period of the years 2016 and 2017. The Czech economy was stable and growing. Figures 3 and 4 display the results for this time period.

Despite the same approach applied in the Figs. 3 and 4, none significant differences between the industry sectors subjected to the analysis were observed. The models predicting financial distress provide comparable results which are not influenced by industry sectors. It cannot be specified in which field the highest and the lowest values were measured. It is, however, evident that Figs. 1 and 2 show different results from those in Figs. 3 and 4. In spite of belonging to the various industries accompanied by different sales volumes, property composition, or financial structure, the final values of the prediction models are similar. These results do not confirm the research hypothesis. The possible reasons causing these discrepancies among the time periods will be described later.

Taffler's approach and Kralicek Quick Test were excluded from the visualization. It is necessary to emphasize that both approaches reached similar results as presented in the Figs. 1, 2, 3 and 4. This refers to the significant differences in the year 2012 but almost no differences in the years 2016 and 2017. Kralicek Quick Test shows the limited differences for the year 2012. This is caused by the inherent model's nature, due to its discrete versus continuous basis used by all the other tested models predicting financial distress.

5 Conclusion

The research dealt with the bankruptcy prediction models constructed in the past. The selected approaches are still applied in corporate predictions and their accuracy seems to be sufficient. Almost 20 models were analyzed and compared to fulfill the aim of this research. In addition to a generally required accuracy and explanatory power, this research was focused on industry specifics. The analyzed models predicting financial distress are used in general although the companies belonging to different industries can have different sales volumes, property and financial structure. These differences may influence financial ratios' values entering the prediction models. It is also a reason why the final values of the prediction models were subjected to the analysis. The verification was carried out on a sample of three industry sectors, such as Construction (CZ-NACE F), Manufacture of machinery and equipment (CZ-NACE 28), and Manufacture of fabricated metal products, except machinery and equipment (CZ-NACE 25).

The significant differences were observed in the data subparts describing the insolvent and the healthy companies of the year 2012. The lowest values were measured in the field of CZ-NACE 28. On the contrary, the highest values—among the insolvent entities—were measured in the field of CZ-NACE F and among the healthy businesses in the field of CZ-NACE 25. These findings are not supported by the third and fourth subparts of the data sample. The companies included in the general samples of the years 2016 and 2017 reached comparable results. No differences were observed between the selected industry branches. In other words, it was not confirmed that the industry specifics have a serious impact on the models predicting financial distress. Although industry branches influence reported financial statements, such as income statement and balance sheet, their effect on the final scores of the prediction models was not observed. As several financial ratios enter into the prediction tools, there are some tradeoffs between the ratios which results on similar final prediction scores. Owing to several discrepancies which occurred in the area of profitability, activity, leverage, or liquidity between different industry sectors, individual indicators should be subjected to further analysis.

The results confirmed the hypothesis of the research conducted on the subsamples from the year 2012. These subsamples represent financial consequences caused by the latest global economic crisis. First, the period of overall economic instability influences industry reactions. Some industries are more flexible than the others and therefore their financial statements and the financial indicators derived are less affected. Secondly, the economic crisis affected various industry sectors differently. Some of them experienced a deeper slump visible on the results observed. What has just been said, can be confirmed by the combination of financial ratios for individual companies and further information such as price movements, production slumps, or changes in payment conditions in different industry sectors.

Acknowledgements The paper is one of the outputs of the research project "Financial characteristics of enterprise in bankruptcy" registered at Grant Agency of Academic Alliance under the registration No. GAAA 10/2018.

References

1. Agarwal, V., Taffler, R.J.: Twenty-five years of the Taffler z-score model: does it really have predictive ability? Account. Bus. Res. **37**(4), 285–300 (2007)
2. Altman, E.I.: Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. J. Finance **23**(4), 589–609 (1968)
3. Altman, E.I.: Corporate Financial Distress and Bankruptcy, 2nd edn. Wiley, New York (1993)
4. Altman, E.I.: Predicting corporate distress in a turbulent economic and regulatory environment. Rassegna Economica **68**(2), 483–524 (2004)
5. Beaver, W.: Financial ratios as predictors of failure. J. Account. Res. **4**(3), 71–111 (1966)
6. Bokšová, J., Randáková, M.: Zveřejňují podniky, které procházejí insolvenčním řízením, své účetní závěrky? (Diskuse ke zveřejňování účetních závěrek). Český finanční a účetní časopis **8**(4), 164–171 (2013)
7. Čámská, D.: Models predicting financial distress and their accuracy in the case of construction industry in the Czech Republic. In: Pastuszková, E., Crhová, Z., Vychytilová, J., Vytrhlíková, B., Knápková, A. (eds.) 7th International Scientific Conference Finance and Performance of Firms in Science, Education and Practice, Tomas Bata University in Zlin, pp. 178–190. Tomas Bata University in Zlin, Zlín (2015)
8. Čámská, D.: Accuracy of models predicting corporate bankruptcy in a selected industry branch. Ekonomický časopis **64**(4), 353–366 (2016)
9. Čámská, D.: Models predicting corporate financial distress and industry specifics. In: Valenzuela, O., Rojas, F., Pomares, H., Rojas, I. (eds.) Proceedings of Papers of International Conference on Time Series and Forecasting (ITISE 2019), Universidad de Granada, pp. 647–656. Godel Impresiones Digitales S.L., Granada (2019)
10. Davidova, G.: Quantity method of bankruptcy risk evaluation. J. Risk Manag. **3**, 13–20 (1999)
11. Doucha, R.: Finanční analýza podniku: praktické aplikace, 1st edn. Vox Consult, Prague (1996)
12. Fairfield, P.M., Rammath, S., Yohn, T.L.: Do industry-level analyses improve forecasts of financial performance? J. Account. Res. **47**(1), 147–178 (2009)
13. Frank, M.Z., Goyal, V.K.: Capital structure decisions: which factors are reliably important? Financ. Manag. **38**(1), 1–37 (2009)
14. Hajdu, O., Virág, M.: Hungarian model for predicting financial bankruptcy. Soc. Econ. Cent. East. Eur. **23**(1–2), 28–46 (2001)

15. Jackson, A.B., Plumlee, M.A., Rountree, B.R.: Decomposing the market, industry, and firm components of profitability: implications for forecasts of profitability. Rev. Acc. Stud. **23**(3), 1071–1095 (2018)
16. Jansone, I., Nespors, V., Voronova, I.: Finanšu un ekonomisko risku ietekme uz Latvijas partikas mazumtirdzniecibas nozares attistibu. Sci. J. Riga Tech. Univ. Econ. Bus. Econ.: Theory Pract. **20**, 59–64 (2010)
17. Jordan, B.D., Westerfield, R., Ross, S.A.: Corporate Finance Essentials. McGraw-Hill, New York (2011)
18. Karas, M., Režňáková, M.: Bankruptcy prediction model of industrial enterprises in the Czech Republic. Int. J. Math. Models Methods Appl. Sci. **7**(5), 519–531 (2013)
19. Kisielinska, J., Waszkowski, A.: Polskie modele do prognozowania bankructwa przedsiębiorstw i ich weryfikacja. Ekonomika I Organizacja Gospodarki Żywnościowej **82**, 17–31 (2010)
20. Klečka, J., Scholleová, H.: Bankruptcy models enunciation for Czech glass making firms. Econ. Manag. **15**, 954–959 (2010)
21. Korol, T., Korodi, A.: Predicting bankruptcy with the use of macroeconomic variables. Econ. Comput. Econ. Cybern. Stud. Res. **44**(1), 201–219 (2010)
22. Kralicek, P., Ertrags- und Vermögensanalyse (Quicktest). http://www.kralicek.at/pdf/qr_druck.pdf. Accessed 15 May 2019
23. Lee, G.K., Alnahedh, M.A.: Industries' potential for interdependency and profitability: a panel of 135 industries, 1988–1996. Strategy Sci. **1**(4), 285–308 (2016)
24. Liou, D.-K., Smith, M.: Macroeconomic variables and financial distress. J. Account. Bus. Manag. **14**, 17–31 (2007)
25. Machek, O.: Long-term predictive ability of bankruptcy models in the Czech Republic: evidence from 2007–2012. Cent. Eur. Bus. Rev. **3**(2), 14–17 (2014)
26. Merkevicius, E., Garšva, G., Girdzijauskas, S.: A hybrid SOM-Altman model for bankruptcy prediction. In: Alexandrov et al. (ed.). ICCS 2006, pp. 364–371 (2006)
27. Neumaierová, I., Neumaier, I.: Výkonnost a tržní hodnota firmy, 1st edn. Grada, Prague (2002)
28. Neumaierová, I., Neumaier, I.: Index IN05. In: Evropské finanční systémy, Masarykova univerzita, pp. 143–148. Masarykova univerzita, Brno (2005)
29. Oztekin, O.: Capital structure decisions around the world: which factors are reliably important? J. Financ. Quant. Anal. **50**(3), 301–323 (2015)
30. Paseková, M., Svitaková, B., Kramná, E., Otrusinová, M.: Towards financial sustainability of companies: issues related to reporting errors. J. Secur. Sustain. Iss. **7**(1), 141–154 (2017)
31. Pitrová, K.: Possibilities of the Altman Zeta model application to Czech Firms. E&M Ekonomika Manag **14**(3), 66–76 (2011)
32. Strouhal, J., Gurtvis, N., Nikitina-Kalamae, M., Li, T.W., Lochman, A.L., Born, K.: Are companies willing to publicly present their financial statements on time? Case of Czech and Estonian TOP100 Companies. In: Culik, M. (ed.) Proceedings of the 7th International Scientific Conference on Managing and Modelling of Financial Risks Modelling of Financial Risks, VSB Tech Univ Ostrava, pp. 731–738. VSB Tech Univ Ostrava, Ostrava (2014)
33. Vlachý, J.: Corporate Finance. Leges, Prague (2018)
34. Wöber, A., Siebenlist, O.: Sanierungsberatung für Mittel- und Kleinbetriebe, Erfolgreiches Consolting in der Unternehmenkrise. Erich Schmidt Verlag GmbH, Berlin (2009)

Stochastic Volatility Models Predictive Relevance for Equity Markets

Per Bjarte Solibakke

Abstract This paper builds and implements multifactor stochastic volatility models where the main objective is step ahead volatility prediction and to describe its relevance for the equity markets. The paper outlines stylised facts from the volatility literature showing density tails, persistence, mean reversion, asymmetry and long memory, all contributing to systematic data dependencies. As a by-product of the multifactor stochastic volatility model estimation, a long-simulated realization of the state vectors is available. The realization establishes a functional form of the conditional distribution, which is evaluated on observed data convenient for step ahead predictions. The paper uses European equity for relevance arguments and illustrational prediction purposes. Multifactor SV models empower volatility visibility and predictability enriching the amount of information available for equity market participants.

Keywords Stochastic volatility · Markov chain monte carlo (MCMC) simulations · Projection-reprojection

1 Introduction

This paper builds and assesses multifactor scientific stochastic volatility (SV) models for the prediction of equity market volatility. Volatility is a measure of dispersion around the mean return of an asset. When the price returns are tightly bunched together (or spread apart), the volatility is small (large). The use of all volatility models entails prediction characteristics for future returns. A volatility model has been used internationally to predict the absolute magnitude of returns, quantiles and entire densities. A special feature of asset volatility is that it is not directly observable. The unobservability of volatility makes it difficult to evaluate the forecasting performance of volatility models. However, knowledge of the empirical properties

P. B. Solibakke (✉)
Faculty of Economics and Management, Norwegian University of Science and Technology,
Larsgårdsvn. 2, 6025 Ålesund, Norway
e-mail: per.b.solibakke@ntnu.no

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_9

of future prices is important when constructing risk management strategies, i.e. *d* portfolio selection, derivatives and hedging, market making and market timing. For all these activities, the predictability of volatility is essential for success. Modern portfolio theory (MPT) suggests that volatility creates risk. Portfolio studies have shown that when volatility increases, risk increases, and portfolio returns decreases. An equity risk manager therefore would want to know the likelihood of future asset and portfolio movements. If a portfolio manager adds more assets to his portfolio, the additional assets diversify the portfolio if they do not covary (correlation less than 1) with other assets in the portfolio. Hence, generally, portfolios imply risk reduction through diversification suggesting asset allocation importance. Mean-variance analysis and the Capital Asset Pricing Model are natural extensions of the portfolio analysis. An equity derivative trader wants to know the volatility that can be expected as contracts mature for both pricing and general risk management activities. The most important use of derivatives is a risk-reduction technique known as hedging, which requires a sound understanding of how to value derivatives and an understanding of which risks should and should not be hedged. Generally, for hedging, an equity risk manager will want to know the contract volatility approaching maturity. The only parameter that requires estimation in the Black-Scholes Model is the volatility. This volatility estimate also may be of use in estimating parameters (u and d) in a binomial model. Ceteris paribus, higher (lower) volatility increases (decreases) derivative prices. Therefore, market participants will sell (buy) both call and put option contract positions that are not part of speculative or hedge positions, if predicted volatility is declining (increasing). In contrast, a portfolio manager may want to buy (sell) a stock or a portfolio before its volatility falls (rises). Finally, a market maker can change his bid-ask spread believing future volatility changes. Normally, the equity markets show that the bid-ask spread increases (decreases) when volatility rises (falls).

Stochastic volatility models have an intuitive and simple structure and can explain the major stylized facts of asset, currency and commodity price changes. The motivation for stochastic volatility is the observed non-constant and frequently changing volatility. Time-varying volatility is endemic in financial markets and market participants who understand the dynamic behaviour of volatility are more likely to have realistic expectations about future prices and the risks to which they are exposed. The SV implementation is an attempt to specify how the volatility changes over time. Bearing in mind that volatility is a non-traded instrument, which suggests imperfect estimates, the volatility can be interpreted as a latent variable that can be modelled and predicted through its direct influence on the magnitude of returns. Risks may change through time in complicated ways, and it is natural to build multifactor stochastic models for the temporal evolution in volatility. The implementation adapts the MCMC estimator proposed by Chernozhukov and Hong [9], claimed to be substantially superior to conventional derivative based hill climbing optimizers for this stochastic class of problems. Moreover, under correct specification of the structural models the normalized value of the objective function is asymptotically χ^2 distributed (and the degrees of freedom is specified). The paper focuses on the Bayesian Markov Chain Monte Carlo (MCMC) modelling strategy used by Gallant and McCulloch [18] and Gallant and Tauchen[1] [14], [19] implementing multivariate statistical models derived from scientific considerations. The method is a systematic approach to generate moment conditions for the generalized method of moments (GMM) estimator [24] of the parameters of a structural model. Moreover, the implemented Chernozhukov and Hong [9] estimator keeps model parameters in the region where predicted shares are positive for every observed price/expenditure vector. Moreover, the methodology supports restrictions, inequality restrictions, and informative prior information (on model parameters and functionals of the model). This article is organized as follows. Section 2 describes the SV methodology. Section 3 presents stylized facts and Sect. 4 concretizes these facts from stochastic volatility models showing two examples, one index and one asset. Section 5 summarizes and concludes.

2 Theory and Methodology

2.1 Stochastic Volatility Models

The SV approach specifies the predictive distribution of price returns indirectly, via the structure of the model, rather than directly. The SV model has its own stochastic process without worries about the implied one-step-ahead distribution of returns recorded over an arbitrary time interval convenient for the econometrician. The starting point is the application of Andersen et al. [2] considering the familiar stochastic volatility diffusion for an observed stock price S_t given by

$$\frac{dS_t}{S_t} = \big(\mu + c\big(V_{1,t} + V_{2,t}\big)\big)dt + \sqrt{V_{1,t}}dW_{1,t} + \sqrt{V_{2,t}}dW_{2,t} \tag{1}$$

where the unobserved volatility processes $V_{i,t}$, $i = 1{,}2$, is either log linear or square root (affine). The $W_{1,t}$ and $W_{2,t}$ are standard Brownian motions that are possibly correlated with corr$(dW_{1,t}, dW_{2,t}) = \rho$. Andersen et al. [2] estimate both versions of the stochastic volatility model with daily S&P500 stock index data, 1953-December 31, 1996. Both SV model versions are sharply rejected. However, adding a jump component to a basic SV model greatly improves the fit, reflecting two familiar characteristics: thick non-Gaussian tails and persistent time-varying volatility. A SV model with two stochastic volatility factors show encouraging results in Chernov et al. [8]. The authors consider two broad classes of setups for the volatility index functions and factor dynamics: an affine setup and a logarithmic setup. The models

[1] The methodology is designed for estimation and inference for models where (1) the likelihood is not available, (2) some variables are latent (unobservable), (3) the variables can be simulated and (4) there exist a well-specified and adequate statistical model for the simulations. The methodologies (General Scientific Models (GSM) and Efficient Method of Moments (EMM)) are general-purpose implementation of the Chernozhukov and Hong [9] estimator.

are estimated using daily data on the Dow Index, January 2, 1953–July 16, 1999. They find that models with two volatility factors do much better than do models with only a single volatility factor. They also find that the logarithmic two-volatility factor models outperform affine jump diffusion models and provide acceptable fit to the data. One of the volatility factors is extremely persistent and the other strongly mean reverting.

This paper's SV model applies the logarithmic model with two stochastic volatility factors [8]. The model is extended to facilitate correlation between the mean and the stochastic volatility factors. The correlation applies the Cholesky decomposition for consistence. The main argument for the correlation modelling is to introduce asymmetry effects (correlation between return innovations and volatility innovations). The formulation of a general SV model for price change processes (y_t) therefore becomes

$$\begin{aligned}
y_t &= a_0 + a_1(y_{t-1} - a_0) + \exp(V_{1t} + V_{2t}) \cdot u_{1t} \\
V_{1t} &= b_0 + b_1\big(V_{1,t-1} - b_0\big) + u_{2t} \\
V_{2t} &= c_0 + c_1\big(V_{2,t-1} - c_0\big) + u_{3t} \\
u_{1t} &= dW_{1t} \\
u_{2t} &= s_1\left(r_1 \cdot dW_{1t} + \sqrt{1 - r_1^2} \cdot dW_{2t}\right) \\
u_{3t} &= s_2\left(\begin{array}{c} r_2 \cdot dW_{1t} + \left((r_3 - (r_2 \cdot r_1))/\sqrt{1 - r_1^2}\right) \cdot dW_{2t} + \\ \sqrt{1 - r_2^2 - \left((r_3 - (r_2 \cdot r_1))/\sqrt{1 - r_1^2}\right)^2} \cdot dW_{3t} \end{array}\right)
\end{aligned} \tag{2}$$

where $W_{i.t}$,$i = 1$, 2 and 3 are standard Brownian motions (random variables). The parameter vector is θ. The r's are correlation coefficients from a Cholesky decomposition[2]; enforcing an internally consistent variance/covariance matrix. Early references are Rosenberg [31], Clark [10], Taylor [35] and Tauchen and Pitts [34]. References that are more recent are Gallant et al. [15, 18, 20], Andersen [1], Durham [12], Shephard [33], Taylor [36], and Chernov et al. [8]. The model above has three stochastic factors. Even jumps with the use of Poisson distributions for jump intensity are applicable (complicates estimations considerably). The paper applies a computational methodology proposed by Gallant and McCulloch [17] and Gallant and Tauchen [19], [20] for statistical analysis of a stochastic volatility model derived from a scientific process[3]. Intuitively, the approach may be explained as follows. First, a reduced-form auxiliary model is estimated to have a tractable likelihood function (generous parameterization). The estimated set of score moment functions encodes important information regarding the probabilistic structure of the raw data

[2] For the Cholesky decomposition methodology see [4].

[3] See www.econ.duke.edu/webfiles/arg for software and applications of the MCMC Bayesian methodology. All models are coded in C/C++ and executable in both serial and parallel versions (OpenMPI).

sample. Second, a long sample is simulated from the continuous time SV model. Using the Metropolis-Hastings algorithm and parallel computing, parameters are varied in order to produce the best possible fit to the quasi-score moment functions evaluated on the simulated data. An extensive set of model diagnostics and an explicit metric for measuring the extent of SV model failure are useful side-products. The scientific stochastic volatility model cannot generate likelihoods, but it can be easily simulated.

2.2 The Unobserved State Vector Using the Nonlinear Kalman Filter

From the prior SV model estimation, a by-product is a long simulated realization of the state vector $\left\{\hat{V}_{i,t}\right\}_{t=1}^{N}$, $i = 1, 2$ and the corresponding $\left\{\hat{y}_t\right\}_{t=1}^{N}$ for $\theta = \hat{\theta}$ Hence, by calibrating the functional form of the conditional distribution of functions given $\left\{\hat{y}_\tau\right\}_{\tau=1}^{t}$; evaluating the result on observed data $\{\tilde{y}_t\}_{t=1}^{n}$; generating predictions for $V_{i,t}$, $i = 1, 2$ through Kalman filtering $y_{t,}$ very general functions of $\{y_\tau\}_{\tau=1}^{t}$ can be used and a huge dataset is available. An SNP model is estimated on the $\hat{y}_t$. The model represents a one-step ahead conditional variance $\hat{\sigma}_t^2$ of $\hat{y}_{t+1}$ given $\{\hat{y}_\tau\}_{\tau=1}^{t}$. Regressions are run of $\hat{V}_{i,t}$ on $\hat{\sigma}_t^2$, $\hat{y}_t$ and $\left|\hat{y}_t\right|$ and lags (generously long) of these series. These functions are evaluated on the observed data series $\{\tilde{y}_\tau\}_{\tau=1}^{t}$, which give values $\tilde{V}_{i,t}$, $i = 1, 2$ for the volatility factors at the original data points.

3 Stylized Facts of Volatility

Modelling and forecasting market volatility have been the subject of vast empirical and theoretical investigation over the past two decades by academics and practitioners. Volatility, as measured by the standard deviation or variance of returns, is often used as a crude measure of total risk. The volatility is not directly observable making it difficult to evaluate the forecasting performance. A good volatility model must be able to capture and reflect the stylized facts. Moreover, a good volatility model should predict volatility for success. The task of forecasting volatility conditional on previously observed data is akin to filtering in Markov-Chained Monte-Carlo (MCMC) analyses[4]. Eliciting dynamics from observables are the one-step-ahead conditional volatility $Var(y_0|x_{-1})$, where $x_{-1} = (y_{-L}, \ldots, y_{-1})$. The volatility can be obtained from standard recursions for the moments of the normal [26]. Filtered volatility is one-step-ahead conditional standard deviation evaluated at data values $\sqrt{Var\left(y_{k_0}|x_{-1}\right)|_{x_{-1}=(\tilde{y}_{t-L},\ldots,\tilde{y}_{-1})}}$ $t = 0, \ldots, n$, where y_t denotes data and y_{k0} denotes

[4] Filtered volatility is a data-dependent concept and the dynamic system must be sampled at the name frequency as the data to determine the density.

the kth element of the vector y_0, $k = 1,\ldots,M$. The volatility application involves estimating an unobserved state variable conditional on all past and present observables. Hence, filtering obtains [16], where y^* is the contemporaneous unobserved variable and x^* is the contemporaneous and lagged observed variables. Applications are portfolio optimization/minimization, option pricing and hedging.

3.1 Tail Probabilities, the Power Law and Extreme Values

The distribution of financial time series (returns) exhibits fatter tails than those of a normal distribution. The distribution for the latent volatility is more lognormal than normal. Hence, financial variables are four times more likely to experience big moves than the normal distribution would suggest. The power law, as an alternative to assuming normal distributions, asserts that it is approximately true that the value of a variable, υ, has the property that when y is large $\text{Prob}(\upsilon > y) = Ky^{-\alpha}$ where K and α are constants. A quick test is a plot of $\ln[\text{Prob}(\upsilon > y)]$ against $\ln y$. Evidence that the power law to hold is that this logarithm of the probability of the series changing more than y standard deviations is approximately linearly dependent on ln y for $y \geq 3$. Furthermore, the extreme value theory (EVT) estimates the tails of the volatility distributions [21]. EVT is a way of smoothing the tails of the probability distribution of daily changes. Value at Risk (VaR) and Expected Shortfall (ES) can be calculated and reflect the shape of the tail of the distribution. High confidence levels VaR and ES are available from EVT.

3.2 Volatility Clustering

Volatility show clustering of periods of volatility, i.e. large (small) movements followed by further large (small) movements (shock persistence). In the financial literature, the lumpiness is called volatility clustering. Hence, a turbulent (tranquil) trading day (period) tends to be followed by another turbulent day (period). The implication is that volatility shocks today will influence the expectation of volatility for many periods in the future (shock persistence) and there are time varying return fluctuations in the markets.

3.3 Volatility Exhibits Persistence

The clustering of large and small movements (of either sign) from price movement processes is a well-documented feature in equity markets. To make a precise definition of volatility persistence let the expected value of the variance of returns k periods in the future be defined as $E_t(r_{t+k} - \mu_{t+k})^2$ where r is the return and μ is the mean.

The forecast of future volatility then depends upon information in today's information set such as today's return. Volatility is said to be persistent if today's return has a large effect on the forecast variance for many periods in the future. A measure of the persistence of volatility is the half-life. That is, the time it takes for the volatility to move half way back towards its unconditional mean following a deviation from it and can be expressed as $\tau = k : \left|h_{t+k|t} - \sigma^2\right| = \frac{1}{2}\left|h_{t+1|t} - \sigma^2\right|$. Alternatively, SV model volatility persistence can be studied by inspection of correlograms (Q-statistics) or the Breusch-Godfrey Lagrange multiplier test. Significant Q-statistics and χ^2 statistics suggest persistence.

3.4 Volatility Is Mean Reverting

Mean reversion in volatility is generally interpreted as meaning that there is a normal level of volatility to which volatility will eventually return. In contrast, volatility clustering (persistence) implies that volatility comes and goes. Hence, mean reversion in volatility means that very long forecasts of volatility should all converge to the same normal level of volatility, no matter when they are made. The implicit interpretation is that mean reversion in volatility shows that current information has no effect on the long run forecast. Hence, periods of high volatility will eventually give way to more normal volatility, and similarly, periods of low volatility will be followed by a rise in volatility. More precisely, mean reversion implies that current information has no effect on the long run forecast. Hence, $p \lim\limits_{k\to\infty} \theta_{t+k|t} = 0,\ \textit{for all } t$, and which is also expressed as $p \lim\limits_{k\to\infty} h_{t+k|t} = \sigma^2 < \infty,\ \textit{for all } t$. Furthermore, note that option prices are generally viewed as consistent with mean reversion. That is, under simple assumptions of option pricing, the implied volatilities of long maturity options are less volatile than short maturity options (loser to long run average volatility).

3.5 Volatility Asymmetry (Leverage)

For equity market returns, it is plausible that positive and negative shocks have a different impact on volatility. This asymmetry is sometimes ascribed to a leverage effect and sometimes to a risk premium effect. For the leverage effect, as the price of a stock rises, its debt-to-equity ratio decreases, lowering the volatility of returns to equity holders. For the risk premium effect, news of increasing volatility reduces the demand for a stock because of general risk aversion among market participants. Hence, the stock value decline is normally followed by an increase in volatility as forecasted by news. Alternatively, price movements are negatively correlated with volatility suggesting that volatility increases (decreases) if the previous day returns are negative (positive) [6, 11]. Moreover, these authors also state that leverage effect happens because the fall (rise) in stock price causes leverage and the financial risk of the firm to increase (decrease).

3.6 Long Memory in Volatility

Financial time series exhibit long memory or persistence for volatility. Bailie et al. [3] states “The presence of long memory can be defined in terms of the persistence is consistent with an essentially stationary process, but where the autocorrelation takes far longer to decay than the exponential rate associated with the ARMA process”. The stochastic volatility (SV) models use long memory for modelling persistence. The autocorrelations for squared returns provide insights into the long memory characteristics of volatility measures. If the autocorrelations remain positive for very long lags, the long memory effect is present [22]. Moreover, explicit SV model volatility must exhibit the characteristics of long memory.

4 European Examples: FTSE100 Index and Equinor Asset

The daily analyses cover the period from the end of 2010 until November 2019, a total of 9 years and 110 consecutive months giving 2,325 returns for the two series. Price series are non-stationary and stationary logarithmic returns from all three series are therefore used in the analysis. Any signs of successful SV-model implementations for the markets indicate non-predictive market features and a minimum of weak-form market efficiency. Consequently, the markets are applicable for enhanced risk management activities.

4.1 Equity Summaries

Summary statistics for the two time-series are presented in Table 1. Both the FTSE100 spot index and the Equinor spot price series have small positive average returns (positive drift). The standard deviation for the index (portfolio) 0.928 is naturally lower than the single asset Equinor asset 1.587 (the index elements have a positive correlation less than 1), reporting lower risk. The maximum (3.9) and minimum (−6.2) numbers confirm lower risk for the FTSE100 index relative to the asset Equinor (a maximum of 8,7 and a minimum of −7.6). The FTSE index reports a negative skewness coefficient indicating that the return distributions are negatively skewed. In contrast, the asset Equinor reports a positive skewness suggesting a positively skewed distribution (more extreme positive price movements). The kurtosis coefficients are relatively high positive for both series (>0), indicating a relatively peaked distributions with heavy tails. The FTSE100 series is peakier than the Equinor series suggesting that the FTSE100 index has more observations close to the unconditional mean. The JB normal test statistics [25] suggest non-normal return distributions. In contrast, the quantile normal test statistics suggest more normal distributed returns. Serial correlation in the mean equation is strong and the Ljung-Box Q-statistic [28] is

Table 1 Characteristics from the European equity markets

	Mean (all)/M (−drop)	Median std.dev.	Maximum/minimum	Moment kurt/skew	Quantile kurt/skew	Quantile normal	Jarque-Bera	Serial dependence	
								Q(12)	$Q^2(12)$
FTSE100 index	0.01117	0.04470	3.9429	3.4296	0.19367	4.1806	1081.5630	29.347	789.21
		0.91763	−6.1994	−0.35950	−0.04916	{0.1236}	{0.0000}	{0.0030}	{0.0000}
	BDS-Z-statistic ($\varepsilon = 1$)				KPSS	Ph-Perron	Augmented	ARCH	RESET
	m = 2	m = 3	m = 4	m = 5	I + Trend	I + Trend	DF-test	(12)	(12;6)
	10.0916	13.3062	15.2696	17.0384	0.02397	−45.0965	−44.6335	305.916	44.0178
	{0.0000}	{0.0000}	{0.0000}	{0.0000}	{0.7238}	{0.0000}	{0.0000}	{0.0000}	{0.0000}
Equinor asset	0.01240	−0.02732	8.6859	2.4184	0.24559	5.4668	523.7562	18.050	475.30
		1.58765	−7.6262	0.15990	0.01852	{0.0650}	{0.0000}	{0.1140}	{0.0000}
	BDS-Z -statistic ($e = 1$)				KPSS	Ph-Perron	Augmented	ARCH	RESET
	m = 2	m = 3	m = 4	m = 5	I + Trend	I + Trend	DF-test	(12)	(12;6)
	10.8927	12.5739	13.8733	14.9152	0.03720	−47.1597	−35.2619	208.747	7.7907
	{0.0000}	{0.0000}	{0.0000}	{0.0000}	{0.3645}	{0.0000}	{0.0000}	{0.0000}	{0.2538}

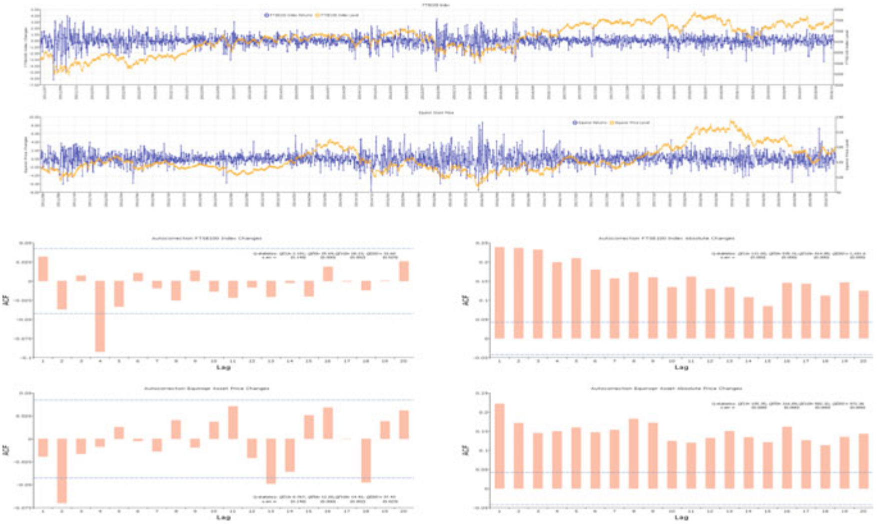


Fig. 1 FTSE100 Index (London) and Equinor Asset Prices (Oslo) for the period 2010–2019

significant for both series. Volatility clustering using the Ljung-Box test statistic for squared returns (Q^2) and ARCH statistics seems to be present. The ADF [13] and the Phillips-Person test statistics reject non-stationary series and the KPSS [27] statistic (12 lags) cannot reject stationary series. The RESET [30] test statistic, covering any departure from the assumptions of the maintained model, is not significant (stability). Finally, the BDS [7] test statistics report highly significant data dependence for all integrals (m). Figure 1 reports prices and returns and correlogram for the returns and squared/absolute returns. The correlogram for returns show only weak dependence while the correlogram for squared and absolute returns indicate substantial data dependence. The price change (log returns) data series (top), show that the level of volatility seems to change randomly but shows a time varying nature typically for financial markets.

4.2 The Stochastic Volatility Models for the European Equities

The y_t is the percentage change (logarithmic) over a short time interval (day) of the price of a financial asset traded on an active financial market. The SV model implementation establishes a mapping between the statistical and the scientific models. The adjustment for actual number of observations and number of simulations is carefully logged for final model assessment. The SV model from Eq. 1 is estimated using efficient method of moments (EMM). The BIC [32] optimal SV model from parallel runs are reported in Table 2. The mode, mean and standard deviation are

Table 2 Scientific stochastic volatility characteristics: θ-parameters

FTSE100 index scientific model				Equinor asset price scientific model			
Parameter values scientific model.			Standard	Parameter values Scientific Model.			Standard
0	Mode	Mean	errors	*0*	Mode	Mean	errors
a0	0.01758	0.02295	0.01339	*a0*	−0.02344	−0.01610	0.03525
al	0.01172	0.00212	0.02107	*al*	−0.05469	−0.06244	0.02330
b0	−0.32422	−0.34477	0.05670	*b0*	0.48438	0.30863	0.19913
b1	0.93945	0.93423	0.01544	*b1*	0.82812	0.80279	0.08810
c1	0	0	0	*c1*	0	0	0
si	0.13477	0.12925	0.01518	*s1*	0.17969	0.16723	0.03999
s2	0.13672	0.13055	0.04115	*s2*	0.14844	0.13971	0.05961
r1	−0.78516	−0.73125	0.06691	*r1*	−0.53125	−0.37728	0.20364
r2	0.51172	0.44628	0.13944	*r2*	0.65625	0.51334	0.25609
Distributed Chi-square (no. of freedom)			χ^2 (4)	Distributed Chi-square (no. of freedom)			χ^2 (3)
Posterior at the mode			−5.2477	Posterior at the mode			−4.6865
Chi-square test statistic			{0.2628}	Chi-square test statistic			{0.1962}

reported. For the two equity markets, a factor SV model produces acceptable model test statistics, reported at the bottom of Table 2. The objective function accuracy is −5.2 and −4.7 for the FTSE100 index and the Equinor asset, respectively, with associated $\cdot^2$ test statistics of 0.26 (4 *df*) and 0.20 (3 *df*). The MCMC log-posterior are reported in Fig. 2. The model does not fail the test of over identified restrictions at the level of 10%, the chains are choppy, and the densities are close to normal, all factors suggesting that the SV model is appropriate for the two equity markets. The long-simulated realization of the state vector, as a-by product of the estimated SV model, establishes a functional form of the conditional distribution. The SNP methodology obtains a convenient representation of one-step ahead conditional variance $\hat{\sigma}_t^2$ *of* $\hat{y}_{t+1}$ *given* $\left\{\hat{y}_\tau\right\}_{\tau=1}^{t}$. Running regressions for V_i on $\hat{\sigma}_t^2$, $\hat{y}_t$ *and* $|\ \hat{y}_\tau|$ and a generous number of lags of theses series, we obtain calibrated functions that give predicted values of $V_{it}|\{y_\tau\}_{\tau=1}^{t}$, $t\ =\ 1,\ 2$ on the observed data series. Figure 3,

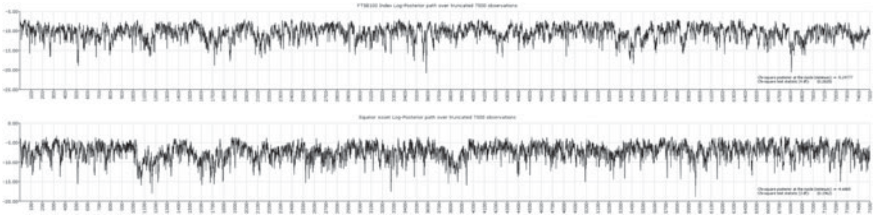


Fig. 2 MCMC posterior chain from 250 *k* optimal SV model (R = 75.000)

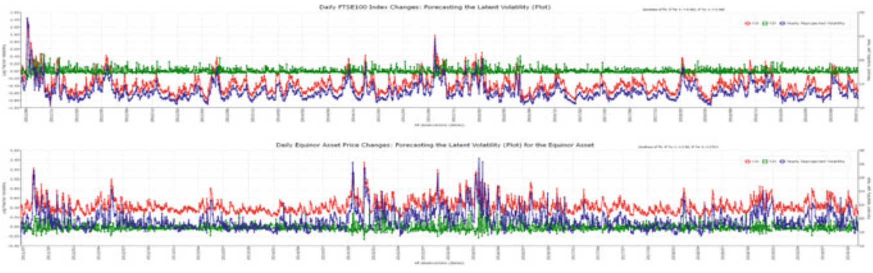


Fig. 3 Conditional volatility from observables and Kalman filtered volatility (daily)

reporting the last 60 days in 2019, shows the two latent volatility factors for the observed data points. The plots indicate that V_1 is slowly moving while V_2 is moving considerably faster. It is quite clear that the slowly persistent factor V_1, leads the re-projected yearly volatility for both series. Figure 3 also reports the ordinary least square number for R^2 for FTSE100 index (Equinor asset) at a level of V_i, where $i =$ 1,2 of 96.2% (82%) and 46.8% (51.3%), respectively. Obviously, the slowly moving V_1 factor, showing persistence, is the main contributor to yearly volatility. V_2 moves much faster showing strong mean reversion, absorbing shocks.

4.3 Volatility Characteristics for the European Equities

The volatility factors in Figs. 3 and 4 seem to model two different flows of information to the equity markets. One slowly mean reverting factor provides volatility persistence and one rapidly mean reverting factor provides for the tails [8]. The factor for the FTSE100 index is clearly moving slower than for the Equinor asset. In contrast to the crash of 1987 which was attributed to a large realization of the mean reverting factor V_2, the period 2011 to 2019 does not show large realization of V_2, but rather much more to the slowly moving factor V_1. In accordance with the plots, the period

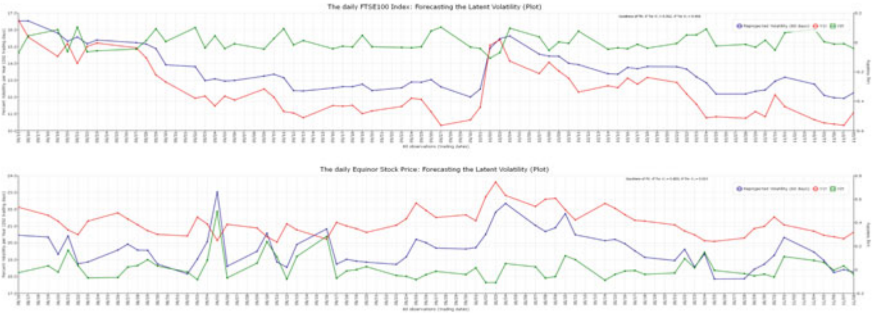


Fig. 4 FTSE100 index (top) and Equinor stock (bottom) factor volatility paths (last 60 days)

from 2011 to 2019 seems to show slow and persistent changes to volatility. However, for the Equinor asset oil shocks have shown some major contributions to volatility. For example, the shock in May 2019 is only temporary and the volatility from the shock, show strong mean reversion (V_2).

Comparing Figs. 1 and 3, the two synchronous plots show that when returns become wider (narrower) volatility increases (decreases). Moreover, turbulent (wide returns) days tend to be followed by other turbulent days, while tranquil (narrow returns) tend to follow other tranquil days (clustering). As should be expected, the volatility is clearly higher for the Equinor asset than for the FTSE100 index. Furthermore, the volatility seems to increase more from negative returns than from positive returns. Volatility densities for the FTSE100 index and the Equinor asset series suggest lognormal densities. As suggested above, the density for Equinor shows both narrower and higher volatility density than the FTSE100 index. Furthermore, the power law $\left(\text{Prob}(\upsilon > x) = Kx^{-\alpha}\right)$ providing an alternative to the normal distributions, seems approximately true for the volatility. Finally, Fig. 5 reports the correlogram for the FTSE100 index and the Equinor asset. The correlograms indicates substantial dependence suggesting both clustering and persistence as well as making volatility predictions more relevant.

Tail properties, the Power law and Extreme values. The power law, an alternative to assuming normal distributions, is applied to the reprojected volatility $\left(e^{(V_1+V_2)}\right)$ for the FTSE100 Index and Equinor asset. The power law asserts that, for many variables, it is approximately true that the value of the variable, $\cdot$, has the property that when x is large $\text{Prob}(\upsilon > x) = Kx^{-\alpha}$ where K and α are constants. The relationship implies that $\ln[Prob(\upsilon > x)] = \ln K - \alpha \ln x$, and a test of whether it holds by plotting $\ln[Prob(\upsilon > x)]$ against $\ln x$. The values for $ln(x)$ and $ln[Prob(v > x)]$ for the FTSE100 index and the Equinor asset show that the logarithm of the probability of a change by more than x standard deviations is approximately linearly dependent in $ln(x)$ for $x \geq 3$. Hence, for both the FTSE100 index and the Equinor asset the power law holds for the re-projected volatility. Regressions show the estimates of K and α are as follows: for FTSE100 (Equinor) $K = e^{-2.274}$ and $\alpha = 2.147$ ($K = e^{-0.379}$ and

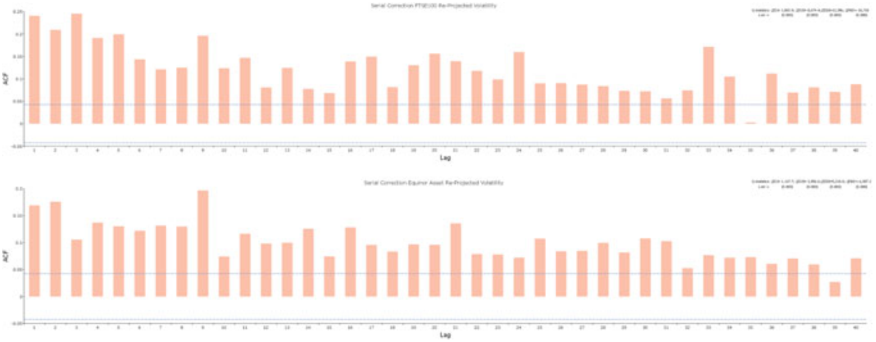


Fig. 5 Conditional volatility from observables and Kalman filtered volatility (daily)

$\alpha = 3.369$). A probability estimate of a volatility greater than 3 (6) standard deviations is $0.103 \cdot 3^{-2.147} = 0.0097\,(0.97\%)$ $\big(0.103 \cdot 6^{-2.147} = 0.0022\,(0.22\%)\big)$ and $0.685 \cdot 3^{-3.369} = 0.0169\,(1.69\%)$ $\big(0.685 \cdot 6^{-3.5369} = 0.0016\,(0.16\%)\big)$ for the FTSE100 index and the Equinor asset, respectively. The extreme value theory takes us a bit further. Setting the u to the 90 percentiles of the filtered volatility series of FTSE100 ($u = 15.55$) and Equinor ($u = 21.65$). The FTSE100 index reports optimal $\beta = 1.648$ and $\zeta = 0.119$ with an associated maximum value for the log-likelihood function of -341.6. The Equinor series reports optimal $\beta = 1.3067$ and $\zeta = 0.0514$ with an associated maximum value for the log-likelihood function of -278.3. The probability that the FTSE100 index re-projected volatility will be greater than 20 (30) is 0.9634% (0.025%). The VaR with 99% (99.9%) confidence limit is 19.92 (25.67). Hence, the 99.9% VaR estimate is about 0.892 times lower than the highest historic re-projected volatility. The 99% (99.9%) expected shortfall (ES) estimate is 22.38 (28.92). Furthermore, for the FTSE100 index, the unconditional probability for a volatility greater than 15.5356 (u) is equal to 0.46%. Similarly, the probability that the Equinor asset re-projected volatility will be greater than 20 (30) is 37.07% (0.9%). The VaR with 99% (99.9%) confidence limit is 24.85 (28.45). Hence, the 99.9% VaR estimate is about 1.005 times higher than the highest historic filtered volatility for the Equinor asset. The 99% (99.9%) ES estimate is 26.265 (30.068). Finally, for the Equinor asset, the unconditional probability for volatility greater than 21.798 (u) is equal to 0.68%. As Var and ES are attempts to provide a single number that summarizes the volatility tails giving the market participants an indication of the risk to which they are exposed. The FTSE100 index shows that a daily volatility greater than 20 is only 0.9634% while the Equinor asset, as a single asset, shows that a daily volatility greater than 20 is 37.06%. Hence, EVT and the power law, reporting VaR and ES values, summarises tail properties that indicate the risk for the market participants. For market participants, inverting the unconditional probability for volatility and setting a 1% limit for the change of unconditional probability, will list associated investments alternatives.

Volatility clustering. The BDS independence test statistic [7] is a portmanteau test for time-based independence in a series. The probability of the distance between a pair of points being less or equal to epsilon (ε) should be constant ($c_m(\varepsilon)$). The BDS test statistics, where ε is one standard deviation and the number of dimensions is 10, reports that for both the FTSE100 index and the Equinor asset, the data strongly rejects the hypothesis that the observations are independent. The FTSE100 index shows a higher BDS dependence than the Equinor asset. Moreover, the SV model reports volatility serial correlation with the SV coefficient b_1. The correlation is much stronger for the FTSE100 index ($b_1 = 0.94$) then for the Equinor asset ($b_1 = 0.83$). The $b_1 > 0.8$ will accommodate volatility clustering. This is also visible in the above Fig. 3 showing longer periods of high/low volatility for the FTSE100 index than for the Equinor asset (choppier).

Persistence in volatility. Figure 4 reports the autocorrelation and partial autocorrelation functions up to 20 lags for the FTSE100 Index and the Equinor asset. The pattern of temporal dependence is different for the two volatility factors, V_1 and

V_2. V_1 shows strong temporal dependence while V_2 shows close to zero temporal dependence. The re-projected volatility $\left(e^{(V_1+V_2)}\right)$ has inherited the temporal dependence from V_1, suggesting strong persistence in volatility. The correlograms show that FTSE100 index show higher correlation for the first lags, 0.940 versus 0.824 for the Equinor asset. However, from lag nine and higher the Equinor asset show higher serial correlations. Running the Breusch-Godfrey serial correlation LM test (Godfrey 1988) also report strong serial correlation up to lag 20 of 1869.76 ($\chi^2(20) = \{0.000\}$) and 1399.42 ($\chi^2(20) = \{0.000\}$) for the FTSE100 index and the Equinor asset, respectively. Hence, the re-projected volatility for both the FTSE100 index and the Equinor asset, show strong volatility persistence.

Volatility is mean reverting. A battery of unit root tests together with a variance ratio test (martingales) are used to test for mean reversion for the re-projected volatility. For example, the FTSE100 index (Equinor) report an ADF statistic of −9.4 (−7.7). Hence, the ADF statistics report significant mean reversion at the 1% level. Furthermore, all unit-root test statistics suggest stationary and mean reverting series. The overlapping variance ratio test [29], examines the predictability of time series data by comparing variances of differences in the data (returns) calculated over different intervals. If we assume the data follow a random walk, the variance of a period difference should be times the variance of the one-period difference. The FTSE100 index (4.399) and the Equinor asset (5.588) both reject that the volatility is a martingale, suggesting mean reversion.

Asymmetry in volatility. The asymmetry and the leverage effects are the negative correlation between the shocks of return and the subsequent shocks on volatility. Hence, after a negative return shock, we expect volatility to increase while after a positive shock on returns we should observe a decrease in volatility. Studying the volatility changes following return shocks gives some information regarding this proposition. Dividing the volatility from positive and negative returns show for the FTSE100 index (Equinor asset) an average increase in volatility from negative shocks of 2.057 (1.912) and from positive shocks of −1.875 (−1.897). Hence, negative return shocks increase average volatility while positive return shocks decrease average volatility. To statistically test for the change in volatility from negative and positive returns, we run an OLS regression on the change in daily volatility on returns and lagged returns. For the FTSE100 index (Equinor asset) the regression reports a coefficient from the returns equal to −2.206 (−0.071) and −2.823 (−1.671) for lagged returns, all significant at the 5% level. That is, the two series show that negative returns seem to increase volatility while positive returns seem to reduce volatility. Furthermore, the correlation coefficients between returns and synchronous (and lagged) re-projected volatility is −0.541 (−0.683), and −0.0164 (−0.6831) for the FTSE100 index and the Equinor asset, respectively, suggesting negative return asymmetry for both series.

Long memory. Long memory is associated with both clustering and persistence. By using fractional differencing with traditional ARMA specifications, the ARFIMA model allows for flexible dynamic patterns for the re-projected volatility. For the FTSE100 index, the ARFIMA (2,d,0) model specification estimate $d = 0.3043$

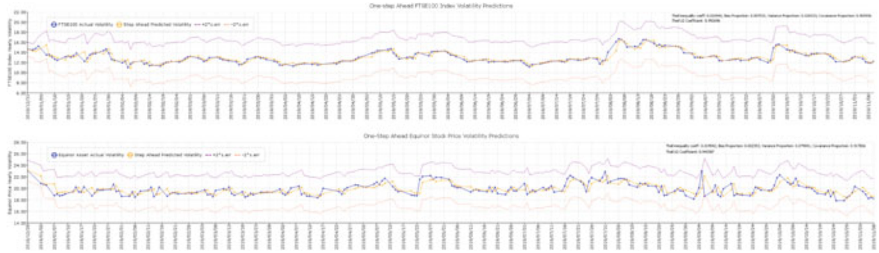


Fig. 6 Static forecasts for the FTSE100 index and the Equinor asset 2019

suggests slow autocorrelations and partial autocorrelations decay (hyperbolically). For the Equinor asset, the ARFIMA (2,d,0) model specification estimate $d = 0.3571$ suggests the same slow autocorrelations and partial autocorrelations decay. The ARFIMA model therefore specifies two slowly decaying series with long-run dependence (long memory).

4.4 Step Ahead Volatility Predictions for European Equities

The SNP methodology obtains a convenient representation of one-step ahead conditional variance $\hat{\sigma}_t^2$ *of* $\hat{y}_{t+1}$ *given* $\left\{\hat{y}_\tau\right\}_{\tau=1}^t$. Running regressions for V_{it} on $\hat{\sigma}_t^2$, $\hat{y}_t$ *and* $|\ \hat{y}_\tau|$ and a generous number of lags of theses series, we obtain calibrated functions that give step ahead predicted values of $V_{it}|\{y_\tau\}_{\tau=1}^t,\ t = 1,\ 2$ at the data points. A static forecast for the FTSE100 index and the Equinor asset is done in Fig. 6. The estimation period is from 2010 to January 1st, 2019 and the static forecasting period from January 1st, 2019 to November 8th 2019. For a "good" measure of fit, using the Theil inequality coefficient (bias, variance and covariance portions) the bias and variance should be small so that most of the bias is concentrated on the covariance proportion. The covariance proportion for re-projected volatility for the FTSE100 index (Equinor asset) is 0.966 (0.918). For the main contributor to re-projected volatility for both series, factor *V_1*, the covariance portion of the Theil inequality coefficient is even higher.

5 Summary and Conclusions

The main objective of this paper has been to characterize a good volatility model by its ability to forecast and capture the commonly held stylized facts about equity market volatility. The stylized facts include such things as heavy tails, persistence, mean reversion, asymmetry (negative return innovations suggest higher volatility), and long memory. The characteristics indicate substantial data dependence in the volatility.

The paper shows that the re-projected volatility contains all these characteristics and that this data dependence suggests an ability for volatility predictions to enhance risk management, portfolio timing and selection, market making and derivative pricing for speculation and hedging in equity markets.

The paper has used the Bayesian M-H estimator and a stochastic volatility representation for European financial equity markets. The methodology is based on the simple rule: compute the conditional distribution of unobserved variables given observed data. The observables are the asset prices and the un-observables are a parameter vector, and latent variables. The inference problem is solved by the posterior distribution. Based on the Hammersley-Clifford [23] theorem, $p(\theta,x|y)$ is completely characterized by $p(\theta|x,y)$ and $p(x|\theta,y)$. The distribution $p(\theta|x,y)$ is the posterior distribution of the parameters, conditional on the observed data and the latent variables. Similarly, the distribution $p(x|\theta,y)$ is the smoothing distribution of the latent variables given the parameters. The MCMC approach therefore extends model findings relative to non-linear optimizers by breaking the "curse of dimensionality" by transforming a higher dimensional problem, sampling from $p(\theta_1,\theta_2)$, into easier problems, sampling from $p(\theta_1|\theta_2)$ and $p(\theta_2|\theta_1)$ (using the Besag [5] formula).

Although price processes are hardly predictable, the variance of the forecast error is clearly time dependent and can be estimated by means of observed past variations. The results suggest that volatility can be forecast. The stochastic volatility models are therefore an area in empirical financial data modelling that is fruitful as a practical descriptive and forecasting device for all participants/managers in the financial services sector, together with a special emphasis on risk management (forecasting/re-projections and VaR/ES), portfolio management and derivative innovations. Irrespective of markets and contracts, Monte Carlo Simulations should lead us to more insight into the nature of the price processes describable from stochastic volatility models. Finally, static predictions of the re-projected volatility suggest a relatively good fit.

References

1. Andersen, T.G.: Stochastic autoregressive volatility: a framework for volatility modelling. Math. Finance **4**, 75–102 (1994)
2. Andersen, T.G., Benzoni, L., Lund, J.: Towards an empirical foundation for continuous-time models. J. Finance **57**, 1239–1284 (2002)
3. Bailie, R.T., Bollerslev, T., Mikkelsen, H.O.: Fractionally integrated generalized autoregressive conditional heteroskedasticity. J. Econometrics **74**, 3–30 (1996)
4. Bau III, D., Trefethen, L.N.: Numerical Linear Algebra, Society of Industrial and Applied Mathemathics. Philadelphia (1997)
5. Besag, J.: Spatial interaction and the statistical analysis of lattice systems (with discussion). J. R. Stat. Soc. Ser. B **36**, 192–326 (1974)
6. Black, F.: Studies of stock market volatility changes. In: Proceedings of the 1976 Meetings of the American Statistical Association, Business and Economics Statitics Section, pp. 307–327 (1976)
7. Brock, W.A., Dechert, W.D., Scheinkman, J.A., LeBaron, B.: A test for independence based on the correlation dimension. Econometric Rev. **15**, 197–235 (1996)

8. Chernov, M., Gallant, A.R., Ghysel, E., Tauchen, G.: Alternative models for stock price dynamics. J. Econometrics **56**, 225–257 (2003)
9. Chernozhukov, Victor, Hong, Han: An MCMC approach to classical estimation. J. Econometrics **115**, 293–346 (2003)
10. Clark, P.K.: A subordinated stochastic Process model with finite variance for speculative prices. Econometrica **41**, 135–156 (1973)
11. Christie, A.A.: The stochastic behaviour of common stock variances: value, leverage and interest rate effects. J. Financ. Econ. **10**, 407–432 (1982)
12. Dickey, D.A., Fuller, W.A.: Distribution of the estimators for autoregressive time series with a unit root. J. Am. Stat. Soc. **74**(366), 427–431 (1979)
13. Durham, G.: Likelihood-based specification analysis of continuous-time models of the short-term interest rate. J. Financ. Econ. **70**, 463–487 (2003)
14. Gallant, A.R., Tauchen, G.: A Nonparametric approach to nonlinear time series analysis, estimation and simulation. In: Brillinger, D., Caines, P., Geweke, J., Parzan, E., Rosenblatt, M., Taqqu, M.S. (eds.) New Directions in Time Series Analysis, Part II, pp. 71–92. Springer, New York (1992)
15. Gallant, A.R., Hsieh, D.A., Tauchen, G.: Estimation of stochastic volatility models with diagnostics. J. Econometrics **81**, 159–192 (1997)
16. Gallant, A.R., Long, J.R.: Estimating stochastic differential equations efficiently by minimum chi-squared. Biometrika **84**, 125–141 (1997)
17. Gallant, A.R., McCulloch, R.E.: GSM: A Program for Determining General Scientific Models, Duk, 84, University (http://econ.duke.edu/webfiles/arg/gsm) (2011)
18. Gallant, A.R., Tauchen, G.: Reprojecting partially observed systems with application to interest rate diffusions. J. Am. Stat. Assoc. **93**(441), 10–24 (1998)
19. Gallant, A.R., Tauchen, G.: Simulated score methods and indirect inference for continuous time models. In: Aït-Sahalia, Y., Hansen, L.P. (eds.) Handbook of Financial Econometrics, North Holland, Chapter 8, pp. 199–240 (2010)
20. Gallant, A.R., Tauchen, G.: EMM: A Program for Efficient Methods of Moments Estimation, Duke University (http://econ.duke.edu/webfiles/arg/emm) (2010)
21. Gnedenko, D.V.: Sur la Distribution limité du terme d'une série aléatoire. Ann. Math. **44**, 423–453 (1943)
22. Granger, C., Ding, Z.: Some properties of absolute returns. Altern. Measure Risk, Ann. Econ. Stat. **40**, 67–91 (1995)
23. Hammersley, J., Clifford, P.: Markov fields on finite graphs and lattices. Unpublished manuscript (1970)
24. Hansen, L.P.: Large sample properties o generalized method of moments estimators. Econometrics **50**, 1029–1054 (1982)
25. Jarque, J.B., Bera, A-K.: Efficient tests for normality, homoscedasticity and serial independence of regression residuals. Econ. Lett. **6**(3), 255–259 (1980)
26. Johnson, N.L., Kotz, S., Balakrishnan, N.: Continuous Univariate Distributions, pp. 1995–2752. Wiley (1970)
27. Kwiatowski, D., Phillips, P.C.B., Schmid, P., Shin, T.: Testing the null hypothesis of stationary against the alternative of a unit root: How sure are we that economic series have a unit root. J. Econometrics **54**, 159–178 (1992)
28. Ljung, G.M., Box, G.E.P.: On a measure of lack of fit in time series models. Biometrika **65**, 297–303 (1978)
29. Lo, A.W., MacKinlay, C.: Stock market prices do not follow random walks: evidence from a simple specification test. Rev Financ. Stud. **1**(1), 41–66 (1988)
30. Ramsey, J.B.: Tests for specification errors in classical linear least squares regression analysis. J. Roy. Stat. Soc. B **31**(2), 350–371 (1969)
31. Rosenberg, B.: The behavior of random variables with nonstationary variance and the distribution of security prices. Unpublished paper, Research Program in Finance, University of California, Berkeley (1972)
32. Schwarz, G.: Estimating the dimension of a model. Ann. Stat. **6**, 461–464 (1978)

33. Shepard, N.: Stochastic Volatility: Selected Readings. Oxford University Press (2004)
34. Tauchen, G., Pitts, M.: The price variability volume relationship on speculative markets. Econometrica 485–505 (1983)
35. Taylor, S.: Financial returns modelled by the product of two stochastic processes: a study of daily sugar prices 1961–79. In: Anderson, O.D. (ed.) Time Series Analysis: Theory and Practice, 1, pp. 203–226. North-Holland, Amsterdam (1982)
36. Taylor, S.: Asset Price Dynamics, Volatility, and Prediction. Princeton University Press (2005)

Empirical Test of the Balassa–Samuelson Effect in Selected African Countries

Joel Hinaunye Eita, Zitsile Zamantungwa Khumalo, and Ireen Choga

Abstract The purpose of this study investigates the validity of the Balassa–Samuelson effect in selected African countries. The kernel of the Balassa–Samuelson (BS) effect is the relationship between productivity and real exchange rate. The study, therefore, estimates the equilibrium real exchange with total factor productivity as the main explanatory variable. The results revealed that Balassa–Samuelson effect holds in the selected African countries. The results show a positive relationship between real exchange rate and productivity. An increase in total factor productivity causes real exchange rate appreciation. An improvement in productivity can cause countries to experience an increase in prices of their products relative to trading partners. The study recommends that the selected African countries should pursue policies that maintain competitive real exchange rate.

Keywords Real exchange rate · Productivity · Balassa–Samuelson effect

1 Introduction

One of the most important hypotheses with respect to the equilibrium real exchange rate level is the so-called Balassa–Samuelson hypothesis (see [1] and Samuelson, [2]); that is, the real exchange rate level is positively correlated with the development degree of the economy because of differential productivity growth between tradable and non-tradable sectors. The real exchange rate is influenced by many factors as stipulated in theories such as the Balassa–Samuelson theorem. The Balassa–Samuelson

J. H. Eita (✉) · Z. Z. Khumalo
School of Economics, University of Johannesburg, Johannesburg, South Africa
e-mail: jeita@uj.ac.za

Z. Z. Khumalo
e-mail: zee9119@gmail.com

I. Choga
Department of Economics, North-West University, Mmabatho, South Africa
e-mail: ireen.choga@nwu.ac.za

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_10

theory postulates the incidence of a positive correlation connecting the real exchange rate and the development of the economy because of differential productivity growth between tradable and non-tradable sectors.

The Balassa–Samuelson (BS) effect results from an extension of the purchasing power parity (PPP). Balassa [1] questioned the validity of the PPP as a theory that explained the determination of the equilibrium exchange rate [3]. The BS postulates that differentials in labour productivity between tradable and non-tradable sectors result in fluctuations of real costs. It also results in fluctuation of relative prices and cause divergences in the real exchange rate [4]. A country with more relative productivity advantage in tradable goods than in non-tradable goods ought to possess a higher real exchange rate [5]. According to [6], the BS effect defines volatility of real exchange rate through differences in productivity between tradable and non-tradable sectors of the economy.

The focus of the BS effect is on productivity difference between the economy and its trading partners. It postulates that productivity growth is generally biased in favour of the tradable goods sector. That means economies that experience relatively more productivity than other economies tend to have higher productivity in tradable compared to the non-tradable sector. According to Montiel [7], if there is higher productivity in the tradable sector, labour will move away from the non-tradable sector. This will increase costs in the non-tradable sector. This implies that in order to sustain profitability in the non-tradable sector, a higher relative price (of non-tradable goods) will be required.

The hypothesis emerged because of the difference in productivity growth among sectors and wages that are generally less differentiated. Normally, productivity grows rapidly in the tradable goods sector than in the non-tradable goods sector. Rapid productivity growth in the tradable goods sector raises wages in all sectors. The prices of non-tradable goods relative to the prices of tradable goods increase resulting in the growth of the overall price level. Moreover, the speed of productivity is faster in developing countries because of their attempt to catch up with developed countries [8].

The Balassa–Samuelson model employs the decomposition of the price level into tradable and non-tradable prices. Hence, the real exchange rate combines the real exchange rate for tradable goods and the ratio of the relative prices of tradable to non-tradable goods in two economies. Higher productivity growth in the tradable sector in one country implies that the relative non-tradable to tradable prices will increase more rapidly [9].

According to Montiel [7], agriculture and manufacturing are normally included in the tradable sector, while service sector is included in the non-tradable sector. The BS effect predicts that countries that have low productivity in the tradable compared to non-tradable goods tend to have lower prices than other countries. This is generally the case for many developing countries. This is the opposite of advanced economies, which tend to have productivity in the tradable sector. An increase in the prices of tradable goods causes a rise in the general price level (including the price of non-tradable goods). The price of non-tradable goods generally rises faster than that of tradable goods. The real exchange rate will appreciate. Poor and low-income countries tend to have low productivity in the tradable sector and this generally tends to reduce the general price level. The real exchange rate will then depreciate. This view is supported by Coudert [10] and Martinez-Hernandez [11]. Under the BS effect or hypothesis, higher profitability in the tradable division of rich nations raises the general level of costs and the genuine trade rates. Low efficiency in the tradable sector of poor nations is normally maintained or reduced to the general level of costs and more devaluated/deteriorated trade rates [11].

There are many studies which investigate the BS effect in advanced and developing economies (such as [12, 13]). Some other studies (such as Kakkar and Yan) computed the resulting real exchange rate misalignment. Others went further to test the effect of misalignment on economic performance [14, 15]. These previous studies examined the BS effect using inappropriate measure of technology or productivity. Relative GDP was used in many of these studies to proxy productivity and technology. The problem with relative GDP is that an increase in this variable should not necessarily be interpreted as a measure of technology.

Hence, it is important to use an appropriate measure of technology or total factor productivity. Contrary to previous studies, this study tests the Balassa–Samuelson effect using a different and appropriate proxy for total factor productivity or technology. This study computes total factor productivity by using the Cobb–Douglas production function. In line with Tintin [16], total factor productivity (TFP) computed using the Cobb–Douglas production function is a better representation of productivity or technology. This was supported by Eita et al. [17] who computed productivity using the production function for African countries. The rest of the study is organised as follows. Section 2 presents the literature review. Section 3 presents the methodology. Section 4 presents the empirical results, while the conclusion and recommendations are presented in Sect. 5.

2 Literature Review

2.1 Introduction

This section presents the theoretical foundations and empirical literature related to the Balassa–Samuelson effect. The empirical literature includes studies from developed and developing countries.

2.2 The Balassa–Samuelson Model

The Balassa–Samuelson model hypothesises that higher productivity differential in production of tradable goods between countries causes great differences in wages and in the prices of services. It also accounts for the pronounced differences between the purchasing power parity and equilibrium real exchange rate. The Balassa–Samuelson model is based on productivity differentials influencing the domestic relative price of non-tradable goods while divergences from PPP display disparities in the relative price of non-tradable goods [18]. Asea and Corden [18] provided an overview of the Balassa–Samuelson model as follows. The Balassa–Samuelson model comprises a small open economy consisting of capital and labour to produce tradable goods (T) which are priced in the world markets and non-tradable goods (NT) priced in the domestic market. Perfect mobility is presumed for capital and labour across all domestic sectors while labour is presumed to be immobile between countries and capital is not restricted internationally. The model also assumes that there is full employment in the economy. The model is presented as follows.

$$L = L_T + \ L_N \tag{1}$$

where the labour in the tradable sector is represented by L_T, while L_N is labour in the non-tradable sectors. To produce tradable and non-tradable goods, inputs of capital (K_T, K_N) and labour (L_T, L_N) are necessary. Linear homogenous functions describe technology in each sector:

$$Y_T = \theta_T K_T^{\beta T} L_T^{\alpha T} \equiv \theta_T L_T f(k_T) \quad \text{and} \quad Y_N = \theta_N K_N^{\beta N} L_N^{\alpha N} \equiv \theta_N L_N f(k_N) \tag{2}$$

where Y_T, Y_N represent the output in the tradable and non-tradable sectors while $k_T \equiv K_T/L_T$ and $k_N \equiv K_N/L_N$ and θ_T, θ_N are stochastic productivity parameters.

The world interest rate i is used as given. The presence of perfect competition equates the world interest rate to the value of the marginal product of capital in each sector:

$$i = \theta_T \beta_T k_T^{(\beta T-1)} \quad \text{and} \quad i = s\theta_N \beta_N k_N^{(\beta_N-1)} \tag{3}$$

where $s = P^N/P^T$ is the relative price of non-tradable goods (the real exchange rate).

$i = \theta_T \beta_T k_T^{(\beta T-1)}$ determines capital-labour in tradable goods sector (k_T). The two factors of production are utilised to obtain the factor price frontier by maximising profit $(F(K, L) - wL - rk)$ which in turn creates factor demand function in each sector. The notion of linear homogeneity allows the wage rate in the tradable sector to be represented by

$$\begin{aligned} w &= \theta_T[f(k_T) - f'(k_T)k_T] \\ &= \theta_T(1 - \beta_T)k_T^{\beta T} \end{aligned} \tag{4}$$

where $f''(k) < 0$ is an increasing function of k, meaning that $i = f'(k)$ is a decreasing function of w and i, therefore, decrease to the factor price frontier, a downward locus on the (w, i) plane with parameter k. Solving for k_T from $(i = s\theta_N \beta_N k_N^{(\beta N-1)})$ and substituting in $(w = \theta_T(1 - \beta_T)(\theta_T \beta_T/i)^{\frac{\beta_T}{1-\beta_T}})$ yields the wage equation:

$$w = (1 - \beta_T)(\theta_T \beta_{T/i})^{\frac{\beta T}{1-\beta_T}} \tag{5}$$

In a small economy, the determination of the wage (w) is reliant on factor productivity in tradable sector. The capital-labour ratio as derived from $i = s\theta_N \beta_N k_N^{(\beta_N - 1)}$ and results in

$$k_N = (s\theta_N \beta_N / i)^{\frac{1}{1-\beta_N}} \tag{6}$$

For perfect competition in the non-tradable sector, the following condition should hold

$$s = \theta_N f(k_N) = ik_N + w$$

(7)

From $(Y_N = \theta_N K_N^{\beta N} L_N^{\alpha N} \equiv \theta_N L_N f(k_N)$,$w = \theta_T(1 - \beta_T)(\theta_T \beta_T / i)^{\frac{\beta_T}{1-\beta_T}}$ and $k_N = (s\theta_N \beta_N / i)^{\frac{1}{1-\beta_N}})$ for given i the relative price of non-tradable goods is

$$\widehat{s} = \alpha N \widehat{w} - \widehat{\theta}_N \tag{8}$$

$$s = \frac{\alpha N}{\alpha T} \widehat{\theta}_T - \widehat{\theta}_N \tag{9}$$

where a hat signals the rate of percentage change. The relative price of non-tradable goods is dependent on the productivity differential in the tradable and non-tradable sectors.

Although the Balassa–Samuelson theory is employed to decipher economic issues by economists and policymakers, it is not without weaknesses. Bergin et al. [19] cited that productivity gains were not only limited to manufactured goods but also included gains from information technology and retail as assumed by the theory. The theory also overlooks services such as information sectors that are now becoming increasingly tradable due to technological advancements. Genius and Tzouvelekas [20] remonstrated the neglect of time-specific factors that potentially influenced the relationship between productivity and real exchange rates. They further mentioned that the assumption of unobservable country-specific factors impartially influencing the projected connection between labour productivity and real exchange rates was restrictive. However, the Balassa–Samuelson theory remains a popular choice amongst economists and policymakers to interpret various applied economic issues.

2.3 Empirical Literature

There is an extensive literature on the Balassa–Samuelson effect or hypothesis. There is a group of empirical studies conducted in developed economies. Ito et al. [21] investigated the Balassa–Samuelson hypothesis in high-growth Asian countries. A generally pronounced Balassa–Samuelson effect was observed in Japan, Korea and Taiwan. The study further suggested that the validity of Balassa–Samuelson hypothesis to an economy depended on the stage of development of that economy. The hypothesis is particularly suited for a rapidly expanding under resourced open economy. The expansion must entail a move from an industrial structure and export composition. However, a growing economy does not imply applicability of the Balassa–Samuelson if the economy has recently emerged from the primary goods exporter or planned economy phase.

Macdonald and Ricci [22] investigated the impact of the distribution sector on the real exchange rate, including the Balassa–Samuelson effect and other macroeconomic variables such interest rates, size of net foreign assets to GDP ratios for ten developed countries (Belgium, Denmark, Finland, France, Italy, Japan, Norway, Sweden, Germany and USA). A panel dynamic ordinary least squares estimator was employed to estimate long-run coefficients. The results revealed growth in productivity and competitiveness of the distribution sector caused an appreciation of the real exchange rate. Using ARDL estimation technique, Chowdhury [23] also found evidence of the Balassa–Samuelson effect in Australia for the period 1990–2003. Égert et al. [24] investigated the Balassa–Samuelson effect in nine Central and Eastern European countries. Panel cointegration techniques were employed and evidence of internal transmission mechanism was found. It was attributed to non-tradable inflation in the open sector because of productivity growth. The results indicated that an increase in productivity causes real exchange rate to appreciate. Kakkar and Yan [25] examined the Balassa–Samuelson effect for six Asian economies. The results indicated further that there was real exchange rate misalignment. The real exchange rate was misaligned.

Sallenave [14] investigated the Balassa–Samuelson effect in a study about the growth effects of real effective exchange rate misalignments for the G20 countries. Similarly, Vieira and MacDonald [15] studied the impact of real exchange rate misalignment on long-run growth for a set of ninety countries with adjustments for the Balassa–Samuelson effect by using real GDP per capita to account for the Balassa–Samuelson effect. They found that exchange rate misalignment impacted economic growth.

Egert et al. [24] explored the hypothesis in the Czech Republic, Hungary, Poland, Slovakia and Slovenia using time series and panel cointegration approaches. The results of the study presented a good application of the hypothesis in these transition economies for the period of 1991 Q1–2001 Q2. However, the study found that productivity growth did not entirely lead to price increments because of the construction of the CPI indexes. DeLoach [26] conducted a study to uncover evidence in support of the Balassa–Samuelson hypothesis. The results revealed a relationship consistent with the Balassa–Samuelson hypothesis, that of a significant long-run relationship between the relative price of non-tradable goods and real output.

Drine and Rault [12] conducted an empirical investigation and tested the validity of the Balassa–Samuelson effect or hypothesis in six Asian countries. A panel data cointegration procedure developed by Pedroni [27, 28] was used and further compared to the traditional Johansen cointegration test. A long-run relationship between real exchange rate and productivity differential was observed under the traditional time series model. However, advanced dynamic panel techniques showed contrary results. This was attributed to the absence of a positive long-run relationship between productivity differential and relative prices.

Tintin [16] investigated the Balassa–Samuelson hypothesis in ten OECD countries for the period 1975 and 2007. A country-specific analysis was conducted through the Johansen cointegration techniques and findings suggested that the BS hypothesis was valid in OECD countries. Gubler and Sax [13] investigated the robustness of the Balassa–Samuelson hypothesis for panel of OECD countries for the period of 1970–2008. The real exchange rate was conditioned on the measures of productivity for both the tradable and the non-tradable sector in addition to control variables such as the terms of trade and government spending share. The DOLS model specifications and the between-dimension group-mean panel FMOLS estimator from Pedroni were employed. The study did not find evidence of the Balassa–Samuelsson hypothesis.

There is also an extensive empirical literature on the relationship between real exchange rate in developing and emerging economies. Choudhri and Khan [29] tested for the Balassa–Samuelson in sixteen developing countries including African countries such as Kenya, Morocco, South Africa and Cameroon. The study showed that traded-non-traded productivity differentials were vital because they impact relative price of nontraded goods, and that the relative price applied a substantial effect on the real exchange rate. Likewise, the terms of trade influence the real exchange rate.

Omojimite and Oriavwote [30] examined the relationship between the Naira real exchange rate and macroeconomic performance and the Balassa–Samuelson hypothesis in Nigeria. The time-series data covered the period 1970–2009 and the Johansen cointegration procedure was employed. The parsimonious error correction model (ECM) results revealed a negative sign and a statistically significant one-period lag value of technological productivity. These results, therefore, implied the existence of the Balassa–Samuelson hypothesis in Nigeria. Increase in productivity causes real exchange rate appreciation in Nigeria.

Tica and Družić [31] investigated the Harrod–Balassa–Samuelson (HBS) effect on fifty-eight empirical papers. The evidence supported the HBS model, these results were influenced by the types of tests applied and set of investigated countries. Funda et al. [32] examined the Balassa–Samuelson effect in Croatia for the period 1998 Q1–2006 Q3. No evidence of the Balassa–Samuelson effect in Croatia was found.

Suleiman and Muhammad [33] conducted a study estimating the long-run effects of real oil price on real exchange rate by means of the Johansen procedure from 1980 to 2010 in Nigeria. The empirical analysis examined the effect of oil price fluctuations and productivity differentials (embodies the Balassa–Samuelson) on the real effective exchange. The result suggested that real oil price had a significant positive effect on the real exchange rate in the long run whilst productivity differentials had a significant negative influence on the real exchange rate. The productivity differentials were expressed against the trading partners of Nigeria. Contrary to Omojimite and Oriavwote's [30] results, this study found no evidence of the Balassa–Samuelson effect in Nigeria shown by the negative and significant coefficient on the productivity differential. The appreciation of the real exchange rate was attributed to improvements in oil prices, not the Balassa–Samuelson effect.

There is a group of studies which use a combination of developed and developing countries to test for Balassa–Samuelson effect. Bahmani-Oskooee and Nasir [34] estimated a random coefficients model permitting country and time-specific productivity effects. They employed an analytic framework expressing an individual country's productivity and real exchange rates relative to the United States (US). The study was for the period 1965–1992 and results revealed an invalid Balassa–Samuelson hypothesis for most African countries and some Latin American countries while it was valid for OECD countries and Asia. In an analysis of the long-run determination of exchange rates using sectoral data in twenty-four developing countries and fourteen OECD economies, Giacomelli [35] found results in support of the Balassa–Samuelson effect. While Faria and León-Ledesma's [36] revealed results unsupportive of the Balassa–Samuelson effect in the long run between two countries (the UK and US, German and Japan and Japan and the US). Genius and Tzouvelekas [20] tested for the Balassa–Samuelson hypothesis on fifty-nine industrialised and developing countries (including African countries such as Rwanda and Ivory Coast amongst others). Results of the study revealed that the hypothesis was invalid in most African countries and some Latin American countries. The hypothesis was held for OECD countries and Asia.

Based on the empirical inconclusiveness established in previous studies, this study investigated the Balassa–Samuelson effect in five African countries. A review of the empirical studies from both developed, emerging and developing economies most of them did not use proper proxies of technology or productivity. Most of these studies used relative real GDP or real GDP growth as a measure of productivity. Contrary to these previous researches, this study computes total factor productivity using the Cobb–Douglass production function as an appropriate measure of productivity.

3 Methodology

3.1 Model Specification

Following an extensive review of the literature, the empirical model is expressed as follows:

$$re_{it} = \alpha_0 + \alpha_1 pr + \alpha_2 tt + \alpha_3 fa + \varepsilon_t \tag{10}$$

where *re* is real exchange rate, *pr* is productivity, *tt* is terms of trade and *fa* is net foreign assets. The weighted average of a country's currency is relative to basket of major currencies as a proxy for *re*. An increase in *re* is appreciation while a decrease will be interpreted as depreciation. An increase in productivity is expected to lead to real exchange rate appreciation. The variable of interest, *pr* captures the Balassa–Samuelson effect, which hypothesises that rapid economic growth is associated with real exchange rate appreciation because of differential productivity growth between tradable and non-tradable sectors. Tintin [16] argues that total factor productivity is a better proxy for technology.

The effect of terms of trade on real exchange rate is ambiguous due to income and substitution effects. If income effect dominates, a rise in terms of trade permits an expansion of absorption and consequently an appreciation of the real exchange rate. However, if the substitution effect dominates, an increase in terms of trade causes real exchange rate depreciation. According to Lane and Milesi-Ferretti [37], net foreign assets are generally taken as cumulative current account of net capital transfers. The transfers are adjusted in order to take into account of capital gains and losses that result from inward and outward foreign direct investment. This also includes portfolio equity holdings. The effect of this variable is expected to be positive. According to Bleaney and Tian [38], the real exchange rate will appreciate if there is an increase in net foreign asset.

3.2 Data Description

The study uses annual data for the period 1991–2016. Five African countries are included in the study. These are Democratic Republic of Congo, Mauritius, Morocco, South Africa and Tunisia obtained from Quantec database. The data in Quantec are sourced from the IMF's International Financial Statistics, World Bank Development Indicators, central banks and statistics organisations of individual countries. The sample period and the countries were selected on the basis of consistent data availability. Real effective exchange rate, terms of trade, net foreign assets, labour, capital are directly available in the Quantec database. Total factor productivity is computed using the Cobb–Douglass production function as previously explained. It is computed as follows:

$$y = AK^{\delta}L^{\gamma}$$

$$A = \frac{y}{K^{\delta}L^{\gamma}} \tag{11}$$

where *y, A, K, L*, $\boldsymbol{\delta}$, $\boldsymbol{\gamma}$ are total output, technology, labour, capital, output elasticities of capital, output elasticities of labour. Total factor productivity is taken as an appropriate proxy for technology.

3.3 Estimation Technique

The Fully Modified OLS Model

The fully modified ordinary least squares (FMOLS) is employed to estimate the equilibrium real exchange rate (BS effect). The FMOLS estimator was developed to estimate directly cointegrating relationships. This is done through making adjustment to the traditional ordinary least squares. It corrects for endogeneity and serial correlation that normally occurs when using the traditional ordinary least squares. Previous studies confirmed that FMOLS is superior compared to other methods of estimating cointegrating relations. Studies such as Cappucio and Lubian and Hagreaves as well as Phillips [39] confirmed the advantages of FMOLS in estimating cointegrating relations and correcting serial correlations and endogeneity. Maddala and Kim [40] outlined the course of the FMOLS. It is important to have cointegration before estimation of the long and short-run empirical results. It is important to mention that the use of FMOLS suggests or implies that it is not necessary for the short run or error correction model.

Unit root test

It is important to mention that the univariate characteristics of the data is the first step before estimation of the empirical model. This involves panel unit root test. The study uses the Levin, Lin and Chu test (LLC Test), Im, Pesaran and Shin test (IPS) to test for unit root. Detailed discussion of these panel unit tests is not available due to space limitation, but can be obtained from the authors on request. If variables are non-stationary, it is important to test whether they are cointegrated. This study uses Kao test in order to establish if there is cointegration.

The Kao Cointegration Test

This study applies Chaiboonsri et al. [41] to test for panel cointegration. The variables as presented in Eq. (10) are assumed to be non-stationary. The detailed discussion of Kao cointegration are presented here because of space limitation, but can be obtained from the authors on request.

If there is cointegration, the real exchange rate model as presented in Eq. (10) will be estimated. The FMOLS as proposed by Hansen and Phillips [42] is estimated and it provides proper cointegration results that correct for endogeneity and serial correlation.

Table 1 Kao Cointegration Test Results

Cointegration test	t-statistic	Probability
Kao Test	−4.050	0.000*

The ADF is the residual-based ADF statistic. The null hypothesis is no cointegration. *Indicates that the estimated parameters are significant at the 5% level

4 Estimation Results

This section presents the empirical results of the stationarity tests, the real exchange rate cointegration test, long-run coefficient, fully modified OLS estimates (FMOLS) and real exchange rate misalignment and macroeconomic performance estimation.

4.1 Panel Unit Root (Stationarity) Tests

The variables were subjected to the LLC and the IPS stationarity tests. The results for panel unit roots are not presented here because of space limitation, but can be obtained from the authors on request. The results show that some variables are stationary while others are non-stationary. Since majority of the variables are non-stationary, it is decided that the next step should be to test for cointegration. Since there is cointegration, the next step is to estimated long-run results using FMOLS.

4.2 Cointegration Test Results

Table 1 presents the Kao panel cointegration test results. The decision rule of this test is rejecting the null hypothesis of no cointegration when the probability value is less than 5%. The results in this study are consistent with this rule, therefore, there is cointegration amongst the variables.

4.3 Long-Run Coefficient

The results in Table 1 indicate the presence of a cointegration relationship amongst the variables. The FMOLS is applied to estimate the long-run *re* model. The results are presented in Table 2.

Table 2 presents the long-run coefficients results of the FMOLS estimator. The results reveal that *pr* is statistically significant and consistent with economic theory. The variable *tt* is statistically significant and consistent with economic theory. The variable *fa* is not statistically significant and is in defiance of economic theory.

Table 2 FMOLS long run—estimation results. Dependent variable: *re*

Explanatory variables	Coefficients
pr	0.138 (0.094)*
tt	−0.665 (0.001)*
fa	−0.001 (0.542)
R-squared	0.920
S. E. of regression	0.200

**p*-values are in parentheses (); *10% statistically significant level; **5% statistically significant level; ***1% statistically significant level. An earlier version of these results in Table 2 was presented by Eita et al. [17]

A 1% increase in *pr* will appreciate the real exchange rate by 0.1% thereby indicating a positive relationship between the two variables as stipulated by economic theory. This indicates that there is evidence of BS effect in these countries. A 1% increase in *tt* will cause the real exchange rate to depreciate by 0.7%.

4.4 Real Exchange Rate Misalignment

Figure 1 presents actual and equilibrium real exchange rate. The computed real exchange rate misalignment is presented in Fig. 2. Figure 1 shows that there were more periods where the real exchange rate was undervalued. This is when compared to periods when the real exchange rate was overvalued. Overvaluation is not appropriate

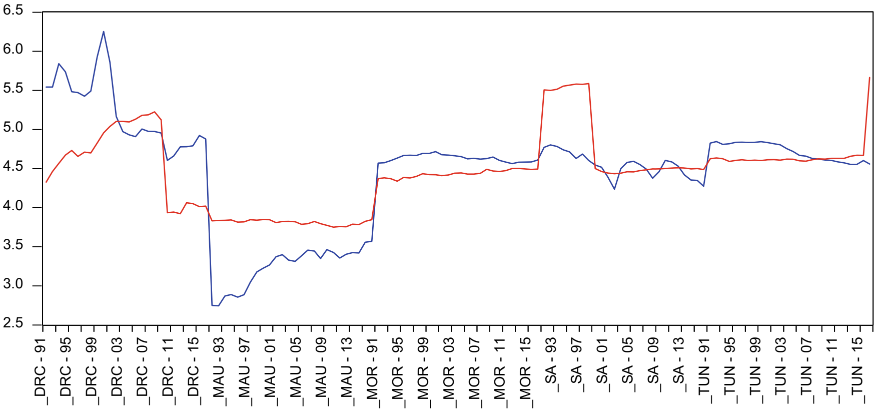


Fig. 1 Actual and equilibrium real exchange rate *Note* DRC, MAU, MOR, SA, TUN denote democratic republic of Congo, Morocco, South Africa and Tunisia. ERER is the equilibrium real exchange rate and RER is the actual real exchange rate. The earlier version of this figure was presented in Eita et al. [17]

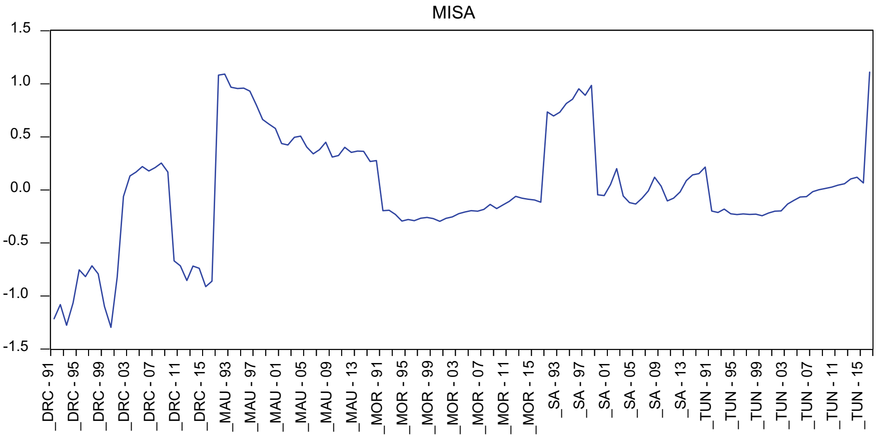


Fig. 2 Real exchange rate misalignment *Note* MISA denotes real exchange rate misalignment. DRC, MAU, MOR, SA, TUN denote Democratic Republic of Congo, Morocco, South Africa and Tunisia. The earlier version of this Figure was presented in Eita et al. [17]

because it has a negative effect on economic growth. This suggests that countries should come up with policies that minimise overvaluation of real exchange rate. This is supported by Gylfason [43] who argues that overvaluation worsens the trade balance. It also causes speculative attacks, increased foreign debt and discourages foreign direct investment (Fig. 2).

5 Conclusion

The study investigates whether the Balassa–Samuelson effect or hypothesis holds for selected African countries. If the hypothesis holds, then there should be a positive relationship between real exchange rate and productivity. This study differs from previous studies in the sense that it uses appropriate measure of productivity. It computed productivity using the Cobb–Douglass production function. The Balassa–Samuelson effect was tested for five selected African countries. The countries are Democratic Republic of Congo, Mauritius, Morocco, South Africa and Tunisia. The relationship between total factor productivity and the real exchange rate is positive. This confirms the validity of the Balassa–Samuelson effect. An increase in productivity in these economies is associated with an appreciation of the real exchange rate in these selected economies.

Undervaluation of the real exchange rate is appropriate for promoting economic growth and development in the selected African countries. These countries need to pursue economic policies in order to promote development and competitiveness of the economy. These countries should come up with policies that help to achieve and maintain a competitive exchange rate.

References

1. Balassa, B.: The purchasing power parity doctrine: a reappraisal. J. Polit. Econ. **72**(6), 584–596 (1964)
2. Samuelson, P. A.: Theoretical notes on trade problems. Rev. Econom. Stat. **46**(2), 145–154 (1964)
3. Moosa, I.: The US-China trade dispute: facts, figures and myths. Edward Elgar Publishing, Cheltenham (2012)
4. Asea, P.K., Mendoza, E.G.: The Balassa-Samuelson model: a general-equilibrium appraisal. Rev. Int. Econ. **2**(3), 244–267 (1994)
5. Mercereau, B.: Real exchange rate in an inter-temporal N-country-model with incomplete markets. ECB Working Paper No. 205 (2003)
6. Romanov, D.: The real exchange rate and the Balassa-Samuelson hypothesis: an appraisal of Israel’s case since 1986. Bank of Israel Discussion Paper, No. 2003.09. Banḳ Yiśra’el, Maḥlaḳat ha-meḥḳar, Jerusalem (2003)
7. Montiel, P.J.: Equilibrium real exchange rates, misalignment and competitiveness in the southern cone, vol. 62. United Nations Publications (2007)
8. Kharas, H.:The emerging middle class in developing countries. OECD Development Centre Working Paper, No (2010)
9. Driver, R., Sinclair, P., Thoenissen, C.: Exchange rates, capital flows and policy. Routledge, New York (2013)
10. Coudert, V.: Measuring the Balassa-Samuelson effect for the countries of central and eastern Europe? Banque de France Bull. Dig. **122**, 23–43 (2004)
11. Martinez-Hernandez, F.A.M.: The political economy of real exchange rate behavior: theory and empirical evidence for developed and developing countries, 1960-2010. Rev. Polit. Econ. **29**(4), 566–596 (2017)
12. Drine, I., Rault, C.: Does the Balassa-Samuelson Hypothesis hold for Asian countries? an empirical Analysis using panel data cointegration tests. Appl. Econometrics Int. Dev. **4**(4), 59–84 (2004)
13. Gubler, M., Sax, C.: The Balassa-Samuelson effect reversed: new evidence from OECD countries WWZ Discussion Paper, No. 2011/09. University of Basel, Basel (2011)
14. Sallenave, A.: Real exchange rate misalignments and economic performance for the G20 countries. Economia Internazionale **1**, 59–80 (2010)
15. Vieira, F.V., MacDonald, R.: A panel data investigation of real exchange rate misalignment and growth. Estudos Econômicos (São Paulo) **42**(3), 433–456 (2012)
16. Tintin, C.: Testing the Balassa-Samuelson hypothesis: Evidence from 10 OECD Countries (Master Thesis). University of Lund, Lund (2009)
17. Eita, J.H., Khumalo, Z.Z., Choga, I.: Productivity and real exchange rate: investigating the validity of the Balassa-Samuelson effect in five African countries. In: Valenzuela, O., Rojas, F., Pomares, H., Rojas, I. (eds.) Proceedings of Papers ITISE 2019 International Conference on Time Series and Forecasting, pp. 39–61. Godel Impressiones Digitales S. L., Granada (2019)
18. Asea, P.K., Corden, W.M.: The Balassa-Samuelson model: an overview. Rev. Int. Econ. **2**(3), 191–200 (1994)
19. Bergin, P.R., Reuven, G., Taylor, A.M.: Productivity, tradability and the long-run price puzzle. NBER Working Paper series, No. 10569. National Bureau of Economic Research, Cambridge, MA (2004)
20. Genius, M., Tzouvelekas, V.: The Balassa-Samuelson productivity bias hypothesis: further evidence using panel data. Agric. Econ. Rev. **9**(2), 31–41 (2008)
21. Ito, T., Isard, P., Symansky, S.: Economic growth and real exchange rate: an overview of the Balassa-Samuelson hypothesis in Asia. In: Krueger, O., Ito, T. (eds.) Changes in Exchange Rates in Rapidly Developing Countries: Theory, Practice, and Policy Issues, pp. 109–132. University of Chicago Press, Chicago (1999)

22. MacDonald, M.R., Ricci, M.L.A.: PPP and the Balassa Samuelson effect: the role of the distribution sector. IMF Working Paper, WP/01/38. International Monetary Fund, Washington (2001)
23. Chowdhury, K.: Modelling the Balassa-Samuelson effect in Australia. Australas. Account. Bus. Finance J. **5**(1), 77–91 (2011)
24. Egert, B., Drine, I., Rault, K.C.: Balassa-Samuelson effect", in Central and Eastern Europe: Myth or reality?. William Davidson Working Paper, No. 483. University of Michigan Business School, Michigan (2002)
25. Kakkar, V., Yan, I.: Real exchange rates and productivity: evidence from Asia. J. Money Credit and Banking **44**(2–3), 301–322 (2012)
26. DeLoach, S.B.: More Evidence in favor of the Balassa-Samuelson Hypothesis. Rev. Int. Econ. **9**(2), 336–342 (2001)
27. Pedroni, P.: Fully modifed OLS for heterogeneous cointegrated panels. Adv. Econometrics **15**, 93–130 (2000)
28. Pedroni, P.: Panel cointegration: asymptotic and finite sample properties of pooled time series tests with an application to the PPP hypothesis. Econometric Theo. **20**(3), 597–625 (2004)
29. Choudhri, E.U., Khan, M.S.: Real exchange rates in developing countries: are Balassa-Samuelson effects present? IMF Staff Pap. **52**(3), 387–409 (2005)
30. Omojimite, B.U., Oriavwote, V.E.: Real exchange rate and macroeconomic performance: testing for the Balassa-Samuelson hypothesis in Nigeria. Int. J. Econ. Fin. **4**(2), 127–134 (2012)
31. Tica, J., Družić, I.: The Harrod-Balassa-Samuelson effect: a survey of empirical evidence. EFZG Working Paper Series, 0607. University of Zagreb, Zagreb (2006)
32. Funda, J., Lukinić, G., Ljubaj, I.: Assessment of the Balassa-Samuelson effect in Croatia. Fin. Theory Pract. **31**(4), 321–351 (2007)
33. Suleiman, H., Muhammad, Z.: The real exchange rate of an oil exporting economy: empirical evidence from Nigeria. FIW Working Paper, No. 72. Dundee Business School, Dundee (2011)
34. Bahmani-Oskooee, M., Nasir, A.B.: Panel data and productivity bias hypothesis. Econ. Dev. Cult. Change **49**(2), 395–402 (2001)
35. Giacomelli, D.S.: Essays on consumption and the real exchange rate. Doctoral Dissertation. Massachusetts Institute of Technology, Massachusetts (1998)
36. Faria, J.R., Leon-Ledesma, M.: Testing the Balassa-Samuelson effect: implications for growth and the PPP. J. Macroecon. **25**(2), 241–253 (2003)
37. Lane, M.P.R., Milesi-Ferretti, M.G.M.: External capital structure: theory and evidence. IMF Working Paper, No WP/00/152. International Monetary Fund, Washington (2000)
38. Bleaney, M., Tian, M. Classifying exchange rate regimes by regression methods. Univ. Nottingham Sch. Econ. Discuss. Pap. **14**(02) (2014)
39. Phillips, P.C.: Fully modified least squares and vector autoregression. Econometrica **63**(5), 1023–1078 (1995)
40. Maddala, G.S., Kim, I.M.: Unit roots, cointegration, and structural change (No. 4). Cambridge University Press, Cambridge (1998)
41. Chaiboonsri, C., Sriboonjit, J., Sriwichailamphan, T., Chaitip, P., Sriboonchitta, S.: A panel cointegration analysis: an application to international tourism demand of Thailand. Ann. Univ. Petrosani Econ. **10**(3), 69–86 (2010)
42. Hansen, B.E., Phillips, P.C.: Estimation and inference in models of cointegration: a simulation study. Adv. Econometrics **8**, 225–248 (1990)
43. Gylfason, T.: The real exchange rate always floats. Aust. Econ. Pap. **41**(4), 369–381 (2002)

Energy Time Series Forecasting

End of Charge Detection by Processing Impedance Spectra of Batteries

Andre Loechte, Ole Gebert, and Peter Gloesekoetter

Abstract During the development of new battery technologies, high production tolerances are likely to occur due to the number of manual manufacturing steps. When putting these prototypes into operation, one of the most critical parameters is the reliable state of charge detection. This can be challenging when parameters like the capacity or the end of charge voltage are not precisely known due to the above-mentioned tolerances. In the majority of cases overcharging should be avoided as it harms the battery. This paper proposes a new criterion for detecting the end of the charging process that is based on the rate of change of electrochemical impedance spectra of the examined batteries. Device parameter fluctuations influence every measurement. Therefore, using the rate of change offers the advantage of using relative values instead of absolute values.

Keywords Electrochemical impedance spectroscopy · Battery analysis · State of charge

1 Introduction

As part of the EFRE-0801585 research project, a battery system was developed that uses rechargeable zinc-air batteries. An important research topic is the end of charge detection, because the battery voltage is not suitable for the state of charge and end of charge detection. The problem is intensified by the fact that the tolerances of the manufactured batteries are still very large due to the novelty of the technology. This publication describes a new procedure for the end of charge detection that works for zinc-air batteries and can also be applied to other cell technologies.

A. Loechte (✉) · O. Gebert · P. Gloesekoetter
Department of Electrical Engineering and CS, University of Applied Sciences Muenster,
Stegerwaldstr. 39, 48565 Steinfurt, Germany
e-mail: a.loechte@fh-muenster.de

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_11

Motivation Manufacturers usually have large tolerances during the break-in process of a new technology. Frequently, the situation is tightened by the lack of process automation. The resulting technological cell properties such as the porosity of the electrodes, the resistance of the contacts or the amount of active materials are therefore not constant and to a greater or less extend unknown [1–3].

These tolerances are particularly problematic when they can lead to dangerous situations. One of the most important exercises is finding a criterion which detects the end of the charging process of the battery. This can be challenging when parameters like the capacity or the end of charge voltage are not precisely known due to the tolerances. Furthermore, new battery types do not necessarily rely on the same stopping criteria. For example, zinc-air secondary batteries do not offer an end of charging voltage. Its charging current is not going to decrease when the battery is completely charged and the charging voltage is held at a fixed value. But instead of de-oxidising zinc oxide, hydrogen is produced [4].

In the majority of cases overcharging should be avoided as it harms the battery [5]. Another even more dangerous consequence is the possibility of an explosion. Especially, lithium-based batteries are known for their need for compatible ambient and charging parameters [6].

Problem There are two problems for detecting the end of charge when working with zinc-air secondary cells. On the one hand, the internal resistance of the batteries is still relatively high compared to other battery technologies, so that manufacturing tolerances and the charging current itself have a high influence on the measured cell voltage [1]. On the other hand, alkaline electrolyte, which has a tendency to electrolyse, is commonly used. The required voltage for activating the electrolysis process is in the range of the cell voltage at the end of a charging process. Therefore, a voltage measurement does not indicate whether the applied current is used to increase the state of charge of the battery or whether it performs an electrolysis [7].

Existing methods generally use fixed voltage limits to determine the state of charge. During operation, however, they are difficult to apply due to fluctuating tolerances. For example, an unknown internal resistance means that the internal cell voltage cannot be determined either. Here, electrochemical impedance spectroscopy is a promising measurement method.

Idea The idea behind the new end of charge criterion is to use the rate of change for electrochemical impedance spectra. Impedance spectra are multiple measurements of the impedance, that is, the AC resistance, at different frequencies. These different measurements can be combined to form a spectrum. The spectra describe the chemical processes within the cell and depend, for example, on the state of charge or the state of health [8]. Of course, the spectra are also influenced by parameter tolerances. Therefore, not the absolute impedance values are evaluated, but their rate of change. Then, absolute parameter fluctuations are less dominant when subtracting one spectrum from another.

2 Data Generation

2.1 Equipment

The measurement setup is shown in Fig. 1. A computer with MATLAB is used to control a PC oscilloscope with a built-in arbitrary waveform generator and two voltage measurement units (In 1 and In 2). During measurement, the waveform generator outputs a sine wave that works as input signal for the current controller [9]. The resulting alternating current component is applied to the battery under test (BUT) and the voltage of a shunt resistor that is equivalent to the current is measured by one of the voltage measuring units. The resulting alternating voltage response of the battery is also measured at once. Then a Discrete Fourier Transform is used to calculate the impedance at the given frequency. This procedure is repeated for a logarithmically distributed set of frequencies between 100 mHz and 5 kHz. In order to increase the precision of the voltage measurement, a microcontroller eliminates the constant component of the battery voltage by a successive approximation. Without this compensation, the alternating component could barely be measured as its amplitude of about 10 mV is quite small compared to the direct component of about 2.45 V. As the minimal voltage range of the 8-bit voltage measuring unit is ±50mV, this results in an effective number of 5.67 bits to measure the alternating component of the voltage response. In a practical implementation, a microcontroller with built-in ADC and DAC will later replace the computer and the oscilloscope, respectively.

Figure 2 shows the circuit of the current controller that is based on the picoEIS impedance analyser introduced by Martin Kiel. The actual current is measured by a shunt resistor and an instrumentation amplifier that references the differential voltage to Ground potential. The output of the instrumentation amplifier is also used by the oscilloscope for measuring the current. An analogue PID controller compensates for the difference between the desired current and the actual current. Since series resistances of batteries are quite small, high stimuli are necessary. Therefore, the OPA549 operational amplifier is used to amplify the current output of the controller [9].

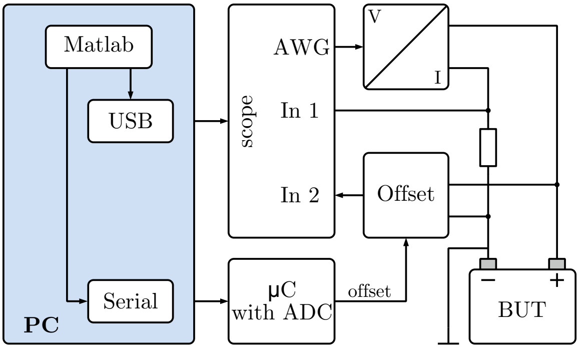


Fig. 1 Structure of measurement equipment

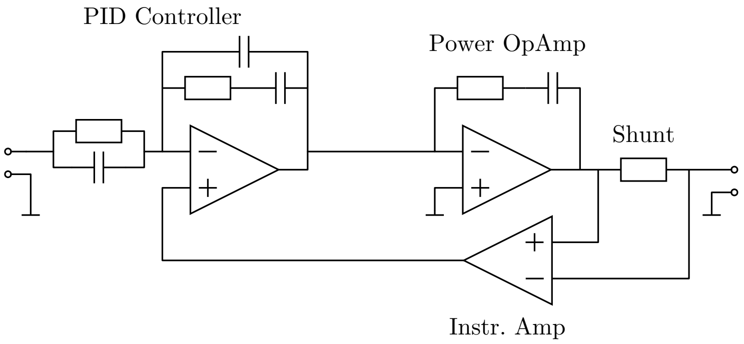


Fig. 2 Schematic diagram of current amplifier

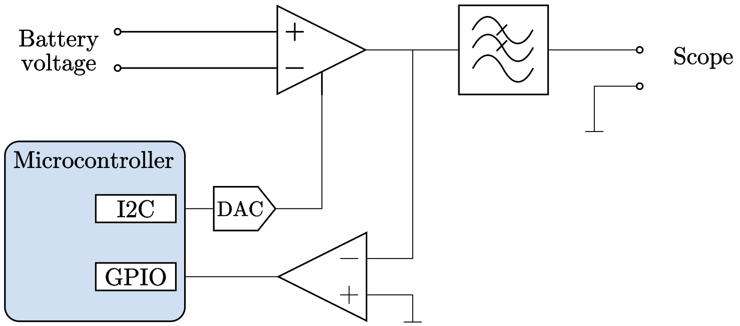


Fig. 3 Schematic diagram of measurement preprocessing

Figure 3 illustrates the offset compensation of the battery voltage signal. An instrumentation amplifier provides high-impedance inputs that are separately connected to the battery electrodes. Furthermore, the amplifier features a reference input that is used to shift the output voltage. The output of the amplifier is connected to a comparator that compares the shifted signal with Ground potential. First, the microcontroller sets the most significant bit of the DAC value which drives the reference signal. Then, it checks whether an overcompensation occurred by monitoring the output of the comparator. If the output signal of the amplifier is still above Ground potential the bit stays set, otherwise it is reset. Then the second most significant bit of the ADC is set and these steps are repeated until all bits have been configured. Finally, a low-pass filter implements an anti-aliasing filter.

2.2 Software

The software for generating data is implemented in MATLAB. A state machine is used for periodically measuring impedance spectra while cycling the battery. Its state

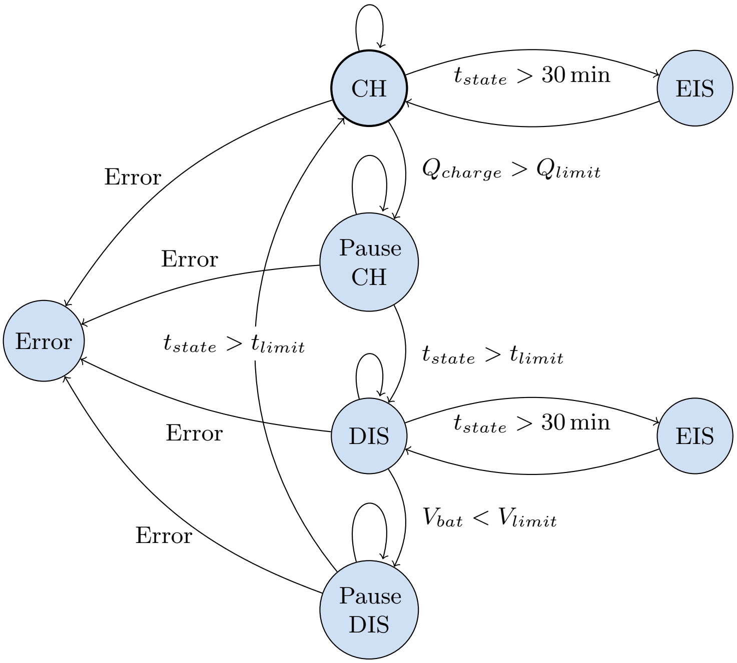


Fig. 4 State diagram of cycling algorithm

diagram is shown in Fig. 4. The machine starts with the charging process (CH). While charging the cell, the charge current and the cell voltage are logged. By disabling the offset compensation of the voltage signal, the oscilloscope is able to measure both signals. Furthermore, impedance measurements are performed every 30 min (EIS). Since the generated data is used to develop a new end of charge criteria, the impressed charge Q_{charge} is evaluated as the end of charge detection during data generation by integrating the charge current. The charging process is stopped when the estimated battery capacity is reached. The discharging process (DIS) starts after a small recovery break (Pause CH). Once again, impedance measurements (EIS) are performed every 30 min and the battery current and voltage are logged. When discharging the battery, the cell voltage V_{bat} can be used as a stopping criteria since a major voltage drop occurs at the end of the discharging process. Then, after a short break (Pause DIS), the charging process starts again. During all these described states, the battery voltage, current and the communication to the hardware equipment is evaluated. In case of an improper behaviour or a communication error, the error state (Error) is executed and the stimulation of the battery is stopped.

The EIS measurement consists of several independent impedance measurements at different frequencies. Each of these measurements starts with compensating the voltage offset. Then, the alternating current is applied to the cell. Now, the sampling of the current and the voltage signal starts for at least 3 periods or 1 s depending on what takes longer. Finally, the alternating stimulus and the offset compensation are stopped. These steps are repeated for a set of different frequencies. The stored results comprise the time, state, transferred charge since start of state, frequencies that were measured and voltage as well as the raw data of momentary current.

3 Data Processing

3.1 Processing of Raw Data

The first step of the data processing is the calculation of impedance values from the raw data of the measured voltage and current signals. Due to the fact that the measuring time of small frequencies lasts up to 30 s, the direct component of the voltage measurement might change during the measurement. The voltage change is particularly large at the beginning of charging or discharging processes due to the initially high chemical diffusion processes. Figure 5 shows exemplarily the voltage signal of such a case.

The error that arises by this reason is minimised by subtracting a linear function that models the development of the direct voltage component. Linear functions are determined by two points. Therefore, the mean values of the first sine period (p_1) and the last sine period (p_2) of a voltage signal is determined. Since the average of both half-waves of one period zeros out, the remaining average value gives the offset value $p_{i,v}$ of that period. The corresponding time component $p_{i,t}$ of the centre point is the mid time point of that period. Therefore, the point (p_1) of the first period is given by its components:

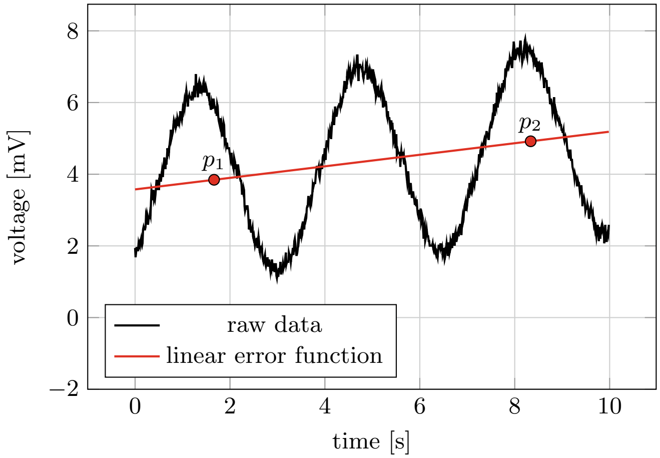


Fig. 5 Measured voltage signal whose direct component increases during the charging process

$$p_{i,v} = \sum_{s=0}^{s_p} \frac{v_s}{s_p},$$

$$p_{i,t} = \sum_{s=0}^{s_p} \frac{t_s}{s_p} = \frac{1}{2 \cdot f}.$$

Here, s implements a control variable that steps through the voltage measurement samples v_s and their corresponding time points t_s. The number of values that correspond to the period is given by s_p and the frequency that is measured is given by f_k. The equation of a line that is described by p_1 and p_2 is then subtracted from the raw measurement data. As one can see in Fig. 6, the error is almost completely eliminated.

Then, the corrected voltage signal and the measured current signal are being Fourier transformed to the transformed signals $\underline{V}$ and $\underline{I}$. Since only the impressed frequency is evaluated, the Goertzel algorithm is used in order to save computational effort [10]. Finally, the actual impedance $\underline{Z}$ of frequency k is calculated by

$$\underline{Z}_k = \frac{\underline{V}_k}{\underline{I}_k}.$$

Several impedances for a set of different frequencies are measured quickly one after the other and can be connected to a spectrum. The resulting spectra of one charging cycle are shown in Fig. 7. The colour of each characteristic specifies the time point of the impedance measurement. Since a constant charging current is used, the colour also represents the state of charge of the battery. Red spectra indicate an empty battery (SoC = 0%) while red spectra belong to a fully charged or even an overcharging battery (SoC = 100%). Each data point of a spectrum corresponds to one measured frequency.

Typically, impedance spectra of batteries consist of several semicircles that are produced by resistance and capacity combinations that model chemical reactions

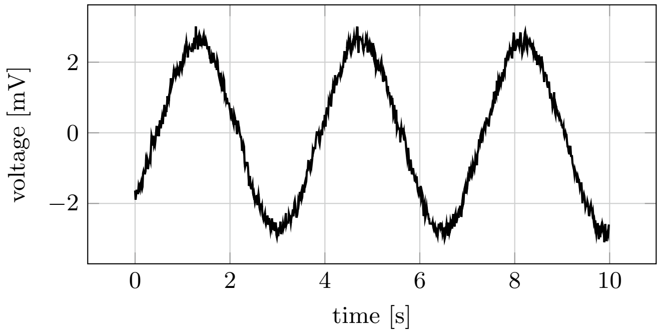


Fig. 6 Adjusted data after subtracting linear error function

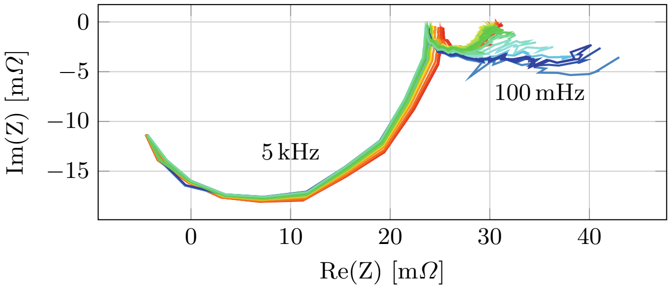


Fig. 7 Impedance spectrum colour weighted from red (SOC = 0) to blue (SOC = 100)

such as the diffusion process and double layer capacities [11–13]. The resulting spectra do not alone depend on the state of charge, but on other parameters like the working point, state of health, temperature, or the oxygen content as well. But the current work focuses on the state of charge. While the differences of the spectra between a full and an empty battery become small for high frequencies (left part of the Nyquist plot), most variation can be found for impedance values building the semicircle on the right-hand side. Therefore, the proposed method generates circle models of the right semicircles and uses the development of their radii [14].

3.2 Processing of Spectra Data

There are two challenges when creating a circle model based on a spectrum. Firstly, the semicircle on the right-hand side needs to be separated from the rest of the spectrum. For this type of battery, the development of the angle ϕ between the impedance values and the real axis is a good feature for determining the splitting point. The development is shown in Fig. 8. The first frequency index $k = 0$ corresponds to the lowest measured frequency. Therefore, the values on the left-hand side in Fig. 8 correspond to values on the right-hand side of Fig. 7. Since the radii of the high-frequency circles tend to be much bigger than the radii of the important semicircles, the angle values decrease strongly when reaching the frequencies of the left-hand semicircle.

The developed criterion is supposed to be robust to new battery prototypes and independent from absolute values. That is why the derivative of the development with respect to the frequency is used. Since a set of fixed frequencies is measured, the derivative can be specified as a difference quotient of two following impedance values with an interval of 1 (diff function). Furthermore, the threshold for finding the index of the splitting point is based on the mean value of the spectrum itself. The index of the splitting point is chosen to be

$$k_{split} = \frac{\partial\phi(Z_k)}{\partial k} > \left[5 \cdot RMS(\frac{\partial\phi(Z_k)}{\partial k}),\right]$$

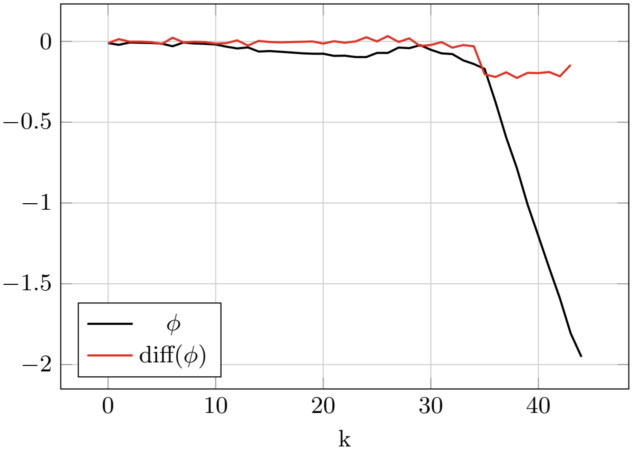


Fig. 8 Development of angles of impedance values of one spectrum

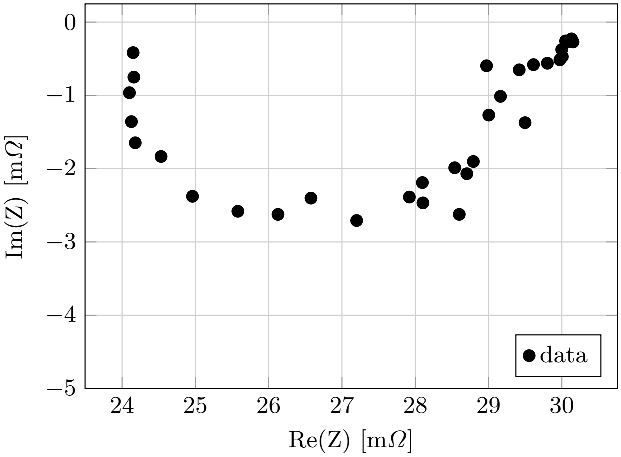


Fig. 9 Resulting spectrum after cutting process

more specifically

$$k_{split} = \phi(Z_k) - \phi(Z_{k-1}) > \left[5 \cdot RMS(\phi(Z_k) - \phi(Z_{k-1}))\right].$$

Figure 9 shows one separated spectrum after the cutting process. As one can see the cutting algorithm works quite well. However, the data is interfered with a lot of noise due to the low-cost measuring setup. Therefore, the second challenge is removing outliers from the spectrum before modelling the circles.

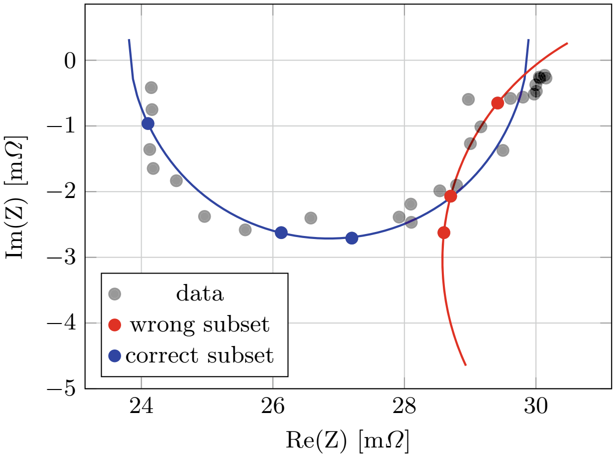


Fig. 10 Visualisation of RANSAC algorithm

Here, the RANSAC (random sample consensus) algorithm gives good results by finding outliers that are not used to calculate the circle model [15]. Figure 10 illustrates the algorithm. First, three impedance points are selected randomly and used to create an initial circle model. These three pints can either lead to a good model (blue points) or to a bad model (red points). Then, all other impedance values are tested against that model. If the distance between an impedance value and the circle model is lower than 2% of the maximum absolute value of that spectrum, it is considered as an inlier and the number of inliers is summed up. Good models are characterised by a high number of inliers. These steps are repeated for 15 sets of randomly chosen starting values. Finally, the algorithm picks the model with the highest number of inliers.

After this, an optimised circle model is determined that uses all the impedance values that are supposed to be inlier. Here, the optimised circle optimises the mean squared error according to Bucher [16]. The relation of a circle is given by

$$(x - x_c)^2 + (y - y_c)^2 = r^2$$

where x_c and y_c denote the centre of the circle and r the radius. Substituting

$$A = x_c^2 + y_c^2,$$

$$B = 2 \cdot x_c,$$

$$C = 2 \cdot y_c$$

results in a linear system of equations:

$$\begin{bmatrix} 1 & -x_1 & -y_1 \\ 1 & -x_2 & -y_2 \\ 1 & -x_3 & -y_3 \\ \vdots & \vdots & \vdots \end{bmatrix} \cdot \begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} x_1^2 + y_1^2 \\ x_2^2 + y_2^2 \\ x_3^2 + y_3^2 \\ \vdots \end{bmatrix}$$

that is solved using the least-squares solution of the system. Finally, the actual radius is determined by inserting the solution into the equations above.

4 Evaluation

The idea of the algorithm is to detect the transition of chemical processes during charging and processes during overcharging. Separately, the spectra of these processes probably do not change that much. During the first phase, zinc is being deoxidised which increases the state of charge of the battery. When the charging process is close to finishing, an attending electrolysis process takes place which decomposes the electrolyte. Since they are completely different reactions, there is perhaps a detectable transition when the overcharging ratio increases. During this phase, the ratio of de-oxidising zinc becomes smaller while the electrolysis process becomes stronger. For that reason, the derivative of the radii with respect to the charged energy is analysed. Since the impedance spectra measurements were time discretely taken with a fixed charging current and a fixed sampling interval of 30 min, the difference between two following radii at measurement index n is used as derivative:

$$r'(n) = \frac{\partial r(n)}{\partial Q_{\text{charged}}(n)} = \frac{r_n - r_{n-1}}{Q_{\text{charged},n} - Q_{\text{charged},n-1}}.$$

The development of the absolute of the derivative during one charging cycle is shown in Fig. 11. It is rather small at the beginning of the charging cycle which means that the variation of the radii during the charging process is indeed quite small. Then, it increases rapidly when 60 A h has been charged. This behaviour probably corresponds to the change of process types that leads to an increased variation of the impedance spectra resulting in a higher derivative of the radii. After 85 A h of charging the variation becomes smaller once again. Here, the de-oxidising process stops completely and only the electrolysis process takes place. Since the ratio of the two processes is not changing anymore, the derivative becomes smaller.

The algorithm determines a threshold value that is based on the derivative values at the beginning of the charging cycle. Even if the examined battery got big tolerances, it is possible to act on the assumption that at least 33% of the aspired capacity

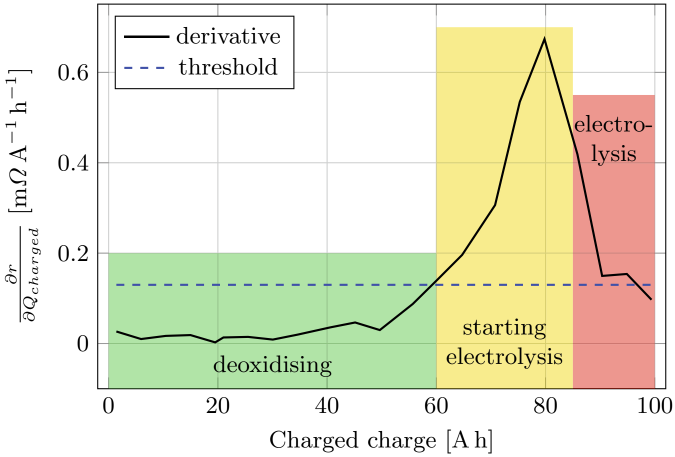


Fig. 11 Development of gradient of radius of the circle models. The characteristic is divided into several chemical processes

is reached. Thus, derivative values from the first third of the number of charging cycles N are used to calculate the threshold Δr_{limit}. More precisely, the algorithm uses the mean of the absolute values in that period. This also has the advantage of independence from absolute values.

$$\Delta r_{\text{limit}} = \sqrt{\frac{3}{N}\sum_{n=1}^{N/3}\left|r'(n)\right|} \cdot 8$$

The resulting threshold value is also included in Fig. 11. Now, a battery is considered to be full if the absolute derivative of the radius is greater than the comparison value:

$$\left|\frac{\partial r(Q_{\text{charged}})}{\partial Q_{\text{charged}}}\right| > \Delta r_{\text{limit}} \rightarrow \begin{cases} True & \text{Battery is full} \\ False & \text{Battery is not full} \end{cases}$$

In the case of this example battery, the target capacity of the produced cell is 100 A h. However, although the battery was charged for 60 h at 2 A resulting in a charged charge of 120 A h, only 60 A h could be taken during the subsequent discharging cycle. Thus, the criterion withstands practical measurements. Figure 12 shows the classification of each spectrum of the charging cycle. As expected the spectra of a charging battery are located densely in a small area. In contrast, the spectra of the overcharging battery vary greatly. The reason for this is that the spectra during the transition are also assigned to this class.

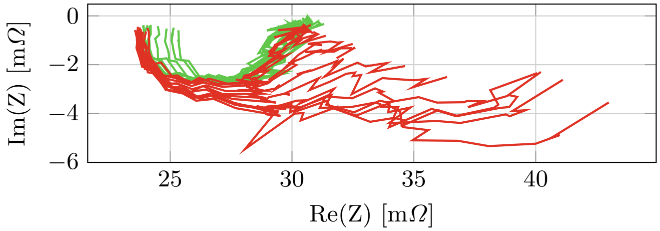


Fig. 12 Classification of impedance spectra into charging (green) and overcharging (red)

5 Conclusion and Outlook

A new criterion that is based on the rate of change of electrochemical impedance spectra for detecting the end of charge of batteries has been proposed. The radius of the most significant diffusion process in the spectrum is used as the decision criterion. For this purpose, the key semicircle is separated and the noise of the measurement data has been removed. By using the rate of change, absolute parameter fluctuations of the batteries can be shortened out. These parameter fluctuations occur mainly in the prototyping phase of development. The criterion was successfully applied to zinc-air battery prototypes. Here, the intended capacity was missed by a large amount. Nevertheless, the algorithm managed to determine the end of charge correctly. This could successfully be verified by massively overcharging a battery and comparing the estimated end of charge point to the actual drawn energy during the following discharge cycle.

Until now, the criterion has only been tested on rechargeable zinc-air batteries. Due to the increased measuring effort, the presented procedure is mainly worthwhile for battery technologies that cannot deduce the state of charge by a simple voltage measurement. Crucial for the application in other cell technologies is the finding and cutting out of decisive diffusion processes in the impedance spectra. Further testing and research are needed here.

The process is also planned to be integrated into a battery management system for rechargeable zinc-air batteries. This requires that both the hardware and the software are integrated into an embedded system. Since the expected impedances and frequencies are now known, we also hope to significantly increase the accuracy of the impedance measurement. If possible, this can reduce the necessary computational effort during preprocessing of the data and, for example, the use of the RANSAC algorithm can be dropped.

References

1. Fenske, D., Bardenhagen, I., Schwenzel, J.: Die Rolle der Gasdiffusionselektroden in der Zink-Luft- und Lithium-Luft-Batterie. Chemie Ingenieur Technik **9**(6), 707–719 (2019)
2. Zhang, X.G.: Secondary Batteries - Zinc Systems - Zinc Electrodes: Overview. In: Encyclopedia of Electrochemical Power Sources, pp. 454–468 (2009)
3. Caramia, V., Bozzini, B.: Material science aspects of zinc-air batteries: a review. Mater. Renew. Sustain. Energy, **3** (2014)
4. Loechte, A., Gebert, O., Kallis, K.T., Gloesekoetter, P.: State estimation of zinc air batteries using neural networks. Neural Comput. Appl. (2018)
5. Sexton, E.D., Nelson, R.F., Olson, J.B.: Improved charge algorithms for valve regulated lead acid batteries. In: Annual Battery Conference on Applications and Advances, vol. 15 (2000)
6. Kaypmaz, T.C., Tuncay, R.N.: Diagnosing overcharge behavior in operation of Li-ion Polymer batteries. In: 2012 IEEE International Conference on Vehicular Electronics and Safety (2012)
7. Fu. J., Liang, R., Liu, G., Yu, A., Bai, Z., Yang, L., Chen, Z.: Recent progress in electrically rechargeable zinc-air batteries. Adv. Mater. 2019, **31** (2018)
8. Barseoukov, E., Macdonald, J.R.: Impedance Spectroscopy: Theory, Experiment, and Applications. Wiley-Interscience (2005)
9. Kiel. M.: Impedanzspektroskopie an Batterien unter besonderer Berücksichtigung von Batteriesensoren für den Feldeinsatz. Aachener Beiträge des ISEA, Aachen (2013)
10. Goertzel, G.: An algorithm for the evaluation of finite trigonometric series. Am. Math. Mon. **65**, 34–35 (1958)
11. Drossbach, P., Schulz, J.: Elektrochemische untersuchungen an kohleelektroden I - die ueberspannung des wasserstoffs. Electrochim. Acta **9**, 1391–1404 (1964)
12. Arai, H., Mueller, S., Haas, O.: AC impedance analysis of bifunctional air electrodes for metal-air batteries. J. Electrochem. Soc. **147** (2000)
13. Helmholtz, H.: Studien ueber electrische Grenzschichten. Annalen der Physik und Chemie **243**(7), (1879)
14. Loechte, A., Gebert, O., Gloesekoetter, P.: End of charge detection of batteries with high production tolerances. In: International Conference on Time Series and Forecasting (ITISE) (2019)
15. Fischler, A.M., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**, 381–395 (1981)
16. Bucher, I.: Circle fit. https://de.mathworks.com/matlabcentral/fileexchange/5557-circle-fit?focused=5059278&tab=function (2004)

The Effect of Daylight Saving Time on Spanish Electrical Consumption

Eduardo Caro, Jesús Juan, Marta Maña, Jesús Rupérez, Carlos Rodríguez, Ana Rodríguez, and Juan José Abellán

Abstract In this work, two analyses are conducted to assess the impact of adopting Daylight Saving Time (DST) on power consumption in the Spanish Electric System. This study was carried out using the short-term electric load forecasting software currently in use in the Spanish Transmission System Operator (TSO). The forecasting software will simulate the case of electrical load in Spain without DST. The results obtained denote that DST may have a positive impact on reducing electric energy demand.

Keywords Electricity demand forecasting · Daylight saving clock change · Spanish electric power system

1 Introduction

Daylight Saving Time (DST) is the practice adopted by many countries worldwide of advancing clocks during summer months (usually from March until October) so that evening sunlight has a longer duration, while sacrificing normal sunrise times. Consequently, DST is a measure to improve the use of available daylight during the summer months, which results in a change in energy consumption.

Technical literature is rich in references concerning the effect of daylight-saving time change on electricity consumption [1, 2]. The effect has been analyzed in several countries and regions, such as Great Britain [3], Indiana [4], Ontario [5], Chile [6], Turkey [7], Southern Norway and Sweden [8], Jordan [9], Kuwait [10], Australia [11], Argentina [12], among others.

E. Caro (✉) · J. Juan (✉) · M. Maña
Universidad Politécnica de Madrid, Madrid, Spain
e-mail: eduardo.caro@upm.es

J. Juan
e-mail: jesus.juan@upm.es

J. Rupérez · C. Rodríguez · A. Rodríguez · J. J. Abellán
Red Eléctrica de España, Madrid, Spain

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_12

Most of the above works indicate that the implementation of DST results in a small reduction of electric energy consumption [1, 2]. In some studies, this effect has been quantified: in Jordan, the load decreases 0.2% in general (reductions for lighting, but increases for heating and cooling purposes) [9]; in Great Britain, in Chile and in Turkey, the reduction is estimated around 0.3% [3], 0.55% [6], and 0.7% [7], respectively. A higher reduction is reported in Southern Norway and Sweden, indicating a decrease at least 1% in both countries [8]. In Ontario, for the evening period, this reduction has been estimated to be 1.5%, approximately [5]. On the other hand, other studies indicate that although the effect on total consumption is negligible, it has a significant impact on the redistribution effect among hours; this is the case of Australia [11]. Finally, other works indicates that this reduction is not clear, or even mixed. This is the case of Argentina, observing an increment of total electric demand between 0.4 and 0.6%, but a decrease in the peak consumption between 2.4 and 2.9% [12].

Some analyses even show that DST implementation results in an increase in energy consumption. This is the case of Kuwait, reporting an increment of 0.07% [10]. In Indiana, an estimated 2–4% increase during the fall season, leads to a 1% increment considering the whole year.

To the best of the Authors' knowledge, not many works have been focused on the Spanish case. In this work, the impact of adopting Daylight Saving Time on consumption in the Spanish electric power system is assessed, using a detailed simulation-based analysis. The simulation has been performed using the short-term electric demand forecasting software currently used by *Red Eléctrica de España, REE* (the Spanish transmission system operator) [13], estimating the most-likely electric consumption without DST in Spain. Obtained results indicate that DST may have a positive impact on reducing electricity demand.

The study conducted in this article is limited to measuring changes in electricity demand on the days immediately following the time change. Firstly, using a demand prediction model, which predicts the hourly demand values for the ten days immediately following the case of DST removal in Spain. Secondly, by studying energy demand in the weeks immediately before and after the time change. If DST favors the saving of electrical energy, a significant reduction should be observed in March when comparing the weeks before and after the change of time. And vice versa, in October.

2 DST Effects on Consumption: Simulation-Based Analysis

In this section, the short-term electric load forecasting software is modified in order to consider the sunlight effect. Then a simulation is performed, comparing the case of (i) DST clock change, which is the real case, and (ii) disregarding the DST effect.

2.1 Procedure

In order to perform the simulation, the load forecasting model must be slightly modified first, to consider the daylight effect in a more realistic way. This procedure comprises two steps: first, the sunset/sunrise times must be computed for Spain. Second, the daylight duration information must be included in the model as an exogenous variable (regressor).

Step (1) Computation of sunset and sunrise times

To obtain the exact time of sunrise and sunset hours, we have made use of the Excel file created by the Department "*Earth System Research Laboratory*" (web www.esrl.noaa.gov) pertaining to the agency "*National Oceanic and Atmospheric Administration*" [14]. This datafile computes the sunrise/sunset moments given any geographical location determined by its latitude and longitude.

In this study, three Spanish cities are considered: Madrid (located in the central zone of the mainland), Barcelona (located in the Western region of the country) and Santiago de Compostela (located in the Eastern region of Spain). The sunrise and sunset times for the aforementioned three cities are provided in Fig. 1, for all de days of the year, considering UTC time.

As it can be observed in Fig. 1, daylength varies throughout the year, and there is a significant difference of sunrise/sunset times for the three selected cities: almost 45 min of difference between Barcelona and Santiago de Compostela. It should be noted that the curves in Fig. 1 are always valid, no matter which year is considered.

In order to validate the previous values, we have accessed to the webpage of the "*Spanish National Astronomical Observatory - National Geographic Institute*", from the Spanish Ministry of Development. In the web [15], a text file can be automatically generated containing the sunrise and sunset times for a specific year of any of the Spanish regions, considering local time. Figure 2 provides the sunrise ("Ort" column) and sunset ("Ocas" column) local time for Madrid during the year 2018.

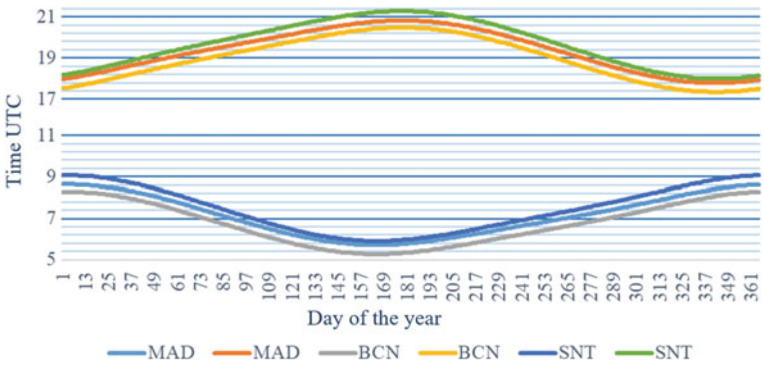
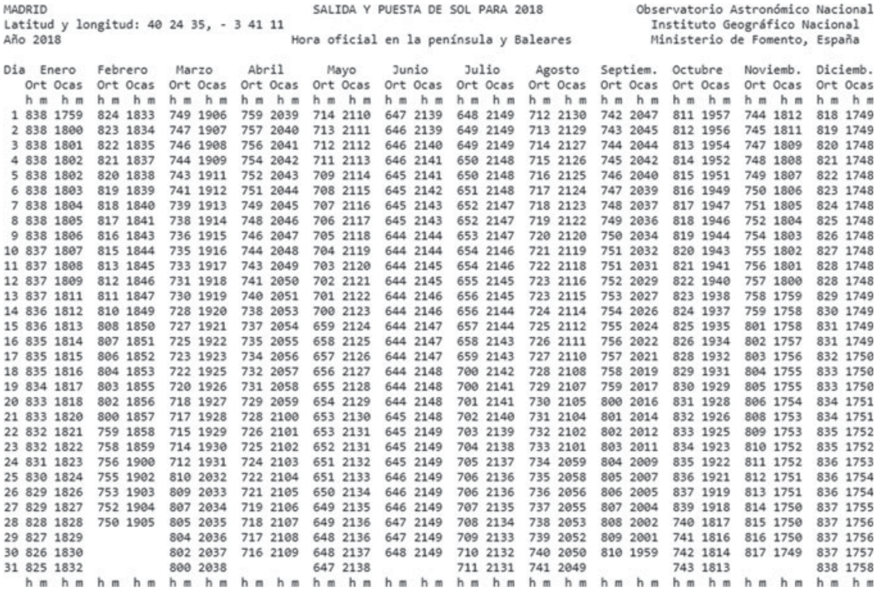


Fig. 1 Sunrise and sunset times for Madrid (MAD), Barcelona (BCN) and Santiago (SNT)



```
MADRID                                          SALIDA Y PUESTA DE SOL PARA 2018              Observatorio Astronómico Nacional
Latitud y longitud: 40 24 35, - 3 41 11                                                         Instituto Geográfico Nacional
Año 2018                                     Hora oficial en la península y Baleares            Ministerio de Fomento, España

Dia  Enero    Febrero    Marzo     Abril      Mayo      Junio     Julio     Agosto   Septiem.  Octubre   Noviemb.  Diciemb.
   Ort Ocas  Ort Ocas  Ort Ocas  Ort Ocas  Ort Ocas  Ort Ocas  Ort Ocas  Ort Ocas  Ort Ocas  Ort Ocas  Ort Ocas  Ort Ocas
   h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m
 1 838 1759  824 1833  749 1906  759 2039  714 2110  647 2139  648 2149  712 2130  742 2047  811 1957  744 1812  818 1749
 2 838 1800  823 1834  747 1907  757 2040  713 2111  646 2139  649 2149  713 2129  743 2045  812 1956  745 1811  819 1749
 3 838 1801  822 1835  746 1908  756 2041  712 2112  646 2140  649 2149  714 2127  744 2044  813 1954  747 1809  820 1748
 4 838 1802  821 1837  744 1909  754 2042  711 2113  646 2141  650 2148  715 2126  745 2042  814 1952  748 1808  821 1748
 5 838 1802  820 1838  743 1911  752 2043  709 2114  645 2141  650 2148  716 2125  746 2040  815 1951  749 1807  822 1748
 6 838 1803  819 1839  741 1912  751 2044  708 2115  645 2142  651 2148  717 2124  747 2039  816 1949  750 1806  823 1748
 7 838 1804  818 1840  739 1913  749 2045  707 2116  645 2143  652 2147  718 2123  748 2037  817 1947  751 1805  824 1748
 8 838 1805  817 1841  738 1914  748 2046  706 2117  645 2143  652 2147  719 2122  749 2036  818 1946  752 1804  825 1748
 9 838 1806  816 1843  736 1915  746 2047  705 2118  644 2144  653 2147  720 2120  750 2034  819 1944  754 1803  826 1748
10 837 1807  815 1844  735 1916  744 2048  704 2119  644 2144  654 2146  721 2119  751 2032  820 1943  755 1802  827 1748
11 837 1808  813 1845  733 1917  743 2049  703 2120  644 2145  654 2146  722 2118  751 2031  821 1941  756 1801  828 1748
12 837 1809  812 1846  731 1918  741 2050  702 2121  644 2145  655 2145  723 2116  752 2029  822 1940  757 1800  828 1748
13 837 1811  811 1847  730 1919  740 2051  701 2122  644 2146  656 2145  723 2115  753 2027  823 1938  758 1759  829 1749
14 836 1812  810 1849  728 1920  738 2053  700 2123  644 2146  656 2144  724 2114  754 2026  824 1937  759 1758  830 1749
15 836 1813  808 1850  727 1921  737 2054  659 2124  644 2147  657 2144  725 2112  755 2024  825 1935  801 1758  831 1749
16 835 1814  807 1851  725 1922  735 2055  658 2125  644 2147  658 2143  726 2111  756 2022  826 1934  802 1757  831 1749
17 835 1815  806 1852  723 1923  734 2056  657 2126  644 2147  659 2143  727 2110  757 2021  828 1932  803 1756  832 1750
18 835 1816  804 1853  722 1925  732 2057  656 2127  644 2148  700 2142  728 2108  758 2019  829 1931  804 1755  833 1750
19 834 1817  803 1855  720 1926  731 2058  655 2128  644 2148  700 2141  729 2107  759 2017  830 1929  805 1755  833 1750
20 833 1818  802 1856  718 1927  729 2059  654 2129  644 2148  701 2141  730 2105  800 2016  831 1928  806 1754  834 1751
21 833 1820  800 1857  717 1928  728 2100  653 2130  645 2148  702 2140  731 2104  801 2014  832 1926  808 1753  834 1751
22 832 1821  759 1858  715 1929  726 2101  653 2131  645 2149  703 2139  732 2102  802 2012  833 1925  809 1753  835 1752
23 832 1822  758 1859  714 1930  725 2102  652 2131  645 2149  704 2138  733 2101  803 2011  834 1923  810 1752  835 1752
24 831 1823  756 1900  712 1931  724 2103  651 2132  645 2149  705 2137  734 2059  804 2009  835 1922  811 1752  836 1753
25 830 1824  755 1902  810 2032  722 2104  651 2133  646 2149  706 2136  735 2058  805 2007  836 1921  812 1751  836 1754
26 829 1826  753 1903  809 2033  721 2105  650 2134  646 2149  706 2136  736 2056  806 2005  837 1919  813 1751  836 1754
27 829 1827  752 1904  807 2034  719 2106  649 2135  646 2149  707 2135  737 2055  807 2004  839 1918  814 1750  837 1755
28 828 1828  750 1905  805 2035  718 2107  649 2136  647 2149  708 2134  738 2053  808 2002  740 1817  815 1750  837 1756
29 827 1829            804 2036  717 2108  648 2136  647 2149  709 2133  739 2052  809 2001  741 1816  816 1750  837 1756
30 826 1830            802 2037  716 2109  648 2137  648 2149  710 2132  740 2050  810 1959  742 1814  817 1749  837 1757
31 825 1832            800 2038            647 2138            711 2131  741 2049            743 1813            838 1758
   h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m  h m
```

Fig. 2 Extract from the text file containing the sunrise and sunset times, from the *Spanish National Astronomical Observatory* (Web page: www.fomento.gob.es/salidapuestasol/2018/Madrid-2018.txt (accessed: 2019 June))

Since this database uses local time, and considering that the daylight-saving change day varies depending on the year, the text files downloaded from [15] are only valid for the specific year considered. In Fig. 2 it can be observed that the daylight saving changes occur in March 25th and October 28th, causing one hour difference of the sunrise/sunset time compared with the previous day.

Step (2) Implementation of the daytime regressor

The short-term electric load forecaster has been modified to include the daytime information. According to the previous plot, depending on the day of the year, the set of hours 6–7–8 a.m. and 6–7–8 p.m. may have sunlight or not. In other words, in Spain, there is always sunlight from 9 a.m. to 5 p.m., no matter the period of the year. Likewise, from 9 p.m. to 5 a.m., there are no sunlight in any day of the year. However, the rest of the hours, depending on the period of the year, may have sunlight or not.

A set of 24 dummy variables (one for each hour) has been created, modeling the daytime effect: $l_{h,d} \in [0, 1]$, where indexes h and d indicate the hour and the day. For each hour h, the parameter $l_{h,d}$ is set to one if the h-th hour for the d-th day has sunlight, $l_{h,d} = 0$ otherwise. Figure 3 provides the values for the parameter $l_{h,d}$ for a whole year and for hours comprised between 6 a.m. and 8 p.m., for the UTC time and local time cases. Parameter $l_{h,d}$ is included in the forecasting model as an exogenous variable (regressor).

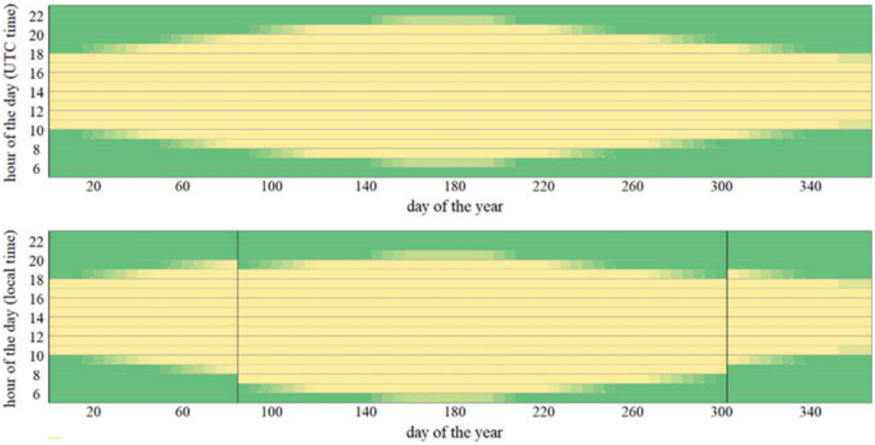


Fig. 3 Values for the parameters $l_{h,d}$ for hours $h \in [6, 22]$, for UTC and local times. Yellow and green color indicate $l_{h,d} = 1$ and $l_{h,d} = 0$, respectively

2.2 *Case Study*

Once the daytime information is included in the model, and considering the actual implementation of the DST effect in the algorithm [13], the Spanish load short-term forecasting software can be used to simulate the effect of considering/disregarding DST. In the following subsections the cases of March and October are studied, for year 2017.

Considering that the predicting model has been designed and created for short-term forecasts (from one to ten days ahead), this study analyzes the effect of DST on the local period close the clock-change day.

DST effect on March

Considering that the clock-change day took place on March 26th, 2017, the Spanish load forecasting model has been used to predict the load behavior in case of: (i) the DST effect and (ii) disregarding this effect. Both cases have been simulated at 0.00 h on March 26th, 2017, generating forecasts from one to ten days ahead.

Figures 4 and 5 provide the forecasted load values for cases considering DST (labeled as '*DST*') and neglecting DST (labeled as '*no DST*'), and the observed load. The difference between curves '*DST*' and '*no DST*' corresponds to the effect of DST removal.

The effect of removing DST can be observed in Fig. 5: electric consumption during 8 p.m. and 9 p.m. increases significantly.

From the daily consumption perspective, Table 1 provides the forecasted daily electric load for both cases (fourth and fifth columns), and the increment of daily demand in case of neglecting DST (sixth column). It can be observed that DST

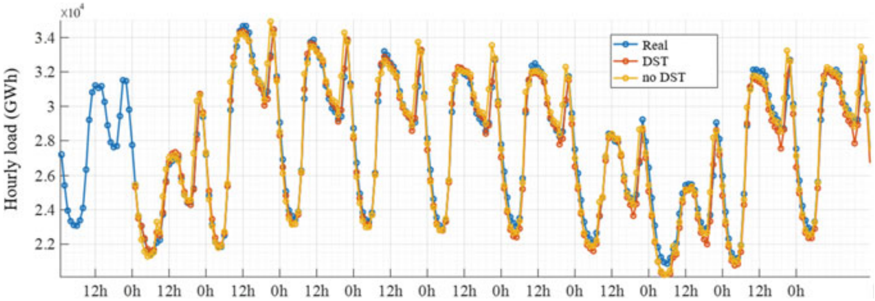


Fig. 4 Observed and forecasted electric load for the period: 25/03/2017–05/04/2017

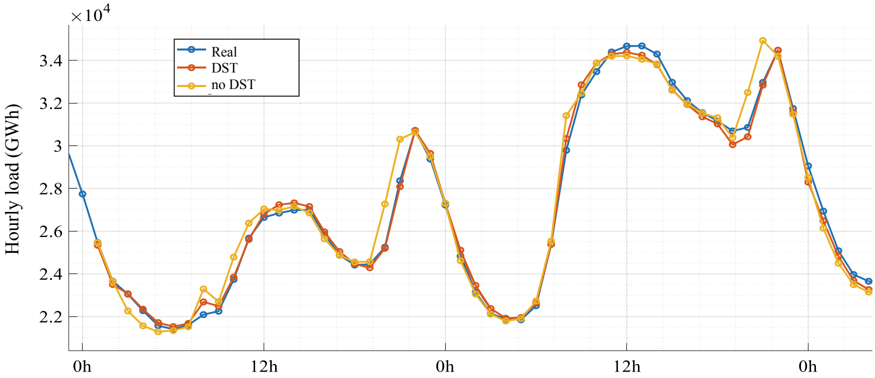


Fig. 5 Observed and forecasted electric load for DST on March 2017

Table 1 Forecasted daily load considering/neglecting DST: March 2017

March			Daily electric load (GWh)		Increment (%)
			DST	no DST	DST versus no DST
Sunday	29/10/2017	d + 1	603.0	606.9	0.64
Monday	30/10/2017	d + 2	711.2	715.2	0.55
Tuesday	31/10/2017	d + 3	708.0	712.4	0.62
Wednesday	01/11/2017	d + 4	696.0	699.2	0.46
Thursday	02/11/2017	d + 5	687.9	694.4	0.95
Friday	03/11/2017	d + 6	675.6	685.7	1.49
Saturday	04/11/2017	d + 7	604.2	612.5	1.38
Sunday	05/11/2017	d + 8	559.0	566.8	1.40
Monday	06/11/2017	d + 9	663.5	673.8	1.55
Tuesday	07/11/2017	d + 10	680.4	690.9	1.54
Average (first five days)			681.2	685.6	0.65
Average (first ten days)			658.9	665.8	1.06

Table 2 Effect of considering/neglecting the DST on hourly load: March 2017

MARCH		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Sunday	d+1	0.5	0.6	-3.5	-3.5	-2.0	-0.9	-0.7	2.7	0.9	3.9	3.0	0.9	-0.9	-0.6	-1.1	-1.3	-0.7	0.4	1.2	8.2	7.9	-0.3	-0.5	0.0
Monday	d+2	-2.0	-1.7	-1.2	-0.6	-0.2	0.4	0.6	3.5	-1.2	0.0	-0.3	-0.5	-0.6	0.1	-0.1	0.1	0.5	0.9	1.1	6.8	6.4	-0.8	-0.3	0.8
Tuesday	d+3	-1.3	-1.3	-0.8	-0.5	0.0	0.5	0.2	2.9	-1.6	-0.5	-0.4	-0.2	-0.2	0.4	0.4	0.4	0.6	1.0	1.3	6.7	6.5	-0.5	-0.2	0.6
Wednesday	d+4	-1.0	-1.1	-1.0	-0.8	-0.5	0.0	-0.2	2.7	-1.7	-0.7	-0.7	-0.7	-0.5	0.2	0.6	0.4	0.3	0.6	1.1	6.3	5.8	-0.5	0.2	1.4
Thursday	d+5	-0.4	-0.5	-0.3	-0.1	0.2	0.7	0.4	2.9	-1.3	-0.5	-0.5	-0.3	-0.1	0.6	0.6	0.8	0.6	1.0	2.1	7.3	6.7	0.1	0.5	1.6
Friday	d+6	0.7	0.5	0.7	0.9	1.3	1.9	1.5	3.6	-0.9	0.4	0.3	0.5	0.7	1.2	1.1	1.0	0.7	1.0	2.2	7.0	6.4	0.4	1.1	1.9
Saturday	d+7	0.7	0.4	0.5	0.7	0.7	1.4	0.9	3.4	0.1	0.8	0.5	0.1	-0.1	0.5	0.4	0.7	0.9	1.3	2.7	7.5	6.7	0.4	0.9	1.5
Sunday	d+8	0.4	0.3	0.1	0.4	0.8	1.3	1.0	3.1	0.4	1.4	0.7	0.3	-0.3	0.3	0.7	0.9	1.1	1.5	3.1	7.3	5.7	-0.2	1.0	2.3
Monday	d+9	0.8	0.7	0.7	0.7	1.1	1.6	1.3	3.2	-0.6	0.5	0.7	0.7	0.7	1.1	1.2	1.3	1.3	1.8	3.5	6.4	4.6	0.0	1.3	2.4
Tuesday	d+10	0.8	0.7	0.7	0.7	1.1	1.6	1.3	3.2	-0.6	0.5	0.7	0.7	0.7	1.1	1.2	1.3	1.3	1.8	3.5	6.4	4.6	0.0	1.3	2.4
Average (first five days)		-0.8	-0.8	-1.4	-1.1	-0.5	0.2	0.1	2.9	-1.0	0.4	0.2	-0.2	-0.5	0.1	0.1	0.1	0.3	0.8	1.4	7.0	6.7	-0.4	-0.1	0.9
Average (first ten days)		-0.1	-0.1	-0.4	-0.2	0.2	0.9	0.6	3.1	-0.6	0.6	0.4	0.2	-0.1	0.5	0.5	0.6	0.6	1.1	2.2	7.0	6.1	-0.2	0.5	1.5

reduces electricity consumption around 0.65–1.06% for the days following the clock-change day in March.

Table 2 shows the effect of DST on Spanish hourly load, for the period around the clock-change day of March 2017. As it can be observed, there is an increment of 6–7% at 8.00–9.00 p.m. Additionally, note that there is a redistribution of loads between 8.00 a.m. and 9.00 a.m.

DST effect on October

As in the previous subsection, the Spanish load forecasting model has been used to predict the load behavior with/without DST. During year 2017, the clock-change day took place on October 29th, 2017. Both cases have been simulated at 0.00 h on October 29th, 2017, generating forecasts from one to ten days ahead.

Figures 6 and 7 provides the forecasted load values for cases considering DST (labeled as '*DST*') and neglecting the DST (labeled as '*no DST*'), and the observed load. The difference between curves '*DST*' and '*no DST*' corresponds to the effect of not changing the clock on October 29th, 2017.

The effect of the clock change can be observed in Fig. 7: electric consumption during 8 p.m. increases significantly. It should be noted that the sunset time changes from 7.20 p.m. (28/10/2017) to 8.20 p.m. (28/10/2017). Consequently, public and private lighting electric consumption commences one hour before.

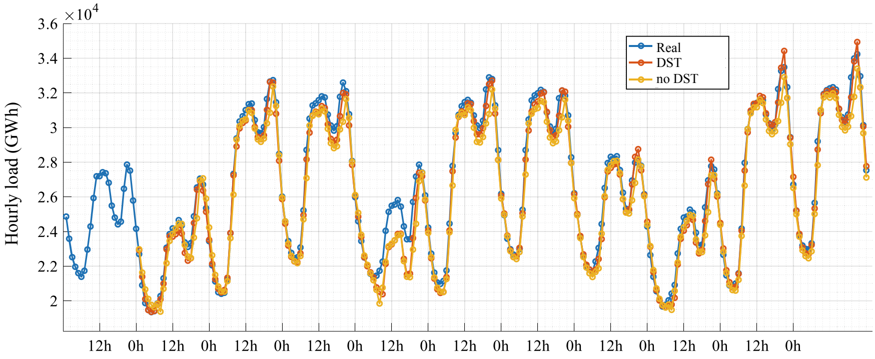


Fig. 6 Observed and forecasted electric load for the period: 27/10/2017–07/11/2017

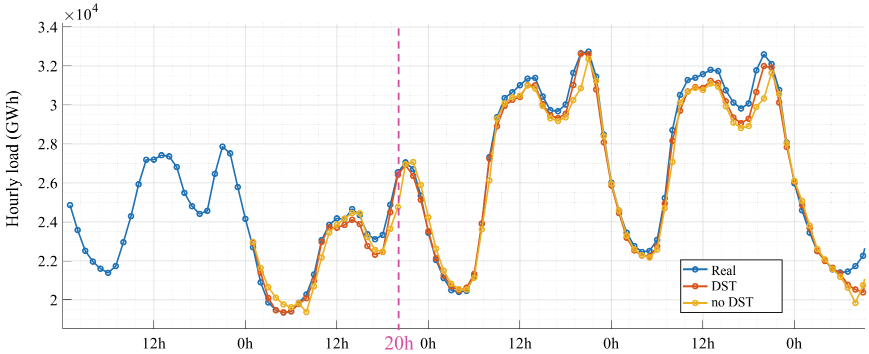


Fig. 7 Observed and forecasted electric load for the DST on October 2017

From the daily consumption perspective, Table 3 provides the forecasted daily electric load for both cases (fourth and fifth columns), and the increment of daily demand in case of removing the clock change in October (sixth column). It can be observed that the clock change increases electricity consumption around 0.65–1.06% for the following days after the clock-change day.

Table 4 provides the local effect of removing the clock change in October 2017 on the hourly demand. As it can be observed, there is a decrement −6% at 8.00 pm. Additionally, note that there is a redistribution of loads between 8.00 a.m. and 9.00 a.m.

Table 3 Forecasted load considering/neglecting the clock change: October 2017

October			Daily electric load (GWh)		Increment (%)
			DST	no DST	DST versus no DST
Sunday	29/10/2017	d + 1	546.2	548.1	0.35
Monday	30/10/2017	d + 2	658.6	655.9	−0.41
Tuesday	31/10/2017	d + 3	671.6	667.1	−0.67
Wednesday	01/11/2017	d + 4	557.0	553.9	−0.55
Thursday	02/11/2017	d + 5	663.9	657.9	−0.90
Friday	03/11/2017	d + 6	679.5	672.1	−1.10
Saturday	04/11/2017	d + 7	607.1	603.3	−0.63
Sunday	05/11/2017	d + 8	554.0	552.7	−0.24
Monday	06/11/2017	d + 9	676.5	665.9	−1.57
Tuesday	07/11/2017	d + 10	698.9	688.0	−1.55
Average (first five days)			619.4	616.6	−0.44
Average (first ten days)			631.3	626.5	−0.73

Table 4 Effect of clock change on hourly load: October 2017

OCTOBER		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Sunday	d+1	0.2	1.1	2.8	3.3	2.1	1.1	0.2	-3.6	-1.4	-3.5	-1.2	0.7	1.4	1.4	2.3	2.0	1.1	0.1	-3.5	-6.3	0.2	2.7	3.0	3.0
Monday	d+2	2.2	1.2	0.7	0.3	-0.5	-0.9	-1.3	-4.1	1.4	0.4	0.4	0.2	-0.1	-0.5	-0.3	-0.5	-0.5	-0.6	-2.5	-5.4	-0.7	1.5	1.1	0.3
Tuesday	d+3	0.6	0.5	0.3	0.1	-0.2	-0.7	-1.0	-3.8	1.4	-0.1	-0.1	-0.3	-0.4	-0.7	-0.9	-0.9	-0.9	-1.3	-2.5	-5.2	-0.9	1.4	0.7	0.1
Wednesday	d+4	0.9	0.5	0.5	0.4	-0.3	-0.8	-0.7	-3.4	1.8	0.5	0.2	0.2	0.0	-0.3	-0.2	-0.6	-0.6	-1.0	-2.8	-5.7	-1.8	0.7	0.2	-0.6
Thursday	d+5	0.3	0.3	0.5	0.3	-0.1	-0.5	-0.5	-3.0	1.5	0.1	-0.2	-0.4	-0.7	-1.2	-1.3	-1.5	-1.6	-1.7	-3.2	-5.3	-1.6	0.9	0.0	-0.7
Friday	d+6	-0.3	-0.2	-0.2	-0.1	-0.7	-1.1	-0.8	-3.1	1.4	-0.4	-0.9	-1.2	-1.3	-1.8	-1.9	-1.4	-1.2	-1.3	-2.7	-4.8	-1.4	1.1	-0.1	-0.9
Saturday	d+7	-0.3	-0.5	-0.7	-0.7	-1.0	-1.1	-0.6	-2.4	1.3	0.2	0.3	0.6	0.7	0.6	0.3	0.0	-0.2	-1.0	-3.4	-5.3	-2.1	0.8	-0.2	-0.8
Sunday	d+8	0.2	0.5	0.5	0.5	0.2	-0.1	0.5	-1.6	2.0	0.5	0.5	0.8	1.1	0.7	0.9	0.8	0.4	-0.6	-3.4	-6.0	-3.2	0.8	0.4	-0.5
Monday	d+9	-0.8	-0.7	-0.7	-0.7	-1.1	-1.6	-1.3	-3.2	0.6	-0.5	-0.7	-0.7	-0.7	-1.1	-1.2	-1.3	-1.3	-1.8	-3.4	-6.0	-4.4	0.0	-1.3	-2.4
Tuesday	d+10	-0.8	-0.7	-0.7	-0.7	-1.1	-1.6	-1.3	-3.2	0.6	-0.5	-0.7	-0.7	-0.7	-1.1	-1.2	-1.3	-1.3	-1.8	-3.4	-6.0	-4.4	0.0	-1.3	-2.4

Average (first five days)	0.9	0.7	0.9	0.9	0.2	-0.4	-0.6	-3.6	0.9	-0.5	-0.2	0.1	0.0	-0.2	-0.1	-0.3	-0.5	-0.9	-2.9	-5.6	-1.0	1.5	1.0	0.4
Average (first ten days)	0.2	0.2	0.3	0.3	-0.3	-0.7	-0.7	-3.1	1.1	-0.3	-0.2	-0.1	-0.1	-0.4	-0.3	-0.5	-0.6	-1.1	-3.1	-5.6	-2.0	1.0	0.3	-0.5

3 Randomized Block Design and Paired Data Analysis

To further complement the previous analysis, another study has been carried out regarding the daylight-saving time effect on the Spanish electric load. The focus of this study has been the analysis of historical data and two different models have been implemented, based on the well-established statistical techniques Randomized Block Design and Paired Data Analysis [16].

3.1 *Period of Study*

Since DST has been implemented in Spain every year since 1974, there is no data available of the electric load when this policy was not applied and, consequently, it is not possible to compare real data of applying and not applying this policy. For this reason, it has been decided to focus the study on a period surrounding both dates of the year in which the clocks are changed, analyzing the data from the days immediately before and after DST was applied.

A period of two weeks prior and two weeks after the implementation of the DST clock change has been considered (see Fig. 8), in order to locally evaluate its potential effect on the electric load.

The reason behind the selection of only four weeks for this analysis is due to some external variables that affect the data during such a period. The comparison between only two weeks would have disregarded important information surrounding the desired date, however, a period of six weeks would have meant studying dates

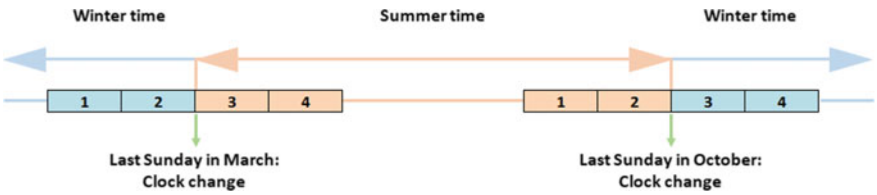


Fig. 8 Four-week period of data used to study DST locally

almost one month prior and after the DST, therefore implying an excessive variation in exogenous factors which could interfere with the obtained results.

3.2 Exogenous Factors Removal

The electric load is highly dependent on the temperature. With higher temperatures the electric consumption increases due to the use of air conditioners, similarly to what happens during the colder months with heating systems. As a consequence, peak consumption is found in summer and winter and a decrease during the rest of the year with milder temperatures.

As shown in Fig. 9, both clock-changes happen when the electric load is adjusting to the varying temperature. In March, the electric consumption diminishes as the temperature rises and, in October, it grows as winter approaches.

In addition to the temperature effect, holidays also play an important role in explaining the daily electric load. National, regional and local holidays reduce economic activity as service and industrial sectors cease their operations. Consequently, a reduction in consumption is observed on holidays and on the days around them.

It should be noted that two national holidays fall on both analyzed four-week periods. Easter is usually celebrated in a week at the end of March or the beginning of April, so, for most years considered, one of the four weeks around the March clock-change would be affected. In addition, November the 1st is All Saints' Day, which coincides with the third week studied for the October clock-change every year.

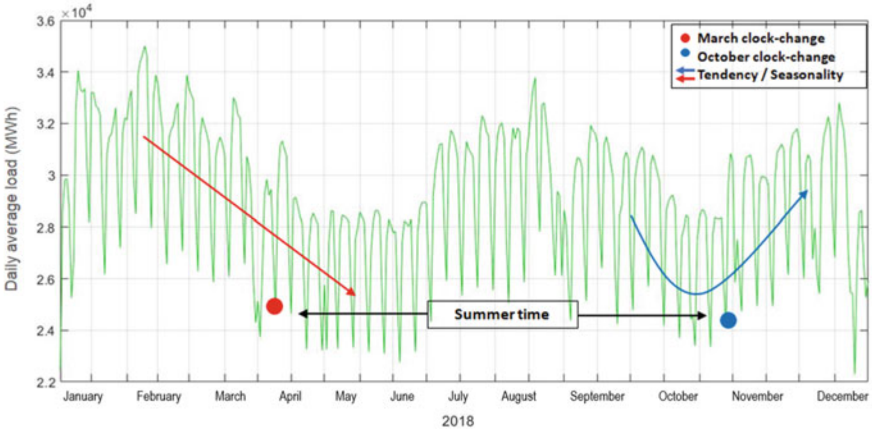


Fig. 9 Monthly load evolution during 2018. Seasonality and clock-changes

3.3 Data Used in This Study

As a result of the previously explained contamination of the historical data due to the effects of temperature variation and holidays, the historical data used for this study has been slightly modified to take these effects into account.

The previously mentioned short-term electric demand forecaster considers both effects of temperature and holidays. The temperature is modeled using the daily average maximum temperature for the ten highest populated cities in the Spanish mainland. For holidays, several variables are used to include the scale, day of the week, affected population, among others. Using this forecaster, the most-likely electricity consumption for the studied eight weeks without these external effects is obtained, providing a dataset of the electric load from 2006 to 2018 free of these external effects, which results in an easier estimation of the DST effect in those periods.

3.4 Implemented Models

If applying the DST policy did contribute to energy savings, an expected decrease in demand should take place after setting the clocks forward on the last Sunday in March, and a consequent increase should take place after setting the clocks back in October. Two models have been implemented to evaluate this hypothesis: a Randomized Block Design and a Paired Data Analysis [13].

The Randomized Block Design can be represented with the following general form:

$$Y_{ij} = \mu + \alpha_i + \beta_j + \varepsilon_{ij}$$

where Y_{ij} represents the electricity consumption in the i-th period (before or after DST) and on the j-th day of the week, parameter μ accounts for the average global effect, α_i picks up the effect for being in the i-th treatment, and β_j is the effect for being in block j.

This generic form was modified to include four levels in factor α_i (one for each week) instead of only two treatments (before DST, after DST). This way, the hypothesis testing will also reveal fluctuations in demand, and the inferences will be more robust.

For the Paired Data Analysis, the same methodology has been applied, but instead of separating the electric demand into two samples, four samples have been considered to conform four groups from which to create comparisons, for each year and day of the week, amongst the paired data (see Fig. 10).

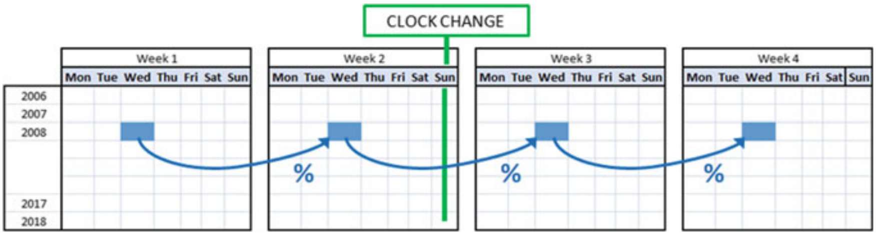


Fig. 10 Paired data analysis comparisons

3.5 *Case Study*

The electric load has been presented in Tables 5 and 6 to back up the inferences. These include the daily demand for each day of the week during the four-week periods.

The implementation of both models and the hypothesis testing of this data have concluded in the following findings:

- The variation in electric load between the first two weeks before DST is insignificant, it is concluded that there is not any significant difference between them.
- Significant differences have been noted between weeks 2–3; and 3–4, respectively. These are shown in Table 7; and Fig. 11 shows daily load values for these weeks.

In Fig. 11 it can be observed that right after the application of the DST policy—when passing from the second to the third week—there is a change in the behavior of electricity consumption after both the March and October clock-changes.

Table 5 Daily demand (GWh) 2006–2018. Four weeks around March DST

Week/Day	Mon	Tue	Wed	Thu	Fri	Sat	Sun
1	739.2	752.6	747.1	746.0	741.9	668.7	611.9
2	729.7	753.5	753.3	749.2	744.2	671.2	612.4
3	730.9	746.7	749.3	737.5	736.8	661.8	605.7
4	723.0	738.9	742.7	735.5	732.2	656.7	599.1

Table 6 Daily demand (GWh) 2006–2018. Four weeks around October DST

Week/Day	Mon	Tue	Wed	Thu	Fri	Sat	Sun
1	700.2	726.1	730.9	730.1	727.0	654.9	594.2
2	710.9	731.2	732.1	730.1	724.0	650.2	592.8
3	714.2	735.5	734.7	737.2	734.5	657.8	598.8
4	721.4	739.6	745.9	746.4	742.6	668.2	614.6

Table 7 Increment (%) in electric demand among the four weeks

Increment among weeks	2–1	3–2	4–3
March DST (%)	0.12	−0.89	−0.82
October DST (%)	0.16	0.85	1.35

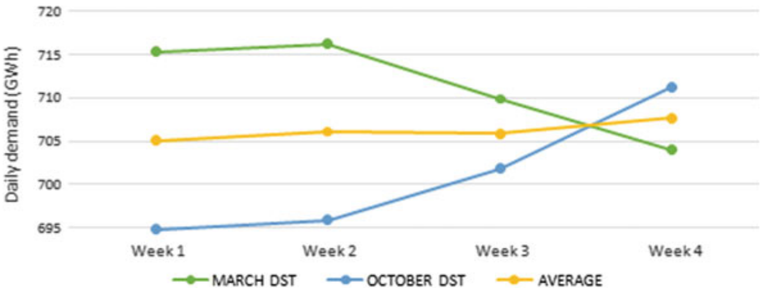


Fig. 11 Average daily demand for each week (GWh)

Both changes have been noted as almost identical in value (decrease of 0.89% by the March clock-change and increase of 0.85% by October's) which could be seen as two symmetrical movements. This is also in accordance with the hypothesis previously stated: the potential energy savings would translate into a reduction of the electric load when clocks go forward for summertime, as opposed to an increase when clocks go back.

Having removed the influence of exogenous factors that could have affected electricity demand during the weeks of the study (temperature and holiday effects), and considering the above results, these variations in demand can be attributed to the application of DST policy on those dates.

4 Conclusions

In this work, the short-term Spanish load forecast model has been slightly modified to consider the daylight duration, and it has been used to simulate the effect of disregarding DST on March 2017, and the effect of disregarding DST in October, 2017.

According to the performed numerical simulations, the DST change in March produces a decrement of electric daily load consumption around 0.6–1.0% in 2017 (decrement of 6–7% between 8.00 p.m. and 9.00 p.m.). On the other hand, the DST change in October causes an increment of daily demand about 0.4–0.7% (increment of 5% at 8 p.m. due to public and private lighting demand).

Additionally, a statistical analysis has been performed during the period 2006–2018, using the techniques of randomized block design and paired data analysis. The obtained results from the analysis report a variation of almost 0.9% in the electric demand after both clock-changes. When setting the clocks forward, the electric power consumption decreases 0.89% and setting them back increases it in 0.85%. Both values are in accordance with what was observed in the numerical simulations.

The above results refer exclusively to the variations in demand observed on the days immediately before and after the time change. It appears that the time shift causes a slight reduction in electricity demand of around 1% during summer time, although with the information available it is not possible to ensure that this reduction is maintained for the whole summer period.

Acknowledgements This work has been funded by *Red Eléctrica de España* (REE) as a R&D project.

References

1. Havranek, T., Herman, D., Irsova, Z.: Does daylight saving save electricity? A meta-analysis. Energy J. **39**(2), 35–61 (2018)
2. Aries, M.B.C., Newsham, G.R.: Effect of daylight saving time on lighting energy use: a literature review. Energy Policy **36**(6), 1858–1866 (2008)
3. Hill, S.I., et al.: The impact on energy consumption of daylight saving clock changes. Energy Policy **38**(9), 4955–4965 (2010)
4. Kotchen, M.J., Grant, L.E.: Does daylight saving time save energy? Evidence from a natural experiment in Indiana. Rev. Econ. Stat. **93**(4), 1172–1185 (2011)
5. Rivers, N.: Does daylight savings time save energy? Evidence from Ontario. Environ. Resour. Econ. **70**(2), 517–543 (2018)
6. Verdejo, H., et al.: Impact of daylight saving time on the Chilean residential consumption. Energy Policy **88**, 456–464 (2016)
7. Karasu, S.: The effect of daylight saving time options on electricity consumption of Turkey. Energy **35**(9), 3773–3782 (2010)
8. Mirza, F.M., Olvar, B.: The impact of daylight saving time on electricity consumption: evidence from Southern Norway and Sweden. Energy Policy **39**(6), 3558–3571 (2011)
9. Momani, M.A., Yatim, B., Ali, M.A.M.: The impact of the daylight saving time on electricity consumption—a case study from Jordan. Energy Policy **37**(5), 2042–2051 (2009)
10. Krarti, M., Hajiah, A.: Analysis of impact of daylight time savings on energy use of buildings in Kuwait. Energy Policy **39**(5), 2319–2329 (2011)
11. Choi, S., Pellen, A., Masson, V.: How does daylight saving time affect electricity demand? An answer using aggregate data from a natural experiment in Western Australia. Energy Econ. **66**, 247–260 (2017)
12. Hancevic, P., and Margulis, D.: Daylight saving time and energy consumption: the case of Argentina (2016)
13. Caro, E., Juan, J., Cara, J.: Estimating periodically correlated models for short-term electricity load forecasting. In: Conference: 37th International Symposium on Forecasting, Cairns, Australia (2017)
14. Earth System Research Laboratory: Pertaining to the agency. National Oceanic and Atmospheric Administration. www.esrl.noaa.gov. Accessed June 2019
15. Observatorio Astronómico Nacional - Instituto Geográfico Nacional: From the Spanish Ministry of Development (Ministerio de Fomento de España). www.fomento.gob.es/salidapuestasol. Accessed June 2019
16. Easterling, R.G.: Randomized block design and a paired data analysis. Wiley (2015). ISBN: 978-1-118-95463-8

Wind Speed Forecasting Using Kernel Ridge Regression with Different Time Horizons

Mohammad Amjad Alalami, Maher Maalouf, and Tarek H. M. EL-Fouly

Abstract Wind speed forecasting is a challenging task due to the high variability of wind data. Thus, advanced forecasting tools and models are required for predicting wind speed and wind power. In this chapter, a powerful non-linear regression method known as Kernel Ridge Regression (KRR) is proposed and adopted for wind speed forecasting. The model performance accuracy is compared with two reference prediction models, namely, the Least Squares (LS) model and the persistent model. For the KRR and LS models, the moving window cross-validation was used. Cross-validation aims to validate whether the model is heading to the right direction or not. A historical wind speed data from East Point, Prince Edward Island, Canadian weather stations was used to validate the models performance for three different forecasting horizons (1 h, 12 h, and 24 h ahead). Results show that forecasts made with the KRR produced the highest accuracy compared to the LS and the persistent models.

Keywords Wind speed forecasting · Kernel ridge regression · Least squares model and persistent model

1 Introduction

Nowadays, the interest in using renewable energy is increasing to mitigate the negative impact of conventional energy resources on the environment. Wind power is considered as one of the most rapidly growing renewable energy resource worldwide.

M. A. Alalami · M. Maalouf (✉)
Digital Supply Chain and Operation Management Center (DSO), Department of Industrial and Systems Engineering, Khalifa University, Abu Dhabi, United Arab Emirates
e-mail: maher.maalouf@ku.ac.ae

M. A. Alalami
e-mail: mohammad.a.alalami@gmail.com

T. H. M. EL-Fouly
Advanced Power and Energy Center (APEC), Department of Electrical Engineering and Computer Science, Khalifa University, Abu Dhabi, United Arab Emirates

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*, Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_13

In 2018, renewable power capacity grew to approximately 2,378 GW globally. For four consecutive years, additions of renewable power generation capacity outpaced net installations of fossil fuel and nuclear power combined. 55% of the renewable capacity was utilized by the installation of solar Photovoltaics (PV), followed by wind power (28%) and hydropower (11%). Overall, renewable energy has grown to account for more than 33% of the world's total installed power generating capacity [1]. Also, the usage of wind power is growing by 30% every year [2]. According to wind energy and green peace organization vision, almost 12% of the electricity generated will be through wind power by 2020 [3].

Energy generated through wind is mainly depended on its speed. Wind speed varies from one site to another depending on various factors. Therefore, wind power is intermittent in nature. This presents a great challenge for power systems operators when large wind power installations are being integrated into their electricity network. Therefore, as the penetration of wind power through the power system increases, the system's operations will be influenced such as generation dispatch and identifying generation reserve needs. This requires accurate forecasting of available wind generation [4]. As the usage of wind power grows dramatically, and due to the fact that the wind speed data is highly variable, multiple obstacles raise such as power system stability and reliability and transmission capacity upgrade. Therefore, many forecasting models were introduced to overcome these challenges [5].

Many methods have been developed for wind speed forecasting that are divided into two categories. The first category is the physical method which aims to find the highest forecast precision via many physical considerations. The other method is the statistical, with the help of machine learning, it focuses on finding the relation with the real time measured wind speed data. Physical method has been known in reflecting better results in long-term forecasts compared to the statistical method that does well in short-term [6]. In order to precisely predict wind speed, powerful models are required. In this chapter, a Kernel Ridge Regression (KRR) model is proposed for the wind speed forecasting procedure. The model performance accuracy, for wind speed forecasting, is compared with two reference prediction models, namely, the persistent (naïve) model and the least squares model [7].

Each forecasting model is examined in three different time horizons reflecting short-term, medium-term, and long-term forecasts. Table 1 shows the time horizons, category and the application for each forecasting time horizon [8].

This chapter is organized as follows: Sect. 2 presents the forecasting models under investigation. Section 3 describes the assessment methodology followed throughout the research. Results and analysis of the three forecasting models with different time horizons are presented in Sect. 4. Finally, conclusion is drawn in Sect. 5.

Table 1 Application of forecasting for different time horizons

Time horizon	Category	Purpose
One hour	Short term	• Planning capacity dispatch • Load increment or decrement actions
Twelve hours	Medium term	• Generator on/off line choices • Operational daily security • Ahead time electricity market
Twenty-four hours	Long term	• Unit commitment decisions • Requirement reserve actions • Obtain a schedule for maintenance • Reduce cost of operation

2 Forecasting Models

2.1 Persistent Model

The persistent model is based on the theory that there is a high correlation between the present and future values of the wind speed. The model uses a simple technique to predict the wind speed of the next hour (next time step). It states that the predicted wind speed of the next hour is the same value as the current observation. This can be modeled using the following generalized linear equation.

$$\mathrm{Y}_{\mathrm{t+b}} = \mathrm{y}_{\mathrm{t}} \tag{1}$$

where $\mathrm{Y}_{\mathrm{t+b}}$ is the predicted wind speed at time t + b, and y_{t} is wind speed observation at time t. This method is widely used by meteorologists as a reference to predict the next hour wind speed. When using a time horizon of 12 h, the model will predict that the wind speed of the upcoming 12 h is all equal to the current wind speed value. Thus, the accuracy of this model reduces with the increase of prediction horizon due to the high fluctuation of wind speed data [9, 10].

2.2 Least Squares Model

The Least Squares (LS) model aims for reducing the squared errors between the forecasted and actual value. As a result, the model finds the best-fit line. Similar to a straight line equation, the mathematical formula of the LS model is as follows [11, 12]:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \varepsilon \tag{2}$$

The β is estimated by minimizing the following equation:

$$\sum_{i=1}^{n} \varepsilon_n^2 = \varepsilon^T \varepsilon = (y - \mathbf{X}\boldsymbol{\beta})^T (y - \mathbf{X}\boldsymbol{\beta}) \tag{3}$$

where $\varepsilon = (\varepsilon_1, \varepsilon_2, \varepsilon_3 \ldots, \varepsilon_n)^T$ is the error, given that the errors have a constant variance, normally distributed, and linearly independent. The Eq. (3) is known as the objective function. Assuming that the (X^TX) is a non-singular matrix, the solution is [13]:

$$\boldsymbol{\beta} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathrm{y} \tag{4}$$

2.3 Kernel Ridge Regression

Unlike the LS model, the Kernel Ridge Regression (KRR) model does not assume linearity of the data but it takes the data into another dimensional space by including a non-linear map ϕ(.). The mapping ϕ(.) is based on the dot product. The kernel function uses the dot product such that the K matrix is equal to the following [14]:

$$\mathrm{K}\left(\mathrm{x_i},\ \mathrm{x_j}\right) = \left(\phi(\mathrm{x_i}),\ \phi\left(\mathrm{x_j}\right)\right) \tag{5}$$

Regression focuses on producing a model y = f(x) that optimally connects independent variables x and dependent variable y. Linear regression assumes the following equation:

$$\mathrm{y} = \mathrm{f(x)} = \mathrm{x}^T\beta + \varepsilon = \mathrm{y}_1 + \varepsilon \tag{6}$$

Gathering all inputs in matrix **X** and all outputs in **y**, the linear regression equation is given by

$$\mathbf{y} = \mathrm{f}(\mathbf{X}) = \mathbf{X}\boldsymbol{\beta} + \varepsilon = \mathrm{y}_1 + \varepsilon \tag{7}$$

In order to minimize the error (ε), the β needs to be optimized

$$\boldsymbol{\beta} = \operatorname{argmin}\left(\mathrm{M}(\boldsymbol{\beta})\right) \tag{8}$$

The total error is minimized by

$$\mathrm{M}(\beta) = \frac{1}{2}\Sigma\varepsilon^2 = \varepsilon\varepsilon^T = \frac{1}{2}(\mathrm{y} - \mathrm{y}_1)^T(\mathrm{y} - \mathrm{y}_1) = \frac{1}{2}(\mathrm{y} - \mathrm{X}\beta)^T(\mathrm{y} - \mathrm{X}\beta) \tag{9}$$

The solution of the optimization problem, $\nabla_{\beta\,\mathrm{M}\,=\,0,}$ is given by the inverse of the data matrix **X**, as shown below

$$\beta = \left(\mathrm{X}^{\mathrm{T}}\mathrm{X}\right)^{-1}\mathrm{X}^{\mathrm{T}}\mathrm{y} \tag{10}$$

To avoid inaccurate estimation of the regression coefficients due to the instability of the outcome of Eq. 10, regularization is used by adding λ which adjusts the penalty term. Thus, ridge regression minimizes the adjusted objective function:

$$\mathrm{M}(\beta) = \frac{1}{2}\varepsilon^{\mathrm{T}}\varepsilon = \frac{1}{2}(\mathrm{y} - \mathrm{X}\beta)^{\mathrm{T}}(\mathrm{y} - \mathrm{X}\beta) + \frac{1}{2}\lambda\beta^{\mathrm{T}}\beta \tag{11}$$

After the optimization process:

$$\beta = (\mathrm{X}^{\mathrm{T}}\mathrm{X} + \lambda\mathrm{I}_{\mathrm{d}})^{-1}\mathrm{X}^{\mathrm{T}}\mathrm{y} \tag{12}$$

where the $\mathbf{I_d}$ is a d by d identity matrix.

The dual form is introduced to spread to non-linear relations. By expressing β as linear combination of the data points:

$$\beta = \mathbf{X}^{\mathbf{T}}\boldsymbol{\alpha} \tag{13}$$

Substituting Eq. 13 back to the regression model in Eq. 7, the following equation is obtained

$$y = \mathrm{X}^{\mathrm{T}}\mathrm{X}\alpha + \varepsilon = \mathrm{G}\alpha + \varepsilon \tag{14}$$

where G is Garmain matrix which equals to $\mathrm{X}^{\mathrm{T}}\mathrm{X}$.

The objective function is minimized to terms of α, the new coefficient vector:

$$\mathrm{f}(\alpha) = \frac{1}{2}(\mathrm{y} - \mathrm{G}\alpha)^{\mathrm{T}}(\mathrm{y} - \mathrm{G}\alpha) + \frac{1}{2}\lambda\alpha^{\mathrm{T}}\alpha \tag{15}$$

The solution is then expressed by

$$\alpha = (\mathrm{G} + \lambda\mathrm{I}_{\mathrm{N}})^{-1}\mathrm{y} \tag{16}$$

KRR is a powerful model when the data is assumed to be non-linear [15, 16].

3 Methodology

In order to examine each model, historical wind data from East Point, Prince Edward Island, Canadian weather stations was used with the following variables wind speed, wind pressure, humidity, and wind direction. Also, each model is tested for three different forecast time horizons (1 h, 12 h, and 24 h ahead). The models were compared using five Key Performance Indicators (KPIs), the Mean Square Error (MSE), the Root Mean Square Error (RMSE), the Mean Absolute Error (MAE), the Mean Absolute Percentage Error (MAPE), and the coefficient of determination (R^2). The KPIs helped to shed a light on the best model as a predictor of the East Point Canadian Wind data.

For the KRR and LS models, the data is divided into two sets, the training, and testing sets. For example, when predicting the next hour of the wind speed, the training set consists of only 24 h and the testing is 720 data points which are equivalent to 30 days. As each data point represents an hour, a day will consist of 24 consecutive data points. Also, the moving window cross-validation is used in order to update the training set [17]. If the training set consists of 24 h, after predicting the next hour, the training set will drop its oldest data point by replacing it with the actual wind speed value. The independent variables for each model are different depending on the R^2 value. All the possible combinations of the independent variables were tested and only the variables with the highest R^2 are selected. The models are developed and the results were generated using MATLAB-R-2017B.

4 Results and Discussion

4.1 One-Hour Ahead Time Horizon

When using one-hour time horizon, the training and testing sets update once every hour. The training set consists of 48 h and the testing set is the proceeding data point. Table 2 shows the coefficient of determination known as the R^2 value and the independent (input) variables for each model when the time horizon is set as one step (hour). The KRR scores the highest R^2 value compared to the persistent and LS models. Therefore, the KRR predictions are closer to the actual value.

Figures 1, 2, and 3 show the relation between the actual and forecasted values of the original data set using the KRR, Persistent, and LS models, respectively, for one-hour time horizon. The closer the data points are toward the linear (red) line, the more accurate the model is. Those figures reveal better predictions of 1-h ahead wind speed data when using KRR model compared to other models.

Table 2 Models accuracy—time horizon = 1

Model	Input variable(s)	R^2 value
KRR	Wind speed, direction, pressure, and humidity	0.97
Persistent	Wind speed	0.89
LS	Wind speed, direction, pressure, and humidity	0.88

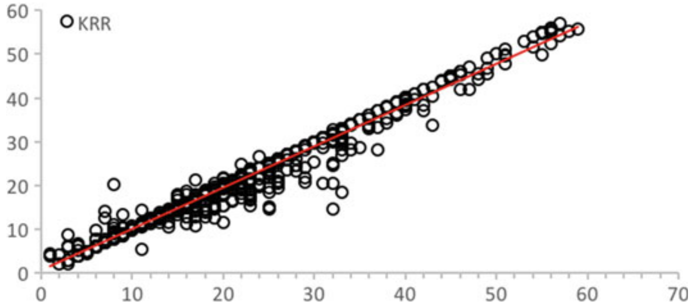


Fig. 1 Actual versus forecasted KRR values

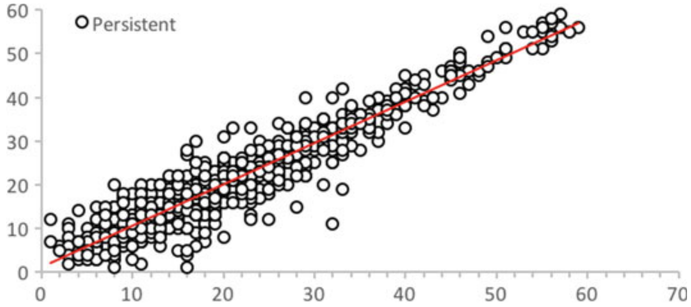


Fig. 2 Actual versus forecasted persistent values

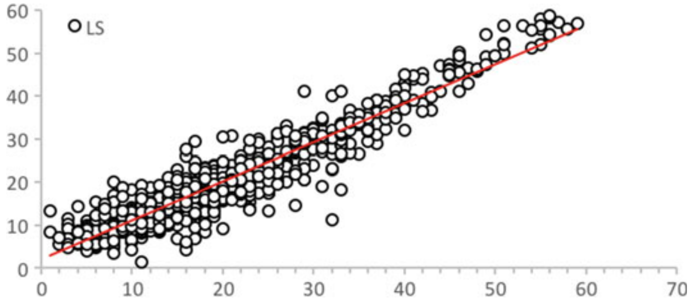


Fig. 3 Actual versus forecasted LS values

The average MSE, MAPE, RMSE, and MAE values of each hour is calculated for each model and averaged for the whole test data. The lower these values are, the more accurate the model is. Table 3 shows the average of all the 720 forecasted values, when testing the three models. The KRR generates the lowest averages of MSE, RMSE, MAE, and MAPE. The second-best model is the persistent model according to the averages of the measurements of error. Figure 4 shows the forecasted values by the three models against the actual data points for four tested days reveal that

Table 3 Average measurements of error—time horizon = 1

Model	MSE	RMSE	MAE	MAPE
KRR	5.00	2.24	1.00	0.07
Persistent	15.53	3.94	2.89	0.22
LS	15.82	3.98	3.00	0.24

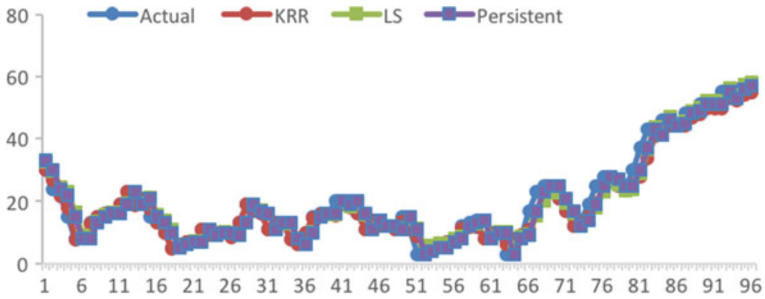


Fig. 4 Sample of actual versus forecasted wind speed—time horizon = 1

the predicted values generated by the KRR model are the closest to the actual wind speed data as the KRR model accurately traces the actual data point.

4.2 Twelve-Hours Ahead Time Horizon

When applying twelve-hours ahead time horizon, the training and testing sets update after predicting twelve hours (once every 12 h or twice daily). Hence, the influence of the fluctuation of the wind speed data on prediction accuracy increases. The training set consists of 168 h and the testing set contains the 12 consecutive proceeding data points. Table 4 presents the R^2 value and input variables for each model when the forecasting time horizon is twelve hours ahead. The KRR generates the highest R^2 value compared to the persistent and LS models. Therefore, the KRR predictions are closer to the actual value compared to the other models.

Figures 5, 6, and 7 present the relation between the actual and forecasted values of the original data set using the KRR, persistent, and LS models, respectively, for

Table 4 Models accuracy—time horizon = 1

Model	Input variable(s)	R^2 value
KRR	Wind speed, direction, pressure, and humidity	0.71
Persistent	Wind speed	0.47
LS	Wind speed, direction, pressure, and humidity	0.29

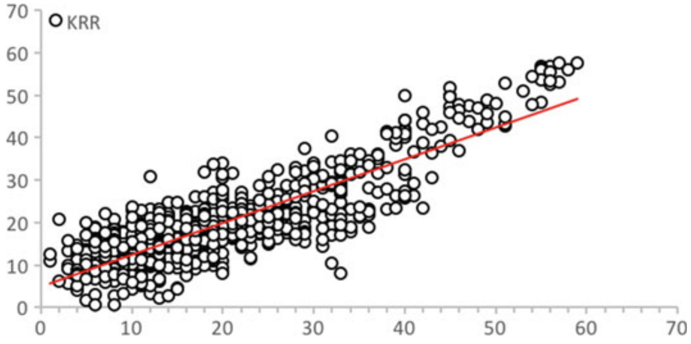


Fig. 5 Actual versus forecasted KRR values

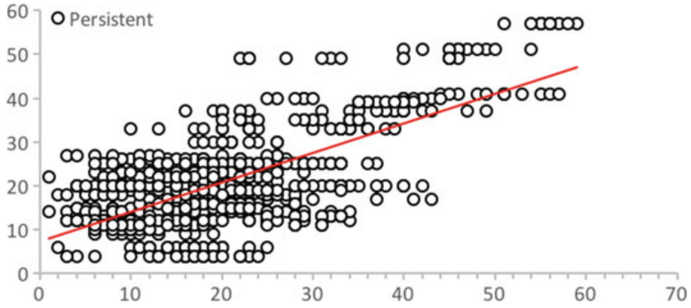


Fig. 6 Actual versus forecasted Persistent values

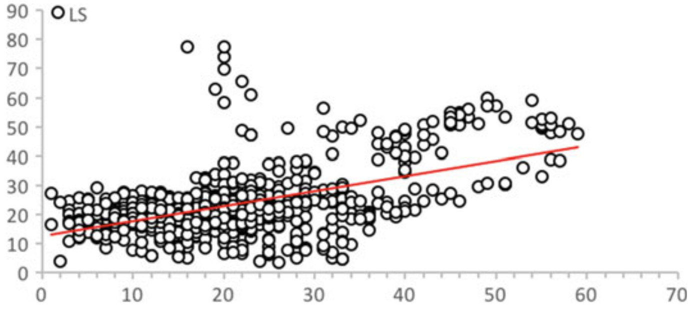


Fig. 7 Actual versus forecasted LS values

twelve-hours ahead time horizon. Once again those figures reveal better predictions of 12-h ahead wind speed data when using KRR model compared to other models.

The MSE, MAPE, RMSE, MAE, and MAPE values of each hour are calculated for each model and averaged for the whole test data. Table 5 shows those average values for all the 720 forecasted points using the three models where the KRR generates

Table 5 Average measurements of error—time horizon = 12

Model	MSE	RMSE	MAE	MAPE
KRR	39.32	6.27	4.85	0.39
Persistent	83.10	9.11	7.16	0.56
LS	126.54	11.25	8.65	0.70

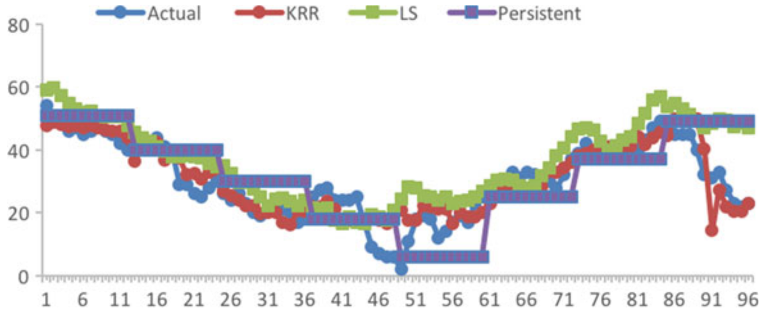


Fig. 8 Sample of the actual versus forecasted wind speed—time horizon = 12

the lowest averages of MSE, RMSE, MAE, and MAP the measurements of error E followed by the persistent model.

Figure 8 presents a sample of the forecasted against the actual wind speed for four days. The forecasted values by the KRR model are the closest to the actual wind speed.

4.3 Day-Ahead (Twenty-Four Hours Ahead) Time Horizon

When using twenty-four-hours ahead time horizon, the training and testing sets update after predicting 24-h points (once every day). Table 6 shows the coefficient of determination value and input variables (independent variables) for each model when the time horizon is twenty-four. The KRR scores the highest value of R^2 value compared to the LS and persistent models. Thus, the KRR predictions are closer to the actual value.

Similar to the previous analyses, Figs. 9, 10, and 11 present the relation between

Table 6 Models' accuracy—time horizon = 24

Model	Input variable(s)	R^2 value
KRR	Wind speed, direction, pressure, and humidity	0.54
Persistent	Wind speed	0.27
LS	Wind speed, direction, pressure, and humidity	0.08

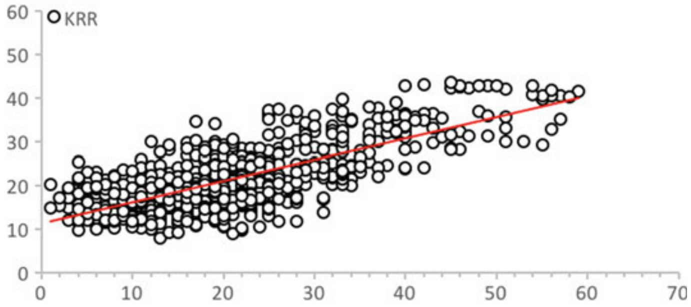


Fig. 9 Actual versus forecasted KRR values

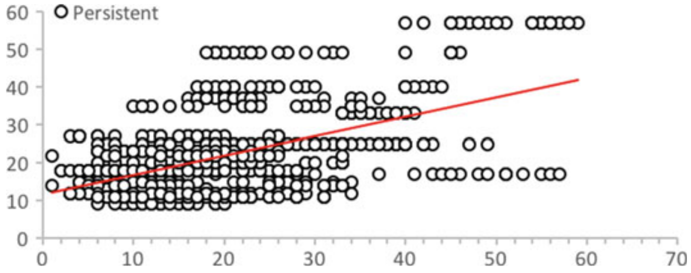


Fig. 10 Actual versus forecasted persistent values

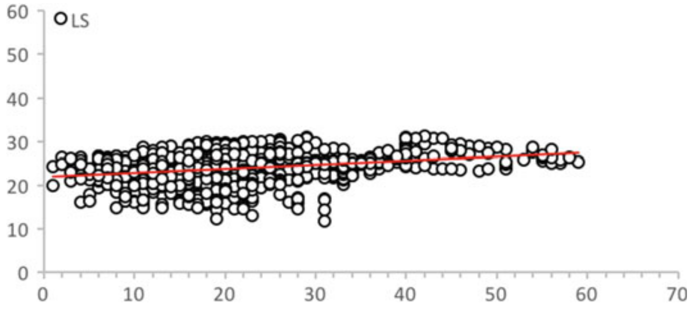


Fig. 11 Actual versus forecasted LS values

the actual and forecasted values of the original data set using the three models for 24-h ahead time horizon. Moreover, Table 7 shows the average values for the MSE, MAPE, RMSE, MAE, and MAPE over the whole test data set of 720 forecasted values using the three models. The results reveal better predictions of 24-h ahead wind speed data when using KRR model compared to other models. Furthermore, KRR generates the lowest averages of MSE, RMSE, MAE, and MAPE. Finally, Fig. 12 presents a sample of the forecasted against the actual wind speed for four

Table 7 Average measurements of error—time horizon = 24

Model	MSE	RMSE	MAE	MAPE
KRR	62.56	7.91	6.42	0.54
Persistent	129.67	11.39	8.80	0.65
LS	131.24	11.46	9.49	0.83

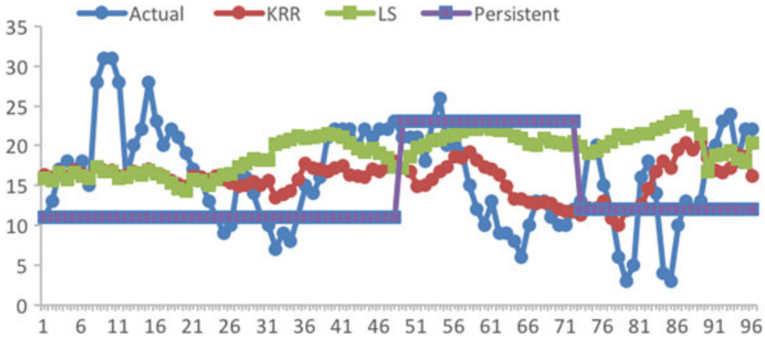


Fig. 12 A sample of actual versus forecasted wind speed—time horizon = 24

days which shows that the forecasted values by the KRR model are the closest to the actual wind speed.

5 Conclusion

Predicting wind speed and wind power is essential to enable high penetration levels of wind energy resources into power systems. Wind speed data is highly variable and requires powerful methods for forecasting purposes. In this chapter, wind speed forecasting based on using the Kernel Ridge Regression (KRR) has been presented and evaluated. The MSE, RMSE, MAE, MAPE, and the R^2 value were set as Key Performance Indicators (KPIs) to compare the forecasting accuracy of the proposed KRR model and two reference models, namely, the least square and persistence models. The accuracy of the proposed forecasting methods is compared for three different time horizons (1 h, 12 h, and 24 h ahead). As expected, it was observed that with the increase of the lead time (horizon), the accuracy of each model decreases. The KRR model generated the highest accuracy compared to the persistence and LS models in all the given time horizons. Thus, KRR models can accurately be used for wind speed prediction.

References

1. Kusch-Brandt: Urban renewable energy on the upswing: a spotlight on renewable energy in cities in REN21's "Renewables 2019 global status report". Resources **8**(3), 139 (2019)
2. Sánchez, I.: Short-term prediction of wind energy production. Int. J. Forecast. **22**(1), 43–56 (2006)
3. Lei, M., Shiyan, L., Chuanwen, J., Hongling, L., Yan, Z.: A review on the forecasting of wind speed and generated power. Renew. Sustain. Energy Rev. **13**(4), 915–920 (2005)
4. Zhu, X., Genton, M.: Short-term wind speed forecasting for power system operation. Int. Stat. Rev. **80**(1), 2–23 (2012)
5. Lund, H.: Large-scale integration of wind power into different energy systems. Energy **30**(13), 2402–2412 (2005)
6. Potter, C., Negnevitsky, M.: Very short-term wind forecasting for Tasmanian power generation. IEEE Trans. Power Syst. **21**(2), 965–972 (2006)
7. Maldonado-Correa, J., Solano, J., Rojas-Moncayo, M.: Wind power forecasting: a systematic literature review. Wind Eng. (2019)
8. Fabbri, A., GomezSanRoman, T., RivierAbbad, J., MendezQuezada, V.: Assessment of the cost associated with wind generation prediction errors in a liberalized electricity market. IEEE Trans. Power Syst. **20**(3), 1440–1446 (2005)
9. Soman, S.S., Zareipour, H., Malik, O., Mandal, P.: A review of wind power and wind speed forecasting methods with different time horizons. In: North American Power Symposium (2010)
10. Mellit, A., Pavan, A., Benghanem, M.: Least squares support vector machine for short-term prediction of meteorological time series. Theoret. Appl. Climatol. **111**(1–2), 297–307 (2012)
11. Ren, Y., Suganthan, P., Srikanth, N.: A comparative study of empirical mode decomposition-based short-term wind speed forecasting methods. IEEE Trans. Sustain. Energy **6**(1), 236–244 (2015)
12. Maalouf, M., Homouz, D.: Kernel ridge regression using truncated newton method. Knowl.-Based Syst. **71**, 339–344 (2014)
13. Cadenas, E., Rivera, W.: Wind speed forecasting in the South Coast of Oaxaca, México. Renew. Energy **32**(12), 2116–2128 (2007)
14. Maalouf, M., Barsoum, Z.: Failure strength prediction of aluminum spot-welded joints using kernel ridge regression. Int. J. Adv. Manuf. Technol. **91**(9–12), 3717–3725 (2017)
15. Gagnis, V., Homouz, D., Maalouf, M., Khoury, N., Polychronopoulou, K.: An efficient method to predict compressibility factor of natural gas streams. Energies **12**(13), 2577 (2019)
16. Maalouf, M., Khoury, N., Laguros, J., Kumin, H.: Support vector regression to predict the performance of stabilized aggregate bases subject to wet-dry cycles. Int. J. Numer. Anal. Meth. Geomech. **36**(6), 675–696 (2011)
17. Rao, J., Wu, B., Dong, Y.: Parallel link prediction in complex network using MapReduce. J. Soft. **23**(12), 3175–3186 (2014)

Applying a 1D-CNN Network to Electricity Load Forecasting

Christian Lang, Florian Steinborn, Oliver Steffens, and Elmar W. Lang

Abstract This paper presents a convolutional neural network (CNN) which can be used for forecasting electricity load profiles 36 hours into the future. In contrast to well established CNN architectures, the input data is one-dimensional. A parameter scanning of network parameters is conducted in order to gain information about the influence of the kernel size, number of filters and number of nodes. Furthermore, different dropout methods are applied to the CNN and are evaluated. The results show that a good forecast quality can already be achieved with basic CNN architectures, the dropout improves the forecast. The method works not only for smooth sum loads of many hundred consumers, but also for the load of single apartment buildings.

Keywords Energy load forecasting · STLF · Neural networks · CNN · Convolutional networks

1 Introduction

There is no dispute in the scientific community that human-induced climate change is real. The effects of climate change are, for example, rising sea levels, an increasing CO_2 concentration in the atmosphere, and more regularly occurring extreme weather events, to name only a few of them [1, 2]. To slow down and stop the global warming, it is crucial to reduce the generation of greenhouse gases, especially in energy production. One of the keys to accomplish a reduction is to establish more renewable energies in the energy market. By doing so, power plants that produce high levels of CO_2, like coal power plants, can in the long term be substituted by renewable energy sources. Another key to minimise the emission of greenhouse gases is to decrease

C. Lang (✉) · F. Steinborn · E. W. Lang
Regensburg Universität, Regensburg, Germany
e-mail: christian3.lang@ur.de

C. Lang · O. Steffens
OTH Regensburg, Regensburg, Germany

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*, Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_14

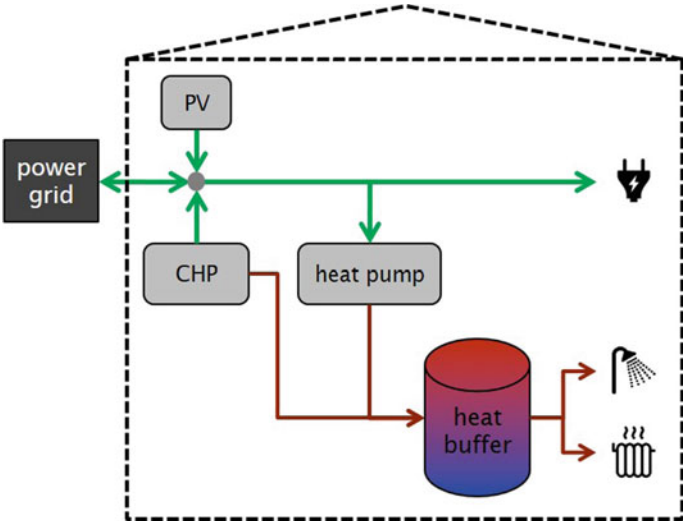


Fig. 1 Schematic of the new heating system. The red lines symbolise heat transport using water and the green lines symbolise electricity transport

the total energy consumption and to increase energy efficiency in consumption and production.

In the research project MAGGIE [3, 4], we try to address all of the above mentioned challenges. The goal of the research project is to energetically modernise existing historic apartment buildings and draft a concept for sector coupling in city districts. In the first step, one exemplary building will be modernised and evaluated. Afterwards, the whole city district will be modernised in a similar manner. In order to decrease the heat consumption of the building the thermal insulation is renewed and in the course of the research project new insulations are in development. In addition, a new heating system (see Fig. 1), with an innovative control system is implemented. This heating system allows increase in energy efficiency and can help integrate renewable energy sources into the power market. The core of the system is a combined heat and power plant (CHP), and a heat pump. All of them generate thermal energy, the heat pump from electricity and the CHP from fuel. The thermal energy is used to heat the water of a buffer storage, which is then, in combination with a heat exchanger, used as process and drinking water. In addition, the CHP generates electricity, as does a photovoltaic system (PV system) installed on the roof of the building, which can then be used to either power the heat pump or supply the habitants with electricity. A connection to the power grid receives surplus electricity and ensures there is always enough electricity available [5].

By utilising both forms of generated energy, the total energy efficiency of the system is nowadays higher than that of a conventional heating system, for example, a gas-fired boiler, in combination with electricity from the power grid [6].

All parts of the energy system are monitored continuously and can be controlled independently and remotely by the control system, which allows one to shift the production and consumption of heat and electricity in time and between the participants of the system by heating and using the water at the needed times. This allows for optimisation of the machine schedules depending on an optimisation target. Those targets can, for example, be self-sufficiency or cost-reduction.

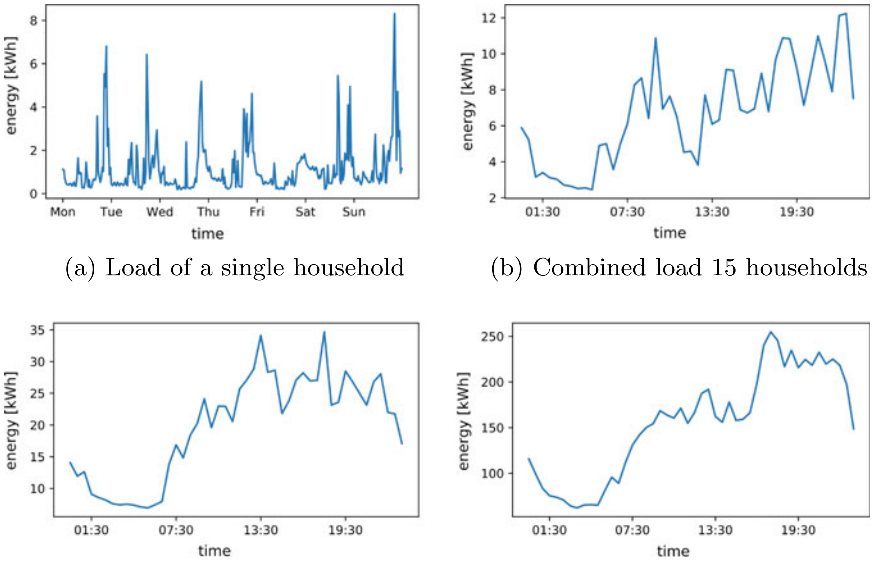
After the modernisation of the entire city district, the energy systems of all houses in the district or even of several districts can act as a virtual power plant (VPP). This VPP can then work as a baseload power plant using electricity from the PV system during the day, from the CHP during the night, and reduce the volatility of the produced electricity from the PV system with the CHP and heat pump. The VPP can thereby integrate photovoltaics in the power grid without the drawback of its volatility. The VPP can also help to stabilise the power grid by using surplus or supplying lacking electricity to the power grid, and therefore, assist with integrating renewable energies into the power market.

The main challenge of the system consists of knowing the electric and heat load of the building and its inhabitants. The loads are crucial for schedule optimisation, as the feed-in into the power grid and the consumption from the power grid have to be reported to the power grid operator in a 15 min grid one day in advance at noon. Deviations in the heat load can be buffered with the heat buffer, deviations in the electric load, however, cannot be buffered in any way. Therefore, the focus of this paper is on forecasting electricity loads.

2 Smart Meter Data

In two directives [7, 8], the EU outlined their decision to establish SmartMeter devices in the energy sector throughout the entire European Union with the aim to enable customers to better monitor and manage their consumptional behaviour. A SmartMeter, in contrast with a conventional electric meter, records the energy consumption at least every 15 min or in even shorter periods. In this paper, the data of the CER Smart Metering project [9] is used. The dataset consists of individual SmartMeter data from over 5 000 Irish homes and businesses recorded for 18 months. In that project, the electricity consumption was measured every 30 min. However, it should be easy to apply the results of our research to data recorded in a 15 min grid.

As the electric load of a single household is highly volatile, and therefore, impossible to predict, sum load time series of 15, 40 and 350 randomly picked households were created. Those time series correspond to a small apartment building, a big apartment building, and a whole city district. Figure 2 shows an exemplary day of the mentioned time series.



(a) Load of a single household

(b) Combined load 15 households

(c) Combined load 40 households

(d) Combined load 350 households

Fig. 2 **a** Shows a load time series of a single household. **b–d** Show each an exemplary day of the combined load time series. The different extracts display how volatile the load of a single household is and that the volatility decreases when households are combined

3 Importance of Time Series Forecasting

The electric load forecast is crucial in order to fully utilise the possibilities of the implemented heating system and similar systems. Without a good forecast, a part of the heat buffer capacity has to be withheld in order to balance the deviation in the electric load by the CHP. The prediction horizon in our case is $h = 72$ samples as 36 h have to be predicted in a 30 min grid.

There are already several publications about time series forecasting and short-term load forecasting (STLF) [10, 11]. However, most of the methods predict either only one or very few time steps in the future, or are applied on load time series of whole cities/states which are, due to the properties of statistics, way smoother than the load time series of one building. Those smooth time series can be described properly with statistical methods when external factors (e.g. the weather) are taken into account. Therefore, their shape and features are also easy for neural networks to learn. None of the methods mentioned in recent publications, however, are designed to predict the electric load of only one building.

In the next chapter, we propose the use of Convolutional Neural Networks (CNN) for time series prediction that can predict several time steps and can handle a volatile input. We report the first results of different network structures and discuss their parameters. Then, we optimise those parameters for forecasting our electricity load time series.

4 Convolutional Neural Networks

Convolutional Networks, in the way they are used nowadays, were first introduced by LeCun et al. [12] for zip code recognition. Since then, they were further developed and are now the standard for image and pattern recognition.

CNNs usually consist of convolutional layers, pooling layers, and fully-connected layers. In the convolutional layers, a set of feature maps, also called activation maps, are created. Each neuron in the feature map is only connected to a subset of neurons in its input layer. All neurons of the feature map share the same weights, thereby reducing the number of parameters significantly compared to a fully-connected neural network. In the most common CNN architectures, pooling layers alternate with convolutional layers. The pooling layer reduces the spatial dimension of the feature maps for the next computational steps in order to minimise the computational load and to avoid overfitting. At the end of the network, after an arbitrary number of the prior layers, fully-connected layers combine the resulting feature maps and return a classification measure. [5, 12, 13].

5 Forecasting with CNNs

CNNs are traditionally used for image and pattern recognition by extracting features from two-dimensional data. In our research, we use a similar architecture for the forecasting of one-dimensional time series. The basic idea is that the convolutional layers extract features. These features are then combined by one or more fully-connected layers, and finally, a forecast is created based on the classification of the last fully-connected layer (see Fig. 3). The pooling layers are omitted because an excessive amount of parameters is not a problem for one-dimensional data and the necessity of pooling layers is questioned in recent research [14].

A forecast can be created in two different ways, either directly or recursively. A direct forecast means the network generates the desired forecast at once. Thus, the number of neurons in the output layer equals the prediction horizon h. When the forecast is generated recursively, only one-time step is predicted by the network. Then, the predicted point is appended to the input data and the first data point of the input is cut off, so that the new input has the required shape. Based on the new input, which is fed back into the neural network, the next point is predicted. This procedure is repeated until h data points are predicted [5].

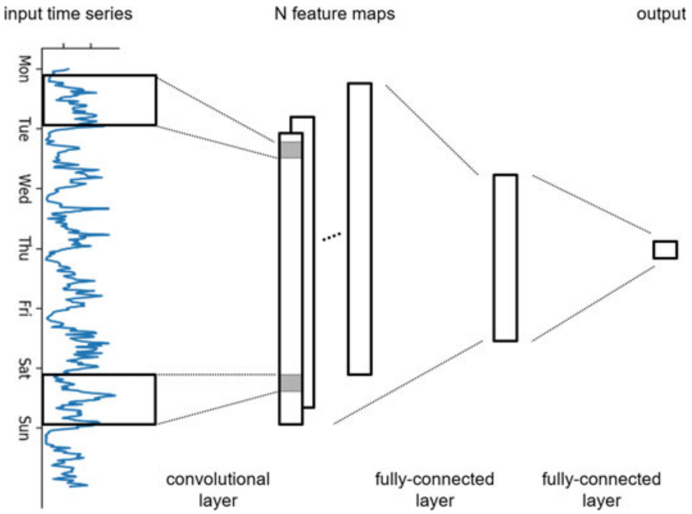


Fig. 3 Principle architecture of the used neural network

6 Evalution of Different Network Structures and Training Parameters

6.1 Development and Analysis of a Basic Forecaster

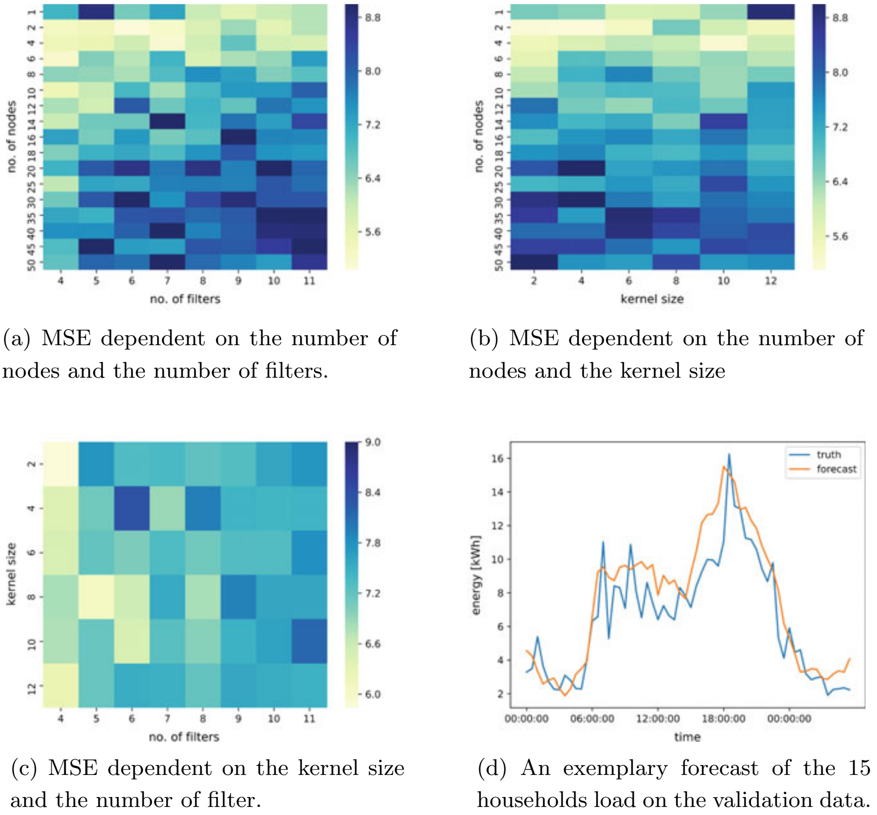
In order to get a better understanding of how the 1D-CNNs process data and how the network architecture influences the results, are the first tests conducted with very basic networks. They are built from one convolutional layer followed by one fully-connected layer which directly calculates the output. The evaluation of these networks yield the basic training parameters that are used throughout the rest of this work.

The data from the SmartMeter Trial is split into a training, a validation, and a test set. The training data set contains data of an entire year. The validation and test set each contains half of the residual data that has about 6 months of data. All the error measures presented are calculated on the test set and represent the error across the whole forecast horizon.

The best results were obtained with mini-batch of size $b = 128$ and epochs $e = 40$. $Nadam$ [15] is used as optimiser and the mean squared error (MSE) as loss-function. When using bigger mini-batch-sizes, unwanted jumps in the training loss were observed regularly, and when using smaller batch-sizes, overfitting occurred early during training.

In the next trials, an additional fully-connected layer was added in between the convolutional layer and the fully-connected output layer (see network architecture in Fig. 3). All output nodes of the new layer are connected to all nodes in every feature map. The additional fully-connected layer improved the forecast quality independently from other network and training parameters. As this simple network already produces promising results, the convolutional layer is varied to further improve the forecasts.

The parameters that were varied are the kernel size, and thereby the receptive field, and the number of filters. They describe how big the filters are, that sample over the time series, and how many filters (each of them creates a feature map) are trained. The used stride length is one. In addition, the number of output nodes of the first fully-connected layer is varied as well to identify how many significant features, from which the forecast is composed, exist. When the number of neurons is chosen too large, the network is prone to overfitting as there are too many trainable variables. When there are not enough neurons, it is not capable of representing all critical features, and therefore, the results deteriorate.



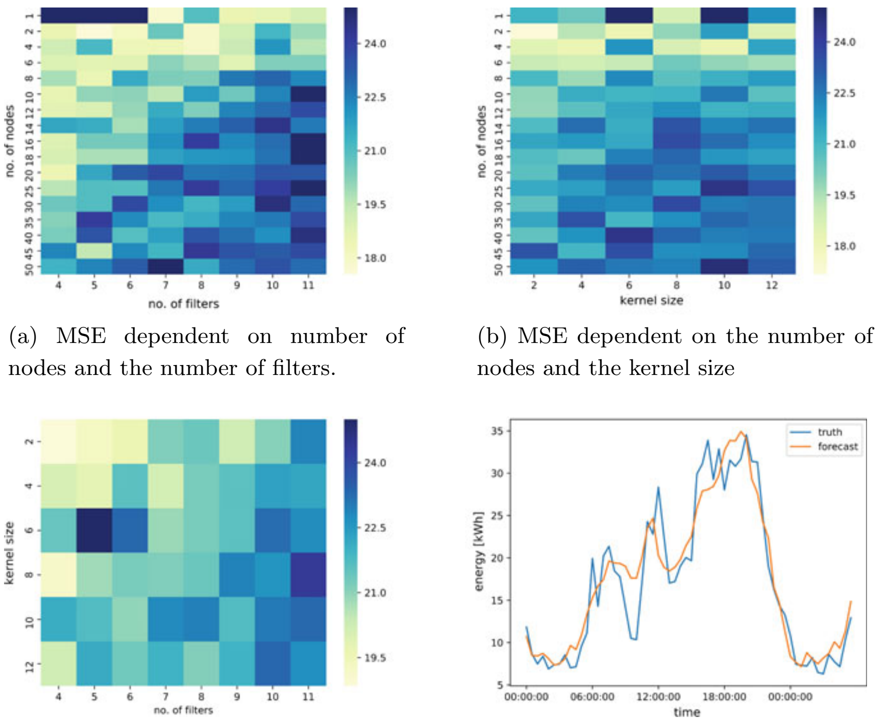
(a) MSE dependent on the number of nodes and the number of filters.

(b) MSE dependent on the number of nodes and the kernel size

(c) MSE dependent on the kernel size and the number of filter.

(d) An exemplary forecast of the 15 households load on the validation data.

Fig. 4 Heatmaps **a**–**c** Show the MSE of the 15 household forecasts. The MSE values are a mean values across the third parameter. **d** Shows a forecast using the trained CNN



(a) MSE dependent on number of nodes and the number of filters.

(b) MSE dependent on the number of nodes and the kernel size

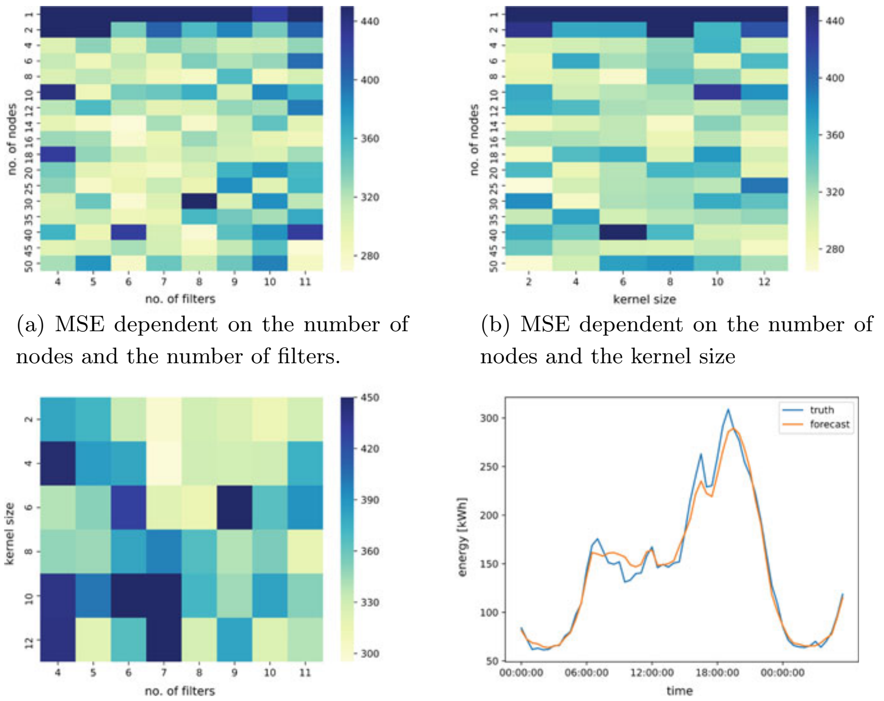
(c) MSE dependent on the kernel size and the number of filter.

(d) An exemplary forecast of the 350 households load on the validation data.

Fig. 5 Heatmaps **a**–**c** Show the MSE of the 40 household forecasts. The MSE values are a mean values across the third parameter. **d** Shows a forecast using the trained CNN

As is apparent from the different heatmaps (see Figs. 4, 5, and 6), the three parameters have a crucial influence on the performance.

On the heatmap plots (b) for 15 households (Fig. 4) and for 40 households (Fig. 5), it can be seen that the best results can be achieved with a rather small number of output nodes of the fully-connected (also called dense) layer between the convolutional and the output layer. With an increasing number of nodes, the forecast results become unreliable, probably overfitting occurs. The heatmaps (a) of both aggregation levels indicate that a large fully-connected layer compensates partially a too small number of filters and vice-versa. This seems to be in particular the case for the 40 household load series. However, when both parameters that are chosen are too big, the MSE increases. Due to the high amount of trainable parameters in the network that come with a large amount of neurons (large dense size) it is preferable to use a small fully-connected layer with a larger number of filters, in order to minimise the computational load. There is no obvious conclusion regarding the kernel size. It seems that a kernel size which is too big or too small has a negative influence on the forecast quality.



(a) MSE dependent on the number of nodes and the number of filters.

(b) MSE dependent on the number of nodes and the kernel size

(c) MSE dependent on the kernel size and the number of filter.

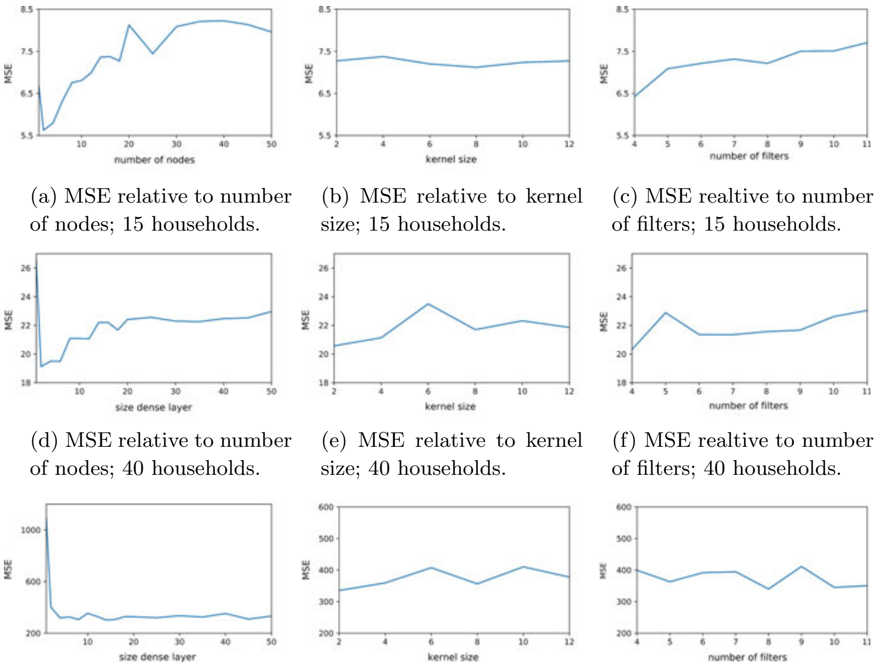
(d) An exemplary forecast of the 350 households load on the validation data.

Fig. 6 Heatmaps **a**–**c** Show the MSE of the 350 household forecasts. The MSE values are a mean values across the third parameter. **d** Shows a forecast using the trained CNN

Those impressions are supported when the effect of only one parameter is inspected (see Fig. 7a–f). In addition, the earlier conclusion that an additional fully-connected layer enhances the forecast quality is confirmed for the load series of 40 households by the significantly worse performance of the network when $dense\_size = 1$. This basically equals a network with only one fully-connected layer.

The heatmaps of MSE of the 350 household load are illustrated in Fig. 6. The influence of the network parameters differs from the 40 household load. In addition to the confirmation that the additional fully-connected layer increases the forecast quality, it also becomes apparent that only with more than two neurons in the fully-connected layer good forecasts are possible. Furthermore, it seems that the number of filters and the kernel size only have a minor influence on the MSE. On heatmap (c), however, it appears that the most accurate forecasts are the ones with smaller kernel sizes. Figure 7g–i supports this assumption.

The heatmaps and the Figs. 7a, d and g suggest that the less volatile the time series, the greater the benefit of an additional fully-connected layer.



(a) MSE relative to number of nodes; 15 households.

(b) MSE relative to kernel size; 15 households.

(c) MSE realtive to number of filters; 15 households.

(d) MSE relative to number of nodes; 40 households.

(e) MSE relative to kernel size; 40 households.

(f) MSE realtive to number of filters; 40 households.

(g) MSE relative to number of nodes; 350 households.

(h) MSE relative to kernel size; 350 households.

(i) MSE relative to number of filters; 350 households.

Fig. 7 The MSE averaged across the other two parameters for the 15 household load in (**a**)–(**c**), for the 40 household load in (**d**)–(**f**), and for the 350 household load in (**g**)–(**i**)

6.2 Improvements to the Basic Forecaster

After obtaining an understanding of the behaviour of the convolutional network, promising combinations of the number of filters, the filter size, and the number of neurons in the first fully-connected layer were further investigated. After running 50 iterations of each combination and analysing the mean squared error, the variance, and the error evolution of the forecast depending on the time-lag to the last known value, the following combinations of kernel size k, number of filters f and dense size d produced the best results

- 15 households: $k = 3$, $f = 8$, $d = 6$
- 40 households: $k = 6$, $f = 8$, $d = 6$
- 350 households: $k = 6$, $f = 8$, $d = 6$

The analysis of the error evolution revealed that the error increases with an increased lag to the last known value. That is to be expected as the larger the time-lag, the more values of the input are already itself predicted by the neural network, and therefore, afflicted with the error.

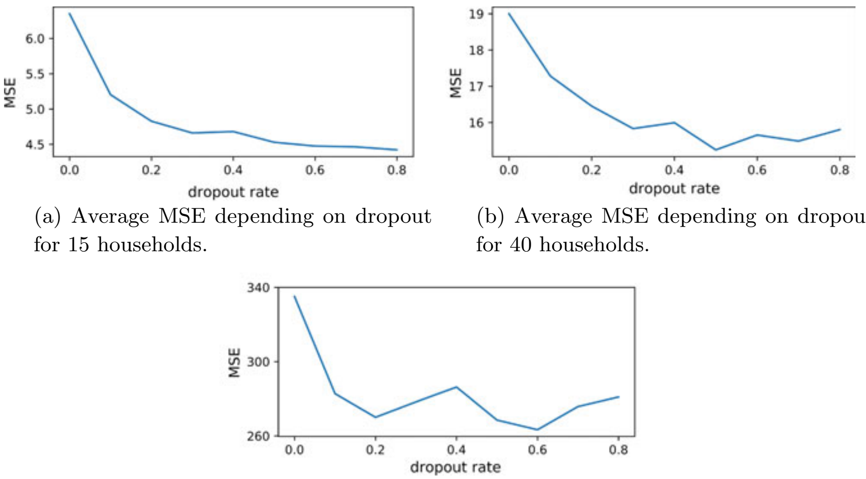
To further improve the network performance, different types of dropouts were implemented and tested on the best performing neural networks. 20 iterations were computed with each dropout rate. Using dropout means some units of the neural network and their connections are dropped (temporarily removed) from the network during training. It is equivalent to sample a thinned network and train that network with the weights being shared between all possible thinned networks. This reduces overfitting by preventing co-adaption of units [16].

Firstly, spatial dropout was tested. When spatial dropout is applied, complete feature maps are randomly dropped during the training in order to prevent the feature maps from co-adapting [17]. Two different layouts were tested—one where the number of filters, and therefore, the number of feature maps, is held constant and a second one where the number of filters is varied according to the dropout rate. That means, if the dropout rate is, for example, 50%, the number of filters doubles. Secondly, a dropout of random units throughout all feature maps was applied. Again, two different layouts were tested—one where the number of units in the first fully-connected layer was kept at 6 independently of the dropout rate and a second one where the number of units was varied the same way the filters were before. The following dropout rates dr were applied to all tests: $dr = [0.2, 0.4, 0.6, 0.8]$.

The results using spatial dropout with an unchanged number of filters were sobering. The MSE of all networks are larger than without dropout. In addition, it is noticeable that the MSE increased with an increasing dropout rate. When the number of filters is varied according to the dropout rate, the networks with a large dropout rate perform slightly better than without dropout, the other networks still perform worse. In conclusion, spatial dropout does not improve the forecast performance significantly. In contrast, the results using a random dropout were way better. As shown in Table 1, the MSE decreases substantially for all three aggregation levels using dropout. When the number of units in the fully-connected layer is changed according to the dropout rate, the quality of the forecasts improves even more. As one can see from Fig. 8, the networks perform the best with a high dropout rate.

Table 1 MSE of the respectively best networks when using a random dropout

	No dropout	Constant dropout	Variable dropout
IRE 15	6.35	4.59	5.42
IRE 40	19.0	15.6	15.2
IRE 350	335	284	263



(a) Average MSE depending on dropout for 15 households.

(b) Average MSE depending on dropout for 40 households.

(c) Average MSE depending on dropout 350 households.

Fig. 8 Evaluation of the effect of different dropout rates on the forecast performance. A medium to large dropout rate gives the best results for all aggregation levels

7 Conclusion

The network parameters number of nodes, number of filters, and kernel size were varied in a wide range. It can be concluded that the right set of parameters depends on the type of time series that is to be predicted.

The 350 household load time series can be forecasted properly with a CNN (see Fig. 6d). When the size of the fully-connected layer chosen is larger than two, the network is quite robust against changes in the number of filters and kernel sizes.

A reliable forecast of the load time series of 40 households is possible with a rather simple CNN when the parameters are chosen correctly (see Fig. 5d). It appears the time series can be described properly with 4–6 features as the best results were obtained with dense size $d = 4 \ldots 6$. Furthermore, it became apparent that with too many training parameters the forecast quality decreases, probably due to overfitting.

Creating a good forecast for the load time series of 15 households is challenging due to the high volatility in the load (see Fig. 4d). Networks with a small number of filters create the best forecast. The benefit of a second fully-connected layer in the neural network is minimal when the volatility is high.

Adding a random dropout to the models improved the forecast quality substantially. The networks performed best with large dropout rates, which implies that without dropout co-adaption between the units is an issue.

The forecasters for the three time series can already outperform the standard load profile, even though the network architecture is quite simple and no external factors have been taken into account yet. For volatile load profiles, simplicity in the network architecture seems to be the key to good forecasting results.

Acknowledgements We thank the Federal Ministry for Economic Affairs and Energy (BMWi) for funding the project MAGGIE.

References

1. Masson-Delmotte, V., Zhai, P., Pörtner, H.O., Roberts, D., Skea, J., Shukla, P.R., Pirani, A., Moufouma-Okia, W., Pan, C., Pidcock, R., Connors, S., Matthews, J.B.R., Chen, Y., Zhou, X., Gomis, M.I., Lonnoy, E., Maycock, T., Tignor, M., Waterfield, T. (eds.): IPCC 2018: global warming of 1.5°C. An IPCC Special Report on the impacts of global warming of 1.5°C above pre-industrial levels and related global greenhouse gas emission pathways, in the context of strengthening the global response to the threat of climate change, sustainable development, and efforts to eradicate poverty (2018)
2. Field, C.B., V. Barros, T.F. Stocker, D. Qin, D.J. Dokken, K.L. Ebi, M.D. Mastrandrea, K.J. Mach, G.-K. Plattner, S.K. Allen, M. Tignor, P.M. Midgley (eds.): IPPC 2012: managing the risks of extreme events and disasters to advance climate change adaptation (2012)
3. Bundesministerium für Wirtschaft und Energie, Energiewende bauen, Solares Bauen: MAGGIE. https://projektinfos.energiewendebauen.de/projekt/energetisch-modernisieren-mit-solaraktiven-baustoffen-und-hybridem-heizsystem/
4. Jüllich, P.: EnArgus, Solares Bauen: MAGGIE. https://www.enargus.de/pub/bscw.cgi/?op=enargus.eps2&q=%2201180590/1%22&v=10&id=539378
5. Lang, C., Steinborn, F., Steffens, O., Lang, E.W.: Electricity Load Forecasting–An Evaluation of Simple 1D-CNN Network Structures. In: 6th International Conference on Time Series and Forecasting, pp. 797–806. Granada (2019)
6. Sterner, M., Stadler, I.: Energiespeicher–Bedarf, Technologien, Integration. Springer Vieweg, Berlin Heidelberg (2014)
7. Directive 2006/32/EC of the European Parliament and of the Council of 5 April 2006 on energy end-use efficiency and energy services and repealing Council Directive 93/76/EEC. https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32006L0032
8. Directive 2009/72/EC of the European Parliament and of the Council of 13 July 2009 concerning common rules for the internal market in electricity and repealing Directive 2003/54/EC. https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32009L0072
9. Commission for Energy Regulation (CER).: CER Smart Metering Project–Electricity Customer Behaviour Trial, 2009–2010 [dataset]. 1st edn, Irish Social Science Data Archive. SN: 0012-00 (2012). www.ucd.ie/issda/CER-electricity
10. Srivastava, A.K., Pandey, A.S., Singh, D.: Short-term load forecasting methods: a review. In: International Conference on Emerging Trends in Electrical Electronics & Sustainable Energy Systems (ICETEESES), pp. 130–138. Sultanpur (2016)
11. Hayes, B., Gruber, J., Prodanovic, M.: Short-term load forecasting at the local level using smart meter data. In: IEEE Eindhoven PowerTech, pp. 1–6. Eindhoven (2015)
12. LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation applied to handwritten zip code recognition. Neural Comput. **1**(4), 541–551 (1989)
13. Aloysius, N., Geetha, M.: A review on deep convolutional neural networks. In: International Conference on Communication and Signal Processing (ICCSP), pp. 0588–0592. Chennai (2017)

14. Springenberg, J.T., Dosovitskiy, A., Brox, T., Riedmiller, M.: Striving for simplicity: the all convolutional net. In: 2nd International Conference on Learning Representations (ICLR). Banff (2015)
15. Dozat, T.: Incorporating Nesterov momentum into Adam. In: 4th International Conference on Learning Representations (ICLR). San Juan (2016)
16. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**, 1929–1958 (2014)
17. Tompson, J., Goroshin R., Jain, A., Lecun, Y., Bregler, C.: Efficient object localization using convolutional networks. In: 28th IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 648–656. Boston (2015)

Long- and Short-Term Approaches for Power Consumption Prediction Using Neural Networks

Juan Carlos Morales, Salvador Moreno, Carlos Bailón, Héctor Pomares, Ignacio Rojas, and Luis Javier Herrera

Abstract This work reviews the challenge of power consumption prediction, approaching both short-term and long-term prediction problems using neural networks. A number of improvements are introduced for both problems using two types of neural nets. For short-term prediction, a modified LSTM network based on direct prediction of four hours horizon is presented, also an alternative model based on Convolutional Neural Networks is also introduced. Different improvements in the short-term prediction by the use of external inputs, and the concatenation of the LSTM sub-net outputs for the prediction, among others, are shown. Finally, long-term forecasting is considered and a modified LSTM model is proposed and trained for that purpose, achieving notable improvements with respect to non-dedicated models.

Keywords Power consumption · Time series prediction · Long-term prediction · Short-term prediction · LSTM networks · Convolutional neural networks

1 Introduction and Problem Description

Neural Networks underwent a terrific revolution with the advent of Deep Learning at the early years of this decade. Feed Forward Neural Networks, Convolutional Neural Networks, and Recurrent Neural Networks and among the most popular types of networks when dealing with a data modeling or pattern recognition tasks. Specifically, for time series prediction, among other problems, Recurrent Neural Networks and their specific forms LSTM and GRU networks, have shown a great performance in their operation [1–3]. Nonetheless, also Convolutional Neural Networks have shown to present interesting capabilities in the extraction of specific patterns from the sequences of data helpful in the prediction of the future [4, 5].

Many works related to GRU and LSTM networks have appeared in the recent literature for power consumption forecasting [1, 4]. Several works deal with household

J. C. Morales · S. Moreno · C. Bailón · H. Pomares · I. Rojas · L. J. Herrera (✉)
Computer Architecture and Technology Department, University of Granada, Granada, Spain
e-mail: jherrera@ugr.es

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*, Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_15

power management and prediction [6]; however, this work deals with a power consumption at a national level. In this regard, the use of additional information for power consumption prediction in the literature [7, 8] has shown to be essential. It can include data such as GDP and other macroeconomic measurements, as well as weather conditions.

Short-term forecasting is usually the most tackled problem in power consumption forecasting. To our knowledge, it is the most critical issue at national power management politics. However, long-term forecasting can help improving long-term decision-making, and perform thorough analyses depending on weather changing conditions for instance. Long-term prediction implies the construction of a model which is able to learn from its own inputs and be robustly accurate in the further horizon. Experience has shown the fact that optimized models for short-term prediction noisily fail in the long term, and additional techniques are needed to attain reasonable models for the long term. To our knowledge few works have dealt with the optimization of models for both objectives short- and long-term forecasting of power consumption.

This work deals with the Iberian Peninsula Power Demand series, specifically the data from January 2009 to June 2016 [9, 10]. It extends a previous conference work [11], and presents improved short- and long-term approaches for the prediction of power consumption. Specifically, a novel LSTM model for short-term prediction of 24 complete values is presented and compared with previous approaches. Also a CNN model for the same problem is presented. Finally an enhanced LSTM network training method is presented for long-term forecasting.

The rest of this work is organized as follows. Section 2 presents the Spanish power consumption dataset. Section 3 presents a brief introduction to Neural Networks and Sect. 4 describes the specific methods proposed for power consumption forecasting. Section 5 presents the results obtained, comparing them with previous works. Finally, in Sect. 6 the conclusions of this work are presented.

2 Data Description

2.1 Power Consumption Dataset

The dataset contains information on the electrical power consumption of the country of Spain at a national level. The evolution of the current Spanish power consumption can be observed at [12]. In this dataset, data consumption is sampled each 10 min. The data available for this study is from the period of January 1, 2009 until June 30, 2016, making for a total of more than 350.000 data points. The specific period considered for this study was the same as that used in previous works [9, 10] . The shape of the series can be seen in Fig. 1, where the whole dataset and a close-up over a winter week and a summer week are shown.

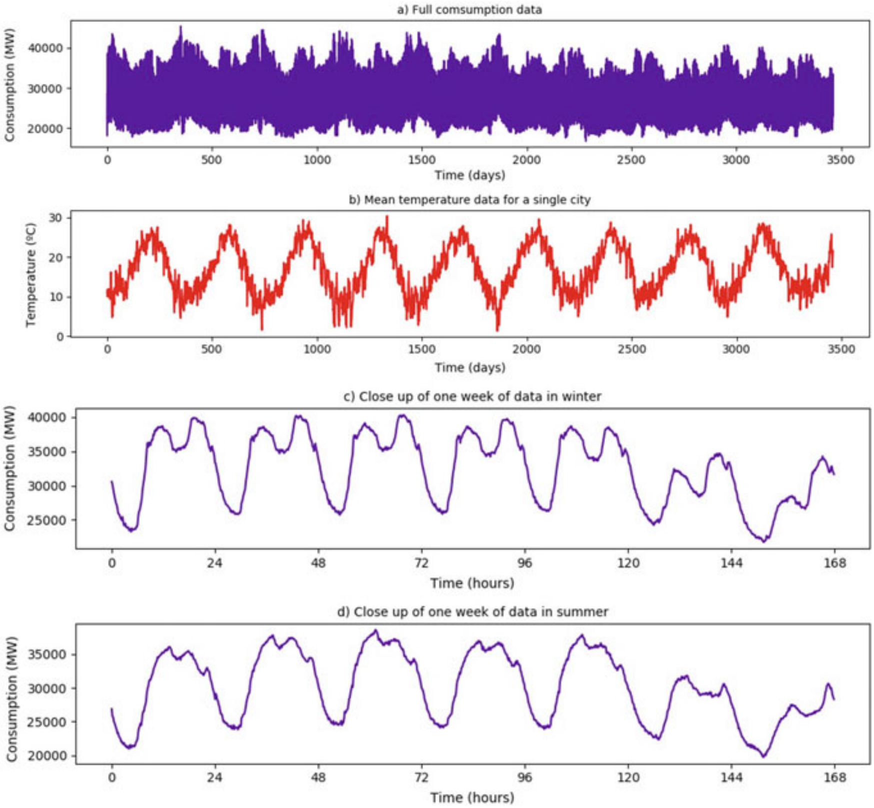


Fig. 1 **a** Power consumption of Spain series from January 1, 2009 to June 30, 2016 measured in MW. **b** Temperature evolution over the full dataset for a single city (Barcelona). **c** Detailed view of the consumption over a winter week (second week of 2009). **d** Detailed view of the consumption over a summer week (27th week of 2019): different daily patterns can be observed depending on the day of the week (lower power consumption in weekends)

The series presents a noticeable daily periodicity, with large minimum over the night and other minimum in the early afternoon. Weekend always presents a lower consumption than usual weekdays. Being this the case, we can also talk about a weekly periodicity. The consumption is also affected by multiple other factors than the day of the week and the time of the day. Whether the day is a holiday or not leads to similar behavior that in weekends. Also, the season show to play an important role, being the consumption higher in winter than in summer. This effect could be due to weather conditions, but also for the fact that in winter there are less sun hours than in summer.

2.2 External Data

Apart from the power consumption, the dataset was extended to consider further information with multiple external variables per day. Each day is marked with the day number within the year, whether the day is a weekend or not and whether the day is a holiday or not. Moreover, daily temperature information (mean, minimum, and maximum temperatures) and precipitation information were collected. Specifically, as mean country values were considered could be misleading due to differences in the weather behavior along the country, 10 of the most inhabited cities of Spain were taken into account. The geographical distribution along the territory was a conditioning criterion for this selection. The cities used are Madrid, Barcelona, Valencia, Sevilla, Coruña, Bilbao, Vizcaya, Málaga, Murcia, and Alicante.

Then, in total for each day, 144 values for the power consumption were collected, plus 43 values as external data: day of the year, day of the week, national holiday, min, mean, and max temperature level and precipitation level for each of the aforementioned 10 cities.

3 Introduction to Neural Networks

Neural networks are computational architectures that try to model connected neurons able to learn from input data. They have shown the ability to solve a large variety of problems within the pattern recognition and prediction areas, with applications in all sciences, such as image classification and processing, signal processing, language processing and synthesis, time series prediction, etc. In their origin, they are built using very small units connected in a particular way that are called neurons or perceptrons.

While artificial neurons are inspired by their biological counterpart, they are far simpler. In brief, a perceptron is modeled using a very basic formula as shown in Eq. 1.

$$y = f\left(\sum_i w_i x_i + b\right) \tag{1}$$

where w_i are the trainable weights, b is the trainable bias, and x_i are the inputs. f is a non-linear function called the activation function. A single perceptron is shown in Fig. 2. The most used activation function is the Rectified Linear Unit (ReLU). The ReLU function is written as $f(x) = \max(x, 0)$.

Summing up, neural networks are essentially combinations of perceptrons rearranged and trained in a particular way that makes the model being able to learn complex information. The most simple neural network architecture is a fully connected feedforward neural network, often called multilayered perceptron. A schematic of the connections is shown in Fig. 3. A classic neural network is trained using the

Fig. 2 Single perceptron

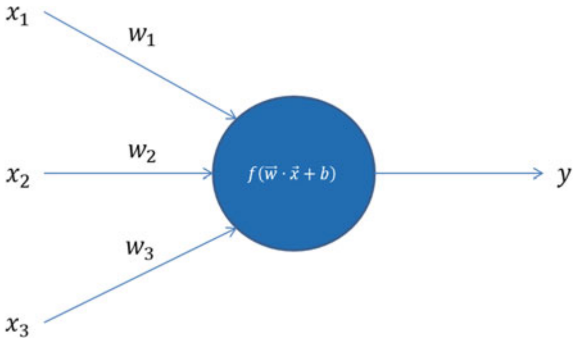
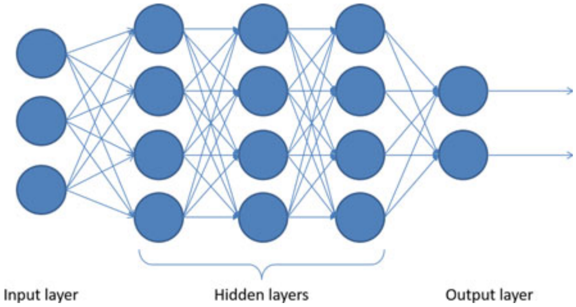


Fig. 3 Multilayered perceptron. In the case of having many layers, we can also talk about Deep Neural Networks



backpropagation of gradients. This works by first defining an error metric for the output and then trying to minimize that error metric using the chain rule, making the derivative with respect to the parameters (weights and biases). After computing the gradients, the parameters are updated using them to minimize the error.

Although neural networks have existed for decades (since 1970, with the introduction to backpropagation), their peak interest has arisen very recently, when our computers became powerful enough to run Deep Neural Networks using very large datasets. Nowadays, neural networks have become the most used machine learning technique and many different architectures have been created that are able to outperform humans in very complex tasks, which for instance is the case of AlphaGo Zero [13] (in the Chinese game of Go) or, more recently, AlphaStar [14] (in the video-game Starcraft). Other architectures such as FaceNet [15] (face recognition model), or the recently released GPT-2 language model [16] present a performance that, in many cases, makes them almost indistinguishable from a human.

Despite Feedforward Neural Networks have shown great performance for countless simple and complex tasks, many different architectures have been created to approach specific complex problems. Two such models are Recurrent Networks (RNNs), being the most well-known model the Long Short-Term Memory Neural Networks (LSTM) and Convolutional Neural Networks (CNNs). Both are summarized next.

3.1 Long Short-Term Memory Neural Networks

Recurrent Neural Networks differ from the previously explained neural networks in that they have an internal memory (hidden state) that evolves after each time step depending on the input received. LSTMs are the evolution of RNNs, being the main difference between LSTMs and RNNs, the existence of a second internal memory that controls in a more precise way how the hidden state changes at a certain time step (see Fig. 4). The existence of the cell state helps in learning not only short-term time dependencies, but long-term ones as well. Thanks to this evolving states, these networks perform really well when learning data with time dependencies. Another advantage of this type of network over other RNN architectures is that they avoid the vanishing gradient problem.

3.2 Convolutional Neural Networks

Convolutional Neural Networks are the state of the art in image and video processing, among other disciplines, due to their ability to learn and extract specific complex patterns from multi-dimensional data (for instance, images and video) with spacial dependencies. Their efficacy is also tested in one-dimensional data, for instance, for signal processing (medical data such as EEG, etc.). For image processing, CNNs traditionally take as input a three-dimensional matrix and produce another three-dimensional matrix as output. This is achieved by applying a convolution operation between the original matrix and a set of filters (see Fig. 5. For example, for an initial (U × V × W) (height, width, depth) matrix, we can apply a set of T filters with size (R × S × W) to obtain an output matrix with size (U × V × T). It is common to

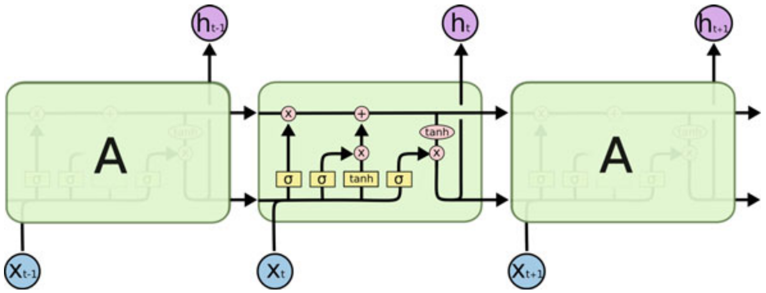


Fig. 4 Basic LSTM architecture, (taken from Colah's blog https://colah.github.io/posts/2015-08-Understanding-LSTMs/). The X and h are the inputs and outputs at each time step. The upper line that connects each time step with the following is the cell state, whereas the bottom one is the hidden state

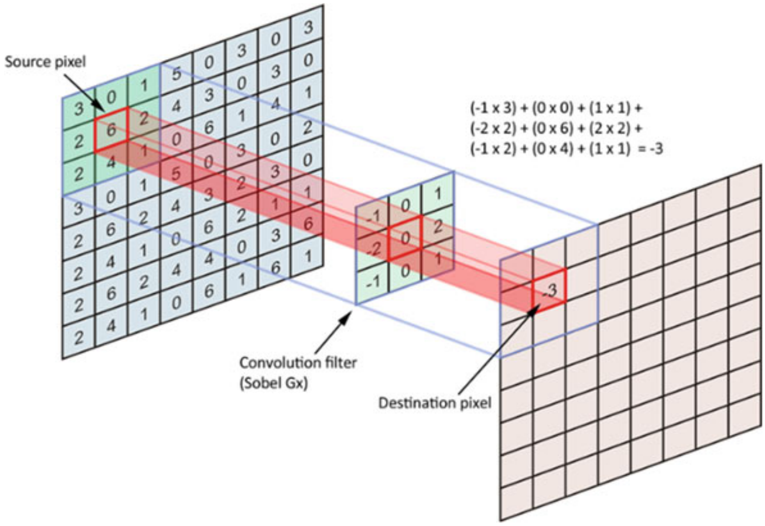

Source pixel
(-1 x 3) + (0 x 0) + (1 x 1) +
(-2 x 2) + (0 x 6) + (2 x 2) +
(-1 x 2) + (0 x 4) + (1 x 1) = -3
Convolution filter
(Sobel Gx)
Destination pixel

Fig. 5 Simple sketch of a convolutional operation [17]. The input data (left) is multiplied by a filter (middle) number by number. The results of all products are added to form a single output digit (right). The empty cells in the input data are considered here as 0s (zero padding). This is used to have an output with the same size as the input. The filter slides through the whole input matrix to compute all outputs

use small filters (R and S are usually 3, 5 or 7), and pooling layers (down-sampling) after the convolution operations, in order to reduce the number of parameters of the network and improve the training and operation speeds.

4 Methodology

As aforementioned, we have divided the prediction problem in two different subproblems of high interest: the short-term prediction and the long-term prediction. For the short-term prediction, we have designed two different types of models, LSTM-based models and CNN-based models. For the long-term prediction, we have designed an LSTM-based model.

For all of the design and training processes, the power consumption data has been normalized to get an N(0,1) normal distribution since normalization of the data is highly important for a good performance of the network. Furthermore, the external data (temperature and precipitation data) has also similarly been normalized.

4.1 Proposed Short-Term LSTM Network

The first methodology used to approach the problem of short-term power consumption prediction is an LSTM neural network.

A first model was designed to take as inputs the 240 values, i.e., 40 h, previous to the starting point of the prediction and predicts the next 24 values, i.e., 4 h. The information was processed in the following way: the 240 input values were split into packets of 4 h (24 values), which were sequentially fed to the LSTM at each time step. But for the last packet of 4 h, none of the previous outputs of the LSTM were collected. This process acts like an encoder and allows the LSTM to build up its hidden state for the prediction of each time step. An approach similar to this one has been used in other areas such as machine translation [18].

After the 10 packets of input data have been fed to the network, the output of the LSTM (the hidden state, 400 values) is collected and concatenated with the external data. Two sets of external data are provided; one corresponding to the start of the prediction and one corresponding to the end of the prediction, making for a total of 88 values. Each set of external data consists of the 43 values previously explained plus the hour at which the data point was taken. The resultant vector is then passed through three fully connected layers to get the final prediction. After this process, the hidden state of the LSTM is reset to the default state (all zeros) for the next prediction. The model has been trained with and without the use of external data to test the improvement in performance.

The training process is performed for 23 epochs, using a batch size of 128 and Adam optimizer. The learning rate used was 5e-3 for the first two epochs. After that, it was diminished to 2.5e-5. Additionally, L2 regularization with weight 1e-5 was used to avoid overfitting.

A sketch of this first LSTM network designed is shown in Fig. 6. Results on this network were provided in [11].

Then, some improvements on this first designed LSTM network were provided and assessed. First, the length of the input was extended, and tests were performed by multiplying the amount of previous data fed to the network. Preliminary test on the train data showed small but significant improvements by doubling the data size to 20 time steps to predict the next step. Second, the approach presented in [6] was used to concatenate not only the LSTM output of the last step, but also the output from all the input steps with the external data to be input to the dense layers. This technique showed to improve the performance of the prediction in the aforementioned work and was also assessed here. Finally, due to the large dimensionality at the output of the LSTM, we decided to use a normalization layer as in [19] to allow the FNN stage to have normalized inputs.

It is to highlight that these modifications imply an increase of the dimensionality of the input of the network and of the input of the dense part of the network, thus making the model slower to train and needing more memory, which can make this specific architecture more difficult to run in slower machines. The extended short-term LSTM model was trained using the same hyperparameters as the ones exposed above.

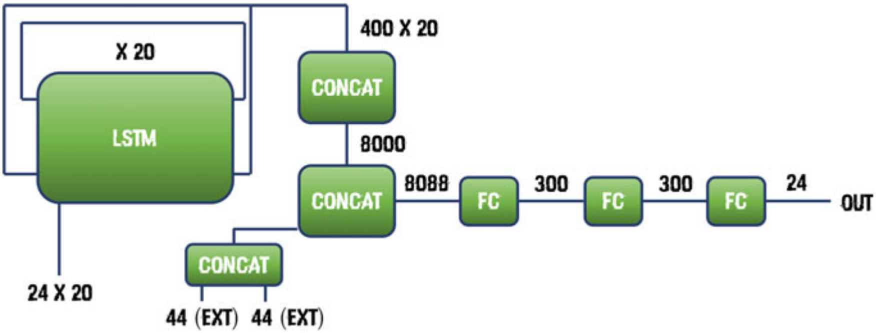


Fig. 6 Proposed LSTM network for short-term forecasting of 24 values (4 h). The sketch corresponds to the extended model that concatenates all outputs

4.2 Proposed Convolutional Neural Network

As explained in Sect. 3.2, CNNs perform especially well when dealing with spatially correlated data. In our case, the power consumption can be stored as a 1D vector where close points have a high correlation. However, it can be also noted that the form of the consumption is correlated between close days. To take advantage of this extra correlation it was decided to distribute the training data in a 2D matrix resembling an image.

Days of the consumption data were stacked in several rows of a matrix. Each row was created by concatenating the m previous values to the hour we want to start the prediction (but corresponding to past days) and the n following values, being n the number of values we want to predict. For the last row, only the m previous values are known, so for the following n a placeholder value (all zeros) is used. In this way, at each column the time stamp of the day for all rows is the same. By organizing the data like this, it is expected that the network is able to use the spatial information not only between close datapoints, but also between close days, to fill the all zeros region with an actual prediction.

This two-dimensional input is then processed by a convolutional layer with kernel size (3 × 11), followed by another two convolutional layers using the same kernel size. Since the dimensionality of the data is not very high, no pooling layers were used between the convolutional layers. The result is then flattened into a vector and, in the same way as in the LSTM model, it is concatenated with 88 values of external data. Finally, the result is processed by three fully connected layers to get the prediction of the n values. The value of n has been set to 24 and the value of m to 120, to cover a full day per row. The number of rows was initially set to three [11]. However, further tests were performed by increasing the size of the previous data window taken. Assessment on these alternatives was performed, and an optimal model taking five rows was selected, as it will be presented in the results section.

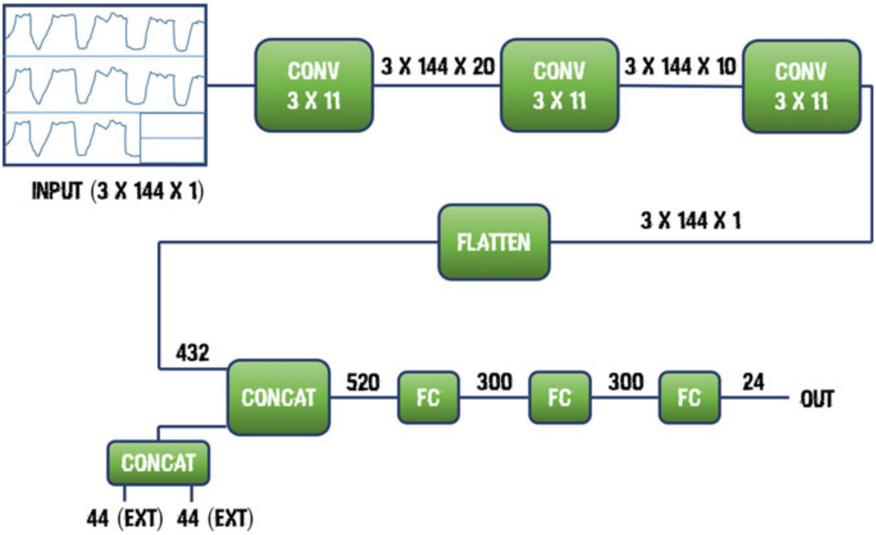


Fig. 7 Proposed CNN network for short-term forecasting of 24 values (4 h)

As in the LSTM model, the network has been trained with and without the external data, to measure the increase in performance. The network was trained for 17 epochs using a batch size of 64 and Adam optimizer. During the first two epochs, the learning rate used was 2e-3 and from there onwards, the learning rate was 5e-5. As in the LSTM model, a L2 regularization with a regularization strength of 1e-5 was used to avoid overfitting.

A sketch of the network is shown below in Fig. 7.

4.3 *Improvements over the LSTM Network for Long-Term Time Series Forecasting*

The two previous model designs have been trained for short-term forecasting; therefore, they are expected to perform properly for predictions in the short term. However, they do not assure a good generalization capability when applying them recursively (that is, using the previous prediction to get the next one) for long-term prediction [20].

A modified long-term LSTM model was designed, still predicting 24 values at each time step. However, the input data consists of the 168 previous values (28 h) instead of 240 and it is not split into different packets, but fed in a single time step. Furthermore, the hidden state of the LSTM is not reset after a prediction, but carried over for the next prediction. Additionally, the network is trained in a recursive way. The 24 predicted values are incorporated as a part of the 168 input values for the

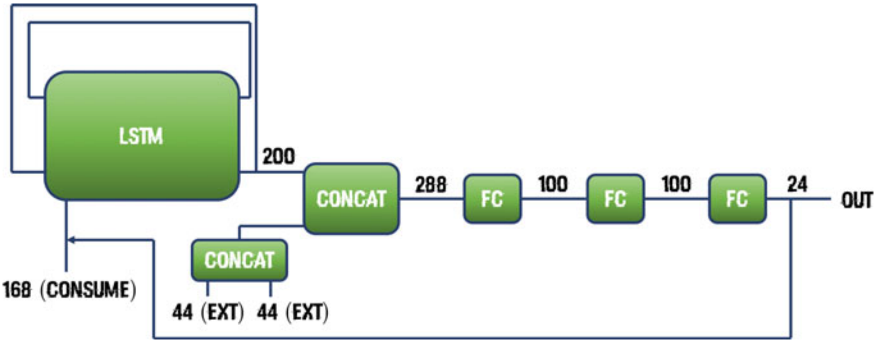


Fig. 8 Proposed LSTM network for long-term forecasting (recursive prediction)

next prediction. In this way, after seven predictions, no true data is being fed into the network (aside from the external data), only the predictions. However, to avoid the training to get very noisy, true data is fed again each 11 time steps.

The rest of the architecture is very similar to the short-term LSTM model. The output of the LSTM (this time of dimension 200 for the sake of lower computational cost) is concatenated with 88 points of the external data and passed through three fully connected layers to get the final 24 value prediction. Unlike the short-term models, this one has not been trained without the external data, since that is the only data it can truly rely upon. The architecture of the network is shown in Fig. 8.

The training process was carried out for 31 epochs using a batch size of 24 and Adam optimizer (the batch size is fixed to be the same as the number of outputs since we only have that number of possible starting points without repeating outputs). During the first 9 epochs, the learning rate used was 5e-4 and from there onwards, the learning rate was reducend to 1e-4. An L2 regularization with a regularization strength of 3e-4 was used to avoid overfitting.

Final tests were performed by multiplying the number of inputs, leading to a very noticeable boost of performance for the long-term prediction when considering double inputs (336 instead of 168). Results on this will be shown and discussed in the results section.

5 Results

All experiments were carried out on a PC with GPU Nvidia GeForce GTX 760M. Codes were implemented under Python 3.6.8 and Tensor Flow 9.0. The performance measure used for the simulations was the Root Mean Square Error (RMSE), as it is considered a standard performance measure for time series prediction, and for the sake of fair comparison with previous works on the same dataset.

A training-val-test subdivision was performed on the dataset using a 80%–10%–10% ratio. Thus, the test set comprised the last 346 daily values of the series, i.e., from the 12 of July 2015 to the 21 of Jun 2016.

Even though the first two models have been trained for short-term prediction and the third one for long-term prediction, we have tested each model for both types of predictions to compare results (adapting the short-term models to predict recursively and disabling the recursive prediction for the long-term model).

5.1 Short-Term Time Series Forecasting

The results obtained for short-term prediction for 4 h (24 values) ahead using the basic LSTM network presented in subsection 4.1 using 10 previous time steps to predict, reached 325.36 and 337.04 of test RMSE, with and without external variables, respectively (see Table 1). The extended LSTM model using 20 previous time steps reached a test RMSE of 315.15. Finally enhanced model by the method provided in [6] finally provided a relevant improved test RMSE of 272.16. CNN model presented in Sect. 4.2 stacing three previous days data reached 346.90 and 328.32 of test RMSE, with and without external variables, respectively. CNN model stacking five days of input data lead to a slight improvement (RMSE 321.94). The long-term model was tested using 28 and 56 h of previous data for each time steps—see next subsection 5.2— with the second case winning by a noticeable margin.

As it can be observed, results show that the base LSTM and base CNN models perform very similar, with a slight advantage of the base LSTM model. The improvements tested in the different networks have led to noticeable increases of performance in the short-term prediction. Moreover, there is also a clear improvement of the mod-

Table 1 4 h prediction

Short term prediction		
Method	Training RMSE	Test RMSE
LSTM(10 prev values)	317.28	337.04
LSTM(10 prev values) + external features (ext.)	301.44	325.36
LSTM(20 prev values) + ext.	297.82	315.15
LSTM(20 prev values) + ext + all outputs	**256.58**	**272.16**
CNN (3 rows)	345.96	346.90
CNN (3 rows) + external features	314.68	328.32
CNN (5 rows) + external features	296.82	321.94
Long term LSTM (28 h)	–	718.82
Long term LSTM (56 h)	–	655.31
DFFNN [9]	–	501.14
Deep Learning approach [10]	–	587.47

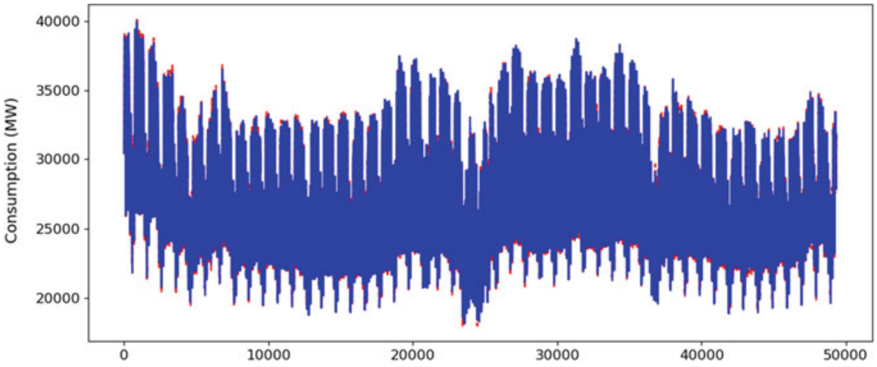


Fig. 9 Short-term LSTM model over the test set. The dashed red line is the real data while the blue line is the prediction

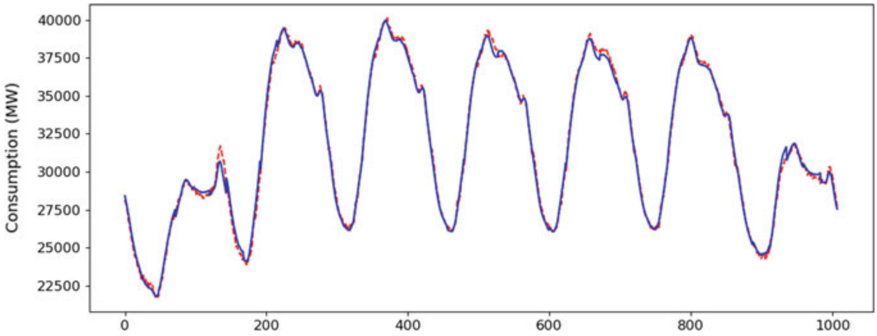


Fig. 10 Close up of the short-term LSTM model (using all outputs) prediction for the first week of the test set. The dashed red line is the real data while the blue line is the prediction

els when using the external data. Finally, results show an improvement over the ones obtained in previous works [9, 10], which consider 24 different feedforward models to predict each of the 24 values. It is to be highlighted that the application of the long-term LSTM model to short-term forecasting led to a RMSE in the test set of 655.31, making it the worst of all of them for short-term prediction, which was expectable as it was not specifically optimized to predict short-term behaviors.

Figures. 9, 10, and 11 show examples of the prediction obtained by the optimized LSTM model using all outputs of the LSTM sub-network.

5.2 Long-Term Time Series Forecasting

Three models were considered for the assessment of neural network architectures for long-term prediction: base short-term LSTM model (see Table 1), base CNN short-term model, and two versions of long-term LSTM model (base and optimized

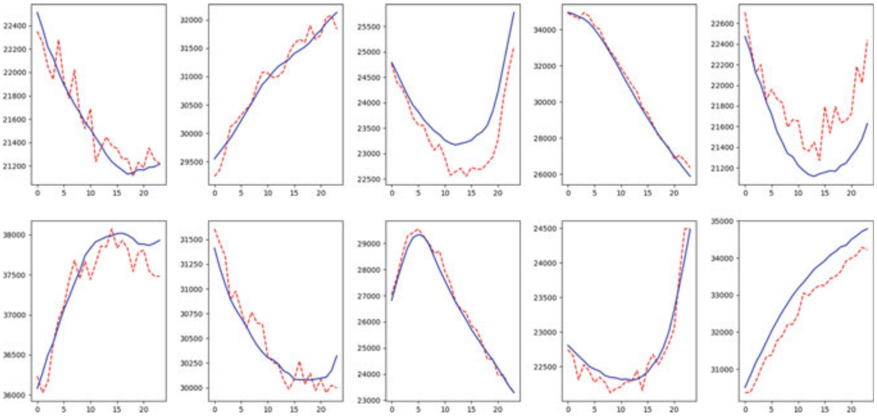


Fig. 11 Close up of the prediction of the short-term LSTM model using all outputs for ten 4 h segments. The dashed red line is the real data while the blue line is the prediction. It is important to notice that the scale is not the same in all graphs

Table 2 Comparison for long-term prediction on the test dataset, among the long-term LSTM, the short-term LSTM, and the CNN short-term prediction models

Long term prediction

Horizon	Short-term model mean RMSE (std.)	CNN model mean RMSE (std.)	Long-term model (28 h) mean RMSE (std.)	Long-term model (56 h) mean RMSE (std.)
One-week-ahead	1366.91 (733.61)	1020.06 (635.74)	925.08 (435.47)	805.29 (149.79)
One-month-ahead	1999.79 (634.33)	1500.14 (1063.10)	1127.46 (400.77)	826.23 (39.53)
Ten-months-ahead	2359.24 (18.06)	2351.03 (933.13)	1286.49 (17.46)	1268.18 (35.58)

according to Sect. 4.3). For the four of them, the external data have been used since trying to predict recursively using no other information would lead to poor results. For long-term prediction, Table 2 shows the results obtained over the test set for one full week ahead (7 days * 6 periods of 4 h = 42 following predictions), one month (+1200 following predictions) ahead and ten months ahead (+12000 following predictions). The RMSE of both the one week ahead and the one month ahead, are averaged by using 310 different days of the test set as the starting point of the prediction. In the case of the ten months ahead, only 40 days have been used as the starting point. Both, the mean RMSE and the standard deviation are shown for each prediction.

As we can be seen in Table 2, the Long-term models are able to outperform the other two models, especially when the prediction horizon gets longer. The CNN model seems to perform better that the Short-term LSTM model for lower horizon predictions, although they get similar results in the ten months ahead prediction. However, we can say that the Short-term LSTM model presents a much higher consistency (reduced std.) than the CNN model when the prediction window gets bigger. It is important to notice that the small value for the deviation of the LSTM models in the ten months prediction is due to the overlapping of the 40 ten-months predictions in almost nine months. This overlapping also happens in the CNN model, but the deviation is still very high because of its bad consistency. We can also see that the 56 h long-term model is much more accurate and consistent than the 28 h one for the one-week-ahead and one-month-ahead predictions, although they perform very similar for the ten-months-ahead prediction.

Regarding these results, one could argue that we are using external data that we cannot possibly know, since they are in the future. While that is true, there is some external data that we know for sure (such as the time of the day or whether the day is a holyday or not). Moreover, for the temperature and precipitation data, we could use the meteorological forecast, which should be accurate enough for, at least, an one week ahead prediction. The ten months ahead prediction is merely here for comparison purposes, since we cannot have a ten months ahead forecast. Moreover, this type of models could be used to assess future long-term consumption under different or varying future weather conditions.

Figure 12 shows a close up of the recursive prediction over the first week of the test set for all the alternatives. Figure 13 shows the worst prediction for the long-term approach on the test dataset.

The long-term recursive prediction is clearly not as good as the normal short-term prediction (see Tables 1 and 2.2), which is to be expected. Nevertheless, the model presents a reasonably good performance since no true consumption data has been fed to the network during the pass through the whole test set (aside from the starting point). This shows that the network is able to learn a lot from the external data when

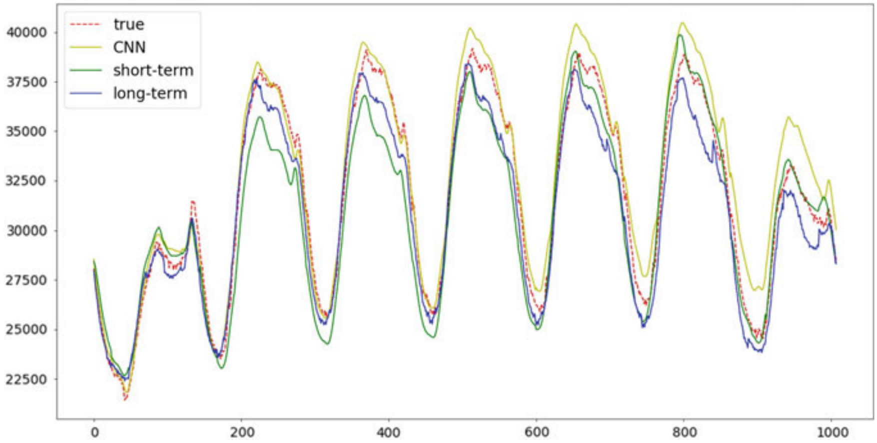


Fig. 12 Close up of the prediction over the first week of the test dataset. The base models for the three alternatives long-term trained LSTM model, short-term LSTM model, and CNN short-term model are compared with the true data

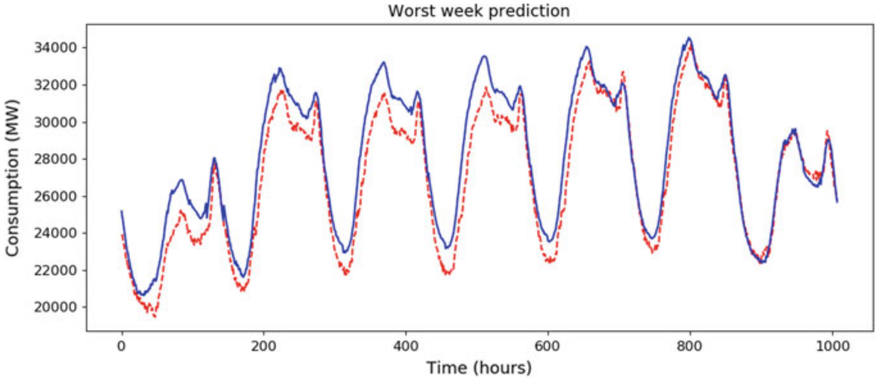


Fig. 13 Long-term LSTM model (56 h) prediction over the worst case on a week ahead prediction. The dashed red line is the real data while the blue line is the prediction

it cannot perfectly rely on the previous consumption data, which is very important since it implies that having an accurate weather forecast can imply a reasonable power consumption prediction.

6 Conclusions

In this work, we have shown multiple deep learning approaches to solve the problem of power consumption prediction. We have presented an LSTM-based and a CNN-based models that focus in predicting the consumption in the short term (4 h), yielding promising results, and improving results on the same series on previous published works. We have also tested the improvement of the networks after adding external data and the improvement of the LSTM when collecting all outputs of the LSTM sub-network to make the final prediction. To address the problem of long-term prediction, we have built an alternative LSTM-based model with a very similar architecture to the short-term-based LSTM but trained in a very different way, showing important improvements over the previously short-term LSTM. Finally, all of the networks, regardless of how they have been trained, have been tested for short- and long-term prediction. Results highlight the importance of the way a network architecture is trained. While the two LSTM models share the same data and almost the same architecture, the way they are trained leads to very different results in each of the two sub-problems, short-term and long-term, presented.

Acknowledgements This research has been possible thanks to the support of project: RTI2018-101674-B-I00 (Spanish Ministry of Economy and Competitiveness –MINECO– and the European Regional Development Fund. –ERDF).

References

1. Angelopoulos, D., Siskos, Y., Psarras, J.: Disaggregating time series on multiple criteria for robust forecasting: The case of long-term electricity demand in Greece. Eur. J. Oper. Res. **275**(1), 252–265 (2019). https://doi.org/10.1016/j.ejor.2018.11.00, https://ideas.repec.org/a/eee/ejores/v275y2019i1p252-265.html
2. Yu, R., Gao, J., Yu, M., Lu, W., Xu, T., Zhao, M., Zhang, J., Zhang, R., Zhang, Z.: LSTM-EFG for wind power forecasting based on sequential correlation features. Future Gener. Comput. Syst. **93**(Environ. Policy Collect. 2015), 33–42 (2019). https://doi.org/10.1016/j.future.2018.09.054, https://app.dimensions.ai/details/publication/pub.1107641229
3. Sagheer, A., Kotb, M.: Time series forecasting of petroleum production using deep LSTM recurrent networks. Neurocomputing **323** (2018). https://doi.org/10.1016/j.neucom.2018.09.082
4. Amarasinghe, K., Marino, D.L., Manic, M.: Deep neural networks for energy load forecasting. In: 2017 IEEE 26th International Symposium on Industrial Electronics (ISIE), pp. 1483–1488 (2017). https://doi.org/10.1109/ISIE.2017.8001465
5. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016), http://www.deeplearningbook.org
6. Yan, K., Wang, X., Du, Y., Jin, N., Huang, H., Zhou, H.: Multi-step short-term power consumption forecasting with a hybrid deep learning strategy. Energies **11**(11) (2018). https://doi.org/10.3390/en11113089, https://www.mdpi.com/1996-1073/11/11/3089
7. Angelopoulos, D., Siskos, Y., Psarras, J.: Disaggregating time series on multiple criteria for robust forecasting: the case of long-term electricity demand in Greece. Europ. J. Oper. Res. **275**(1), 252–265 (2019). https://doi.org/10.1016/j.ejor.2018.11.003, http://www.sciencedirect.com/science/article/pii/S0377221718309287
8. Li, H., Mao, X., Zhu, L., Yao, Y., Tan, J.: Saturation load forecasting based on long short-time memory network. In: 2018 2nd IEEE Conference on Energy Internet and Energy System Integration, pp. 3355–3360 (2018). https://doi.org/10.1109/EI2.2018.8582222
9. Torres, J.F., Gutiérrez-Avilés, D., Troncoso, A., Martínez-Álvarez, F.: Random hyper-parameter search-based deep neural network for power consumption forecasting. In: Rojas, I., Joya, G., Catala, A. (eds.) Advances in Computational Intelligence, pp. 259–269. Springer International Publishing, Cham (2019)
10. Torres, J., Galicia de Castro, A., Troncoso, A., Martínez-Álvarez, F.: A scalable approach based on deep learning for big data time series forecasting. Integr. Comput.-Aided Eng. **25**, 1–14 (2018). https://doi.org/10.3233/ICA-180580
11. Morales, J.C., Moreno, S., Bailon, C., Pomares, H., Rojas, I., Herrera, L.J.: Long and short term prediction of power consumption using LSTM networks. In: Valenzuela, O., Rojas, F., Pomares, H., Rojas, I. (eds.) International Conference on Time Series and Forecasting 2019, pp. 914–926 (2019)
12. https://demanda.ree.es/visiona/home
13. Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., et al.: Mastering the game of go without human knowledge. Nature **550**(7676), 354–359 (2017). https://doi.org/10.1038/nature24270, http://dx.doi.org/10.1038/nature24270
14. Vinyals, O., Babuschkin, I., Czarnecki, W.M., Mathieu, M., Dudzik, A., Chung, J., Choi, D.H., Powell, R., Ewalds, T., Georgiev, P., et al.: Grandmaster level in starcraft ii using multi-agent reinforcement learning. Nature **575**(7782), 350–354 (2019). https://doi.org/10.1038/s41586-019-1724-z, https://doi.org/10.1038/s41586-019-1724-z
15. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: a unified embedding for face recognition and clustering, pp. 815–823 (2015). https://doi.org/10.1109/CVPR.2015.7298682
16. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language models are unsupervised multitask learners (2019)
17. Taherkhani, F., Dawson, J., Nasrabadi, N.M.: Deep sparse band selection for hyperspectral face recognition (2019)

18. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, vol. 27, pp. 3104–3112. Curran Associates, Inc. (2014). http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf
19. Ba, J., Kiros, J., Hinton, G.: Layer normalization (2016)
20. Herrera, L., Pomares, H., Rojas, I., Guillén, A., Prieto, A., Valenzuela, O.: Recursive prediction for long term time series forecasting using advanced models. Neurocomputing **70**(16), 2870–2880 (2007). https://doi.org/10.1016/j.neucom.2006.04.015, http://www.sciencedirect.com/science/article/pii/S0925231207001622. Neural Network Applications in Electrical Engineering Selected papers from the 3rd International Work-Conference on Artificial Neural Networks (IWANN 2005)

Forecasting Complex/Big Data Problems

Freedman's Paradox: A Solution Based on Normalized Entropy

Pedro Macedo

Abstract In linear regression models where there are no relationships between the dependent variable and each of the potential explanatory variables-a usual scenario in real-world problems-some of them can be identified as relevant by standard statistical procedures. This incorrect identification is usually known as Freedman's paradox. To avoid this disturbing effect in regression analysis, an info-metrics approach based on normalized entropy is discussed and illustrated in this work. As an alternative to traditional statistical methodologies currently used by practitioners, the simulation results suggest that normalized entropy is a powerful procedure to identify pure noise models.

Keywords Big data · Info-metrics · Regression · Variable selection

1 Introduction

Freedman [2, p. 152] alerts for a potential problem in regression analysis when stating "[…] in a world with a large number of unrelated variables and no clear a priori specifications, uncritical use of standard methods will lead to models that appear to have a lot of explanatory power." Through simulation studies and asymptotic theory it is demonstrated some technical features of this misleading interpretation, including the behavior of the t-test, the F-test, and the coefficient of determination, R^2. In a regression model where does not exist relationships between independent/explanatory variables and the dependent variable, Freedman [2] shows that, if there are many explanatory variables in the model, the R^2 will be high and some explanatory variables can be easily considered relevant variables through common significance tests.

P. Macedo (✉)
CIDMA - Center for Research and Development in Mathematics and Applications, Department of Mathematics, University of Aveiro, 3810-193 Aveiro, Portugal
e-mail: pmacedo@ua.pt
URL: https://cidma.ua.pt/

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*, Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_16

The concept of normalized entropy belongs to info-metrics [4], a research area at the intersection of statistics, computer science, and decision theory, where the maximum entropy principle established by Jaynes [8, 9] plays a central role. Maximum entropy provides a simple tool to make the best prediction (i.e., the one that is the most strongly indicated) from the available information and it can be seen as an extension of the Bernoulli's principle of insufficient reason. Jaynes [8, pp. 622–623] recognizes the importance of the Shannon [13] entropy measure as a criterion for the "amount of uncertainty" and presents a magnificent statement: "[…] in making inferences on the basis of partial information we must use that probability distribution which has maximum entropy subject to whatever is known. This is the only unbiased assignment we can make […]." The merit of the maximum entropy principle is unquestionable, although perhaps not yet fully recognized; see, for example, Soofi [14, p. 1244].

Although there are many other methodologies for variable selection (e.g., stepwise family, best subsets, Bayesian model averaging, cross-validation, lasso and its generalizations like elastic net and oscar), usually requiring a lot of computation effort, this work is intended only to illustrate the use of normalized entropy, which requires just one (and simple) analysis of the sample, in the context defined by Freedman [2] (pure noise models). Even though subsequent estimation and validation procedures will depend on the characteristics of the resulting models (e.g., influential observations, collinearity, heteroscedasticity) and the criteria of the researcher (e.g., interpretation, prediction accuracy, precision), different maximum entropy estimation procedures along with variable selection can be easily implemented in more general regression models; see, for example, Golan [3] and Golan et al. [5].

To illustrate how normalized entropy can be used to avoid the abovementioned disturbing effect in regression analysis, the generalized maximum entropy (GME) and generalized cross entropy (GCE) estimators are briefly presented in Sect. 2, along with the definition of normalized entropy. The remaining article is organized as follows: in Sect. 3 the simulation studies are implemented; some conclusions and topics for future research are given in Sect. 4.[1]

2 Maximum Entropy Estimators and Normalized Entropy

Consider a linear regression model defined as

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{e}, \tag{1}$$

where $\boldsymbol{y}$ denotes a $(N \times 1)$ vector of noisy observations, $\boldsymbol{\beta}$ is a $(K \times 1)$ vector of unknown parameters to be estimated, $\boldsymbol{X}$ is a known $(N \times K)$ matrix of explanatory variables, and $\boldsymbol{e}$ is the $(N \times 1)$ vector of random disturbances, typically assumed to have a conditional expected value of zero and representing spherical disturbances.

[1] This work is an extension of the conference paper [11].

Golan et al. [5, pp. 86–93] proposed a reformulation of the linear regression model in (1) as

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{Z}\boldsymbol{p} + \boldsymbol{V}\boldsymbol{w}, \tag{2}$$

where

$$\boldsymbol{\beta} = \boldsymbol{Z}\boldsymbol{p} = \begin{bmatrix} \boldsymbol{z}_1' & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{z}_2' & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{z}_K' \end{bmatrix} \begin{bmatrix} \boldsymbol{p}_1 \\ \boldsymbol{p}_2 \\ \vdots \\ \boldsymbol{p}_K \end{bmatrix}, \tag{3}$$

with $\boldsymbol{Z}$ a $(K \times KM)$ matrix of support spaces and $\boldsymbol{p}$ a $(KM \times 1)$ vector of unknown probabilities to be estimated, and

$$\boldsymbol{e} = \boldsymbol{V}\boldsymbol{w} = \begin{bmatrix} \boldsymbol{v}_1' & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{v}_2' & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{v}_N' \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \\ \vdots \\ \boldsymbol{w}_N \end{bmatrix}, \tag{4}$$

with $\boldsymbol{V}$ a $(N \times NJ)$ matrix of support spaces and $\boldsymbol{w}$ a $(NJ \times 1)$ vector of unknown probabilities to be estimated. In this reformulation, each β_k, $k = 1, 2, \ldots, K$, and each $e_n, n = 1, 2, \ldots, N$, are viewed as expected values of discrete random variables $\boldsymbol{z}_k$ and $\boldsymbol{v}_n$, respectively, with $M \geq 2$ and $J \geq 2$ possible outcomes, within the lower and upper bounds of the corresponding support spaces. Additional details can be found in Golan [4], Chap. 13.

To illustrate reparameterizations (3) and (4), suppose a simple linear regression model ($K = 2$), only four observations ($N = 4$), and consider the support spaces as $[-10, 10]$ and $[-1, 1]$, respectively, for all the parameters and all the errors, with $M = 5$ and $J = 3$. Thus, with symmetric supports centered on zero and equally spaced points,

$$\boldsymbol{Z}\boldsymbol{p} = \begin{bmatrix} -10 & -5 & 0 & 5 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -10 & -5 & 0 & 5 & 10 \end{bmatrix} \begin{bmatrix} p_{11} \\ p_{12} \\ p_{13} \\ p_{14} \\ p_{15} \\ p_{21} \\ p_{22} \\ p_{23} \\ p_{24} \\ p_{25} \end{bmatrix},$$

and

$$\boldsymbol{V}\boldsymbol{w} = \begin{bmatrix} -1\ 0\ 1 & 0\ 0\ 0 & 0\ 0\ 0 & 0\ 0\ 0 \\ 0\ 0\ 0 & -1\ 0\ 1 & 0\ 0\ 0 & 0\ 0\ 0 \\ 0\ 0\ 0 & 0\ 0\ 0 & -1\ 0\ 1 & 0\ 0\ 0 \\ 0\ 0\ 0 & 0\ 0\ 0 & 0\ 0\ 0 & -1\ 0\ 1 \end{bmatrix} \begin{bmatrix} w_{11} \\ w_{12} \\ w_{13} \\ w_{21} \\ w_{22} \\ w_{23} \\ w_{31} \\ w_{32} \\ w_{33} \\ w_{41} \\ w_{42} \\ w_{43} \end{bmatrix}.$$

For the linear regression model expressed in (1), the generalized maximum entropy (GME) estimator is given by

$$\underset{\boldsymbol{p},\boldsymbol{w}}{\operatorname{argmax}} \left\{ -\boldsymbol{p}' \ln \boldsymbol{p} - \boldsymbol{w}' \ln \boldsymbol{w} \right\}, \tag{5}$$

subject to the model constraints,

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{Z}\boldsymbol{p} + \boldsymbol{V}\boldsymbol{w}, \tag{6}$$

and the additivity constraints for $\boldsymbol{p}$ and $\boldsymbol{w}$, respectively,

$$\begin{aligned} \mathbf{1}_K &= (\boldsymbol{I}_K \otimes \mathbf{1}'_M)\boldsymbol{p}, \\ \mathbf{1}_N &= (\boldsymbol{I}_N \otimes \mathbf{1}'_J)\boldsymbol{w}, \end{aligned} \tag{7}$$

where $\otimes$ represents the Kronecker product. On the other hand, with the same restrictions, the generalized cross entropy (GCE) estimator is given by

$$\underset{\boldsymbol{p},\boldsymbol{w}}{\operatorname{argmin}} \left\{ \boldsymbol{p}' \ln \left(\frac{\boldsymbol{p}}{\boldsymbol{q}_1} \right) + \boldsymbol{w}' \ln \left(\frac{\boldsymbol{w}}{\boldsymbol{q}_2} \right) \right\}, \tag{8}$$

where $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ are vectors with prior information concerning the parameters and the errors of the model, respectively.

The estimators generate the optimal probability vectors $\widehat{\boldsymbol{p}}$ and $\widehat{\boldsymbol{w}}$ that can be used to form point estimates of the unknown parameters and the unknown errors, through the reparameterizations (3) and (4) defined previously. It is important to note that the GME estimator is a particular case of the GCE estimator, when the prior information is expressed as a uniform distribution (vectors $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$). In view of the fact that ill-posed real-world problems seem to be the rule rather than the exception, these estimators have acquired special importance in the set of statistical techniques, by allowing statistical formulations free of restrictive and unnecessary assumptions.

Additionally, to measure the information content of the signal component in a particular model, Golan et al. [5, p. 93, p. 165] defined normalized entropy as

$$S(\widehat{\boldsymbol{p}}) = \frac{-\widehat{\boldsymbol{p}}' \ln \widehat{\boldsymbol{p}}}{K \ln M} \tag{9}$$

in the GME estimator, and

$$S(\widehat{\boldsymbol{p}}) = \frac{-\widehat{\boldsymbol{p}}' \ln \widehat{\boldsymbol{p}}}{-\boldsymbol{q}_1' \ln \boldsymbol{q}_1} \tag{10}$$

in the GCE estimator context. This measure lies between zero (no uncertainty) and one (perfect uncertainty). Concerning variable selection, it is interesting to note that if all the $\boldsymbol{z}_k$ in $\boldsymbol{Z}$ are defined uniformly and symmetrically around zero, then $S(\widehat{\boldsymbol{p}}_k) \approx 1$ implies $\beta_k \approx 0$, because $\widehat{\boldsymbol{p}}_k$ is uniformly distributed. Thus, a variable corresponding to $S(\widehat{\boldsymbol{p}}_k) \approx 1$ has no information content and should be excluded from the model.

Some advantages of this procedure are presented by Golan et al. [5, p. 176]: it is simple to perform, even for a large number of variables (just one analysis of the sample is needed, which represent important computational advantages; it does not require the evaluation of 2^K models); allows the use of non-sample information (through the supports in GME or the vectors with prior information in GCE); is free of asymptotic requirements; involves a shrinkage rule that reduces mean squared error; allows to account for model misspecifications and model uncertainty; and it can be implemented for well- and ill-posed models.

Additional details on maximum entropy estimation, normalized entropy, simulation studies, properties, and asymptotic theory can be found in Conceição Costa and Macedo [1], Golan [4], Golan et al. [5] and Mittelhammer et al. [12].

3 Simulation Studies and Discussion

For comparison purposes, the simulation studies conducted in this work follow the same structure of the ones performed by Freedman [2]. Different matrices are created with 100 rows and 51 columns. All the entries are independent observations generated from the standard normal distribution. To establish a multiple regression model, the first 50 columns are considered as the explanatory variables and the last column as the dependent variable. Given this construction, all the regression coefficients should be considered statistically not significant by the standard t-test. However, as expected, this won't be the case.

Freedman [2] performed two successive model estimations: in the first one are identified the number of coefficients that are statistically significant at the 25% (representing an exploratory analysis) and the 5% (representing a confirmatory analysis) levels; in the second one, only the variables whose coefficients are significant at the 25% level enter to the regression model and the number of coefficients that are statistically significant at the 25% and the 5% levels are identified again. All the results are misleading, in particular on the second stage, where are identified between one and nine statistically significant coefficients in the models (depending on the simulation), at the 5% significance level.[2]

To illustrate variable selection using normalized entropy, the GME and GCE estimators are performed with four different supports: $[-100, 100]$, $[-10, 10]$, $[-5, 5]$, and $[-2, 2]$ for all the parameters. The supports are defined as closed and bounded intervals in which each parameter is restricted to belong. Since there is empirical evidence that different supports provide different results in terms of variable selection, four supports (with five points each) are tested in this work, reflecting different levels of prior information about the parameters.

For each error support, the three-sigma rule is used, considering the standard deviation of the noisy observations (usual procedure in GME literature by using a sample scale statistic), with three points. The number of points in the supports is usually between three and seven, since there is likely no significant improvement in the estimation with more points in the supports.

Regarding the GCE estimator, and following Golan et al. [5, p. 166], which state that "If we believe that potential extraneous variables with zero coefficients exist in the linear statistical model specifications, it would seem reasonable to shrink those close to zero more than others [...]," a vector with prior information is defined as $\boldsymbol{q}_1 = [0.1, 0.2, 0.4, 0.2, 0.1]$ for all the parameters, which will accomplish the idea of additional shrinkage. As mentioned by Golan et al. [5], the priors take over as the solution when they are consistent with the data (this feature of the GCE estimator is revealed in the results).

Due to space limitations, only three models are discussed next.[3] Taking into account the number of regression coefficients that are considered statistically significant at the 25% significance level in the first stage, in the second stage, the three models will have only 18, 14, and 14 variables, respectively. Table 1 presents the number of regression coefficients statistically significant, at different significance levels, in the first stage, for the three models.

Considering the three usual significance levels evaluated in literature, in the first stage, with models including 50 variables each, six, seven, and eight coefficients are considered statistically significant at 10% level, respectively, in Model 1, Model 2, and Model 3. Additionally, four coefficients are considered statistically significant at 5% and none of them is considered statistically significant at 1% level, in Model 1 and Model 2. Regarding Model 3, seven coefficients are considered statistically significant at 5% and two are considered statistically significant at the 1% significance level.

Table 2 presents the number of regression coefficients statistically significant, at different significance levels, in the second stage. Is this second estimation, between seven and eight coefficients are considered statistically significant at 5% and between

[2] The term "statistically significant" is used here following the work of Freedman [2]. For a discussion concerning the use of this expression, see Hurlbert et al. [7].

[3] As will be noted later, the conclusions are qualitatively similar among the several simulated models conducted in this work.

Table 1 Number of coefficients statistically significant (first stage)

	Significance levels						
	1%	2%	3%	4%	5%	10%	25%
Model 1 (50 variables)	0	2	3	4	4	6	18
Model 2 (50 variables)	0	2	2	2	4	7	14
Model 3 (50 variables)	2	4	5	5	7	8	14

Table 2 Number of coefficients statistically significant (second stage)

	Significance levels						
	1%	2%	3%	4%	5%	10%	25%
Model 1 (18 variables)	2	3	5	5	7	11	16
Model 2 (14 variables)	2	3	6	7	7	7	12
Model 3 (14 variables)	5	7	7	8	8	8	12

two and five coefficients are considered statistically significant at 1% level. The worst scenarios occur is Model 1, where 11 coefficients are considered statistically significant at the 10% level, and in Model 3 with five coefficients that are considered statistically significant at the 1% significance level.

Since the models are pure noise, the results are disturbing because they suggest relationships that do not exist between explanatory variables and the dependent variable. Considering that this is an usual procedure adopted by practitioners and some recent alternatives in the literature require a lot of computation effort (e.g., lasso family, cross-validation, glmulti R Package), the following part of the work intends to illustrate how normalized entropy can easily avoid Freedman's paradox with one (and simple) analysis of the sample.

Table 3 presents the normalized entropy (truncated to four decimals) of the models, $S(\widehat{\boldsymbol{p}})$, considering different supports for GME and GCE estimators. It is interesting to see that all values are near one, indicating no information content of the signal in the models (in both stages). This information clearly contradicts the information of R^2 that always presents high values, particularly in the first stage, indicating the presence of models with explanatory power.

Now, and to investigate the information content of each variable, a more detailed analysis is developed and all the $S(\widehat{\boldsymbol{p}}_k)$ for each model are also computed. The results are reported through boxplots, from Figs. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12.

Although in some scenarios, especially in the ones with supports of lower amplitude, slightly lower normalized entropy values are obtained, the values are always very high. Note that the y-axis in versions B are defined just between 0.97 and 1.00, and normalized entropy values range between zero and one, as represented in versions A. The performance of the normalized entropy procedure in terms of variable selection, which is observed in Table 3, as well as in Figs. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, is also achieved in all the other simulated models conducted in this work. The results are qualitatively the same.

Table 3 Normalized entropy for the three models

			Supports			
			[−100, 100]	[−10, 10]	[−5, 5]	[−2, 2]
GME	Model 1	50 variables	0.9999	0.9998	0.9994	0.9972
		18 variables	0.9999	0.9997	0.9991	0.9954
	Model 2	50 variables	0.9999	0.9998	0.9994	0.9972
		14 variables	0.9999	0.9997	0.9991	0.9951
	Model 3	50 variables	0.9999	0.9998	0.9993	0.9969
		14 variables	0.9999	0.9996	0.9987	0.9929
GCE	Model 1	50 variables	0.9999	0.9998	0.9995	0.9982
		18 variables	0.9999	0.9998	0.9994	0.9969
	Model 2	50 variables	0.9999	0.9998	0.9995	0.9982
		14 variables	0.9999	0.9998	0.9993	0.9967
	Model 3	50 variables	0.9999	0.9998	0.9995	0.9981
		14 variables	0.9999	0.9997	0.9991	0.9951

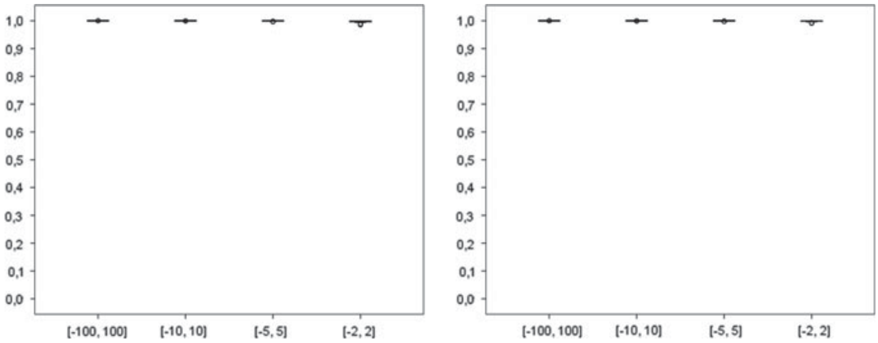


Fig. 1 $S(\widehat{\mathbf{p}}_k)$ with GME (left) and GCE (right) in Model 1 (50 variables)—A

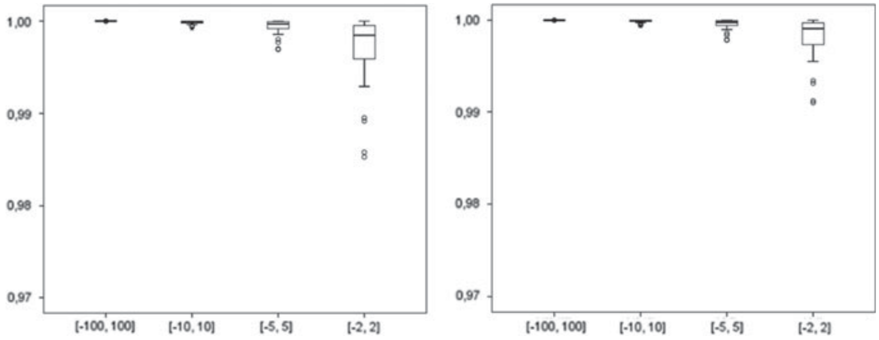


Fig. 2 $S(\widehat{\mathbf{p}}_k)$ with GME (left) and GCE (right) in Model 1 (50 variables)—B

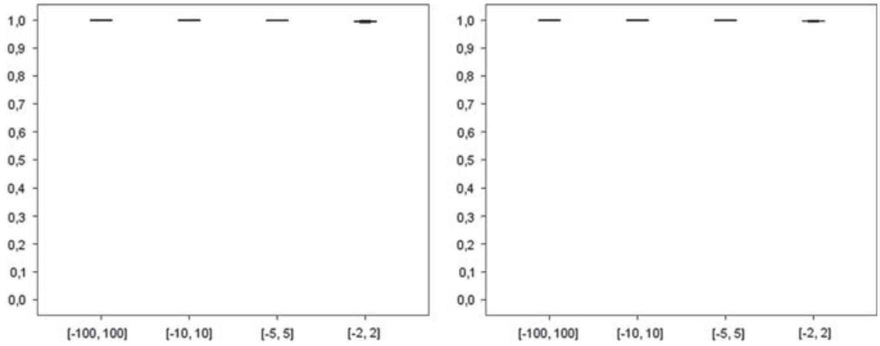


Fig. 3 $S(\widehat{\mathbf{p}}_k)$ with GME (left) and GCE (right) in Model 1 (18 variables)—A

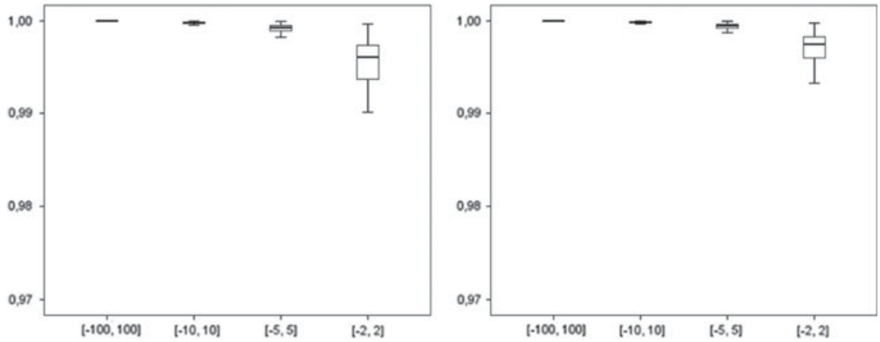


Fig. 4 $S(\widehat{\mathbf{p}}_k)$ with GME (left) and GCE (right) in Model 1 (18 variables)—B

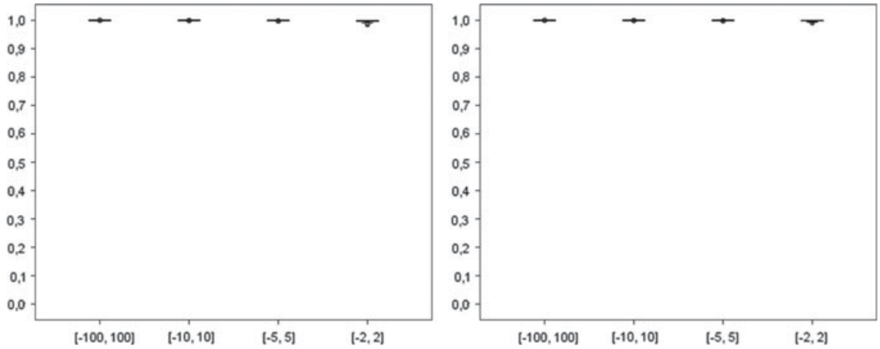


Fig. 5 $S(\widehat{\mathbf{p}}_k)$ with GME (left) and GCE (right) in Model 2 (50 variables)—A

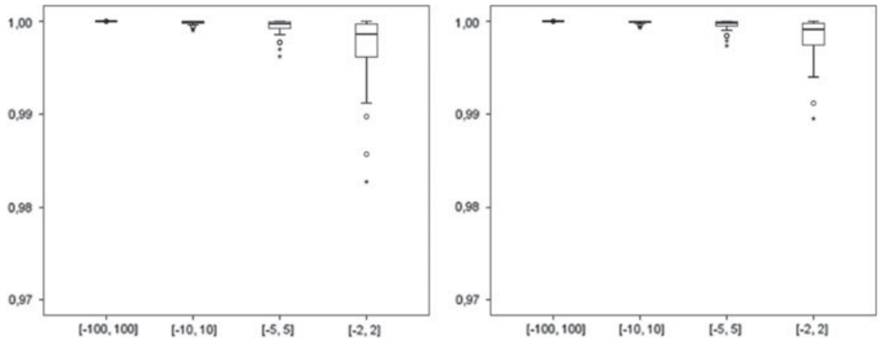


Fig. 6 $S(\widehat{\mathbf{p}}_k)$ with GME (left) and GCE (right) in Model 2 (50 variables)—B

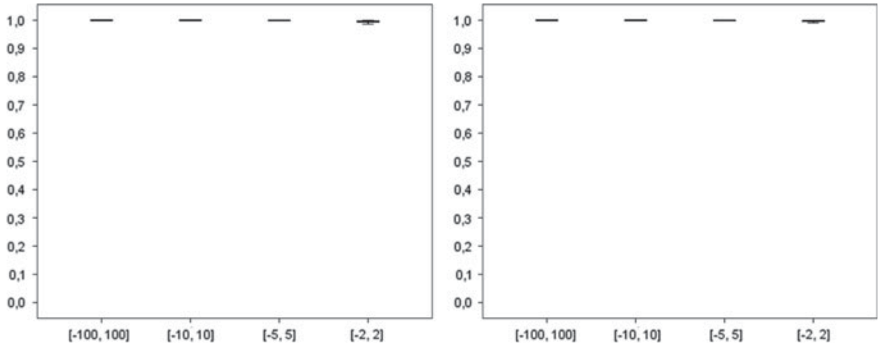


Fig. 7 $S(\widehat{\mathbf{p}}_k)$ with GME (left) and GCE (right) in Model 2 (14 variables)—A

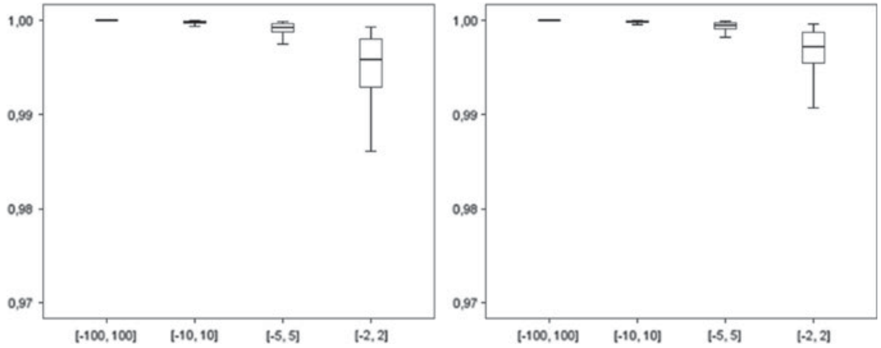


Fig. 8 $S(\widehat{\mathbf{p}}_k)$ with GME (left) and GCE (right) in Model 2 (14 variables)—B

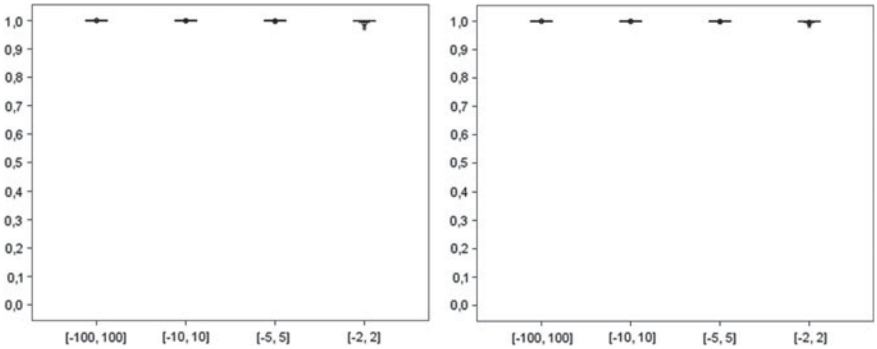


Fig. 9 $S(\widehat{\mathbf{p}}_k)$ with GME (left) and GCE (right) in Model 3 (50 variables)—A

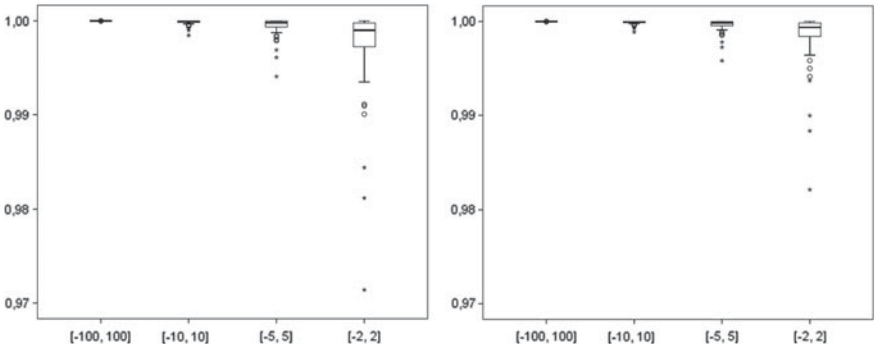


Fig. 10 $S(\widehat{\mathbf{p}}_k)$ with GME (left) and GCE (right) in Model 3 (50 variables)—B

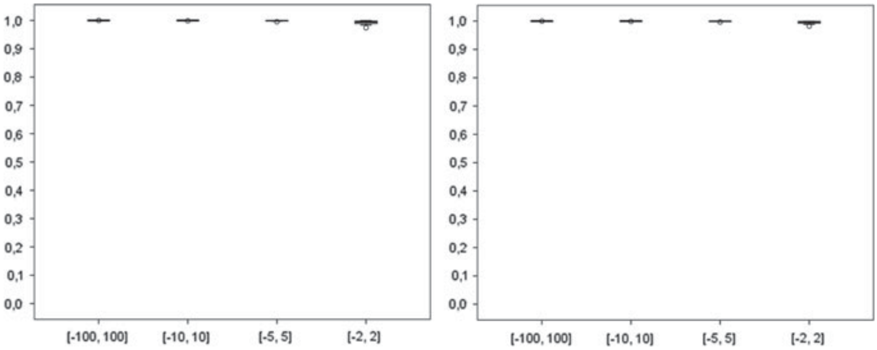


Fig. 11 $S(\widehat{\mathbf{p}}_k)$ with GME (left) and GCE (right) in Model 3 (14 variables)—A

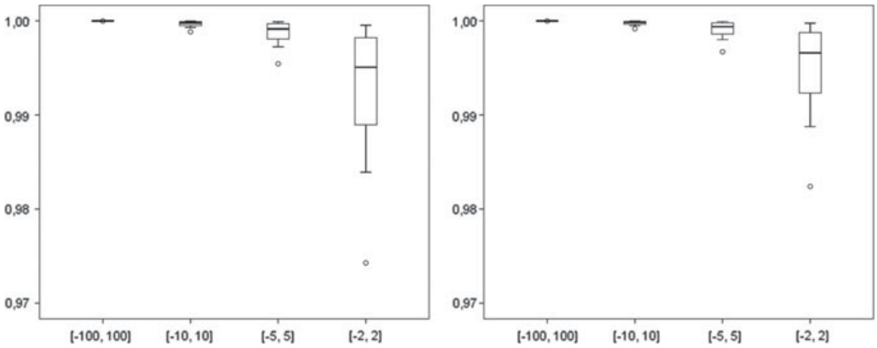


Fig. 12 $S(\widehat{\mathbf{p}}_k)$ with GME (left) and GCE (right) in Model 3 (14 variables)—B

The results in Table 3 suggest no information content of the signal in the models, regardless the supports considered or the maximum entropy estimator used. This is an interesting result because, under the conditions of the previous models, in the first stage, when $N \to \infty$ and $K \to \infty$, so that $K/N \to \rho$, where $0 < \rho < 1$, then the R^2, a standard procedure usually evaluated by practitioners, tends to ρ, and the ratio of the number of relevant variables by N tends to $\alpha\rho$, where α represents the significance level considered; see Freedman [2].

Taking into account that a variable corresponding to $S(\widehat{\boldsymbol{p}}_k) \approx 1$ has no information content (it is considered irrelevant) and should be removed from the model, the analysis of Figs. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 suggests the exclusion of all the variables, in both stages. Although in some scenarios, namely, in the one with the support defined as $[-2, 2]$, lower normalized entropy values are obtained, almost all of them are greater than 0.98.[4] Indeed, if the criterion of inclusion considered by Golan et al. [5, p. 165] is applied, $S(\widehat{\boldsymbol{p}}_k) \leq 0.99$, a few variables are considered relevant when the support $[-2, 2]$ is used, although the number of incorrect inclusions is lower when the GCE estimator is applied, as expected given the prior information considered. As mentioned previously, the priors take over as the solution when they are consistent with the data.

Naturally, without a formal rule to define a cutoff value, the identification of "relevant" variables (with "relevant" information content) can be considered difficult in the cases with normalized entropy values "near" one. Nevertheless, regarding this possible concern, is it really necessary a cutoff value? Is it not sufficient the evaluation of the information embodied in the normalized entropy? Possible answers to these open questions should always take into account, although in a different perspective, the theoretical discussions provided by Hurlbert et al. [7] and Wasserstein and Lazar [15], where some recommendations to statisticians are provided,

[4] There are only two exceptions (0.13% of the total), with values 0.971 and 0.974, approximately.

namely, to eliminate the choice of specific significance levels or to abolish the use of the terms "statistically significant," when p-values are interpreted in hypothesis testing.[5]

4 Conclusions and Future Research

The results in this work suggest that the evaluation of normalized entropy is a promising approach to avoid the disturbing effect in regression analysis described by the Freedman's paradox. Future research on the definition of the supports and in the amount of pressure around zero, established by the prior information vector for the GCE estimator, should be accomplished, along with the comparison with recent methodologies (e.g., lasso and its generalizations). Future research should also include an optimization procedure to cope with large-scale data, using the conditional maximum entropy formulation proposed by Mittelhammer et al. [12]. As a final remark, a MATLAB code to compute normalized entropy using the GME estimator can be easily obtained from the code available in Macedo [10]; see Appendix.

Acknowledgements This work is supported by The Center for Research and Development in Mathematics and Applications (CIDMA) through the Portuguese Foundation for Science and Technology (FCT—Fundação para a Ciência e a Tecnologia), references UIDB/04106/2020 and UIDP/04106/2020.

Appendix: MATLAB code

To adapt the code available in Macedo [10], the first line of the original code can be replaced, for example, by

```
function [b3,nep,nepk]=nentropy(Y,X)
```

Suppose now a model, for example, with $K = 6$ and consider all the supports in $\boldsymbol{Z}$, for example, as $[-10, 10]$. Lines 38–71 are replaced by

```
intg=[-10,10;-10,10;-10,10;-10,10;-10,10;-10,10];
```

Lines 116–132 are replaced by

[5] It is important to note that s-values can be much more useful than p-values; e.g., Greenland [6]. Information measures based on Shannon's work [13] are very attractive in statistical inference.

```
p=a(1:dp)';
b3=Z*p;
nep=(-p'*log(p))/(k*log(m));
nepk=zeros(k,1);
for i=1:k
    pos=(i-1)*m+1;
    nepk(i,1)=-p(pos:pos+m-1)'*log(p(pos:pos+m-1))/log(m);
end
```

All lines with comments and features related to the original code should be eliminated. Other changes can be made; e.g., the number of points in the supports, which are 5 and 3, by default; lines 74 (`m=5`) and 86 (`j=3`).[6]

References

1. Conceição Costa, M., Macedo, P.: Normalized entropy aggregation for inhomogeneous large-scale data. In: Valenzuela, O., Rojas, F., Pomares, H., Rojas, I. (eds) Theory and Applications of Time Series Analysis. ITISE 2018. Contributions to Statistics, pp. 19–29. Springer, Cham (2019)
2. Freedman, D.A.: A note on screening regression equations. Am. Stat. **37**(2), 152–155 (1983)
3. Golan, A.: A simultaneous estimation and variable selection rule. J. Econ. **101**, 165–193 (2001)
4. Golan, A.: Foundations of Info-Metrics: Modeling, Inference, and Imperfect Information. Oxford University Press, New York (2018)
5. Golan, A., Judge, G., Miller, D.: Maximum Entropy Econometrics: Robust Estimation with Limited Data. Wiley, Chichester (1996)
6. Greenland, S.: Valid P-values behave exactly as they should: some misleading criticisms of P-values and their resolution with S-values. Am. Stat. **73**(S1), 106–114 (2019)
7. Hurlbert, S.H., Levine, R.A., Utts, J.: Coup de Grâce for a tough old bull: "Statistically Significant" expires. Am. Stat. **73**(1), 352–357 (2019)
8. Jaynes, E.T.: Information theory and statistical mechanics. Phys. Rev. **106**(4), 620–630 (1957)
9. Jaynes, E.T.: Information theory and statistical mechanics. II. Phys. Rev. **108**(2), 171–190 (1957)
10. Macedo, P.: Ridge regression and generalized maximum entropy: an improved version of the Ridge-GME parameter estimator. Comm. Stat. Simul. Comput. **46**(5), 3527–3539 (2017)
11. Macedo, P.: Freedman's paradox: an info-metrics perspective. In: Proceedings of ITISE 2019, pp. 665–676. Godel Impresiones Digitales S.L., Granada (2019)
12. Mittelhammer, R., Cardell, N.S., Marsh, T.L.: The data-constrained generalized maximum entropy estimator of the GLM: asymptotic theory and inference. Entropy **15**, 1756–1775 (2013)
13. Shannon, C.E.: A mathematical theory of communication. Bell Syst. Tech. J. **27**(3), 379–423 (1948)
14. Soofi, E.S.: Capturing the intangible concept of information. J. Am. Stat. Assoc. **89**(428), 1243–1254 (1994)
15. Wasserstein, R.L., Lazar, N.A.: The ASA's statement on p-values: context, process, and purpose. Am. Stat. **70**(2), 129–133 (2016)

[6] The author accepts no responsibility for damages resulting from the use of the code. The code is provided as it is, and its users assume all the responsability when using it. There is no warranty of any kind.

Mining News Data for the Measurement and Prediction of Inflation Expectations

Diana Gabrielyan, Jaan Masso, and Lenno Uusküla

Abstract In this chapter, we use high frequency multidimensional textual news data and propose an index of inflation news. We utilize the power of text mining and its ability to convert large collections of text from unstructured to structured form for in-depth quantitative analysis of online news data. The significant relationship between the household's inflation expectations and news topics is documented and the forecasting performance of news-based indices is evaluated for different horizons and model variations. Results suggest that with optimal number of topics a machine learning model is able to forecast the inflation expectations with greater accuracy than the simple autoregressive models. Additional results from forecasting headline inflation indicate that the overall forecasting accuracy is at a good level. Findings in this chapter support the view in the literature that the news is good indicators of inflation and are able to capture inflation expectations well.

Keywords Inflation · Inflation expectations · News data · Machine learning · Text mining · Topic modelling

1 Introduction

Household surveys of inflation often indicate that the perception of the current inflation differs substantially from the actual values of inflation. Similarly, expectations about the future expectations differ strongly from the surveys of professional forecasters and the implied inflation rates of financial markets (for evidence see, e.g.

D. Gabrielyan (✉) · J. Masso
University of Tartu, Tartu, Estonia
e-mail: diana.gabrielyan@ut.ee

J. Masso
e-mail: jaan.masso@ut.ee

L. Uusküla
Bank of Estonia, Tallinn, Estonia
e-mail: lenno.uuskyla@eestipank.ee

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_17

Coibion et al. [1]). Potential reason for the difference is that households and firms obtain only very partial information while doing everyday purchases and aggregating the information is very costly. Imperfect information in turn affects adversely the formation of expectations.

Subjective inflation now casts and expectations are built through personal experiences, prior memories of inflation, and various other sources of information. One primary source of information is public media, and it is well established that consumers rely largely on it when thinking about overall price changes [2, 3]. Media covers a lot of news on prices and price developments.

In this chapter, we explore online news as a novel data source for capturing and measuring inflation perceptions by utilizing the power of text mining and its ability to convert large collections of text from unstructured to structured form. We propose a novel index of inflation news that provides a real-time indication of the price developments. Such index of inflation news captures and summarizes well the information used in the formation of expectations.[1] The delay in the publishing of official statistics, frequency of available survey-based inflation expectations and risks contained in the high frequency market-based forecasts[2] highlight the importance and need for such an indicator. Our main contribution is, therefore, using the novel source of information to prove that online news can provide a real-time and accurate indication of consumer's expectations on inflation.

Machine learning is considered to be a very promising avenue for academic and applied research. Although its applications are already actively used in many disciplines and research areas, it is still relatively new to economics. One modern strand of machine learning is text mining—the computational approach to processing and summarizing large amounts of text, which would be far more difficult to read, even impossible, for any single person. Extracting information from novel sources of data, such as social media (e.g. Twitter, Google) or public media (e.g. online news, communication reports) allows analysis and different kind of understanding of economic relationships (e.g. consumer behaviour), therefore, contributing to policymaking and forecasting. See for, example, Tuhkuri [5], D'Amuri and Marcucci [6], Yu et al. [7], Nyman et al. [8].

Another contribution of this work is to forecast the inflation in real-time using machine learning methods. The importance of inflation forecasting for rational decision-making is well established in the literature along with the common knowledge that improving upon simple models is quite challenging. According to Medeiros et al. [9], most of this literature, however, ignores the recent machine learning advances. In their work, they show that with machine leaning and data-rich models

[1] As Nimark and Pitschner [4] note, since no agent has resources to monitor all events potentially relevant for his decisions, news is preferred delegates for information choice to monitor the world on their behalf. And since news mainly reports selection of events, typically major ones, coverage becomes more homogenous across different outlets.

[2] Market-based expectations are available daily but include risk premia. Survey-based expectations are published monthly. For example, for the United Kingdom, the quarterly Consumer trends data are typically published around 90 days after the end of the quarter. See https://www.ons.gov.uk/economy/nationalaccounts/satelliteaccounts/bulletins/consumertrends/apriltojune2019.

improving inflation forecasts is possible. Their LASSO and Random Forest models are able to produce more accurate forecasts than the standard benchmark models, e.g. autoregressive models. Similarly, Garcia et al. [10] find that high dimensional models perform very well in inflation forecasting in data-rich environments. Our findings from LASSO regressions support these findings: for inflation expectations, the short-term forecast errors are smaller than those of the autoregressive models. The analysis also identifies the optimal number of news topics for predicting up to five quarters ahead inflation expectations to be either four or five, thus suggesting that the LASSO regression using optimal number of topics and best value of regularization parameter results in simpler model, which doesn't compromise the model performance. These results are, however, not robust for longer forecasting horizons and for different values of the regularization parameter. In additional results, when forecasting headline inflation, we find that the LASSO models fail to improve upon the benchmark models but demonstrate similar forecasting accuracy.

The rest of the chapter is organized as follows. Section 2 describes the data sources and methodology. Sections 3 and 4 provide results and an application in forecasting respectively. Section 5 concludes.

2 Methodology

2.1 Data

The process of building the inflation expectations index can be divided into data collection part and analysis of the data. This section describes both the data collection and its analysis by means of text mining. Our inflation news indicator is built from the article data of one of the UK leading newspapers,[3] Guardian, business section over the last 15 years. The choice of the news outlet is due relevance to our research in terms of content and readership, as well as the availability of open-source data. In May 2013, Guardian was the most popular UK newspaper website with 8.2 million unique visitors per month and in April 2011, it was the fifth most popular newspaper in the world.[4]

Any news in Guardian is public and readable by anyone by default. The Guardian API is a public web service for accessing all the content the Guardian creates, categorized by tags and sections. Users can query content database for articles with full content by tags and sections. The data comes in unstructured form, that is, the data is in a text form and does not have a given structure. Overall, we collected around 20,000 documents and 32 million terms from January 2004 to January 2019, which is sufficient amount of data to conduct our analysis. We only fetch articles from the business section, since this is the most relevant section for economic topics in general. In

[3] See https://www.pressgazette.co.uk/uk-newspaper-and-website-readership-2018-pamco/. In addition, see https://pamco.co.uk/pamco-data/latest-results/ for comparison among UK newspapers.

[4] Guardian.co.uk most read newspaper site in UK in March. www.journalism.co.uk.

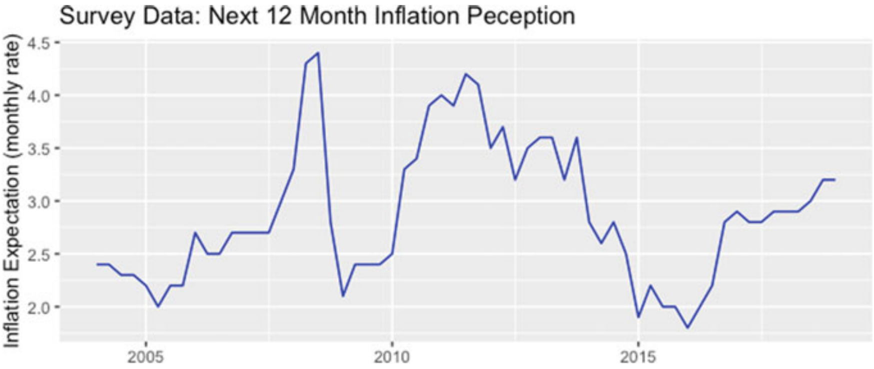


Fig. 1 Next 12-month inflation expectations, quarterly, growth from the previous period

addition, articles were also filtered based on subjectively chosen keywords, which in our opinion are relevant to inflation expectations topic. Namely, they are price, price increase, expensive, cheaper, cost, expense, bill, payment, oil, petrol, gas, diesel.

In addition to the novel data source, we also use the official inflation expectations data, which is taken from the Bank of England Inflation Attitude Surveys and reflects public's attitude towards the inflation for the next 12 months.[5] Figure 1 plots the results of this survey, that is, the quarterly inflation expectations for the UK from 2014 until 2019. Augmented Dickey–Fuller test is used to determine the presence of unit root and hence to understand whether the series are stationary or not. As such, we find that inflation expectations data is non-stationary and is transformed to stationary by first differencing.

Lastly, for robustness analysis, we also use actual inflation statistics from the UK National Office for Statistics, which reflect the Consumer Price Index including owner occupiers' housing costs (CPIH). In the analysis, we use both the annual 12-month rate, as well as the quarterly 3-month rate.

2.2 Text Pre-processing

We start with pre-processing, which a set of activities performed on the corpus. This way, the unstructured text is transformed into structured form, the dimensionality of the data is reduced, noise is eliminated, and we get more understandable data. Below are the text mining steps applied in this analysis. We follow text mining's bag-of-words[6] approach, which means all words are analyzed as a single token and

[5] Survey respondents were asked the question 'Q.1 Which of these options best describes how prices have changed over the last 12 months'? and their results of inflation attitude were summarized by the median response.

[6] In text mining, vector representations of text are called bag-of-words representations.

their structure, grammar or order is not used in the analysis.[7] We mostly follow suggestions for pre-processing by Bholat et al. [12], at the same time adding more steps and more advanced methods.

First step when extracting the data from news database is to remove any images and links contained in the articles and convert any information contained in the article into an appropriate format. Duplicate and empty entries should also be accounted for and such documents are removed. This can be done either manually or using Echkely [13] method. In our analysis, we use R language's built-in commands for duplicate and empty documents removal. We then break down the document into tokens, that is, we split the documents into words, numbers, symbols, etc. This is called tokenization and is done by using blank spaces or punctuation marks as delimiters. Next, all the words are converted into lower case and punctuation is removed. This is an important step, otherwise same words, such as Price and price, which are written in upper and lowercase, respectively, will be interpreted as different words. The downside is, however, because when written in uppercase, Price may refer to an individual with the last name Price and the lowercase may refer to the cost of something. We assume, though, that across all articles and words, it is more likely that the article's message is about prices as costs, rather than a person with last name Price.

Next crucial step is removing the stop words, otherwise these words will appear in the frequently used words and will give incorrect picture of the core meaning of the document. The list of these words is provided in the beginning of the analysis and includes common words in the English language that do not contain any information relating to the article. Examples of such words are the, like, can, I, also, are, in, on, this, that, gmt, pm, etc. To reduce the dimensionality further, we use word stemming, which involves cutting off affixes and suffixes and reducing all words to their respective word stems. This is a form of linguistic normalization, where part-of-speech of each word is identified, and each word is converted into its base form, e.g. nouns, verbs, pronouns with same base into base word (e.g. reporting, reported and reporter will be reduced to report).

The last step of the pre-processing is defining the Document-Term Matrix (DTM) based on the cleaned text and computing the most common words across all the documents. DTM lists all occurrences of words in the corpus by document. In the DTM, the documents are represented by rows and the terms (or words) by columns. This step also includes calculation of Term Frequency Inverse Document Frequency or Term Document Frequency (TDM), which allows reducing the dimension of DTM by removing all words which are less frequent, since the TDM measures how important are all the words in the full corpus in explaining single articles by assigning scores to each word. We, therefore, remove the sparse terms, i.e. terms occurring only in very few documents. These are the tokens which are missing from more than 90% of the documents in the corpus.[8] The remaining 900 000 stems with the highest TDM

[7] Comparison between bag-of word approach and other techniques is given in Cambria and White [11].

[8] Maximal allowed sparsity is in the range from 0 to 1. For this paper, the sparsity was chosen equal to 0.9, which means the token must appear in at least 10% of the documents to be retained.

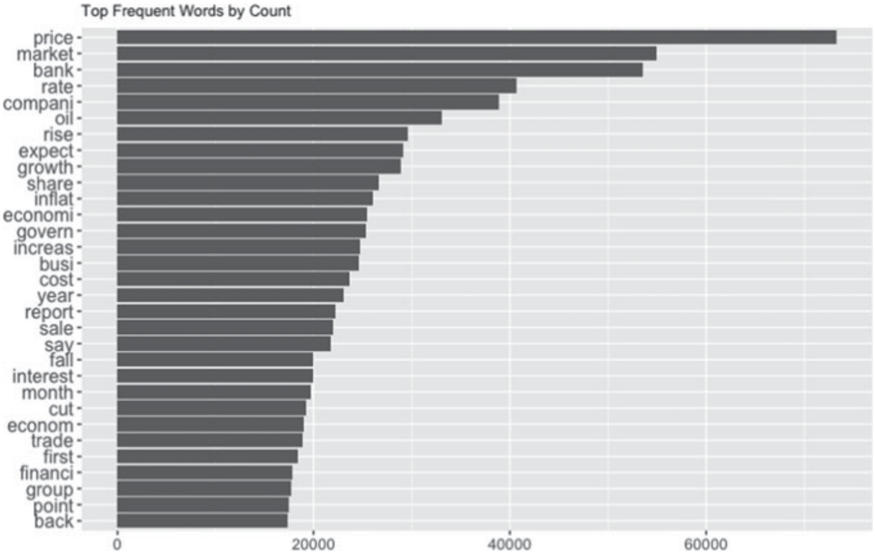


Fig. 2 Top frequent words and their counts. The words are presented in stemmed form

score are used in the final analysis. Frequency counts of the top 31 words in their stemmed form, that is the number of times those words appear in the final sample, are plotted in Fig. 2.

2.3 Topic Modelling

The pre-processing results in a data frame which consists of the words used in the text and their frequencies. These words consist of a document-term matrix, where each row of the matrix is a unique term and each column is a unique document. To proceed to building the index, topics need to be extracted from the DTM. Topic modelling is the statistical approach for discovering topics from the collection of text documents. In other words, it is the process of looking into a large collection of documents and identifying clusters of words based on similarity, patterns and multitude. Since any document can be assigned to several topics at a time, the probability distribution across topics for each document is, therefore, needed.

Latent Dirichlet Allocation (LDA)[9] is a statistical model that identifies each document as a mixture of topics (related to multiple topics) and attributes each word to one of the document's topics, therefore, clustering words into topics. With LDA method, it is possible to derive their probability distribution by assigning probabilities to each

The sparsity value can be modified to higher or lower value, but that affects the number of terms remained in the corpus.

[9] Detailed description of the LDA approach is provided in Blei et al. [14].

word and document. Assigning words and documents to multiple topics also has the advantage of semantic flexibility (ex. the word ‘rate’ can relate both to inflation and unemployment topic). The term ‘latent’ is used because the words are intended to communicate latent structure, the topic of the article, while the Dirichlet term is used because the topic mixture is drawn from a conjugate Dirichlet prior in order to ensure sparsity in the underlying multinomial distribution. Thorsrud [15] notes that LDA shares many features with Gaussian factor models, with the difference being that factors here are topics and are fed through a multinomial likelihood. In LDA, each document is given a probability distribution and for each word in each document, a topic assignment is made. The joint distribution of topic mixture θ, a set of N words w is given by

$$p(\theta, \mathrm{z}, \mathrm{w}|\alpha\,, \beta) \;=\; p(\theta|\alpha\,) * \prod\nolimits_{n=1}^{N} p(z_n|\theta\,) * p(w_n|z_n, \alpha\,) \tag{1}$$

where parameters α and β are k-vectors with components greater than zero, with k being the dimensionality of Dirichlet distribution, that is the directionality of topic variable z. In addition, the topic distribution of each document is as θ ~ *Dirichlet* (α) and term distribution is modelled by z_n~ *Dirichlet* (β), while N ~ *Poisson* (ξ).

LDA model’s goal is, therefore, to estimate θ and φ in order to define which words are important for which topic and which topics are important for a given document. For α and β, the higher they are, the more likely each document will contain a mixture of several topics instead of a single topic and the more likely each topic will contain a mixture of several of the words and not just single words. More technical specifications on the LDA model and topic modelling in general in Blei [16] and Griffiths and Steyvers [17]. In our research, we used LDA model with Gibbs sampling. To choose the number of topics, that provide the best statistical decomposition of the Guardian corpus, we use maximum likelihood method and find the model with the best score. To note, different model iterations and different parameters for α and β result in different document clustering. The goal is finding unknown patterns, therefore, there is no perfect value for topic number and the solution will most likely differ for different values. The best solution is to try different values of topics to find the optimal topic distribution across the documents that will match our intuition. As such, we classified 50 topics.

One other characteristic of LDA procedure is that it does not assign names to the topics. We do so ourselves, based on the most frequent words computed for the given topic and based on our subjective understanding of the topics, the economy and perceived economic relationships among them. Exact name, however, plays minor role in the actual analysis or results. LDA results in a vector indicating the distribution of topics in each document and most popular/relevant words within them. The topics resulting from LDA modelling along with the top 10 frequent words are provided in Appendix 1.

3 Results

For each document within a day, five most popular words are identified, and their daily frequency is calculated. This allows counting also the frequency of each topic for a given day. At this step, our results of topic decompositions and distribution are used to build the new high-frequency index that will capture the intensity of inflation expectations. The index is built for every day, that is, we build daily time series using Guardian's business articles for each day. To do so, we first collect together all articles for a given day into one document, grouping them into one plain text for each day. Next, based on the first ten most frequent words in each topic the article's daily frequency is calculated. In other words, the frequency is calculated for the given day as the raw count of frequencies with which the most common words in each topic appear on that day. For example, to understand the intensity of how many times the word 'vote' has been used on 22 June 2016 (the day of the Brexit vote), we will aggregate all the Guardian articles for that day as one big text document, then calculate the number of times the word 'vote' appears in the text. Here the Brexit can be our topic and the 'vote' is the term.

The news volume $I(t)$ of given topic z is given by

$$I_z(\mathrm{t}) = \sum\nolimits_{d \in I(t)} \sum\nolimits_{w} N(d, w, z), \tag{2}$$

where $N(d, w, z)$ is the frequency with which the word w tagged with topic z appears in document d. These time series $I_z(\mathrm{t})$ are measures of volume, that is, they measure the intensity of given topic for given time period, that is, for given day. Figure 3

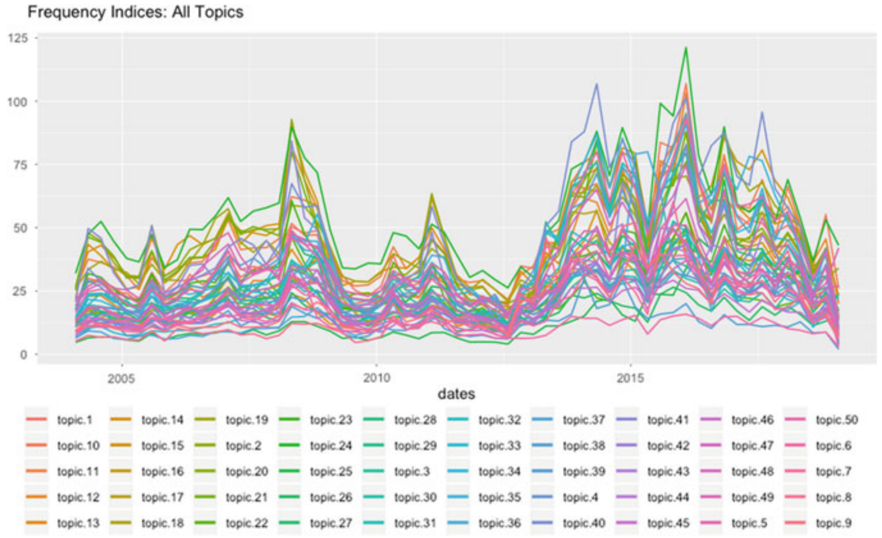


Fig. 3 Frequency time evolution for all topics

illustrates our main results, that is, the 50 news topics identified from the Guardian news dataset and their frequencies over the period from 2004 to 2019.

We find that some of index series are non-stationary and consequently transform them to stationary series by differencing. Augmented Dickey–Fuller test is used to determine the presence of unit root and hence understand if the series are stationary or not. As such, some of the indices are evaluated as non-stationary and are transformed to by differencing.

4 Application in Forecasting

The first task is to filter information from the list of variables and select more relevant components. It is highly inefficient to use all the topic indices for predicting in such a rich dataset, as some of the regressors may be imparting redundant information. Therefore, number of topics N is too high and there is a definite multicollinearity present among the topic indices, as can also be observed from Fig. 3. To reduce dimensionality and tackle the issue of multicollinearity,[10] we use another machine learning method for variable selection. Least Absolute Shrinkage and Selection Operator (LASSO) method automates variable selection by reducing the coefficients of some features to zero, while keeping those that have the most impact on the dependent variable. LASSO's main goal is finding β that minimizes (3) with constraint $\sum_{j=1}^{p} |\beta_i| \leqslant t$.

$$\sum\nolimits_{i=1}^{N} \left(\pi_{t}^{t+h} - \sum\nolimits_{j=1}^{p} \beta_i\, x_{ij}\right)^2 + \lambda \sum\nolimits_{j=1}^{p} |\beta_i| \tag{3}$$

π_t^{t+h} is the inflation (expectations) for the next h quarter, N is the simple linear mapping of p indices built using (2) and $x_{i,j}$ are the lagged indices built from the news data. λ is the shrinkage parameter and controls the strength of penalty finding the model with the smallest number of predictors that also gives a good accuracy. Therefore, the number of variables to be removed is decided by the shrinkage parameter λ, which is chosen using cross-validation. Once the topic indices are selected, we forecast the inflation expectations by building a model using a direct forecast approach as given below

$$\pi_{t}^{t+h} = \alpha + \mathrm{a} * \pi_{t-1}^{t-1+h-1} + \sum\nolimits_{n=1}^{N} b_n * x_{n,t-1} + u_t, \tag{4}$$

$\pi_{t-1}^{t-1+h-1}$ the lagged value for the same horizon as for the inflation expectations π_{t}^{t+h}. N is the number of indices built from news data, b_n are vectors of unknown parameters, $x_{n,t}$ are the lagged indices and u_t is the forecasting error. We call the Eq. (4) a News-Based Model (NBM). It is common practice to fit a model using training data, and then to evaluate its performance on a test data set. Forecast horizon

[10] LASSO is very robust against multicollinearity, see Friedman et al. [18].

h is also the length of the out-of-sample period (i.e. fitted values on the training set) and will be varied from 1 to 12 to compare the forecasts at different horizons and find the 'optimal' horizon defined by the lowest forecasting error. Since all of the data in this analysis is quarterly, *h* is measured in quarters. For benchmarking, we use naïve AR (1) model on inflation expectations and compare the Root Mean Squared Errors (RMSE).

Table 1 reports the normalized results of estimating (5) and an AR (2) with different forecast horizons relative to simple AR (1) model. The first column of the table shows the forecast horizon, the second column (*n_min*) shows the number of variables (topics) selected by LASSO regression and the last two columns show the Root Mean Squared Errors (RMSE) for each of the applied models. It can be seen that generally, the RMSEs are small, varying from 0.02 to 0.76, while the forecast errors are the lowest when forecasting the next one or two period expectations using the news data. In this case, the LASSO model outperforms both the naïve AR (1) and AR (2) forecasts in terms of accuracy.

Several interesting observations can be made from Table 1. Firstly, LASSO models select different number of topics that are relevant for inflation expectations prediction for different forecast horizons. Out of our fifty topics compiled by the LDA method, LASSO selects three to six topics depending on the forecast horizon. Lagged value of the inflation expectations is always included among selected regressors and is always significant. The adjusted R-squared statistic is informative and for some horizons is as high as 70%. Thus, the selected news topic, as well as the past values of inflation expectations, explain a relatively large fraction of the variation in the households' inflation expectations. One to two quarters ahead expectations can be forecasted with five topic indices as regressors, while the longer forecasts of eleven and twelve quarters can be forecasted with the best accuracy when only three relevant topics are employed in the regression. It can also be observed that the longer the forecast horizon, the lower the forecast accuracy, which is intuitive. Figure 4 visualizes the

Table 1 RMSEs of *h*-period inflation expectations forecasts using LASSO and AR (2) models. Errors are normalized relative to AR (1) benchmark

h	n_min	RMSE_LASSO_MIN	RMSE_AR2
1	5	0.6	6
2	5	0.7	1.9
3	6	0.9	1.8
4	5	0.8	1.8
5	5	0.8	1.8
6	5	1.0	1.6
7	5	1.1	1.7
8	5	1.0	2.1
9	5	1.2	2.9
10	4	1.6	1.9
11	3	1.3	1.8
12	3	1.5	1.8

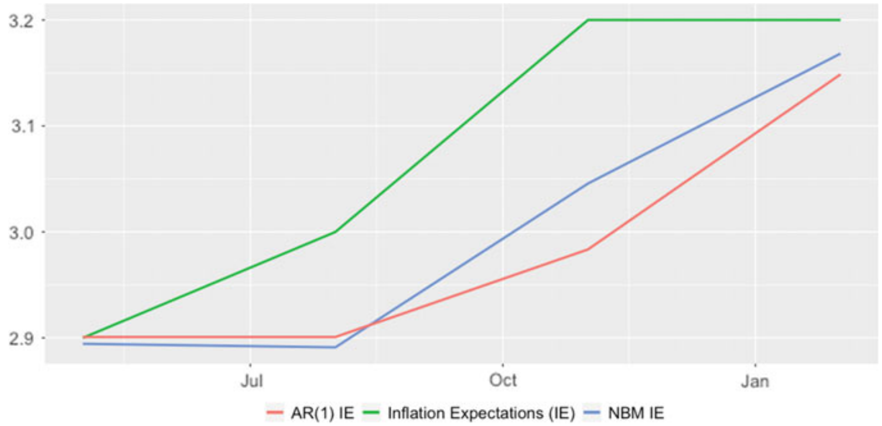


Fig. 4 One year ($h = 4$) ahead inflation expectations (IE) with the fitted values from LASSO (NBM IE) and AR (1)

results for one year ahead inflation expectations with the fitted values from NBN and AR (1).

Our results were not robust when controlling and comparing different values of regularization parameter in the LASSO regression. There are different ways to choose the optimal value of lambda by cross-validation. Our results, in Table 1, weere based on the smallest value of lambda from the cross-validation results. Table 2 compares the accuracy obtained with LASSO regression using different values of lambda shrinkage parameters against the benchmark autoregressive models. First column is the forecast horizon, while following three columns report the number of regressors selected by LASSO for different values of lambda. Among selected topics for all three variations of lambda, first lag of inflation expectations is selected. Column RMSE_LASSO_MIN uses the value of lambda that is equal to the minimum value of lambda chosen by cross-validation, while column RMSE_LASSO_LSE is based on the model where lambda is within one standard error. Column RMSE_LASSO_BIC is based on the lambda which is chosen using information criterion, while last two columns show the errors for benchmark AR (1) model AR (2) model. All errors are normalized relative to AR (1). Given the sparsity across normalized errors for different forecast horizons, as well as in the number of topics selected by LASSO, it can be noted that LASSO models other than that based on its minimum value are less accurate and fail to outperform the naïve models.

The model obtained from RMSE_LASSO_LSE includes less topics but shows poor forecasting performance. Similarly, the model from RMSE_LASSO_BIC includes even more predictors, particularly in the intermediate horizons, however, shows worse performance. In the shorter forecasting horizons, the number of chosen topics is four, which is closer to five from the minimum lambda model, and the forecast accuracy improves. These analyses demonstrate that the optimal number of topics to predict inflation expectations up to five quarters ahead is between four

Table 2 RMSEs of *h*-period inflation expectations forecasts using different values of lambda in LASSO model, as well as AR (1) and AR (2) models. All values are normalized relative to AR (1) benchmark

h	n_min	n_lse	n_bic	RMSE_LASSO_MIN	RMSE_LASSO_LSE	RMSE_LASSO_BIC	RMSE_AR1	RMSE_AR2
1	5	2	4	0.6	3.4	0.4	1	6
2	5	2	4	0.7	1.8	0.7	1	1.9
3	6	2	48	0.9	3.5	5.9	1	1.8
4	5	2	50	0.8	3.7	6.1	1	1.8
5	5	2	4	0.8	4.5	0.7	1	1.8
6	5	2	44	1.0	4.6	7.1	1	1.6
7	5	2	47	1.1	5.1	7.4	1	1.7
8	5	2	41	1.0	5	7.5	1	2.1
9	5	2	41	1.2	4.9	6.8	1	2.9
10	4	3	2	1.6	1.3	1.5	1	1.9
11	3	2	2	1.3	1.3	1.5	1	1.8
12	3	2	2	1.5	1.5	1.6	1	1.8

and five. This also suggests that the LASSO regression, using minimum lambda as the best lambda, results to simpler model without compromising much the model performance on the test data.

It is also of interest to look how the same news data and model can be used to predict the headline inflation. We computed forecast errors for different horizons and models compared to benchmark AR (1) for annual rate of inflation and its quarterly rate. Results, not included in this chapter, but available from authors upon request suggest that while the LASSO model built using pre-selected news topics does not outperform the benchmark models, it can, however, be used as a forecasting model with similar forecast accuracy as those naïve models. This means that the model obtained with LASSO regression does at least as good a job fitting the information in the data as the more complicated one.

5 Conclusions

In this chapter, we proposed a novel index of inflation news that provides a real-time indication of the price developments. Such index of inflation news captures and summarizes well the information used in the formation of expectations. We then document the significant relationship between the households' inflation expectations and news topics and evaluate the news' forecasting performance using out-of-sample validation. We use machine learning's LASSO model and different values of regularization parameters to find the optimal number of topics that provide the best accuracy for the inflation expectations forecasts. Our results suggest that when using the best value for lambda and the optimal number of topics, the LASSO models are able to forecast the inflation expectations with greater accuracy than the simple benchmark models, such as AR (1) and AR (2). However, the predictive relationship between the headline inflation and the news topics is not as strong. Yet, both for the quarterly and annual rates of actual inflation, we find similar forecasting accuracy as the benchmark models. The obtained accuracy remains good enough, and the LASSO regression does at least as good a job fitting the information in the data.

These findings are in accordance with our main hypothesis that the news is good indicators of inflation and inflation expectations and are able to capture them well. Our results also support the view of Medeiros et al. [9] and Garcia et al. 10, that high dimensional models have better forecasting power than the simple naïve models, hence confirming that our methodology results in a model with as good forecasting power as existing simple models, if not better. This also highlights the importance of media as an information source for households' expectation formation.

Appendix 1: The List of Topics with Their Most Frequent Words

Topic	Top frequent words							
Topic 1	peopl	say	world	work	money	year	famili	man
	mani	made	univers	live	person	use	life	school
Topic 2	price	rise	increas	cost	higher	high pressur	push	demand
	record	risen	warn	rais	month	soar	level	expect
Topic 3	execut	chief	compani	sharehold	chairman	director	manag	board
	group	former	sir	busi	boss	meet	year	head
Topic 4	custom	compani	servic	mobil	use	phone	busi	call
	charg	network	card	oper	offer maketr	vodafon	onlin	internet
Topic 5	problem	caus	day	system	safeti	report	work	damag
	respons	issu	affect	manag	failur	made	two	delay
Topic 6	airlin	passeng	cost	fuel	travel	flight	ryanair	airport
	oper	rail	air	train easyjet	transport	carrier	british	ticket
Topic 7	car	vehicl	sale	diesel	industri	emiss	manufactur	engin
	test	volkswagen	model	compani	ford	scandal	carmak	petrol
Topic 8	european	europ	germani	eurozon	countri	franc	german	spain
	ireland	french	euro	crisi	market	irish	govern	debt
Topic 9	govern	parti	elect	polit	minist	presid	bill	support
	leader	prime	labour	conserv	democrat	state	call	polic
Topic 10	market	stock	point	trade	investor	index	ftse	day
	close	wall	fall	share	street	hit	trader	fear
Topic 11	per	cent	say	term	much	look	like	even
	believ	see	current	think	long	less	come	around

(continued)

(continued)

Topic	Top frequent words						
Topic 12	china	world	global	countri	chines	trade market	japan	economi
	develop	steel	demand	commod	import	india	econom	emerg
Topic 13	energi	gas	price	custom	british	bill compani	market	supplier
	big	centrica	power wholesal	six	electr	suppli	ofgem competit	household
Topic 14	sector	manufactur	survey	servic	growth	show report	order	data
	busi	markit	output	month	activ	factori	firm	construct
Topic 15	inflat	price	rise	fall	consum	rate	figur	food
	pressur	index	cpi	economist	cost month	expect	annual	data
Topic 16	sale	retail	store	christma	shop	street	high	consum
	spend	cloth	chain	trade	onlin	shopper	good	decemb
Topic 17	london	citi	local	say	peopl	centr	year	build
	home	open	street	shop	place	around	area	busi
Topic 18	wage	unemploy	peopl	pay	rise	job employ	incom	labour
	work	household	live	earn	real	growth	averag	increas
Topic 19	profit	group	compani	expect	busi	result	quarter	first
	share	half	analyst	revenu	sale	chief	perform	execut
Topic 20	photograph	updat	say	morn	report	today	here	news
	market	bank	novemb	point	hit	show	mdash	data
Topic 21	hous	price	market	properti	home	mortgag	averag	buyer
	rise	rate	month	increase	nationwid	interest	number	estat
Topic 22	oil	compani	shell	gas	north	well	sea	explor
	drill	reserv	product	field	gulf	oper	spill	project
Topic 23	food	product	price	produc	british	farmer	brand	pub
	drink	use	year	good	include	industri	milk	wine

(continued)

(continued)

Topic	Top frequent words							
Topic 24	share	investor	invest	fund	valu	market	compani	stock
	price	sell	asset	manag	cash	capit	stake	buy
Topic 25	brexit	pound	trade	trump	vote	britain	sterl	relat
	june	leav	uncertainti	deal	busi	import	warn	dollar
Topic 26	investig	compani	fine	court	case	claim	alleg	legal
	charg	law	regul	action	author	inform	offic	rule
Topic 27	tesco	supermarket	price	store	sainsburi	morrison	retail	asda
	chain	sale	market	custom	discount	cut	food	shopper
Topic 28	job	worker	union	staff	cut	industri	work	plan
	strike	busi	loss	unit	employ	close	employe	compani
Topic 29	ecb	bank	rate	draghi	central	polici	bond	eurozon
	programm	euro	eas	cut monetari	market	inflat	mario	govern
Topic 30	pay	payment	cost	insur	bonus	paid	claim	receiv
	cash	total	annual	report	cover	salari	bill	fee
Topic 31	australia	cost	drug	note	relat	year	australian	author
	product	research	health	much	suggest	use	two	lower
Topic 32	tax	busi	corpor	compani	revenu	govern	pay	account
	avoid	bill	duti	rule	chang	benefit	invest	make
Topic 33	bank	financi	lloyd	barclay	credit	loan	capit	rbs
	loss	scotland	mortgag	hsbc taxpay	test	money	crisi	lender
Topic 34	economi	financi	crisi	recess	debt	econom	credit	warn
	world	year	global	fall	cut	even	mani	risk
Topic 35	growth	quarter	economi	gdp	figur	econom	consum	first
	spend	month	economist	three	recoveri	expect	second	rise

(continued)

(continued)

Topic	Top frequent words							
Topic 36	greec	greek	minist	debt	bailout	athen	meet	financ
	deal	govern	tsipra	june	talk	reform	creditor	februari
Topic 37	rate	market	fed	feder	reserv	interest	expect	rais
	hike	economi	rise	meet central	polici	yellen	job	point
Topic 38	russia	russian	gas	countri	suppli	Gazprom	ukrain	state
	pipelin	europ	western	control	import	world	moscow	secur
Topic 39	govern	pension	contract	scheme	servic	royal	public	mail
	busi	cost	compani	work	privat	provid	sector	project
Topic 40	rate	bank	interest	inflat	england	polici	monetari	economi
	committe	rise	mpc	cut	rais	king	governor	economist
Topic 41	oil	price	barrel	crude	opec	product	suppli	demand
	produc	saudi	brent	day	cut output	global	petrol	high
Topic 42	govern	budget	spend	public	chancellor	cut osborn	britain	georg
	treasuri	financ	brown	deficit	economi	econom	plan	fiscal
Topic 43	septemb	fall	month	august	juli	octob	drop	fell
	show	figur	expect	june	declin	record	report	continu
Topic 44	get	think	thing	even	look	dont	seem	happen
	know	want	might	good	big	that	way	realli
Topic 45	regul	commiss	propos	review	need	rule chang	competit	plan
	industri	allow	report	concern	system	author	use	want
Topic 46	energi	power	climat	invest	chang	renew	fuel	generat
	nuclear	plant	govern	carbon	use	project	build	electr
Topic 47	say	carney	mark	need	financi	England	question	governor
	think	independ	issu	committe	make	busi	speech	get

(continued)

(continued)

Topic	Top frequent words							
Topic 48	share	group	compani	ftse	target	analyst	mine	news
	point	close	posit	lower	price	buy	higher	fell
Topic 49	deal	compani	offer	bid	group	share	takeov	sharehold
	firm	stake	merger	talk	yesterday	two	acquisit	own
Topic 50	expect	forecast	growth	market	remain	continu	next	report
	risk	current	like	lower	see	predict	fall	level

References

1. Coibion, O., Gorodnichenko, Y., Kumar, S.: How do firms form their expectations? New survey evidence. Am. Econom. Rev. **108**(9), 2671–2713 (2018)
2. Blinder, A.S., Krueger, A.B.: What does the public know about economic policy, and how does it know it? Brook. Pap. Econ. Act. Econ. Stud. Program Brook. Inst. **35**(1): 327–397 (2004)
3. Curtin, R.: What U.S. consumers know about the economy: the impact of economic crisis on knowledge? In: Proceedings of the 3rd OECD World Forum on Statistics, Knowledge and Policy: Charting Progress, Building Visions, Improving Life: OECD, 27–30 Oct 2009, Busan, Korea
4. Nimark, K.P., Pitschner, S.: News media and delegated information choice. CEPR Discussion Papers 11323, C.E.P.R. Discussion Papers (2018)
5. Tuhkuri, J.: Forecasting unemployment with Google Searches. ETLA Working Papers No 35 (2016)
6. D'Amuri, F., Marcucci, J.: The predictive power of Google searches in forecasting US unemployment. Int. J. Forecast. **33**(4):801–816 (2017)
7. Yu, L., Zhao, Y., Tang, L., Yang, Z.: Online big data-driven oil consumption forecasting with Google trends. Int. J. Forecast. (2018)
8. Nyman, R., Gregory, D., Kapadia, S., Ormerod, P., Tuckett, D., Smith, R.: News and narratives in financial systems: exploiting big data for systemic risk assessment. Mimeo (2015)
9. Medeiros, M.C., Vasconcelos, G.F.R., Veiga, Á., Zilberman, E.: Forecasting inflation in a data-rich environment: the benefits of machine learning methods. J. Bus. Econ. Stat. (2019)
10. Garcia, M,. Medeiros, M.C., Vasconcelos, G.: Real-time Inflation forecasting with high-dimensional models: the case of Brazil. Int. J. Forecast. 33, 679–693 (2017)
11. Cambria, E., White, B.: Jumping NLP curves: a review of natural language processing research, proceedings of research. IEEE Comput. Intell. Mag. **9**, 48 (2014)
12. Bholat D.M., Hansen S., Santos P.M., Schonhardt-Bailey Sh.: Text mining for central banks. SSRN Electron. J. **33**, 1–19 (2015)
13. Eckley, P.: Measuring economic uncertainty using news-media textual data, MPRA Paper, No. 64874 (2015)
14. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**: 993–1022 (2003)
15. Thorsrud, L.A.: Words are the new numbers: a newsy coincident index of business cycles. J. Bus. Econ. Stat. (2018)
16. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet Allocation. J. Mach Learn. Res. **3**, 993–1022 (2003)
17. Griffiths T.L., Steyvers M.: Finding scientific topics. Proc National Acad. Sci. **101**(Supplement 1), 5228–5235 (2004)
18. Friedman, J., Hastie, T., Tibshirani, R.: The Elements of Statistical Learning, vol. 1. Springer Series in Statistics. Springer, Berlin (2001)

Big Data: Forecasting and Control for Tourism Demand

Miguel Ángel Ruiz Reina

Abstract In this study, innovative forecasting techniques and data sources from Big Data are used for the study of Hotel Overnight Stays for Spain, from January 2018 to June 2019. The unstoppable development of the Tourism sector with the application of Big Data technologies, allow to make efficient decisions by economic agents. In this work, the use of the data collected from the Google Data Mining tools allows to obtain knowledge about Hotel Tourism Demand in Spain. The analysis carried out meets the four basic principles of Big Data analysis: volume, velocity, variety and veracity. In this setting, the methodology used corresponds to ARDL models, and ECM models being developed Granger-Causality extended to seasonality. The first one explains easily when economic agents will make their decisions; while the second one allows forecasting for short-term and long-term. This fact means that tourist offers and demands can be perfectly adjusted at every moment of the year. As a criterion for the selection of models, the innovative Matrix U1 Theil is proposed, this allows to quantify how much a model is better than another in terms of forecasting.

Keywords Big data · Forecasting · Google trends

1 Introduction

The use of massive data in a digital environment has led to a disruptive change in the developed economies of the world. Before the appearance of the Big Data concept, the amount of data collected already exceeded the ability to process and analyze data. The generation of massive data by the millions of device users and data analysis have created an unsuspected digital economy decades ago [1].

The "Tourism Industry" [2], generates a quantity of data to be analyzed. This sector increasingly has a greater weight in the Gross Domestic Products (GDP) and turn generates externalities in economic agents [3].

M. Á. R. Reina (✉)
Department of Economic Theory and Economic History, PhD Program in Economics and Business, University of Málaga, S/N, Plaza Del Ejido, 29013 Málaga, Spain
e-mail: ruizreina@uma.es

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_18

This paper introduces a modern unexplored analysis of the data generated on the internet network for the Spanish tourism accommodation market by country of origin. Innovative modelling of data processing from primary data sources (official sources) with secondary sources from Big Data (Google Trends—GT) is introduced following four basic principles of analysis: volume, velocity, variety and veracity. GT analyzes the shift of searches throughout the time and reveal consumer intentions.

The main objective of this paper is to obtain forecasting on Hotel Overnight Demand in Spain (HODS) from January 2018 to June 2019, by establishing a causality model for monthly data. The multivariate method developed of Autoregressive Distributed Lags with seasonal variables (ARDL + seasonality) uses as an explanatory variable for HODS a search interest rate (generated by GT) and seasonal dummies variables for monthly data by country of origin. This second contribution is a very relevant fact since tourism agents will be able to make efficient decisions in the tourism market. To explain causation relations, the Granger-Causality test extended with seasonality is developed and modelling we will be able to identify when consumer interest occurs. Ultimately, a criterion for the selection of new models, such as Matrix U1 Theil, has been developed, and it will be applied in this paper [4]. The forecasting is compared with univariate techniques such as Seasonal Autoregressive Moving Average (SARIMA) and the relatively new non-parametric technique Singular Spectrum Analysis (SSA).

The remainder of this research is as follows: Sect. 2 provides a review of the existing literature on the forecasting of Tourism Demand, influenced by the techniques of every epoch; in Sects. 3 and 4, data analysis is initially carried out along with the methodological development and information criteria. The use of the criterion for the selection of predictive models based on Theil's index is considered a great contribution to the literature. In Sect. 5 an empirical analysis is carried out verifying the application of the proposed methodology. Section 6 shows the conclusions and future lines of research for Data Scientists and some economic implications. Finally, there is a section for the bibliographical references used.

2 Literature Review

Data science is a fundamental field for the exploitation and generation of knowledge to make decisions in efficiency. In the bibliographic research carried out the appearance of these new datasets from open data such as Google could modify the culture and business in the Tourism field [5].

Tourism Demand is caused by multiple exogenous factors and techniques have focused on obtaining robustness and dynamic modelling, scalability and granularity [6]. The variety of Big Data studies has been applied to Tourism research, making a great improvement in the area [7]. Traditionally these studies have been influenced by the techniques of the moment [8–11]. However, researchers have found the need for greater integration between computational and scientific fields [12].

In our study, we will carry out an analysis with novel techniques and will be compared with most used techniques, a contribution of this study is the use of Big Data [13], tools summarized in an index of relevance provided by GT.

2.1 Forecasting Methods Using Google Search Engines (Google Trends)

Previous researchers such as Lu and Liu [14], found correlations between Internet search behaviour and the flows produced by tourists. Shimshoni et al. [15] concluded that 90% of the categories analyzed are predictable, making a great contribution to the scientific literature (categories: Socio-Economics fields).

Using the R programming and developing several examples in which the GT tool is used, it is worth mentioning the study of Choi and Varian [16], to analyze the tourism demand in Hong Kong. They obtained models with high explanatory capacity (on average $R^2 = 73\%$) using ARDL. Gawlik et al. [17] concluded that the GT search popularity evolution offers a useful predictor of tourism rates for a series of arrivals of Hong Kong. For the Charleston region (USA), practical and interesting applications were found on the use of search engine data. The main limitation is that it was done only in one city [18].

To carry out Chinese Tourists' forecasting, Yang, Pan et al. [19], proposed and demonstrated the valence of the use of search engines based on web searches comparing Baidu search engines with those of GT. In this sense, with data obtained through GT, comparing purely autoregressive models with ARDL models with seasonal dummy variables, short-term results were obtained for the case of Vienna with data from images, words search or videos on YouTube [20].

Studies from the use of GT have meant an improvement in predictions for the Caribbean area. Autoregressive Mixed-Data Sampling models represent an improvement over SARIMA (Seasonal Autoregressive Integrated Moving Average) and AR for 12-months predictions [21].

The study of the tourist flows from Japan to South Korea has been examined with the construction of the Google variable combining the lowest Mean Square Error (MSE) or the absolute average of forecast errors for monthly data. Finding the best results for the model that uses Google data [22].

In the case of tourist flows from Spain, Germany, UK and France, Google data was used with the construction of indicators through Dynamic and SARIMA models [23]. For tourist arrivals in the city of Vienna [24], Google Analytics data was extracted using Bayesian methods. In the case of Puerto Rico, the volume of searches has been studied to predict the hotel demand of non-residents with a Dynamic Linear Model. The results showed improvements in forecasting time horizons greater than 6 months [25]. Google data has been used for the flow of tourists in Portugal [26] and tourists flow in Spain [27].

Irem Önder [28] compared forecasting models with web and/or image search indices regarding two cities (Vienna and Barcelona) and two countries (Austria and Belgium). Tourist Arrivals in Prague was analyzed by Zeynalov [29], with the objective to assess whether GT were useful for forecasting tourists' arrivals and overnight stays in Prague with weekly data. The results confirm that predictions based on Google searches are advantageous for policymakers and businesses operating in the Tourism sector.

The online behaviour of hotel consumers for the United States of America was researched with Discrete Fourier Transformation using data from GT, with empirical evidence for its use in marketing strategies [30].

In the case of Amsterdam, it has been investigated by Rödel [31], on forecasting Tourism Demand using keywords related to "Amsterdam" in GT. With the development of Big Data technology in the last decades have emerged collaborative economy companies [32]. They have carried out studies on a vacation rental company that operates worldwide but reducing it to results from the Iberian Peninsula. In 2018, a study was published on the online and offline behaviour of consumers, for US restaurants with Google and Baidu search engine data. [33].

The data provided by Google use an index that summarizes the interest of the search words, in the case of data from Baidu. Li et al. [34], developed an index of interest with data from Baidu. Demonstrating the forecasting capacity of Dynamic Factor Model (GDFM) to forecast tourist demand in a destination for Monthly Beijing tourist volumes from January 2011 to July 2015. A relevant study using Machine Learning algorithms is the one developed by Sun et al. [35], using criteria for the selection of models such as Normalized Root Squared Error (NRMSE) and MAPE, in addition to using the Diebold-Mariano criterion to determine if the prediction differences are significant.

Measures of forecasting. As observed above, the Tourist Industry has had an interest in the past, in the present and in the future, and it will continue to have it. Mainly because it is an industry signal of the evolution of the service economy. So, the modelling used is very diverse, one aspect to be taken into account has been the criteria of information on the selection of models. It has been observed in the literature review the use of Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE); Theil's index [36–39]; Symmetric Mean Percentage Error (SMAPE) [40]. Some authors developed the RMSE ratio [41, 42], and in this article, we will develop the Matrix U1 Theil as a criterion for the selection of forecasting models [4]. This method allows quantifying the gain of the use of one methodology versus another.

To summarize the review of the literature, we can say that new models have been used in Data Science. In this work, new methodologies are developed, such as the improved Ganger causality test for seasonal data. Dynamic models have been developed to analyze the forecasting capacity in the short and long-term. Big Data tools have been used from one of the largest search engines in the world and a decision matrix on predictive capacity has been developed for different time horizons.

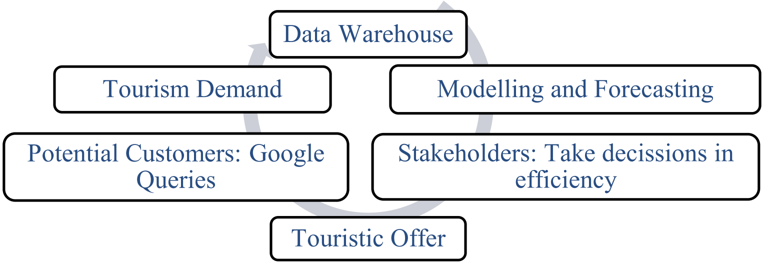

Data Warehouse
Modelling and Forecasting
Stakeholders: Take decissions in efficiency
Touristic Offer
Potential Customers: Google Queries
Tourism Demand

Fig. 1 Data life cycle and efficiency decision scheme. Own elaboration

3 Methodology

In this section, the scheme (see Fig. 1), of the cycle between offer and demand in tourism has been developed under four basic principles of Big Data. Specifically, in our paper, the objective is modelling and forecasting, however, we will suppose ad hoc the data from the Data Warehouse [43]. In this sense, the data will come from official sources of the INE[1] and Google[2]. So, all of the Extraction, Transformation and Loading—ETL [44], work will come from the data engineering of these entities. The main objective is to make efficiencies predictions based on knowledge to improve the user experiences of Tourism Demand and the offers of the stakeholders.

3.1 Modelling and Forecasting Evaluation

In this paper, ARDL + seasonality model is proposed and its application with data from Big Data architectures is analyzed. This modelling allows to know how HODS is generated through the searches of Google users (by country of origin). The purpose of this model is to know the causality relationship and to be able to make forecasts. To analyze the relationship between Granger causality and seasonality a test is developed. To evaluate the forecasting capacity is developed Matrix U1 Theil by country of origin. This matrix is developed to evaluate forecasting capabilities in order to obtain a comparative dimensionless measure among models. For a more in-depth detail of the predictions made, the reader can refer to the references of SARIMA [45] and Singular Spectrum Analysis [46]. All models are made for different scenarios and forecast comparisons are made for different time horizons h = 3, 6, 12, 18.

Granger causality and seasonality testing: ARDL and ECM. We develop the test proposed by Granger [47] and discussed by Montero [48], to detect the causality, since it is not observed with the simple analysis of correlation.

[1] INE: Instituto Nacional de Estadística (Spain). The National Statistics Institute (Spain).

[2] www.google.com.

The model considered by Granger is for two variables (y_t, x_t). Due to the great influence of seasonality [49], in the Tourism sector, the following equation is proposed with HAC covariance method which determines the robust standard error for parameters estimated

$$\ln(y_t) = \beta_0 \ln(x_t) + \sum_{j=1}^{m} \beta_j \ln\left(x_{t-j}\right) + \sum_{j=1}^{m} \alpha_j \ln\left(y_{t-j}\right) + \sum_{i=1}^{12} \delta_i w_i + \varepsilon_t' \tag{1}$$

where w_i is a deterministic seasonal dummy ($i = 1, \ldots, 12$) component and for monthly data is defined as follows:

$$\begin{aligned} w_1 &= -1, \textit{for others}\, w_i = 0 \\ w_1 = -1, w_2 &= 1\, \textit{for others}\, w_i = 0 \\ w_1 = -1, w_3 &= 1\, \textit{for others}\, w_i = 0 \\ &\vdots \\ w_1 = -1, w_{12} &= 1\, \textit{for others}\, w_i = 0 \end{aligned}$$

The use of HAC covariance method guarantees the efficiency of the parameters estimated. Once obtained ε_t', this will be distributed as white noise.

The decision of causality with seasonal effects (Testing linear restrictions for parameters of x_{t-j} and w_i) is asymptotically ($T \geq 60$) as Chi-squared [50].

The most general expression of a dynamic model named ARDL[3] (m, n) with seasonal components is as follows [51, 52]:

$$\gamma(L) \ln(y_t) = \delta(L) \ln(x_t) + \sum_{i=1}^{12} \alpha_i w_i + \varepsilon_t \tag{2}$$

With the interest of evaluating the dynamic persistence of an effect on the exogenous variable at a certain moment, the Error Correction Model (ECM regression or ARDL Error Correction Regression) is constructed. The ECM[4] regression is as follows:

$$\Delta \ln(y_t) = \delta_0 \Delta \ln(x_t) + \sum_{j=1}^{n} \lambda_j \Delta \ln\left(x_{t-j}\right) + \sum_{j=1}^{m} \delta_j \Delta \ln\left(y_{t-j}\right)$$

[3] m is the number of endogenous variables y_t(HODS); n is the number of exogenous variables x_t(Google Queries). ln is the Natural Logarithm. (L) is the Lag operator. Stability conditions: if inverted roots are $|\gamma(L)| < 1$.

[4] Granger-Engle representation theorem and parameters are estimated in two stages. Consistency and Efficiency of estimators are fulfilled.

$$-\gamma(L)\big[\ln(y_{t-1})-\beta\ln(x_{t-1})\big]+\sum_{i=1}^{12}\alpha_i w_i+\varepsilon_t \tag{3}$$

In this model, short-term effect is represented by parameters of first variables differentiated, while long-term effects $|\gamma(L)| < 1$ are represented by Correction Error term. According to Zivot [53], if long-term effect is not statically significant, cointegration does not exist. The long-run multiplier is defined as $\beta = \frac{\delta(L)}{\gamma(L)}$

Forecasting Evaluation: Theil's measures. To verify the forecasting accuracy of different models, we adopted an evaluation criterion to compare the out-sample forecasting performance. We will work with the inequality index of Theil [36]

$$U_1=\frac{\left[\frac{1}{h}\sum_{h=1}^{18}\left(y_{T+h}-\hat{y}_{T+h}\right)^2\right]^{1/2}}{\left[\frac{1}{h}\sum_{h=1}^{18}\left(y_{T+h}\right)^2\right]^{1/2}+\left[\frac{1}{h}\sum_{h=1}^{18}\left(\hat{y}_{T+h}\right)^2\right]^{1/2}} \tag{4}$$

Ratio Theil's (RT's) is designed to comparisons between predicted variables with horizons h = 3, 6, 12,18.

$$RT's_{y_{it},y_{jt}}=\frac{U_1^{y_{it}}}{U_1^{y_{jt}}} \tag{5}$$

In the mathematical interpretation of the RT's, three situations are described according to the predictive capacity of models: if the RT's is equal to one, both models have the same explanatory capacity; if the ratio is greater than one, this would indicate that the denominator's model has a better explanatory capacity than that of the numerator; if the ratio is less than one, the numerator's model has better predictive results than the denominator.

4 Data

The Data of the number of HODS has been collected by INE. For the number of tourists in Spain, by country of origin, the dataset from the first month of 2010 to June of 2019, was obtained. In the grouping of nationalities, the name of "Resident abroad" should be noted. This includes all foreign nationalities except for the 5 main nationalities described in the table (Germany, France, Italy, Netherlands, UK, USA).

According to the data represented in Fig. 2, the average of Residents Abroad was 16,180,005.75 in the period cited. The maximum number of hotel occupancy was recorded in August 2017, with 29,594,071 and the minimum 11.887.105 in January 2010.

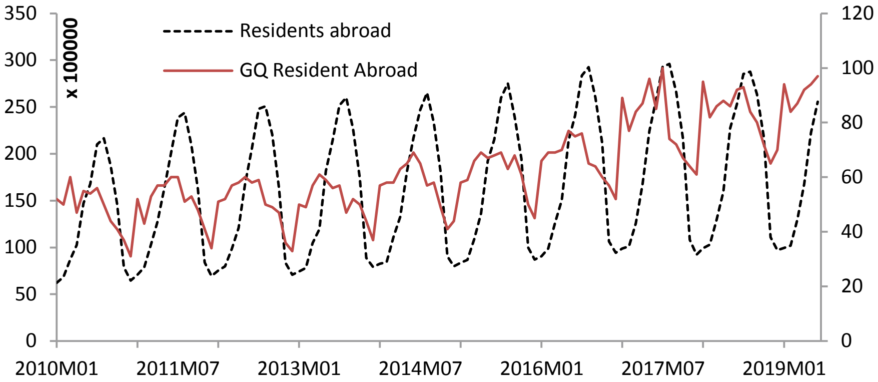


Fig. 2 number of HODS and keyword "visit Spain" for Resident abroad (Jan. 2010–June 2019). Own elaboration

To obtain data from Google, the Big Data tool called GT has been used. Previously GT tools have been used to make forecasts as is cited in the literature review. The lowest interest occurred in December of the year 2010. Analyzing the data obtained of interest for the keyword or Google Query (GQ) "visit Spain", the greatest worldwide interest of the word was in May 2017, just with three periods of advance to the maximum historical overnight stays in Spain.

With the observation of the maximum and minimum values of both series analyzed, it is observed graphically that searches on the Internet are made with at least one period in advance.

Table 1 displays a summary of variables selected by nationalities: Hotel demand and GQ. According to the two series selected, it is worth mentioning that only the variable "Google Queries" in the case of Residents abroad (and USA HODS) meets the hypothesis of normality at 95% confidence (Jarque-Bera). As for stochastic trends (ADF test), all nationalities have unitary roots in Hotel demand and only three cases have been found in which there is evidence of unit root: they are the Google Queries of the Residents abroad, UK and USA. Regarding the stationarity in variance (KPSS), a more stationary behaviour is observed in the Hotel Demand variable for all nationalities including Residents abroad. On the other hand, in the Google queries variable, there is a clearly non-stationary behaviour in the series of Residents Abroad, UK and USA.

5 Empirical Results

The empirical results obtained from the application of the previously proposed methodology section are briefly summarized in the following text. In this paper of

Table 1 Mean and stationary analysis of HODS and keyword “visit Spain” sample period Jan. 2010–December 2017. *P*-values in brackets. Own elaboration

	Mean	Jarque-Bera	ADF	KPSS
Hotel demand				
Residents abroad	16,180,005.75	10.03 (0.01)	−1.50 (0.52)	0.49
Germany	3,846,629.63	13.23 (0.00)	−2.35 (0.15)	0.08
France	1,231,000.87	16.41 (0.00)	−1.36 (0.59)	0.51
Italy	711,484.83	76.14 (0.00)	−1.01 (0.74)	0.10
Netherlands	645,451.61	8.25 (0.02)	−0.60 (0.86)	0.43
UK	4,113,511.96	11.72 (0.01)	−1.55 (0.50)	0.35
USA	437,373.41	5.22 (0.07)	1.19 (0.99)	0.67
Google queries (*GQ*)				
Residents abroad	62.62	4.98 (0.08)	1.53 (0.99)	1.07
Germany	47.02	7.71 (0.02)	−3.70 (0.00)	0.53
France	44.29	8.41 (0.01)	−10.67 (0.00)	0.65
Italy	28.28	29.93 (0.00)	−9.51 (0.00)	0.49
Netherlands	38.04	25.15 (0.00)	−9.16 (0.00)	0.49
UK	41.54	11.38 (0.00)	−0.76 (0.81)	1.14
USA	57.11	8.23 (0.01)	1.29 (0.99)	0.95

predictive techniques, we will focus expressly on the dynamic model with explanatory variables of Internet searches (“visit Spain”) and seasonal factors. The Granger-Causality test extended to seasonality confirms this hypothesis at least within 95% of confidence. As usual in the literature, the forecasting is carried out for time horizons h = 3, 6,12,18 months. Moreover, this article considers the training period from January 2010–December 2017 and out-sample period from January 2018–June 2019.

The results obtained through the Granger causality test including seasonal factors have determined that the number of HODS could be explained by the number of searches generated on the internet and by a systematic seasonality (Fig. 3).

The ECM with seasonality obtained for residents abroad is as follows (lags selected under Akaike Info Criterion):

$$\Delta \ln\left(\hat{y}_t\right) = \underset{(0.00)}{-0.28}\, \Delta \ln(x_t) - \underset{(0.03)}{0.13}\left[\ln(y_{t-1}) - \underset{(0.00)}{0.55}\ln(x_t)\right] + \sum_{i=1}^{12} \hat{\alpha}_i w_i + \hat{\varepsilon}_t$$

$$\textit{Sample}:\ 2010M\,1\,2017M\,12\,R^2 = 0.9888$$

$$\sum_{i=1}^{12} \hat{\alpha}_i w_i = \underset{(0.03)}{-22.41}\, w_1 \underset{(0.02)}{+1.86}\, w_2 + \underset{(0.01)}{2.05}\, w_3 + \underset{(0.01)}{2.08}\, w_4 + \underset{(0.00)}{2.23}\, w_5 + \underset{(0.01)}{2.13}\, w_6$$

$$+ \underset{(0.01)}{2.11}\, w_7 + \underset{(0.02)}{2.01}\, w_8 + \underset{(0.04)}{1.81}\, w_9 + \underset{(0.06)}{1.65}\, w_{10} + \underset{(0.18)}{1.16}\, w_{11} + \underset{(0.08)}{1.48}\, w_{12}$$

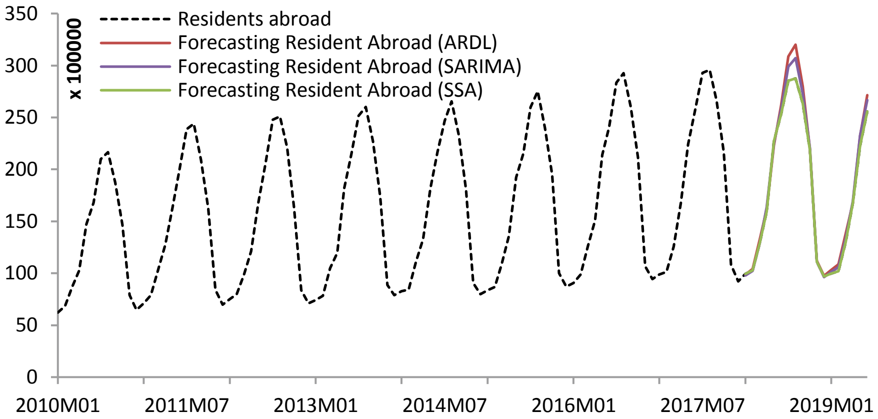


Fig. 3 Out-sample forecast HODS h = 18 (Jan. 2018–Jun. 2019). Own elaboration

In the model defined for the HODS resident abroad variable, two aspects stand out (p-values in brackets): firstly, the existence of a cointegration relationship; second, the strong influence of seasonality. Table 2 shows models and results for HODS by country of origin.

It emphasizes, on the one hand, that all models show a long-term relationship (except for the UK) with a 95% confidence level (USA with 90%). On the other hand, all models are affected by the monthly seasonality, highlighting the fact that the German country of origin every month is significantly different from zero.

Once the results of the three forecasting models cited in the methodology section have been obtained by nationalities of tourists who visit Spain, the RT's can be applied to quantify which model is better in predictive terms.

The results of the forecasting accuracy (see Table 3), depend on the time horizon used and the country of origin analyzed.

In general, we can say that SARIMA models have obtained better results than SSA models (except the Netherlands with h = 12, 18). On the other hand, when comparing with the ARDL causal models with seasonality, the diversity of the results does not

Table 2 Summary of ARDL + seasonality models by country of origin for HODS. Sample Jan. 2010–December 2017. The table shows no relevant seasonality (months). Own elaboration

Hotel demand	ARDL	EC term (Prob)	Seasonality	R^2
Germany	(2,0)	−0.34 (0.00)	–	0.97
France	(4,0)	−0.06 (0.03)	2, 10,11, 12	0.97
Italy	(2,1)	−0.11 (0.01)	9, 10, 11	0.97
Netherlands	(4,2)	−0.12 (0.01)	11, 12	0.96
UK	(1,1)	−0.07 (0.09)	7, 8,9,10,11, 12	0.99
USA	(3,0)	−0.10 (0.05)	2, 8, 10, 11, 12	0.97

Table 3 Matrix U1 Theil forecasting evaluation (Jan. 2018–June 2019): RT's by country of origin. Own elaboration

h	Ratio theil	Residents Ab.	Ger.	France	Italy	Net.	UK	USA
3	SSA/SARIMA	236.84	5.57	5.08	2.33	9.86	5.88	3.48
	ARDL/SARIMA	2.44	0.83	0.98	1.51	1.41	0.84	0.88
6	SSA/SARIMA	76.62	1.47	2.71	1.73	1.75	4.64	2.92
	ARDL/SARIMA	1.46	1.01	0.74	1.30	0.48	1.47	0.85
12	SSA/SARIMA	33.45	1.55	3.53	4.10	0.73	1.58	1.39
	ARDL/SARIMA	1.66	0.79	1.14	2.24	0.43	1.15	0.63
18	SSA/SARIMA	33.50	1.33	3.79	1.96	0.69	1.79	1.00
	ARDL/SARIMA	1.57	0.67	1.11	1.71	0.38	1.05	0.37

allow us to conclude which model has the best forecasting capacity. With a time horizon of 3 months, SARIMA presents the best results in three nationalities of origin (Residents abroad, France, UK), for the rest they have obtained better results of forecasting with ARDL seasonally. For a 6-month time horizon, the best results of ARDL with seasonality have been obtained for France and the Netherlands, against SARIMA. For the 12-month and 18-month time horizons, the gains from using ARDL models with seasonality are observed in the German and Netherlands nationalities. For the rest of the cases, the SARIMA models are superior to those analyzed in this paper.

6 Conclusions

In this paper, the importance of Forecasting modelling and historical analysis carried out in the literature review has been highlighted. The four dimensions of Big Data have been discussed: *volume*, the technologies coming from Google tools for data ETL have allowed analyzing the main markets of origin tourism in Spain; *velocity*, related to the volume of data, the data engineering provided by Google technologies allow us to monitor the Tourism Demand search intentions of the main nationalities who visit Spain; variety, the use of primary data source (INE) and secondary (Google) have allowed build knowledge based on the data. This last one is a novel aspect in the analysis since the users show their interest through the search of information on the Internet; *veracity* of the data verified through the cointegration contrasts carried out. They have allowed modelling the forecasts of Spanish hotel demand by country of origin.

In addition, this article has used more common techniques (SARIMA or ARDL) with a novel technique named SSA. The contribution, in particular, can be divided into the following points:

1. A Granger causality test extended to seasonality has been developed. In the literature, it was usual to perform only the contrast between endogenous and exogenous variables.
2. A criterion of the model's selection based on the predictive capacity of the models has been developed (RT´s). In previous literature work, the gain in the use of models has not been quantified. Theil ratio quantifies the gain between pairs of models.
3. Related to the previous point, Econometric modelling with data from Big Data technologies does not guarantee an improvement in forecasting capacity. It has been demonstrated by the main nationalities who visit Spain.
4. Concerning the dynamic models with seasonality, we have empirically demonstrated that hotel demand decisions are made with at least a period in advance.
5. Cointegration relationship has been revealed expressed in the ECM model.

We can conclude that the models used in this work improve the explanatory capacity of causality (R^2 close to 1) and cointegration relationships have been demonstrated, provide seasonal knowledge in decision making for the Spanish Tourism Demand. According to the results obtained, it is not possible to conclude that there is a gain in terms of forecasting by the use of tools from Big Data engineering; in contrast to what some authors claim [35]. The econometric interpretation of causality models and the economic interpretation can facilitate an adjustment of the offer in terms of prices or even advertising to the agents interested in visiting Spain. This article has been the basis of future research in which data from Big Data technologies are used to make efficient decisions. The theoretical framework could be developed in fields where online markets are relevant. The preferred frameworks for this type of analysis could be Finance, Automotive, Insurance or any sort of market which implies searches on the internet network and this is translated into a quantification of the final decision of the consumer.

References

1. García, J., Molina, J.M., Berlanga, A., Patricio, M.Á., Bustamante, Á.L., Padilla, W.R.: Ciencia de Datos: Técnicas Analíticas y Aprendizaje Estadístico. Un enfoque práctico. Alfaomega, Tarragona (2018)
2. Juul, M.: Tourism and the European Union: Recents Trends and Policy Developments. European Parliamentary Research Service (2015)
3. Pegg, S., Patterson, I., Vila Gariddo, P.: The impact of seasonality on tourism and hospitality operations in the alpine. Int. J. Hosp. Manage. **31**, 659–666 (2012)
4. Ruiz-Reina, M.Á.: Big Data: does it really improve forecasting techniques for tourism demand in Spain?. In: ITISE 2019: International Conference on Time Series and Forecasting on Proceedings of Papers, pp. 694–706. Godel Impresiones Digitales S.L. Granada (2019)
5. Jansen, B.J.: Review of "The search: how Google and its rivals rewrote the rules of business and transformed our culture". Inform. Process. Manage. Int. J. **2**(5), 1399–1401 (2006)
6. Wu, D.C., Song, H., Shen, S.: New developments in tourism and hotel demand modeling and forecasting. Int. J. Contemp. Hosp. Manage. **29**(1), 507–529 (2017)

7. Li, J., Xu, L., Tang, L., Wang, S., Li, L.: Big data in tourism research: a literature review. Tour. Manag. **68**, 301–323 (2018)
8. Li, C., Song, H., Wit, S.: Recent developments in econometric modeling and forecasting. J. Travel Res. **44**(1) (2005)
9. Song, H., Li, G.: Tourism demand modelling and forecasting: a review of Recent research. Tour. Manag. **29**(2), 203–220 (2008)
10. Peng, B., Song, H., Crouch, G.I.: A meta-analysis of international tourism. Tour. Manag. **45**, 181–183 (2014)
11. Xiaoying Jiao, E., Li Chen, J.: Tourism forecasting: a review of methodological developments over the last decade. Tour. Econ. **20**(10), 1–24 (2018)
12. Mariani, M., Baggio, R., Fuchs, M., Höepken, W.: Business intelligence and big data in hospitality and tourism: a systematic literature review. Int. J. Contemp. Hosp. Manage. (2018)
13. Silva, E.S., Hassani, H., Heravi, S., Huang, X.: Forecasting tourism demand with denoised neural networks. Ann. Tour. Res. **74**, 134–154 (2019)
14. Lu, Z., Liu, N.: The guiding effect of information flow of Australian tourism website on tourist flow: process, intensity and mechanism. Hum. Geogr. **22**(5), 88–93 (2007)
15. https://www.researchgate.net/publication/238115677_On_the_Predictability_of_Search_Trends. Last accessed 06 Nov 2019
16. https://static.googleusercontent.com/media/www.google.com/es//googleblogs/pdfs/google_predicting_the_present.pdf. Last accessed 06 Nov 2019
17. http://cs229.stanford.edu/proj2011/GawlikKaurKabaria-PredictingTourismTrendsWithGoogleInsights.pdf. Last accessed 06 Nov 2019
18. Pan, B., Wu, D.C., Song, H.: Forecasting hotel room demand using search engine data. J. Hosp. Tour. Technol. **3**(3), 196–210 (2012)
19. Yang, X., Pan, B., Evans, J.A., Benfu, L.: Forecasting Chinese Tourist volume with search engine data. Tour. Manage. (2015)
20. Onder, I., Gunter, U.: Forecasting tourism demand with Google trends: the case of Vienna. Tour. Anal. (2015)
21. Bangwayo-Skeete, P., Skeete, R.W.: Can Google data improve the forecasting performance of tourist arrivals? Mixed-data sampling approach. Tour. Manage. **46**, 454–464 (2015)
22. Park, S., Lee, J., Song, W.: Short-term forecasting of Japanese tourist inflow to South Korea using Google trends data. J. Travel Tour. Market. **34**(3), 357–368 (2017)
23. Artola, C., Pinto, F., de Pedraza, P.: Can internet searches forecast tourism inflows. Int. J. Manpower **36**(1), 103–116 (2015)
24. Gunter, U., Onder, I.: Forecasting city arrivals with Google analytics. Ann. Tour. Res. **61**, 199–212 (2016)
25. Rivera, R.: A dynamic linear model to forecast hotel registrations in Puerto Rico using Google Trends data. Tour. Manag. **57**, 12–20 (2016)
26. Dinis, G., Costa, C., Pacheco, O.: The use of Google trends data as proxy of foreign tourist inflows to Portugal. Int. J. Cult. Digital Tour. **3**(1), 66–75 (2016)
27. Camacho, M., Pacce, M.J.: Forecasting travellers in Spain with Google's search volume indices. Tour. Econ. **24**(4), 434–448 (2017)
28. Önder, I.: Forecasting tourism demand with Google trends: accuracy comparison of countries versus cities. Int. J. Tour. Res. **19**(6), 1–39 (2017)
29. Zeynalov, A.: Forecasting tourist arrivals in Prague: Google econometrics. Munich Personal RePEc Archive (2017)
30. Liu, J., Li, X., Guo, Y.: Periodicity analysis and a model structure for consumer behavior on hotel online search interest in the US. Int. J. Contemp. Hosp. Manage. **29**(5), 1486–1500 (2017)
31. Rödel, E.: Forecasting tourism demand in Amsterdam with Google Trends. Master Thesis (2017)
32. Palos-Sanchez, P.R., Correia, M.B.: The collaborative economy based analysis of demand: study of Airbnb case in Spain and Portugal. J. Theor. Appl. Electron. Commerce Res. **13**(3), 85–98 (2018)

33. Tang, H., Qiu, Y., Liu, J.: Comparison of periodic behavior of consumer online searches for restaurants in the U.S. and China based on search engine data. IEEE Access (2018)
34. Li, X., Pan, B., Law, R., Hyang, X.: Forecasting tourism demand with composite search index. Tour. Manage. **59**, 57–66 (2017)
35. Sun, S., Wei, Y., Tsui, K.-L., Wang, S.: Forecasting tourist arrivals with machine learning and internet search index. Tour. Manag. **70**, 1–10 (2019)
36. Theil, H.: Econ. Forecasts Policy (1958)
37. Theil, H.: Appl. Econ. Forecasting (1966)
38. Bliemel, F.W.: Theil's forecast accuracy coefficient: a clarification. J. Mark. Res. **10**(4), 444–446 (1973)
39. Ahlburg, D.A.: Forecast evaluation and improvement using theil's decomposition. J. Forecasting **3**(3), 345–351 (1984)
40. Tofallis, C.: A better measure of relative prediction accuracy for model selection and model estimation. J. Oper. Res. Soc. **66**(8), 1352–1362 (2015)
41. Hassani, H., Webster, A., Simiral Silva, E., Heravi, S.: Forecasting U.S. tourist arrivals using optimal singular spectrum analysis. Tour. Manage. **46**, 322–335 (2015)
42. Hassani, E.S., Antonakakis, N., Filis, G.: Forecasting accuracy evaluation of tourist arrivals. Ann. Tour. Res. **63**, 112–127 (2017)
43. Dedić, Stanier: An evaluation of the challenges of multilingualism in data warehouse development. In: 18th International Conference on Enterprise Information Systems—ICEIS 2016 (2016)
44. Dunning, T., Friedman, E.: Time series databases: new ways to store and access data. O'Reilly Media (2014)
45. Box, G.E.P., Jenkins, G.M., Reinsel, G.C.: Time Series Analysis, Forecasting and Control. Wiley, USA (2008)
46. Golyandina, N., Korobeynikov, A., Zhigljavsky, A.: Singular Spectral Analysis with R. Springer (2018)
47. Granger, C.: Investigating causal relations by econometric models and cross spectral methods. Econometrica **37**(3), 424–438 (1969)
48. Montero, R.: Test de Causalidad. Documentos de Trabajo en Economía Aplicada. Universidad de Granada, España (2013)
49. Vergori: Forecasting tourism demand: the role of seasonality. Tour. Econ. **18**(5), 915–930 (2012)
50. Buse, A.: The likelihood ratio, Wald, and Langrange multiplier test: an expository note. Am. Statitician **36**(3), 153–157 (1982)
51. Hylleberg, S., Engle, R., Granger, C., Yoo, B.: Seasonal integration and cointegration. J. Econometrics **44**, 215–238 (1990)
52. Nkoro, E., Uko, K.: Autoregressive Distributed Lag (ARDL) cointegration technique: application and interpretation. J. Stat. Econometric Methods **5**(4), 63–91 (2016)
53. Zivot, E.: The power of single equation tests for cointegration when the cointegrating vector is prespecified. Econometric Theory **16**(3), 407–439 (2000)

Traffic Networks via Neural Networks: Description and Evolution

Alexandros Sopasakis

Abstract We optimize traffic signal timing sequences for a section of a traffic network in order to reduce congestion based on anticipated demand. The system relies on the accuracy of the predicted traffic demand in time and space which is carried out by a neural network. Specifically, we design, train, and evaluate three different neural network models and assert their capability to describe demand from traffic cameras. To train these neural networks we create location specific time series data by approximating vehicle densities from camera images. Each image passes through a cascade of filtering methods and provides a traffic density estimate corresponding to the camera location at that specific time. The system is showcased using real-time camera images from the traffic network of Goteborg. We specifically test this system in reducing congestion for a small section of the traffic network. To facilitate the learning and resulting prediction we collected images from cameras in that network over a couple of months. We then use the neural network to produce forecasts of traffic demand and adjust the traffic signals within that section. To simulate how congestion will evolve once the traffic signals are adjusted we implement an advanced stochastic model.

Keywords Neural network · Traffic signals · LSTM · GRU · SAE · Filter.

1 Traffic Networks and Their Diverse Impact

Continued economic growth and urbanization trends represent an overarching challenge for cities. Rising traffic congestion is an inescapable reality for large and developing metropolitan areas across the world typically costing between 1 and 4% of their respective national GDP. Surprisingly, 11 of the top 15 cities in 2018, with the highest number of hours spent in congestion globally are in Europe as can be seen in Fig. 1.

A. Sopasakis (✉)
Department of Mathematics, Lund University, 22100 Lund, Sweden
e-mail: alexandros.sopasakis@math.lth.se

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_19

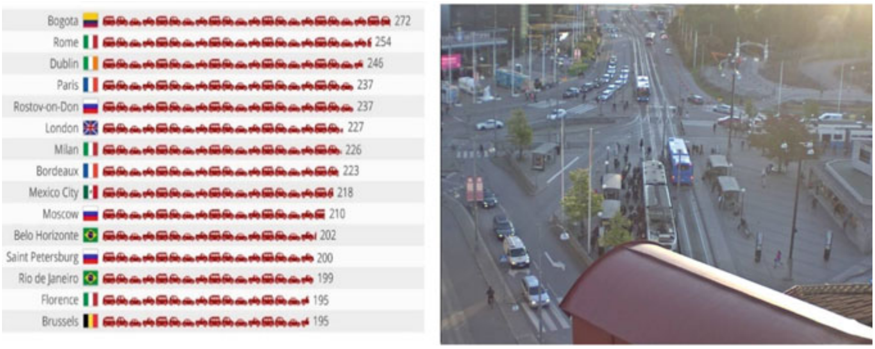


Fig. 1 **(Left)** Number of hours spent in traffic congestion per driver per year. Top 15 cities in the world. Table computed using the Inrix dashboard. **(Right)** Typical morning commute congestion as seen from one of our cameras in the city of Goteborg

Traffic congestion imposes massive costs on governments, transportation companies, and drivers due to time loss, increased pollution rates, and higher incidence of accidents. Each year, congestion costs in Europe amount to 100 billion euros or about 1% of the EU's GDP. Similar costs are shared for cities in the USA and the world. Without effective action, the problem will worsen. An average size taxi company, with 500 vehicles, spends around 197.000 euros per year while idling in congestion and lose approximately 94.000 h of its driver's productivity per year. It is estimated that traffic congestion contributes up to 20% of total CO_2 emissions. Traffic related air pollution has been associated with asthma and respiratory diseases and cardiovascular diseases. This air pollution is the cause behind millions of premature deaths each year.

Traffic is a chaotic phenomenon and as such very hard to predict. Traffic scientists agree [1–3] that there is a gap in theory related to the formation and appearance of traffic congestion and resulting jam. Errors in classical modeling methods [2, 4] are known [3, 5], to increase to levels that render predictions useless, when the number of vehicles is 35% (150 veh/lane/km) dense or higher. It is well accepted [1, 2] and demonstrated in [6], that the state of the art methods involve huge errors. In fact, it is shown that the larger the number of vehicles involved, the larger the error will be for the currently used state of the art methods [2, 3]. Specifically, it is found [6] that the state of the art solution is approximately 22% wrong when the traffic is dense; e.g., when it matters the most.

Describing traffic interactions is intrinsically a multi-scale problem making a description of the resulting evolution difficult. In essence, information at very small spatial and temporal scales can profoundly impact intermediate- and large-scale behavior. Fluctuations in the dynamics can play a dominant role [7] in the system evolution as is evident in long time simulations [8] and asymptotic analysis in a linearized stochastic PDE limit [9]. As a result, resolving the microscopic dynamics is critical. The most widely used methods to produce detailed solutions of traffic

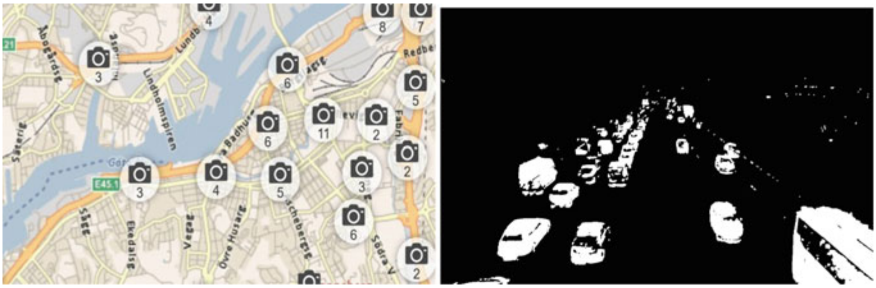


Fig. 2 **(Left)** Overview of a part of the traffic network and locations of (multiple) cameras where some of the images and data are collected from. Example from Goteborg downtown. Data provided by Trafikverket. **(Right)** Filtering one of the images for processing and counting (see Sect. 3 for details.)

models are lattice-based methods involving stochastic Monte Carlo [8, 10] or Cellular Automaton [1–3] techniques. However, not all such methods have a systematic mathematical methodology to model stochasticity.

We present an end to end system which uses images arriving in real time from traffic cameras which through filtering and machine learning methods can produce predictions of yet to appear congestion at different locations and time instances for a city network. As with any machine learning methodology, the process partly relies on amassing a large amount of data from which to train our specialized neural networks. Specifically, we process images, such as the one shown in Fig. 1, from the city of Goteborg, with a series of fast filtering methods in order to remove uninteresting features and focus information. Each such estimate of vehicular density is used to create a time series record of densities. A step in that cascading filtering process can be seen in Fig. 2. The resulting time series are then used to train our neural networks. In return, the neural networks will eventually be asked to produce forecasts of traffic densities into the future.

We begin in Sect. 2, by providing an overview of how neural networks (NN) can be used for describing complex dynamics from data. We focus our presentation on NN architectures most suited for time series data such as the ones we collected for this work. We discuss in Sect. 3, our data collection approach, processing, and specific filtering methods used. In Sect. 4, we present results from training and forecasting for each of the three neural networks chosen. Finally, in Sect. 5, we propose an application of this system into congestion management via dynamic traffic signal control based on real-time forecasts of this neural network. We end with a discussion of the results in that same section.

2 Neural Networks for Time Series Analysis

Neural Networks (NN) are collections of nodes or neurons within a given layer which are connected with neurons from other layers. A NN is nothing more than a collection of all such layers including an input layer, as well as an output layer. The importance of these connections between the neurons is described by unknown parameters called weights. Computing the values for those weights is achieved by using input data for which the resulting output is typically known. This is called training. The main purpose of machine learning algorithms is to compute the values of these weights, and therefore, reduce the error between prediction and reality. The number of layers within a network constitutes how deep that network is.

There are many different types of neural networks which are designed to learn from time series data. Most but not all of those go under the name of recurrent NN (RNN). RNNs consist of a design architecture which has a clear time component. Specifically, each of the connections between layers and respective nodes in a RNN can be thought as a next iteration or a time step. This is what makes RNNs so amenable to time series data and eventual pattern analysis once the network is trained. RNNs are known to suffer from a number of shortcomings [11, 12]. In general, RNNs suffer from short memory. In other words, RNNs cannot remember data information which they learned many time steps previously.

One of the most popular improvements of RNN type networks which does not suffer from the vanishing or exploding gradient problem is the Long Short-Term Memory (LSTM). So an LSTM is a recurrent type NN with improved functionality. A schematic of an LSTM network can be found in Fig. 5 of [13]. In the next section, we discuss LSTMs, as well as a number of other relevant adaptations of RNNs.

2.1 The Three Neural Networks Chosen

In this study, we employ three diverse NN: Long Short-Term Memory NN (LSTM), Gated Recurrent Units (GRU), and Stacked AutoEncoders (SAEs). We refer to figures in [13], for relevant schematics of their architecture.

A SAEs network is able to filter information by breaking down data into its essential elements. Once the network learns which are the elements which best describe the data, then it can easily reproduce that data. A SAEs is a reverse version of an autoencoder (AE) network. The way a AE network works is by *reducing* the number of nodes within successive hidden layers. AEs essentially create a bottleneck within their hidden layers. Effectively, AEs are forced to learn a compressed version of the data containing only the most important of its features. After repeated feeding of many such time series the network would eventually learn which are the most important features in the data set and use only those to describe the data.

A SAE on the other hand is the opposite of an AE [14, 15]. So instead we now ask the network to understand the data by finding representations of its patterns through many more nodes within each of its hidden layers. This allows us to account for every feature within the dataset through a direct representation of its importance in the SAEs hidden layers. A relevant schematic in Fig. 5 of [13], shows a single hidden layer with a larger number of nodes than the input (or the output) layer.

An issue for SAEs, however, is that after training such a network the obvious solution is the identity. An identity would imply that for any input data we would get back as output the same input data. Clearly, such training is useless since the network did not learn any important features for our data. This is actually what would happen if we teach the network by simply feeding it the input data. So, how do we avoid getting an identity network back through the training? We achieve this by requiring that the network activates only a subset of its hidden cells during each of its training sessions [15, 16]. So, in the essence, we teach a sparse set of our network at a time. We furthermore supplement the training by minimizing a corresponding sparse set of the feedback error [15].

Finally, the third NN we implement is a gated recurrent unit (GRU) which is a variation of the LSTM type network. The main difference is that GRUs have one less gate than LSTM networks do. Specifically, as we discussed above, LSTM networks control information proliferation by using an input, an output, and a forget gate. Instead, GRUs have a reset and an update gate. The reset gate has a similar functionality as the forget gate in LSTMs although it is located in a different place within the GRU architecture. The update gate, however, is the one determining how much information to input from the previous layer. Since GRUs lack an output gate they are, in theory, able to proliferate their full information state between successive cells.

2.2 Network Design Specifics

We tested a large number of different designs for each of the above networks [13]. We experimented with the number of layers, number of nodes per layer, as well as how the data is fed in batches, in order to improve learning for each of the networks. The best of these designs are presented below in Tables 1 and 2, for each of the LSTM, GRU, and the SAEs networks, respectively. In these designs, the parameters chosen (or otherwise called hyper-parameters) for each network are considered to be optimal. In the final dense layer used in each network, we implement a Sigmoid [13] activation function.

Table 1 **(Left)** This LSTM network consists of $16896 + 33024 + 65 = 49985$ parameters. Dropout layers are also implemented between the hidden layers. **(Right)** This GRU network consists of a total of 37505 parameters

Network	LSTM			GRU		
Layers	First	Second	Dense	First	Second	Dense
Nodes	64	64	1	64	64	1
Input	288	64×288	64	288	64×288	64
Output	64×288	64	1	64×288	64	1
Parameters	16896	33024	65	12672	24768	65

Table 2 This SAEs network ended up with more than 800,000 parameters and is the largest of the three networks implemented in this work

SAEs	Dense 1	Dense 2	Dense 3	Dense 4	Dense 5	Dense 6
Nodes	400	400	400	400	400	400
Parameters	116,001	160,801	160,801	160,801	160,801	116,001

3 Data Filtering, Training and Simulations

All neural networks require vast amounts of data for training. It is well established that some of the best neural networks will fail if insufficient amounts of data are used for training. In contrast, an average network could produce great results if vast amounts of data is available to train it. So the amount of data is paramount toward the success or failure of any neural network model.

In that respect, we have collected a large amount of still images from several fixed camera locations in the traffic network at the city of Goteborg, in Sweden. There are hundreds of cameras overlooking the traffic network of Goteborg, as can be seen in Figs. 2 and 3. Some of those record videos while others record still images taken every minute. We collected and used for training all such still images for a number of months. There have been several instances, however, where some of the images were lost due to temporary camera malfunction at a specific location or other hardware/software issues. Overall, however, the data size was sufficiently large to allow our NN to succeed in training at several of those camera locations.

3.1 Images and Processing

Image processing is needed in order to ascertain vehicle density for each road section overlooked by the cameras. This is achieved by fast filtering methods. The reason that these methods need to be fast is that images from hundreds of cameras arrive

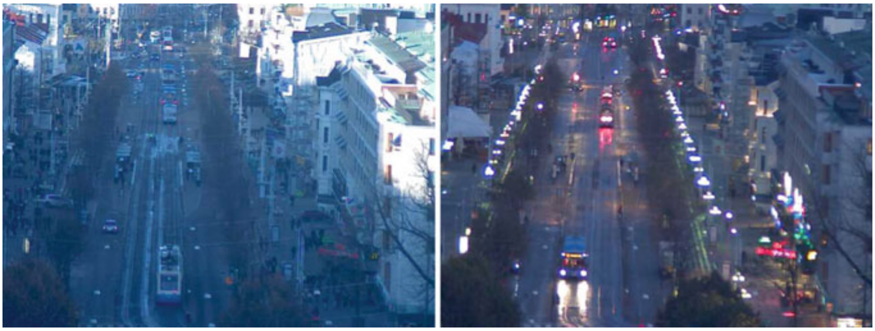


Fig. 3 Viewpoints from cameras in the city of Goteborg during day and night

at one minute intervals, and therefore, the algorithms counting density must be both accurate and speedy.

We process each image by imposing four different procedures in sequence: masking, sharpening, blurring, and finally using a threshold. The last step, involving the threshold, produces a binary output over each pixel in an image thus making it possible to estimate traffic density by simply counting zeros and ones.

We use a mask to focus our density estimation methods into a specific region of the nonmoving camera image. Such a mask can be seen in green in the left part of Fig. 4. Our algorithms, therefore, disregard all other information in the image and only process the pixels within that green region.

3.2 Filtering Cascade

We begin processing by performing edge detection for the masked region of the image. One of the fastest such algorithms is the well-known Canny edge detection by Canny [17]. However, the method is prone to errors if the image is not clear enough. Thus, we preprocess each image by first performing a low-pass denoising procedure. The method consists of applying a Gaussian function [13] on each 5×5 pixel square region of the image. This results in a smoothing effect which makes Canny edge detection later produce much more accurate results. Such a procedure is typically called low-pass filtering in contrast to the high pass filter imposed by Canny edge detection which we discuss below.

Edge detection is responsible for finding the outlines of objects in the image. Accurate edge detection is critical for accurate density counting—occurring at the end of the image processing cascade of treatments each image must undergo. An efficient method to achieve both of the above objectives, accuracy, and efficiency is to compute gradients between neighboring pixels in the image. Image gradients for an image F are computed through the function $\sqrt{F_x + F_y}$. We then verify the existence of edges or remove them by providing a threshold. If the gradient is below

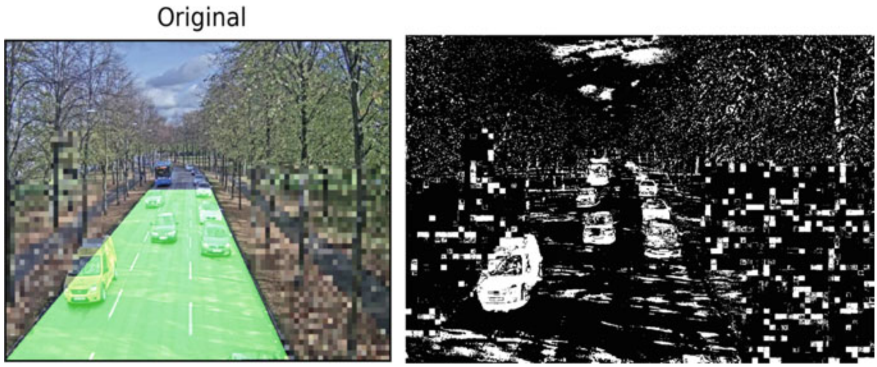


Fig. 4 Original image (left) and corresponding Canny edge detection analysis (right) in real time from a camera in Goteborg

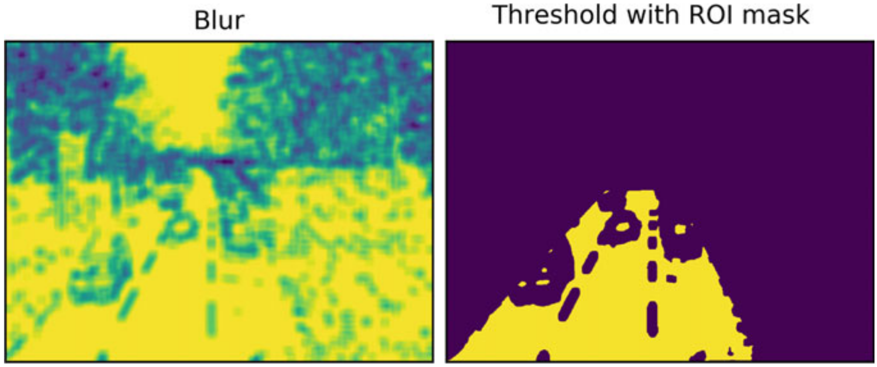


Fig. 5 Further processing of image from Fig. 4 above in order to produce the resulting density estimate for this camera location. Image undergoes blur and threshold analysis for the area masked in green of the original image in Fig. 4

the threshold we eliminate the edge, otherwise, if it is above the threshold, we keep the edge. We refer to [13, 17], for more details. An example of such a detection is also seen in the right part of Fig. 4.

The next step in the cascade is to fill-in the pixels between edges in order to allow more accurate density counting. This is effectively done with a procedure introducing blur into the image. This is equivalent to a low-pass filtering of the image. To achieve this, we reuse the Gaussian filter which we have applied in the beginning of the procedure in order to de-noise the image. The result of this blur procedure can be seen in the left part of Fig. 5. Although blurring an image helps, it is also clear from that image that there still exists much variation in pixels colors. As a result, the final step in the procedure uses a threshold in order to decide which pixels are actually part of the object and which are not. The result is a binary type of image where each pixel can only have one of two possible values. This makes it easier to count the density of objects within the masked region of the image as can be seen in the right part of Fig. 5.

4 Training and Simulations

We use images from each of our camera locations in order to built time series traffic densities for each such location based on the filtering methodologies outlined in the previous section. In this section, we train each of our three machine learning NN using the time series data created from those cameras.

Training and validation results are similar to those presented in Fig. 7 of [13], where the networks are trained under similar conditions. Overall the loss function quickly achieves values below 0.005 for all three networks. In the cases of LSTM and SAEs, the training loss is in fact lower than the validation loss rather quickly in the learning process while the GRU reaches this value a lot later in the process.

4.1 Traffic Density Forecasting

There are a number of ways to produce forecasts at different future time instances. We chose to train a neural network for each such future time instance. Alternatively, we could have also trained one neural network for only one short time prediction and then use that to produce forecasts further along in time. For example, it is possible to train a network to predict traffic density in 5 min and then reuse that network together with the 5 min prediction in order to also predict the traffic density in 10 min. The reason we did not do that is that errors can quickly accumulate in such forecasts. Instead, therefore, we train different networks for each of our time predictions. We, therefore, trained networks which can provide predictions at 5, 10, 15, 30, 60, and 120 min for a number of the camera locations. We present some of those in Fig. 6.

A number of similar results are also available in [13] where the real-time capabilities of such a system are also evident. In particular, the capabilities of this system are put to the test there since they are shown to be able to also predict rare events such as for instance an unusual once a year traffic commute such as a Black Friday (see Fig. 10 in [13]).

5 Traffic Signal Timings and Applications

In this section, we explore applications of such a neural network-based real-time forecasting system towards traffic congestion reduction by adjusting existing traffic signal timing sequences in real time based on camera input.

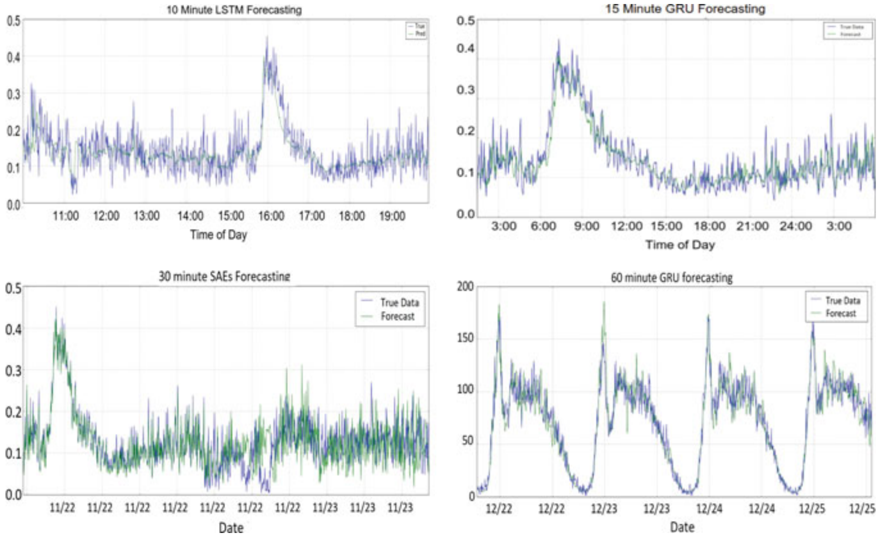


Fig. 6 Traffic density forecasts (green) versus reality (blue) for 10 (LSTM based), 15 (GRU based), 30 (SAEs), and 60 (GRU based) minutes, respectively. Flow vs time. In the last case, we present the real-time capabilities of the methodology. Specifically, we produce 1 h predictions continuously in real time over a period of 6 different days

Specifically, we explore here whether it is possible to use such a trained neural network system in order to *dynamically* adjust traffic light switching. The system will allow the signals to adapt to upcoming traffic conditions in order to avoid or reduce expected congestion phenomena for the monitoring region. The monitoring region consists of a sequence of traffic lights in a section of the traffic network of the city of Goteborg. Changing the actual timing for the traffic lights in that part of the network is not feasible at this stage. As a result, we instead simulate how traffic flow will be affected due to the proposed adjustment of the traffic light timings.

5.1 Stochastic Markov Model

To measure and visualize the ensuing traffic evolution once the traffic lights have changed, we implement advanced stochastic simulations such as those in [18]. The models within these works have been shown [18, 19], to faithfully represent a complicated vehicle to vehicle interactions, lane changing, flow, and velocity fluctuations, etc.

We outline here a *two-dimensional* microscopic lattice-free stochastic process which will set the foundation for the Monte Carlo simulations which we carry out in the next section. We referred [18], for more details.

We define domain $D := \mathbb{T}^2 = [0, 1)^2$ representing a multi-lane roadway. We assume for now that all vehicles have the same size and occupy a space $V_i = V_r(\mathbf{x}_i)$ with radius r around their centers $\mathbf{x}_i \in D$. We split the spatial domain $D = O \cup E$ into an empty set E and an occupied set O. Set O is comprised of the disjoint union of all sets occupied by vehicles in our domain $O = \cup_{i=1}^{k} V_i$. Set $E = \cup_{i=k+1}^{k+l} E_i$ is simply the complement of O. We let I_O and I_E represent the index sets for O and E, respectively. Assuming that k vehicles interact on the spatial domain then $D = O \cup E = V_1 \cup V_2 \cup V_3 \cup \ldots V_k \cup E_{k+1} + \cdots + E_{k+l}$.

We now construct the microscopic stochastic process $\{\sigma\}_{t \geq 0}$ on D. We can define a spin-like variable $\sigma_t(i) \equiv \sigma(i)$ on those sets as follows:

$$\sigma(i) = \begin{cases} 1 \text{ if } V_i\text{, i.e., vehicle exists at index set } i \in I_O, \\ 0 \text{ if } E_i\text{, i.e., there is no vehicle at index set i, e.g., } i \in I_E, \end{cases} \tag{1}$$

where $1 \leq i \leq k + l < M$ assuming k vehicles and l empty sets. Note the although the number of sets can change over time there will always exist an upper bound M for that number.

We denote the configuration of spins on the lattice by $\sigma = \{\sigma(i) | 1 \leq i \leq k + l < M\}$. Note that a spin configuration σ is an element of the configuration space $\Sigma = \{0, 1\}^{k+l}$ and that the size of this space can change in time as vehicles enter or exit the roadway.

We follow the classical development [7–9] of a stochastic process in defining the corresponding inter-particle potential J. Using our set infrastructure, local interactions between vehicles are described from

$$J(i - j) = \frac{1}{(2L + 1)^d} F\left(\frac{1}{2L + 1} |\mathbf{x}_i - \mathbf{x}_j|\right), \ i, j \in I_O \tag{2}$$

where I_O is the index set for O. We let $F : \mathbb{R} \to \mathbb{R}$ with $F(r) = F(-r)$ and $F(r) = 0$ if $|r| \geq 1$. For simplicity, we assume uniform potentials and take $F(r) = J_0 =$ constant for $|r| \leq 1$. For now, we assume uniform potentials and let J_0 to be a constant. The interaction radius for these dynamics is denoted by L.

Cylinder functions $\{c(\mathbf{x}, \pm 1, \cdot); \mathbf{x} \in \mathbb{T}^2\}$ are implemented to describe the rates of evolution for the lattice-free (LF) stochastic process σ_t in a two-dimensional space $\mathbb{T}^2$. Following ideas from lattice-based (LB) particle interactions [8, 18], we propose the rate by which vehicles enter or exit at location i on the roadway to be given by,

$$c(i, \sigma) = c_d \sigma(i) \exp(-\beta U(i, \sigma)) + c_a w(i)(1 - \sigma(i)), \tag{3}$$

where the potential function U is provided below in (5). One of the important differences, however, is the inclusion of the weight function $w(i)$. This function is related to the empty space still available in the domain/roadway for vehicles to enter.

Similarly, in order to allow a vehicle to move and interact with other vehicles within the roadway, we equip our microscopic stochastic process with diffusion dynamics [8, 18],

$$c(i, j, \sigma) = c_{se} w(j)(1 - \sigma(j))\sigma(i)e^{-\beta U(i,\sigma)} + c_{se} w(i)(1 - \sigma(i))\sigma(j)e^{-\beta U(j,\sigma)}. \tag{4}$$

The potential function U implemented in both (3) and (4) describes the length of the vehicle to vehicle interactions [8] and is described by

$$U(i, \sigma) = \sum_{j=1}^{k} J(i - j)\sigma(j), \tag{5}$$

with J from (2).

Here c_a, c_d, and c_{se} are adsorption, desorption, and diffusion constants, respectively, and involve the inverse of the characteristic time of the stochastic process. These constants are usually calibrated from actual data and are based on car velocities, driver reaction times, etc. We use data from camera images to calibrate these constants following the ideas in [18].

5.2 Stochastic Model to Simulate Traffic Dynamics

The model presented in Sect. 5.1, has been extensively tested against real data and shown to effectively reproduce a number of important multi-lane road features. We present some such comparisons here in Fig. 7, and refer to [18, 19], for more examples.

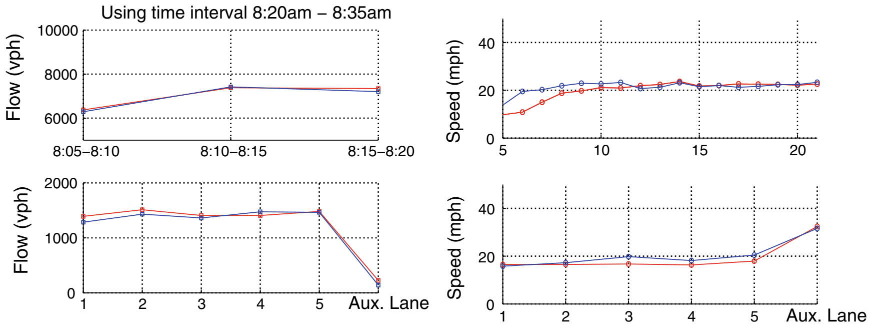


Fig. 7 Comparisons between real and stochastic model simulations of flow and speed over time, location, and number of lanes. Agreement is shown in all cases. The stochastic simulation model from [18], is implemented to produce the predicted quantities

The simulations overflows and vehicle speeds produced by the stochastic model in Fig. 7, seem to produce acceptable results over short periods of time when compared against actual traffic. In [18, 19], comparisons are also included against other well-known simulation packages such as VISIM. The stochastic model is shown to perform better than VISIM there as well when compared against reality.

Based on these tests, we choose this microscopic stochastic model for the simulations which we undertake in the next Sect. 5.3. Our aim in the next section is to compare traffic congestion before and after adjusting timing sequences for traffic lights in a small part of the traffic network which we monitor. Specifically, we use the stochastic model in order to simulate traffic evolution after adjusting the traffic light timings for that part of the traffic network and compare it against reality.

5.3 Traffic Signal Assignment and Heat Map

In all simulations presented in this section, we use the trained GRU neural network. This network has produced reasonable estimates of the number of vehicles expected to arrive at each of the camera locations. An effective time horizon for use of this forecasting system can range from a few minutes and up to 2 h. Even rare traffic events, such as an unusual Black Friday commute, are possible to predict in advance as shown in [13].

We specifically train our neural network model to produce 30 min predictions of traffic density for the region of Haga, in Goteborg, as shown in Fig. 8. We train the neural network to anticipate traffic density on each of the locations where cameras are placed for the region presented in Fig. 8. We present in the left part of that figure a 10 min average of the actual traffic density between the hours of 7:30 and 7:40 a.m. Using the stochastic model from Sect. 5.2, we then simulate a 10 min evolution of density and present its average in the right part of that same figure.

Although this is only a first test case, it is sufficient to convey the question of the study: is it possible to improve traffic congestion, at least for a small part of a traffic network, given sufficiently accurate predictions of expected traffic demand? These first positive results seem to suggest that it is worth investigating this further. In other words, if the neural network can produce a reasonable expectation of the number of vehicles which will arrive on the monitoring roads of that part of the network we might be able to adjust the traffic lights in order to improve traffic characteristics of interest such as flow, velocity, fuel consumption, etc. Clearly, we do not expect that all of those quantities can be optimized at once. In some cases, optimizing one of them may be detrimental to another. The main message, however, here is that it should at least be possible to choose one such quantity and optimize it based on the methodology outlined above for a limited time window. A longer study is needed in order to understand the extent of our estimation errors, as well as the sensitivity of other quantities of interest, such as the length of the time windows for prediction and simulation.

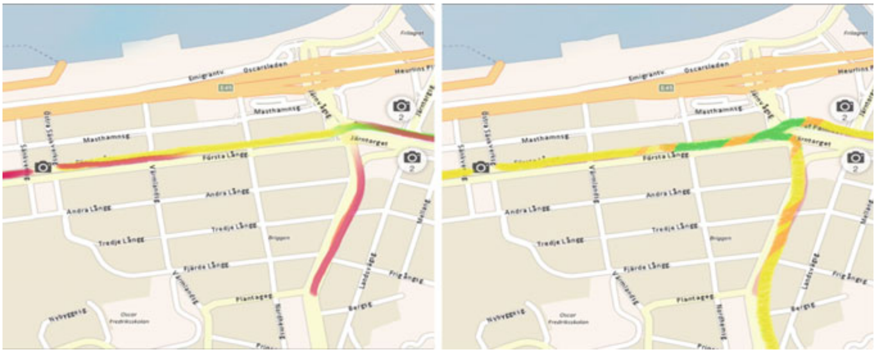


Fig. 8 A heat map which displays the traffic density based on information from cameras for a section of the Goteborg traffic network. Colors indicate different vehicle densities and corresponding velocities. Red is used for stopped vehicles, yellow for reduced velocities, and green for free-flowing vehicles. On the **left** actual traffic density averages over a 10 min window between the hours of 7:30 and 7:40 a.m. On the **right** simulated traffic density averages over the same 10 min after adjusting traffic light timings in order to reduce anticipated road congestion. We implemented the stochastic model from [13], as discussed in Sect. 5.2, to simulate traffic evolution for that section of the network based on new traffic light timings. This dynamic signal adjustment based on the predicted traffic demand seems to alleviate the stop and go traffic waves (left) as shown in our simulation (right)

References

1. Helbing, D., Hennecke, A., Shvetsov, V., Treiber, M.: Micro and macro simulation of freeway traffic. Math. Comp. Modell. **35**, 517 (2002)
2. Schadschneider, A.: Traffic flow: a statistical physics point of view. Physica A **312**, 153 (2002)
3. Schreckenberg, M., Wolf, D.E.: Traffic and Granular Flow. Springer, Singapore (1998)
4. Nagel, K., Schreckenberg, M.: A cellular automaton model for freeway traffic. J. Phys. I **2**, 2221 (1992)
5. Tossavainen, O., Work, D.: Markov chain Monte Carlo based inverse modeling of traffic flows using GPS data. Netw. Heterogen. Media **8**(3), 803–824 (2013)
6. Sopasakis, A., Katsoulakis, M.A.: Stochastic modeling and simulation of traffic flow: ASEP with Arrhenius look-ahead dynamics. SIAM J. Appl. Math. **66**(2), 921–944 (2005)
7. Katsoulakis, M.A., Plecháč, P., Sopasakis, A.: Numerical analysis of coarse-grained stochastic lattice dynamics. SIAM J. Numer. Anal. **44**(1), 2270–2296 (2006)
8. Katsoulakis, M.A., Majda, A.J., Vlachos, D.G.: Coarse-grained stochastic processes for microscopic lattice systems. Proc. Natl. Acad. Sci. USA **100**(3), 782–787 (2003)
9. Krug, J., Spohn, H.: Universality classes for deterministic surface growth. Phys. Rev. A **38**, 4271 (1988)
10. Tympakianaki, A., Koutsopoulos, H., Jenelius, E.: c-SPSA: Cluster-wise simultaneous perturbation stochastic approximation algorithm and its application to dynamic origin-destination matrix estimation. Transp. Res. Part C **55**, 231–245 (2015)
11. Fu, R., Zhang, Z., Li, L.: Using LSTM and GRU neural network methods for traffic flow prediction. Chin. Assoc. Autom. **324–328**, 2017 (2017)
12. Jeffrey, E.L.: Finding structure in time. Cogn. Sci. **14**(2), 179–211 (1990)
13. Sopasakis, A.: Traffic demand and longer term forecasting from real-time observations. In: Proceedings ITISE-2019, pp. 1247–1259. Springer, Granada (2019)
14. Lv, Y., Duan, Y., Kang, W., Li, Z., Wang, F.Y.: Traffic flow prediction with big data: a deep learning approach. IEEE Trans. Intell. Transp. Syst. **16**(2), 865–873 (2015)

15. Ranzato, M., Poultney, C., Chopra, S., LeCun, Y.: Efficient learning of sparse representations with an energy-based model. In: Proceedings of NIPS (2007)
16. Makhzani, A., Frey, B.: K-sparse autoencoders (2013). arXiv preprint arXiv:1312.5663
17. Canny, F.J.: A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. **8**(6), 679–698 (1986)
18. Sopasakis, A., Katsoulakis, M.A.: Information metrics for improved traffic model fidelity through sensitivity analysis and data assimilation. Trasnp. Res. Part B: Methodol. **86**, 1–18 (2016)
19. Sopasakis, A.: Lattice free stochastic dynamics. Comm. Comput. Phys. **12**(3), 691–702 (2012)

Time Series Analysis with Computational Intelligence

A Comparative Study on Machine Learning Techniques for Intense Convective Rainfall Events Forecasting

Matteo Sangiorgio, Stefano Barindelli, Valerio Guglieri, Riccardo Biondi, Enrico Solazzo, Eugenio Realini, Giovanna Venuti, and Giorgio Guariso

Abstract In the last decades, the great availability of data and computing power drove the development of powerful machine learning techniques in many research areas, including the ones, as the meteorology, where traditional conceptual models were usually adopted. In this work, we analyze the performance obtained by different techniques in the forecasting of intense rainfall events. A linear classifier, the logistic regression, is used as a benchmark in order to fairly evaluate more complex nonlinear tools: a support vector machine, a deep neural network, and a random forest. Our analysis focuses on both the accuracy and computing effort necessary to identify these models. The nonlinear predictors are proved to outperform the linear baseline model. Under a computational perspective, both neural network and random forest turn out to be more efficient than the support vector machine. The study area we considered is composed of the catchments of four rivers (Lambro, Seveso, Groane, and Olona) in the Lombardy region, Northern Italy, just upstream from the highly urbanized metropolitan area of Milan. Data of intense convective rainfall events from 2010 up to 2017 (more than 600 events) have been used to identify and test the four considered predictors.

Keywords Nowcasting · Logistic regression · Random forest · Support vector machines · Deep neural networks · Global navigation satellite system · Zenith tropospheric delay

M. Sangiorgio (✉) · G. Guariso
Department of Electronics, Information, and Bioengineering, Politecnico di Milano, Milan, Italy
e-mail: matteo.sangiorgio@polimi.it

S. Barindelli (✉) · V. Guglieri · E. Solazzo · G. Venuti
Department of Civil and Environmental Engineering, Politecnico di Milano, Milan, Italy
e-mail: stefano.barindelli@polimi.it

R. Biondi
Dipartimento di Geoscienze, Università degli Studi di Padova, Padua, Italy

E. Realini
Geomatics Research & Development srl (GReD), Lomazzo (CO), Italy

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_20

1 Introduction

In recent years, researchers have adopted novel machine learning tools to a wide range of different applications. Among these, a challenging task is to try to improve the performances of traditional physically based models in weather forecasting [1–5]. A huge advantage of the black-box models is that they well fit the need for fast real-time algorithms for nowcasting applications and early-warning systems for natural hazards. Moreover, the computational effort needed to run a Numerical Weather Prediction (NWP) model is much higher than the one required by a machine learning algorithm.

Intense rain events develop locally and are difficult to be predicted because of the coarse spatial resolution of NWP models compared to the area interested by the single convective cell. This kind of rainfall events is characterized by peculiar local environmental conditions and requires solving physical and meteorological equations at a fine spatial and temporal scale, easily becoming a hard task both in terms of time and computational power. The scope of this work is to implement different machine learning techniques to forecast these local, intense rain events, and to compare their performances.

The considered area, located in the Lombardy region, Northern Italy, is composed of the hydrological basins of four torrential rivers (Lambro, Seveso, Groane, and Olona). The latitudes of the considered basin span from 45.37 N to 45.93 N, longitude from 8.77 E to 9.40 E, and the extension is almost 1400 km^2. This is a high-risk territory due to the high frequency of severe and short thunderstorms [6, 7], which usually trigger flash floods. The situation is even more critical due to the presence of the highly urbanized metropolitan area of Milan, where the flows coming from the four rivers are drained, causing severe damage. In 2014, for instance, floods produced damages evaluated in several tens of million euros in the Milan municipality.

In this work, the machine learning tools that have been implemented and compared are Logistic Regression (LR), Support Vector Machine (SVM), Deep Neural Network (DNN), and Random Forest (RF). In order to fairly compare these models, we feed each of them with the same set of input variables. Each model returns the prediction about the occurrence of an intense rain event as output.

In addition to the classical meteorological variables (temperature, pressure, humidity, wind speed), we also included within the input variables the Zenith Tropospheric Delay (ZTD), which is a proxy of water vapor in the atmosphere, a fundamental variable in rain events genesis [8–11] and storm development [12, 13]. Some recent works showed that this variable could be effectively used to boost the performance of black-box models trained for weather forecasting related tasks [14–23]. It is important to specify that ZTD is a commonly available side result of the calibration procedure of global positioning systems.

First, we briefly describe the functioning of the considered machine learning techniques with a special focus on the differences between them. We then evaluate their performances in terms of accuracy and the computational effort required in the training and inference phase.

2 Materials and Methods

2.1 Dataset Description

The input dataset consists of meteorological variable values with a temporal resolution of 10 min. Temperature, atmospheric pressure, wind (intensity and direction), and relative humidity are used to feed the prediction algorithm together with the ZTD retrieved by Como GNSS station (45.8021 N, 9.0953 E, 246 m a.s.l.), that belongs to the European Permanent Network. GNSS raw observations have been post-processed with the precise point positioning technique [24], an approach that allows processing a single station without the need for relying on a network of receivers. Intuitively, the ZTD is the delay introduced in the GNSS signal propagation due to the presence of the atmosphere [25]. A component of this delay is caused by gases in hydrostatic equilibrium, i.e., Zenith Hydrostatic Delay (ZHD), and a component caused by atmospheric water vapor, i.e., Zenith Wet Delay (ZWD). Since the first term, mainly affected by the orography of the region of interest, has very small fluctuations in time, the ZTD could be considered a proxy of the content of water vapor along the vertical direction of the GNSS receiver [26]. Although the ZTD is referred to the zenith direction, it depends on the delay introduced along each single line of sight between the receiver and the satellites in view. Several studies have shown that there is a strong correlation between ZTD and the occurrence of rain events [8, 27, 28], and between the atmospheric integrated water vapor and intense storms [29, 30, 13]: the condensation of water vapor leads to the formation of raindrops. Figure 1 shows an intense convective thunderstorm on the study area (left) and the damages generated by this event in the north of Milan (right).

Each sample of our dataset is composed of an input vector that reports the meteorological variables and the corresponding binary output: “1” for the occurrence of an intense rainfall event and “0” otherwise. For this specific case study, a rainfall event has been classified as intense if it persists on the study area for more than 25 min and its radar reflectivity is greater than 50 dBZ, as recorded by the Thunderstorm Radar Tracking (TRT) algorithm [31–33] (a radar-based tool able to track convective cells inside a thunderstorm system). The dataset spans from 2010 to 2017, eight years of data that have been used to identify and evaluate the machine learning algorithms.

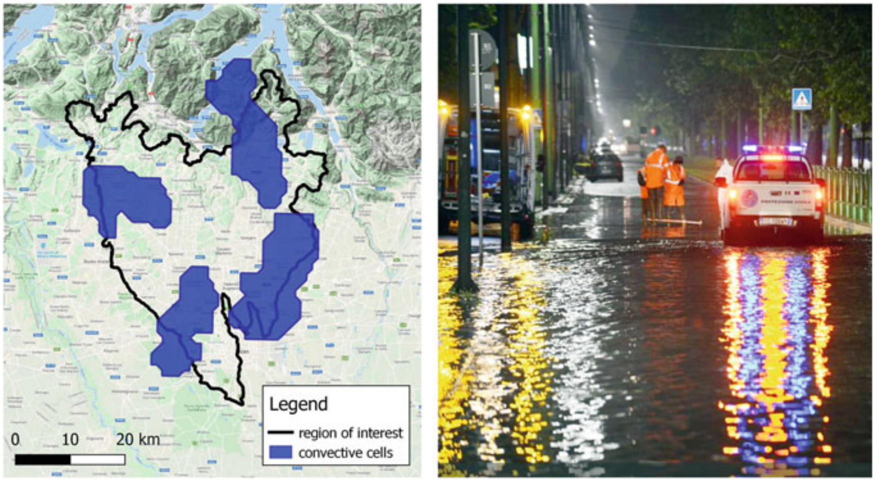


Fig. 1 Convective rainfall events occurring on the May 11, 2017, 10:45 p.m. (left), and causing an overflow of the Seveso river at Niguarda, North of Milan, starting from 11:30 p.m. A picture of the following morning (www.milano.corriere.it) (right)

2.2 Machine Learning Techniques

The problem considered here is usually known as binary classification. At each time step, the future occurrence of an intense rainfall event is predicted, given the current atmospheric conditions. Following a traditional supervised learning approach, the classifiers are trained using a dataset where each sample is qualified by its features (input) and is already categorized with the actual occurrence of a critical rainfall event (output).

First, we implemented a logistic regression: a linear classifier that splits the input space with a hyperplane and classifies each sample based on its position relative to this linear decision boundary [34], as shown in Fig. 2 (top row). Due to its simplicity, linear regression is usually used as a benchmark to evaluate the performance of more complex classifiers.

Given the complexity of the processes taking place in the atmosphere, which are well known for their nonlinear behavior, an advanced model able to deal with the nonlinearity of the physical system may provide better results as reported in previous works (for a review of the topic, see [35–37] and the references mentioned there).

In the machine learning literature, there are three main strategies to separate categories in a nonlinear way. The first strategy we consider is a kernel-based classifier, so-called since it makes use of the kernel trick [38]. The most popular of these models is the support vector machines [39], which project the data from the input space to a new high-dimensional space applying kernels (usually Gaussian) and then search for the linear manifold maximizing the margin [40] between the classes, as shown in Fig. 2 (second row). Note that this linear boundary can be mapped back to the

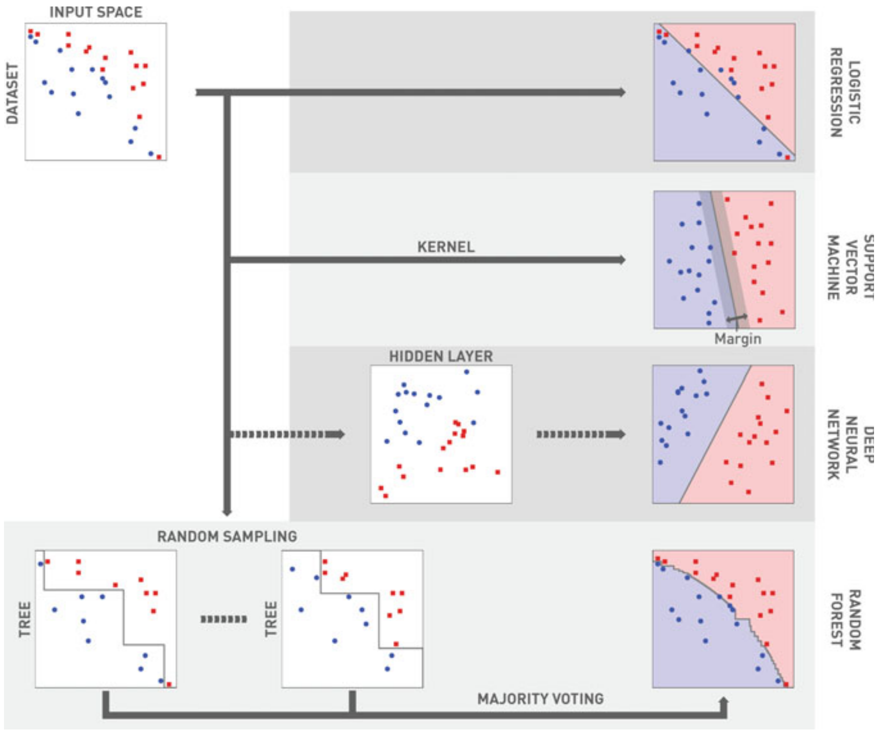


Fig. 2 Schematic representation of how the different machine learning techniques divide the input space. The grey row on top is dedicated to LR. The second and the third to SVM and DNN, respectively. The last represents the processes of random sampling and majority voting which characterize the RF

input space obtaining the corresponding nonlinear one. The main drawback of this approach is that the kernel trick becomes expensive under a computational point of view when the dataset is composed of numerous samples.

An alternative to kernel-based classifiers is deep learning [41], which became widely used in the last decade. Deep learning makes use of multi-layer (deep) artificial neural networks. These architectures are inspired by the structure of the human brain and are made of nodes (called neurons), organized in layers. The first hidden layer performs a nonlinear transformation of the input space, and the same procedure is repeated between subsequent hidden layers. In the end, the output layer computes the Boolean output through a normalized exponential activation function (*softmax*), which performs the actual classification [42]. In the network described above, the information flows from the input layer through the hidden layers to the output layer, without any loop. Such neural architecture is traditionally named feed-forward neural network. The layers composing the DNN considered in this work are fully connected (or dense): every neuron in one layer is connected to every neuron in the previous layer. The design of a DNN is not trivial due to the high number of hyper-parameters that define its structure (nonlinear activation functions shape, number of hidden layers, number of neurons for each layer) and the characteristics of the training process (learning rate, batch size, regularization rate). Since we are dealing with a classification task, we considered the binary cross-entropy as loss function and the overall classification accuracy as validation metrics. Early stopping and L2 norm weight regularization have been used to avoid overfitting on training data. As it happens for SVMs, one can map the linear hyperplane of the last layer to the input space, obtaining a nonlinear decision boundary (see Fig. 2, third row).

The third alternative exploits decision tree-based algorithms, a random forest classifier [43]. A RF is an ensemble of classification trees: each tree recursively divides the input space using thresholds. For this reason, the feature space in each tree is separated by orthogonal hyperplanes, which results in a box-like decision boundary [44–46]. A single classification tree is not trained on the whole training set; it is built considering a random (both on instances and features) subset of the training dataset. The algorithm that builds the tree operates with a top-down procedure, choosing at each step the variable that performs the best split of the data. Once all the trees have been identified, the final output of the RF is computed adopting a majority voting system, as reported in the last row of Fig. 2.

As it is common practice in the identification of machine learning models, the dataset has been split into training (years from 2010 to 2015), validation (2016), and test (2017) sets. The first subset, the training set, is used to compute the optimal values of the parameters. The second, the validation set, serves to tune the hyper-parameters and to define the complexity of the model structure. To find the best combination of hyper-parameter values, we implemented a traditional grid search approach. The third, the test set, is not involved in the identification process, and is employed to fairly evaluate the performances of the model only once all the parameters and hyper-parameters have been definitively fixed.

We adopted the Scikit-learn library [47] implementation for LR, SVM, and RF. The code for the DNN has been written in Keras [48] with TensorFlow backend [49].

3 Results

During the training process, the classifiers do not return a Boolean output directly, but a value in the range from 0 to 1. After the training, it is necessary to investigate which is the proper value of the threshold delimiting the two classes. The default value for the threshold is 0.5, meaning that if the output is below 0.5, the sample is classified as "No Thunderstorm", while if it is greater than 0.5, the predicted category will be "Thunderstorm". Unluckily, many times this is not the best choice and it is necessary to perform an analysis of the result obtained plotting the Receiver Operating Characteristic (ROC) curve of the model [50]. Each model's curve is obtained changing the value of the threshold separating the two classes from 0 to 1 with a certain step size. The curve relative to a random classifier would be the bisector as reported in Fig. 3. The one of a perfect model would be a step-wise linear

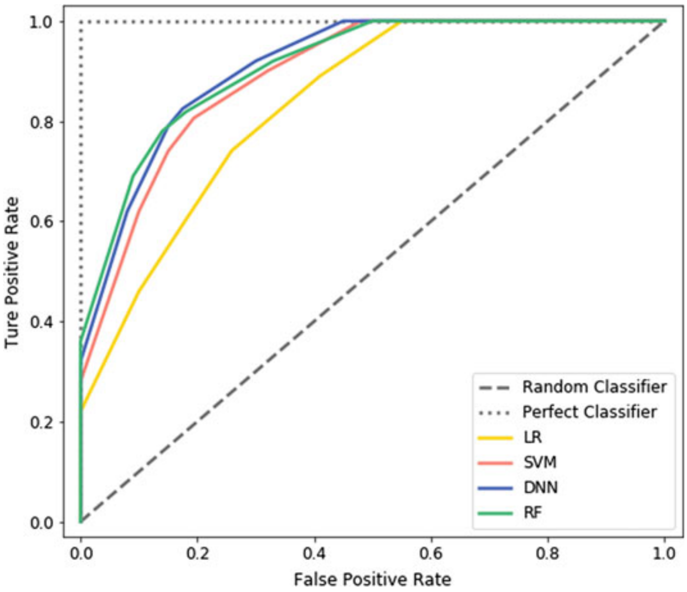


Fig. 3 ROC curves of the four considered predictors. The x-axis reports the false positive rate, the y-axis the true positive rate (also called sensitivity)

function connecting the points (0, 0), (0, 1), and (1,1).

The analysis of the ROC curves allows comparing the models removing the dependence on the value of the threshold and selecting a proper value of the threshold considering the tradeoff between true positive rate and false-positive rate. The traditional criteria to fairly compare different models basing on their ROC curves consist of evaluating the Area Under the Curve (AUC) [51]. When the AUC is close to 0.5, the classifier behaves on average as a random classification [52]. Conversely, when the AUC is close to one, the model has performance similar to a perfect classifier. In the case here considered, the LR has an AUC equal to 0.83. Adopting a nonlinear classifier, the AUC increases to 0.89 for the SVM, and 0.91 for DNN and RF.

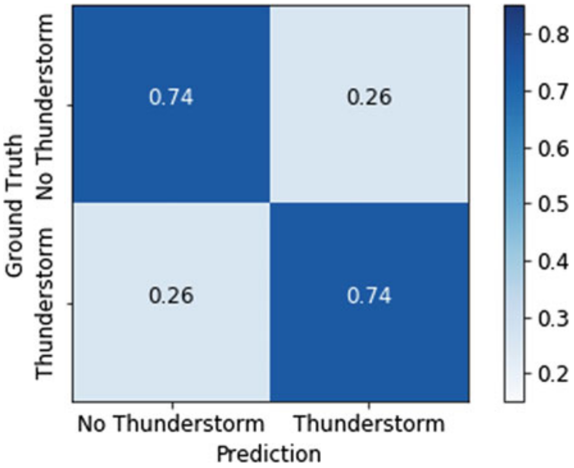
Figure 3 is also useful to visualize the tradeoff between the fraction of true positives and the fraction of false positives.

The analysis of the ROC curves allows selecting a value of the threshold which is appropriate for the considered application. For instance, it is possible to select the threshold which balances the true negative and true positive rate. As it is easy to demonstrate with basic arithmetic computation, these points are those at the intersection between the ROC curve and the line connecting (0, 1) and (1, 0).

The LR we used as baseline guarantees an overall accuracy of 74.1 %, and the corresponding confusion matrix is reported in Fig. 4.

As already stated in the previous section, due to the nonlinear nature of the processes which take place in the atmosphere, it is really unlikely that a simple linear classifier, as LR, turns out to be the best approach to deal with the thunderstorm

Fig. 4 Confusion matrix obtained with the LR



forecasting. The idea expressed above is confirmed by the performances obtained with the nonlinear models we implemented. The overall accuracy of these models increases by 6–8 % with respect to the LR. Figures 5, 6, and 7 report the confusion matrices for SVM, DNN, and RF, respectively.

In the last analysis, we couple the accuracy in the classification task with the computational effort required by the training process, expressed in terms of computing time. The technical specification of the computer used for the case study was, respectively, Intel(R) Core (TM) 17-4770 CPU @ 3.40 GHz and Intel(R) HD Graphics 4600. Figure 8 reports the training time on the logarithmic horizontal axis (lower values are better), and the overall accuracy on the vertical axis (values close to 1 are better). As expected, the model that requires the lower computational effort is the LR (0.03 min). The three nonlinear models have a more complex structure and

Fig. 5 Confusion matrix obtained with the SVM

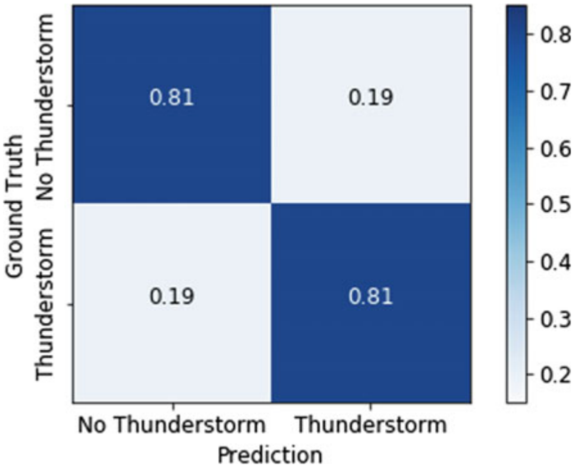


Fig. 6 Confusion matrix obtained with the DNN

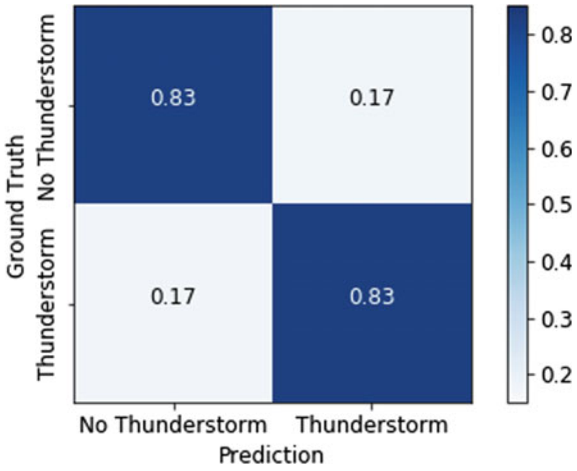
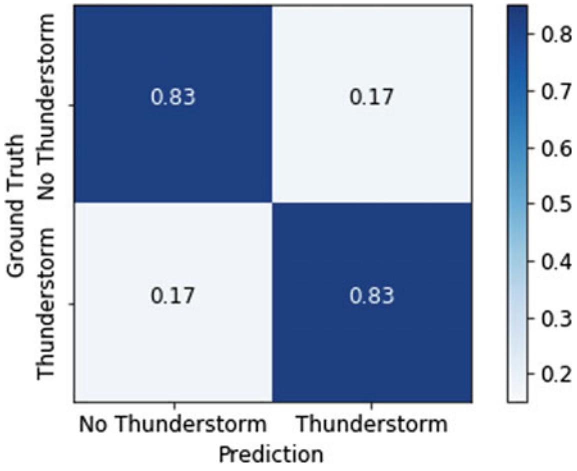


Fig. 7 Confusion matrix obtained with the RF



they require to adopt demanding identification algorithms. The training time is still limited for RF (0.66 min), and DNN (2.10 min), while it increases dramatically for the SVM (23.51 min), due to the well-known issue when applying the kernel trick on thousands of samples. The SVM is critical also when we switch to inference mode (i.e., predicting the outcome of a new sample); it takes more than 5 min to produce a new output. This issue is particularly critical in real-time applications, as the alert system that will be built starting from the model implemented in this work. Under this perspective, the other models (LR, DNN, and RF) are much more suitable for real-time applications, because they can predict a new output almost instantly.

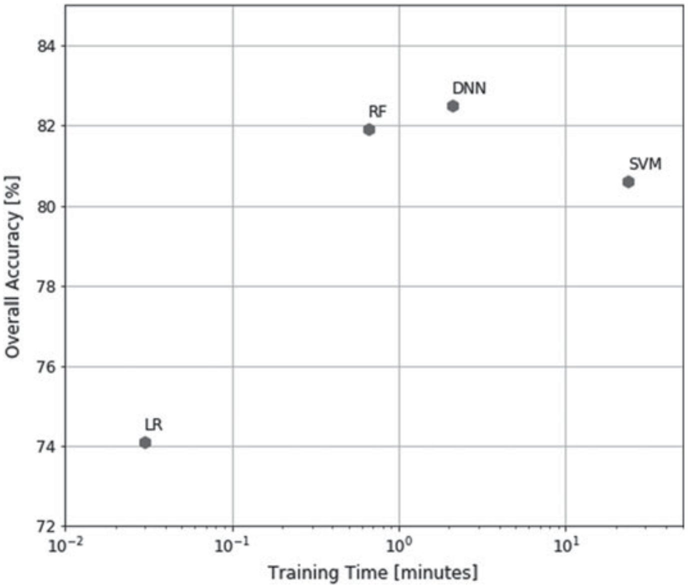


Fig. 8 Evaluation of the performances of the four algorithms in terms of overall accuracy (higher is better) and training time (lower is better). The four points are relative to the four considered algorithms

4 Conclusion

In this work, we showed how different machine learning techniques perform in the prediction of severe rain events. First, we briefly described the processes which occur behind the scenes in different techniques: a linear model, a kernel-based classifier, an artificial neural network, and a tree-based architecture. We then analyzed the performance of such models in terms of predictive power (using the ROC curves and the confusion matrices) and of computational effort required by the identification process.

The results showed that the accuracy of the three nonlinear models is definitely superior to that of the LR, reaching the maximum with DNN and RF. The fastest model to be calibrated is the LR, confirming again its benchmark capabilities in terms of velocity and easiness of implementation. LR predictive accuracy is lower than the one provided by the nonlinear competitors. This trend is due to the fact that the complex physical and chemical phenomena taking place in the atmosphere usually exhibit nonlinear behaviors.

The analysis that combines both the accuracy and the computational effort demonstrated that the SVM is a dominated solution: both the DNN and RF provide greater predictive power and require less time for the training process. The issue relative to the time required by the SVM is not limited to the training phase, but it strongly affects also the inference phase, and would probably limit the application of this technique for alert systems and other tasks which require to use it in real-time.

We can conclude that the most promising machine learning models to be used in the considered nowcasting meteorological application are the RF and the DNN. It should be pointed out that, among these two, the DNN shows higher capabilities in terms of customization (different architectures and hyper-parameters). In particular, implementing recurrent neural architectures would allow to explicitly take into account the temporal dimension of the process, further boosting the DNN predicting power.

Acknowledgements This work is part of the Lombardy-based Advanced Meteorological Predictions and Observations (LAMPO) project (http://www.geolab.polimi.it/projects/lampo-project). We would like to thank MeteoSwiss, in particular Dr. Luca Nisi, for providing the output data of the TRT algorithm.

References

1. Salman, A.G., Kanigoro, B., Heryadi, Y.: Weather forecasting using deep learning techniques. In: International Conference on Advanced Computer Science and Information Systems (ICACSIS), Depok, West Java, 10–11 October 2015, pp. 281–285 (2015)
2. Xingjian, S.H.I., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K., Woo, W.C.: Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In: Advances in Neural Information Processing Systems (NIPS), Montréal, Canada, 7–12 December 2015, pp. 802–810 (2015)
3. Hernández, E., Sanchez-Anguix, V., Julian, V., Palanca, J., Duque, N.: Rainfall prediction: A deep learning approach. In: International Conference on Hybrid Artificial Intelligence Systems (HAIS), Seville, Spain, 18–20 April 2015, pp. 151–162 (2016)
4. Gope, S., Sarkar, S., Mitra, P., Ghosh, S.: Early prediction of extreme rainfall events: a deep learning approach. In: Industrial Conference on Data Mining (ICDM), New York, USA, 18–20 July 2016, pp. 154–167 (2016)
5. Cramer, S., Kampouridis, M., Freitas, A.A., Alexandridis, A.K.: An extensive evaluation of seven machine learning methods for rainfall prediction in weather derivatives. Expert Syst. Appl. **85**, 169–181 (2017)
6. Davini, P., Bechini, R., Cremonini, R., Cassardo, C.: Radar-based analysis of convective storms over Northwestern Italy. Atmosphere **3**, 33–58 (2012)
7. Sangiorgio, M., Barindelli, S.: Spatio-temporal analysis of convective storms tracks in a densely urbanized Italian basin. ISPRS Int. J. of Geo-Inf. Stage of publication: under review
8. Barindelli, S., Realini, E., Venuti, G., Fermi, A., Gatti, A.: Detection of water vapor time variations associated with heavy rain in northern Italy by geodetic and low-cost GNSS receivers. Earth Planets Space **70**, 28 (2018)
9. De Haan, S.: Assimilation of GNSS ZTD and radar radial velocity for the benefit of very-short-range regional weather forecasts. Q. J. Roy. Meteorol. Soc. **139**, 2097–2107 (2013)
10. Dousa, J., Vaclavovic, P.: Real-time zenith tropospheric delays in support of numerical weather prediction applications. Adv. Space Res. **53**, 1347–1358 (2014)
11. Benevides, P., Catalão, J., Miranda, P.M.A.: On the inclusion of GPS precipitable water vapour in the Nowcasting of rainfall. Nat. Hazards Earth Syst. Sci. **15**, 2605–2616 (2015)
12. Trenberth, K.E.: Framing the way to relate climate extremes to climate change. Clim. Change **115**(2), 283–290 (2012)

13. Guerova, G., Dimitrova, T., Georgiev, S.: Thunderstorm classification functions based on instability indices and GNSS IWV for the Sofia Plain. Remote Sens. **11**(24), 2988 (2019)
14. Benevides, P., Catalão, J., Nico, G., Miranda, P.: Evaluation of rainfall forecasts combining GNSS precipitable water vapor with ground and remote sensing meteorological variables in a neural network approach. In: Remote Sensing of Clouds and the Atmosphere XXIII. International Society for Optics and Photonics, p. 1078607 (2018)
15. Benevides, P., Catalão, J., Nico, G.: Neural network approach to forecast hourly intense rainfall using GNSS precipitable water vapor and meteorological sensors. Remote Sens. **11**(8), 966 (2019)
16. Sangiorgio, M., Barindelli, S., Biondi, R., Solazzo, E., Realini, E., Venuti, G., Guariso, G.: Improved extreme rainfall events forecasting using neural networks and water vapor measures. In: Proceedings of the International conference on Time Series and Forecasting (ITISE), Granada, Spain, 25–27 September 2019, Vol. 2, pp. 820–826 (2019)
17. Mawandha, H.G., Kishimoto, M., Oishi, S.: GNSS-based PWV application for short term rainfall prediction in mountainous region. IOP Conf. Ser.: Earth Environ. Sci. **355**(1), 012070 (2019)
18. Manandhar, S., Lee, Y.H., Meng, Y.S.: GPS-PWV based improved long-term rainfall prediction algorithm for tropical regions. Remote Sens. **11**(22), 2643 (2019)
19. Manandhar, S., Dev, S., Lee, Y.H., Meng, Y.S., Winkler, S.: A data-driven approach for accurate rainfall prediction. IEEE Trans. Geosci. Remote Sens. **57**(11), 9323–9331 (2019)
20. Manandhar, S., Lee, Y.H., Meng, Y.S., Yuan, F., Ong, J.T.: GPS-derived PWV for rainfall nowcasting in tropical region. IEEE Trans. Geosci. Remote Sens. **56**(8), 4835–4844 (2018)
21. Liu, Y., Zhao, Q., Yao, W., Ma, X., Yao, Y., Liu, L.: Short-term rainfall forecast model based on the improved Bp–nn algorithm. Sci. Rep. **9**(1), 1–12 (2019)
22. Yao, Y., Shan, L., Zhao, Q.: Establishing a method of short-term rainfall forecasting based on GNSS-derived PWV and its application. Scientific Rep. **7**(1), 1–11 (2017)
23. Zhao, Q., Liu, Y., Ma, X., Yao, W., Yao, Y., Li, X.: An improved rainfall forecasting model based on GNSS observations. IEEE Trans. Geosci. Remote Sens. (2020)
24. Kouba, J., Héroux, P.: Precise point positioning using IGS orbit and clock products. GPS Solutions **5**(2), 12–28 (2001)
25. Kleijer, F.: Troposphere modeling and filtering for precise GPS leveling (2004)
26. Bevis, M., Businger, S., Herring, T.A., Rocken, C., Anthes, R.A., Ware, R.H.: GPS meteorology: remote sensing of atmospheric water vapor using the global positioning system. J. Geophys. Res.: Atmos. **97**(D14), 15787–15801 (1992)
27. Sato, K., Realini, E., Tsuda, T., Oigawa, M., Iwaki, Y., Shoji, Y., Seko, H.: A high-resolution, precipitable water vapor monitoring system using a dense network of GNSS receivers. J. Disaster Res. **8**(1), 37–47 (2013)
28. Brenot, H., Neméghaire, J., Delobbe, L., Clerbaux, N., De Meutter, P., Deckmyn, A., Delcloo, A., Frappez, L., Van Roozendael, M.: Preliminary signs of the initiation of deep convection by GNSS. Atmos. Chem. Phys. **13**(11), 5425–5449 (2013)
29. Choy, S., Wang, C., Zhang, K., Kuleshov, Y.: GPS sensing of precipitable water vapour during the March 2010 Melbourne storm. Adv. Space Res. **52**(9), 1688–1699 (2013)
30. Bonafoni, S., Biondi, R.: The usefulness of the Global Navigation Satellite Systems (GNSS) in the analysis of precipitation events. Atmos. Res. **167**, 15–23 (2016)
31. Rotach, M.W., Ambrosetti, P., Ament, F., Appenzeller, C., Arpagaus, M., Bauer, H.S., Behrendt, A., Bouttier, F., Buzzi, A., Corazza, M., Davolio, S.: MAP D-PHASE: Real-time demonstration of weather forecast quality in the Alpine region. Bull. Am. Meteorol. Soc. **90**(9), 1321–1336 (2009)
32. Hering, A.M., Morel, C., Galli, G., Sénési, S., Ambrosetti, P., Boscacci, M.: Nowcasting thunderstorms in the Alpine Region using a radar based adaptive thresholding scheme. In: Proceedings of the Third European Conference on Radar Meteorology (ERAD), Visby, Sweden, 6–10 September 2004, pp. 206–211 (2004)
33. Hering, A.M., Sénési, S., Ambrosetti, P., Bernard-Bouissières, I.: Nowcasting thunderstorms in complex cases using radar data. In: WMO Symposium on Nowcasting and Very Short Range Forecasting, Toulouse, France, 5–9 September 2005, vol. 2, no. 14 (2005)

34. Cox, D.: R: The regression analysis of binary sequences. J. Roy. Stat. Soc.: Ser. B (Methodol.) **20**(2), 215–232 (1958)
35. Cheng, C., Sa-Ngasoongsong, A., Beyca, O., Le, T., Yang, H., Kong, Z., Bukkapatnam, S.T.: Time series forecasting for nonlinear and non-stationary processes: a review and comparative study. IIE Trans. **47**(10), 1053–1071 (2015)
36. Mosavi, A., Ozturk, P., Chau, K.W.: Flood prediction using machine learning models: literature review. Water **10**(11), 1536 (2018)
37. Camporeale, E.: The challenge of machine learning in space weather: nowcasting and forecasting. Space Weather **17**, 1166–1207 (2019)
38. Aizerman, M.A.: Theoretical foundations of the potential function method in pattern recognition learning. Autom. Remote Control **25**, 821–837 (1964)
39. Cristianini, N., Shawe-Taylor, J.: An introduction to support vector machines and other kernel-based learning methods. Cambridge university press (2000)
40. Boser, B.E., Guyon, I.M., Vapnik, V.N.: A training algorithm for optimal margin classifiers. In: Proceedings of the Fifth Annual Workshop on Computational Learning Theory, pp. 144–152. ACM (1992)
41. Bengio, Y.: Learning deep architectures for AI. Found. Trends Mach. Learn. **2**(1), 1–127 (2009)
42. Goodfellow, I., Bengio, Y., Courville, A.: Convolutional networks. In: Dietterich, T. (ed.) Deep Learning. MIT Press, Cambridge, Massachusetts, London, England, pp. 321–359 (2016)
43. Ho, T. K.: Random decision forests. In: Proceedings of 3rd International Conference on Document Analysis and Recognition, vol. 1, pp. 278–282. IEEE (1995)
44. Biau, G., Scornet, E.: A random forest guided tour. Test **25**(2), 197–227 (2016)
45. Pancerasa, M., Sangiorgio, M., Ambrosini, R., Saino, N., Winkler, D. W., Casagrandi, R.: Can advanced machine learning techniques help to reconstruct barn swallows' long-distance migratory paths? In: Artificial Intelligence International Conference (A2IC), Barcelona, Spain, 21–23 November 2018, pp. 89–90 (2018)
46. Pancerasa, M., Sangiorgio, M., Ambrosini, R., Saino, N., Winkler, D.W., Casagrandi, R.: Reconstruction of long-distance bird migration routes using advanced machine learning techniques on geolocator data. J. R. Soc. Interface **16**(155), 20190031 (2019)
47. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al.: Scikit-learn: machine learning in python. J. Mach. Learn. Res. (2012)
48. Chollet, F.: Keras Documentation. (web: keras.io) (2015)
49. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S.: Tensorflow: large-scale machine learning on heterogeneous distributed systems (2016). arXiv preprint arXiv:1603.04467
50. Fawcett, T.: An introduction to ROC analysis. Pattern Recogn. Lett. **27**(8), 861–874 (2006)
51. Bradley, A.P.: The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recogn. **30**(7), 1145–1159 (1997)
52. Hand, D.J.: Measuring classifier performance: a coherent alternative to the area under the ROC curve. Mach. Learn. **77**(1), 103–123 (2009)

Long Short-Term Memory Networks for the Prediction of Transformer Temperature for Energy Distribution Smart Grids

F. J. Martinez-Murcia, J. Ramirez, F. Segovia, A. Ortiz, S. Carrillo, J. Leiva, J. Rodriguez-Rivero, and J. M. Gorriz

Abstract The near future of energy is shaped by a plethora of heterogeneous sources and growing demand. This poses new challenges for energy production and distribution, in which it will be essential that Medium Voltage/Low Voltage (MV/LV) distribution networks are planned, operated and monitored in a manner analogous to what transmission networks have been doing for decades. In this context, a precise tool for anticipating transformer overload and potential network problems is of paramount importance. Here, a system that can predict transformer temperature—a critical indicator of potential problems in a transformer—is key to the development of versatile and autonomous grid control strategies that enable more intelligent energy distribution. Understanding how this and other transformer measures relate is of fundamental importance to predict and prevent possible network failure. In this paper, we propose a transformer temperature prediction system based on long-term memory network (LSTM) that uses data from the previous 100 min to predict the transformer temperature for the next 100 min. The system is able to predict with a low error the temperature value using only the active power of the three transformer lines along with the ambient temperature. This makes it possible to discover trends towards anomalous temperature values in different transformers and act accordingly by planning a redistribution of the workload, avoiding possible incidents or service interruptions.

Keywords MV/LV · Temperature · Time-series prediction · LSTM · Regression · Energy distribution

F. J. Martinez-Murcia (✉) · A. Ortiz
Department of Communications Engineering, University of Malaga, Malaga, Spain
e-mail: fjmm@ic.uma.es

J. Ramirez · F. Segovia · J. Rodriguez-Rivero · J. M. Gorriz
Department of Signal Theory, Networking and Communications,
University of Granada, Granada, Spain

S. Carrillo · J. Leiva · J. Rodriguez-Rivero
Endesa Distribución, Madrid, Spain

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_21

1 Introduction

The future of energy is evolving towards a very heterogeneous scenario, in which new energy consumption and production patterns will emerge. In contrast to the traditional "one-size-fits-all" approach, new concepts in energy generation and transmission need to be designed to work in a diverse set of different scenarios and demand for energy will be increasingly diverse in its sources and uses, requiring new approaches to the transmission and distribution of electricity [1]. The way Medium Voltage/Low Voltage (MV/LV) distribution networks are planned, operated and monitored will evolve in an analogous way to what transport networks have been doing for decades. Here, the distributor goes from a mere distribution asset manager to being the operator of the network. This inevitably implies that the voltage levels are provided with much more intelligence than hitherto [2], involving a whole spectrum of digital technologies: sensors, local controllers, devices, supervision, smart metres, broadband communications, Control and Data Acquisition (SCADA) devices and Energy Management Centres (EMCs) that implement advanced data processing software, optimal control or workload prediction, among others.

Within this context, the Monitoring and Advanced Control (MONICA) initiative provided solutions for MV and LV distribution networks such a new state estimator of MV/LV networks. New initiatives like the Spanish Preventive Analysis of Smart Grid with real Time Operation and Renewable Assets Integration (PASTORA) project—a follow-up to MONICA—are key in order to advance in the development of flexible, reliable and efficient networks capable of absorbing the maximum renewable generation at the lowest cost. For this purpose, the project proposes, among others, the development of real-time information processing tools and analysis of historical series for prediction of possible device overload. In this context, a system capable of identifying patterns of anomalous behaviour of the network could act preventively with regard to incidents and breakdowns, improving the quality of service of the energy supplier.

Particularly, the treatment of historical data of the MONICA project could allow the system to predict anomalies in the distribution network, especially with regard to the temperature of transformers, one major key indicator of malfunctioning. The prediction of possible anomalies in the temperature of the transformers could help predict network failure, triggering a series of security containment protocols to prevent overload and optimize energy distribution in this context of heterogeneous power generation and demand. Thus, a series of actions have been directed towards the construction of an anomalous temperature early warning system (SATTA).

To do so, it is important to correctly characterize the variables that affect the network, especially the transformer temperature. Here, the Wiener-Granger causality (G-causality) [3–5] could provide relevant information about the flow of information between time-series variables that operate in the transformer. The G-causality is defined by two assumptions: (i) a cause occurs before its effect and (ii) the knowledge of a cause improves the prediction of its effect. It was originally developed for Auto-Regressive (AR) modelling of stochastic processes by means of a statistical description of fused observed responses. This could help us to identify the perfect candidate variable that G-cause the variations in transformer temperature, allowing for a better modelling.

In this regard, there exist a vast literature of time-series prediction algorithms, ranging from classical Auto-Regressive (AR) [6] methods to complex machine learning regression techniques like Support Vector Machines (SVM) [7, 8]. The current wave of neural network architectures has revolutionized the classification and regression paradigm [9, 10], with many applications in fields such as image recognition [10], generative models [11] or biomedical image analysis [12], among others. Within this context, the recent advances in recurrent neural networks—networks with feedback links—have paved the way for newer applications in time-series analysis and prediction using either Convolutional Neural Networks (CNNs) [13] or the Long Short-Term Memory (LSTM) [14] cells, which have experienced a major growth in the last years with many applications in, among others, stock market prediction [15], speech recognition [16, 17] or even music composition [18].

In this paper, we propose a recurrent neural network architecture based on LSTMs in order to predict temperature levels of a transformer from a series of power and temperature variables. In Sect. 2, we propose the methodology that combines feature selection via covariance matrices and LSTM networks for prediction, as well as the dataset used. In Sect. 3, we describe the evaluation procedure and present and analyse the results. Finally in Sect. 4, we draw some conclusions about the proposed system.

2 Data and Methodology

2.1 Data Acquisition

Data used in the preparation of this article was provided by ENDESA, the largest electric utility company in Spain. It was obtained during the MONICA (acronym for Advanced Monitoring and Control) project, with the fundamental objective of developing a technology that allows real-time monitoring and diagnosis of medium and low voltage distribution networks, with an approach similar to that which has traditionally existed in transmission networks (high voltage). The data consists of yearly acquisitions of different variables at the transformation centres in southern Spain. It comprises a large and variable number of measures, including active and reactive power delivered by the transformer, reactive, capacitive and inductive energy, intensity, phase, voltage and temperatures.

In this work, we use the 16 transformers which provide information about Transformer Temperature (TT), a critical variable to measure potential incidences and anomalous behaviour, including transformer overload. Since the signals were recorded with non-uniform period, the data was subsequently resampled to 12 samples/hour (or a time-step τ of 5 min), corresponding to the mode of the data distribution. A simple linear interpolation between consecutive samples was used for this procedure.

2.2 Wiener-Granger Causality Analysis

The Wiener-Granger causality [3] is used in this work to model the G-causal relationships between different transformer variables, following the methodology in [5]. It is based on the principles of cause-before-effect and that the addition of a cause improves the prediction ability of an outcome. In a nutshell, a variable X is said to G-cause a variable Y if the past of X fused with the one of Y helps predict the future of Y more accurately than only using the past of Y.

Let us note $\mathbf{X} = [X_1, X_2, \ldots]$ and $\mathbf{Y} = [Y_1, Y_2, \ldots]$, two jointly distributed vector stochastic processes. **Y** G-causes **X** if and only if **X**, conditional on its own past, is dependent of the past of **Y**. This can be easily interpreted if the prediction of future values of **X** based on its on past can be improved when using past values of **Y**. That is what we will consider "causality", as in many other examples on the bibliography [19].

The G-causality assumes a p-th order VAR model for the underlying processes, from which $\mathbf{u}$ of length m is a realization of a discrete-time stationary vector stochastic process $\mathbf{U}_1, \mathbf{U}_2, \ldots$. The model can be therefore defined as

$$\mathbf{U}_t = \sum_{k=1}^{p} \mathbf{A}_k \mathbf{U}_{t-k} + \epsilon_t \tag{1}$$

where the real-valued matrices $\mathbf{A}_k$, of size $n \times n$, are the regression coefficients, and ϵ_t are the error terms, a n-dimensional iid stochastic process. $\mathbf{A}_k$ and ϵ_t are assumed to be time-independent (stationarity).

The time-domain unconditional G-causality is based on the VAR model described before. There, the G-causality from one to another jointly distributed multivariate processes $\mathbf{U}_{1,t}$ to $\mathbf{U}_{2,t}$ can be defined as the improvement in the prediction of $\mathbf{U}_{1,t}$ when the past of $\mathbf{U}_{2,t}$ is included by early information fusion in the VAR model. We can note this causality as $F_{\mathbf{U}_{2,t} \to \mathbf{U}_{1,t}}$. This causality uses the restricted VAR model of Eq. 1 [4] and evaluates it on the process $\mathbf{U}_{1,t}$ with an extended model:

$$\mathbf{U}_{1,t} = \sum_{k=1}^{p} \mathbf{A}'_{1,k} \mathbf{U}_{1,t-k} + \sum_{k=1}^{p} \mathbf{A}'_{21,k} \mathbf{U}_{2,t-k} + \epsilon'_{1,t} \tag{2}$$

where the residuals of the covariance matrix are

$$\Sigma(\epsilon'_{1,t}) \equiv \mathrm{Cov}\left(\epsilon'_{1,t}\right) \tag{3}$$

and $\mathbf{A}_{21,k}$ contains the dependence of $\mathbf{U}_{1,t}$ on the past of $\mathbf{U}_{2,t}$ given its own past.

2.3 Recursive Neural Networks

Although there exist many examples of time-series processing using neural networks such as Restricted Boltzmann Machines or CNNs [13, 20], Recursive Neural Networks (RNNs) are the state of the art for time-series prediction and analysis. RNNs are a subtype of neural architectures specially designed for temporal processing, in which some type of memory or “state” is held within the network. Recursive means that unlike typical feedforward networks [12, 21], it has feedback connections. They are usually arranged in “cells” that hold some memory of the past events in order to provide activations. Network architectures based on Long Short-Term Memory cells (LSTM) or Gated Recurrent Units (GRU) are becoming commonplace in applications such as stock market prediction [15], speech recognition [16, 17] or even music composition [18].

2.3.1 Long Short-Term Memory Cell

The Long Short-Term Memory (LSTM) [14] is a recurrent architecture. The architecture contains a memory activated via a “forget” gate that, together with an input and output gates, regulates the flow of the information and whether they are relevant for the output or not. A schema of an LSTM cell is shown in Fig. 1.

LSTM networks are particularly good for the analysis and prediction of time-series data. Within the architecture proposed in Fig. 1, the equations that govern the behaviour of the unit can be summarized as follows:

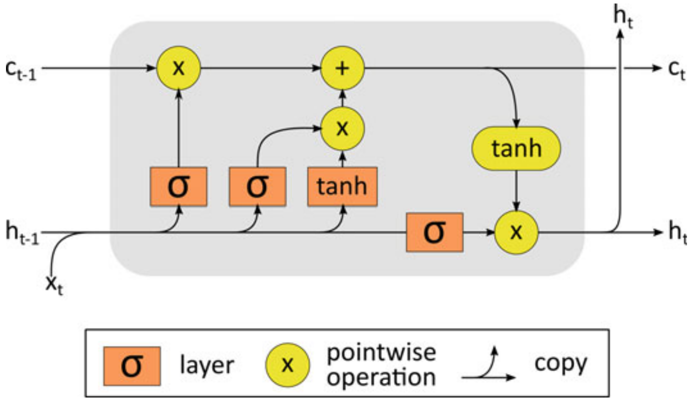


Fig. 1 Structure of a Long Short-Term Memory (LSTM) cell. Refer to the legend for understanding the layers and operations applied

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \tag{4}$$

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \tag{5}$$

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \tag{6}$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \tag{7}$$

$$h_t = o_t \circ \sigma_h(c_t) \tag{8}$$

where $\sigma_g(x)$ and $\sigma_c(x)$ are the sigmoid and hyperbolic tangent activation functions. $x_t \in \mathbb{R}^d$ and $h_t \in \mathbb{R}^h$ are the input and output (also known as hidden state) vectors of the LSTM unit, of length d and h, respectively, $f_t \in \mathbb{R}^h$ the forget gate's activation vector, $i_t \in \mathbb{R}^h$ the input gate's activation vector, $o_t \in \mathbb{R}^h$ the output gate's activation vector, $c_t \in \mathbb{R}^h$ the cell state vector and $W \in \mathbb{R}^{h\times d}$, $U \in \mathbb{R}^{h\times h}$ and $b \in \mathbb{R}^h$ the weight and bias parameters of the different layers implemented in each gate.

2.3.2 Network Architecture

In this work, we used an LSTM network composed of two LSTM cells of 100 and 50 units connected to a dense layer of 20 units. The network is trained with different combinations of the candidate variables in the last τ, and the output is intended to predict TT with a maximum lapse of 20τ. The predicted TT is therefore in a range between 5 and 100 min from the current instant, fed by the variable data of the 100 min previous to the current instant.

3 Results and Discussion

3.1 Evaluation

Our system is trained with the data of each of the 15 available transformers, using the selected variables over the 20 previous time steps. The trained system is then tested with two different approaches:

- Predictive ability **on the training transformer**: A time-series cross-validation (CV) [22, 23], in which the time series is divided into progressive sequential batches (see Fig. 2) and all previous batches are used to predict the next one, is used to estimate the performance.
- Generalization ability **over other transformers** (GEN): In this case, the system is trained with one transformer, and then the predictive ability over other transformer's data is tested.

As for the transformer variables used, we provide a new set of candidate variables in Sect. 2.2, according to criteria of G-causality. We always use the values between -1 and -20 ts from $t = 0$ for the prediction at $t = 0$. The Root-Squared Mean

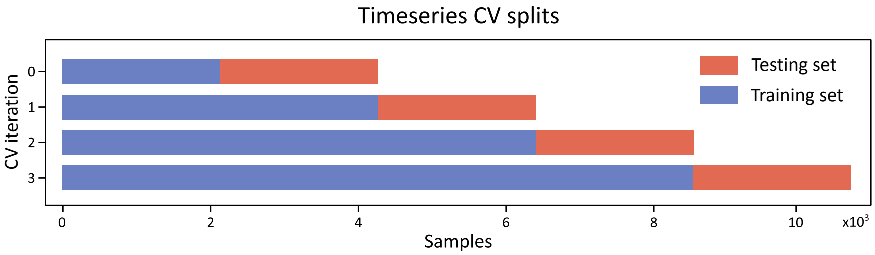


Fig. 2 Example of a time-series fourfold cross-validation split

Error (RMSE) and its standard deviation (when several values are aggregated, e.g. within folds or over different transformers) are provided as to measure the quality of prediction of the model.

3.2 *Candidate Variables*

In order to estimate a new set of candidate variables for the prediction of TT, we estimate the G-causality in the different transformer datasets. By doing this, we obtain the results shown in Fig. 3. There we see that the TT is mainly related to the reactive

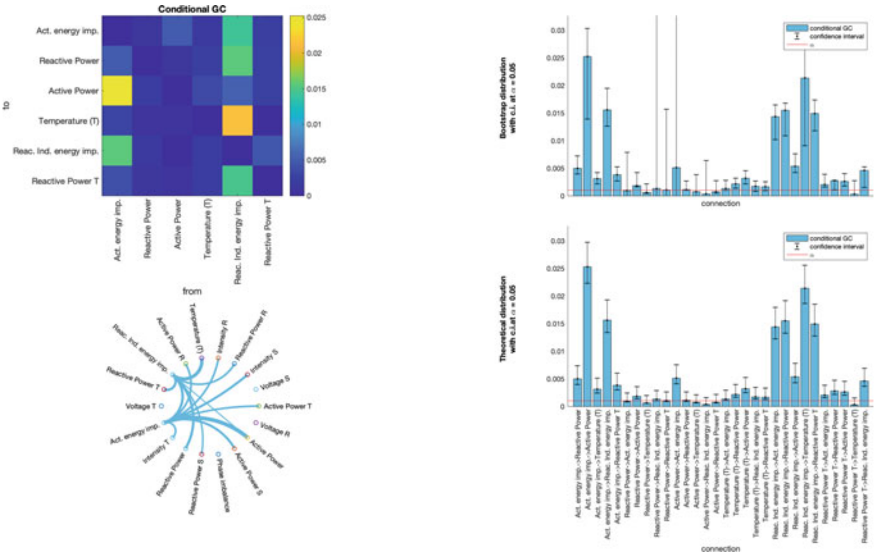


Fig. 3 Granger causality analysis for the variables of interest related to Transformer Temperature (TT)

inductive energy. However, this variable is not present in all the transformers on the database. However, this variable is strongly dependent on the reactive and active powers (PR and PA), and in the active energy (EA). When looking at the available variables in all transformers of the dataset, we reduced the candidate variables to two: reactive power (PR) and active power (PA), because not in all cases we obtained the three-line powers separately and the active energy is not always provided.

We also included the Ambient Temperature (TA) present in the subset of transformers used in this article, given that they help in modelling the low-frequency (days or even month) variations of the TT. Since this can only be obtained through a more complete measurement of ambient temperatures, it is important to include this in our modelling. So the final set of candidate variables is composed of PR, PA and TA.

3.3 Prediction Results

Using different combinations of the candidate variables, we trained a temperature LSTM model using each of the 16 transformers. We used the last 20τ to predict the next 20τ, equivalent to a period that ranges from 5 to 100 min. The performance, within the two experiments described in Sect. 3.1 (cross-validation and generalization ability), is shown in Table 1.

Two main trends can be shown in this table. First, the tendency that the generalization error is always higher than the CV error. This is coherent which what was expected, since there are differences in the location of the different transformers, and there exists also a high heterogeneity in the equipment used to measure, something that is to be unified in the PASTORA project. TRF-1 and TRF-3 are clearly outliers in both experiments, as it can be seen for an extremely low or high CV error (for TRF-3 and TRF-1, respectively) and an anomalous GEN error over the rest of transformers.

When comparing the models that use TA and those that do not, there is a clear improvement in the former. The TA prediction models achieved a significantly better performance when compared to the baseline, suggesting that TA is indeed enhancing its performance. This is even more clear when looking at the AVG* performance. A closer look at the predictions of the model (see Fig. 4) reveals a possible explanation. Whereas the PA or PR is enough to model the high-frequency (day-level) variations of temperature in the transformers, and possible peaks due to malfunctioning, a major, long-term contribution to the temperature of the transformer seems to be the weather conditions at the specific location of the transformer. That may explain why the higher frequency variations are correctly modelled in the left figure (only PA), but the lower frequency trends of the TA allow our system to provide a much better prediction, even when generalizing to another transformer (trained with TRF-9, predicting TRF-2).

There is almost no difference in the prediction error when varying the prediction steps. Figure 5 shows that the error increases with τ within the CV experiment (above) as it may be expected, but this effect is far less evident in the GEN experiment (below).

Table 1 RMSE and average RMSE (AVG) results for all the experiment and transformers, using different combinations of the input features (PA, PR and TA)

Exp.	TRF	PA	PA + TA	PR	PR + TA	PA + PR	PA + PR + TA
CV	TRF-0	3.76 (1.09)	1.14 (0.59)	3.78 (0.85)	1.36 (0.69)	3.77 (0.78)	1.23 (0.60)
	TRF-1	5.56 (6.33)	4.78 (5.59)	5.74 (6.76)	4.49 (5.14)	5.69 (6.68)	4.95 (5.77)
	TRF-2	3.40 (0.66)	1.22 (0.74)	2.97 (0.63)	1.34 (0.64)	2.81 (0.52)	1.38 (0.82)
	TRF-3	**0.76 (1.41)**	**0.80 (1.40)**	**0.73 (1.43)**	**0.72 (1.43)**	**0.74 (1.42)**	**0.74 (1.42)**
	TRF-4	3.70 (0.79)	1.10 (0.68)	2.97 (0.53)	1.54 (0.75)	3.32 (0.71)	1.57 (1.28)
	TRF-5	3.84 (1.25)	1.83 (1.25)	4.11 (1.25)	2.37 (0.77)	3.88 (1.14)	2.36 (1.25)
	TRF-6	4.88 (1.71)	2.67 (1.33)	4.06 (0.57)	2.67 (1.07)	4.33 (1.18)	2.95 (1.45)
	TRF-7	3.45 (1.01)	2.08 (1.13)	3.25 (0.95)	2.15 (1.14)	3.15 (1.17)	2.26 (1.24)
	TRF-8	4.16 (0.99)	1.78 (0.89)	4.15 (0.86)	1.76 (0.69)	3.97 (0.94)	1.92 (0.97)
	TRF-9	4.00 (1.10)	1.47 (1.01)	4.42 (1.02)	1.40 (1.00)	4.15 (0.86)	1.56 (1.04)
	TRF-10	3.73 (1.12)	1.76 (1.29)	3.01 (1.06)	1.98 (1.11)	2.97 (0.99)	1.82 (1.36)
	TRF-11	3.89 (1.12)	1.68 (1.04)	3.87 (1.23)	2.11 (0.78)	3.75 (1.21)	1.83 (1.24)
	TRF-12	3.56 (1.17)	2.04 (1.02)	3.70 (1.40)	2.70 (0.67)	3.65 (1.32)	2.51 (1.39)
	TRF-13	3.34 (0.90)	1.40 (0.68)	3.51 (0.84)	1.81 (0.51)	3.06 (0.88)	1.70 (0.81)
	TRF-14	3.91 (0.98)	2.13 (1.29)	3.77 (1.28)	2.17 (1.33)	3.66 (1.19)	2.17 (1.41)
	TRF-15	3.93 (1.13)	2.15 (1.22)	3.48 (0.54)	2.18 (1.19)	3.43 (0.68)	2.41 (1.28)
	*AVG**	*3.82 (1.15)*	*1.75 (1.12)*	*3.65 (1.07)*	*1.97 (1.01)*	*3.56 (1.08)*	*1.98 (1.26)*
GEN	TRF-0	7.49 (3.59)	4.11 (2.60)	7.25 (3.05)	4.15 (3.25)	7.48 (4.04)	4.19 (2.63)
	TRF-1	13.87 (4.20)	8.04 (2.18)	14.10 (4.32)	6.41 (1.90)	14.09 (4.29)	10.29 (2.56)
	TRF-2	7.21 (4.44)	4.28 (3.78)	6.95 (4.40)	4.38 (3.49)	7.38 (4.68)	5.56 (4.34)
	TRF-3	16.08 (3.36)	13.73 (2.69)	10.97 (7.48)	10.15 (6.64)	14.27 (3.48)	10.71 (4.13)
	TRF-4	7.41 (5.02)	4.20 (3.67)	7.06 (5.17)	4.20 (2.52)	7.55 (5.26)	4.42 (3.94)
	TRF-5	8.20 (5.97)	6.18 (5.72)	8.72 (6.06)	4.69 (4.36)	9.08 (5.91)	7.66 (6.06)
	TRF-6	7.15 (2.98)	5.54 (1.64)	7.49 (2.99)	6.06 (1.55)	7.05 (2.92)	5.69 (1.63)
	TRF-7	8.48 (5.84)	7.60 (5.93)	8.42 (6.32)	8.35 (6.34)	8.89 (7.26)	8.45 (6.38)
	TRF-8	7.58 (5.25)	4.93 (4.26)	7.35 (5.50)	4.55 (4.02)	7.47 (5.38)	5.11 (4.43)
	TRF-9	**6.97 (4.17)**	**4.09 (3.22)**	**6.94 (4.80)**	**4.07 (3.00)**	**7.02 (5.36)**	**4.15 (3.20)**
	TRF-10	8.10 (5.79)	4.93 (4.89)	8.36 (6.99)	5.65 (5.38)	8.49 (6.81)	5.35 (4.91)
	TRF-11	8.39 (6.01)	5.79 (5.35)	8.38 (6.68)	4.65 (4.22)	8.44 (6.24)	6.76 (5.59)
	TRF-12	9.57 (6.97)	7.18 (6.05)	8.74 (6.49)	6.54 (5.40)	9.36 (6.71)	8.31 (6.23)
	TRF-13	7.70 (5.26)	5.66 (5.50)	7.71 (5.58)	5.76 (5.18)	7.88 (5.57)	6.16 (5.55)
	TRF-14	8.38 (2.48)	5.98 (1.70)	9.36 (2.59)	5.91 (1.73)	8.94 (2.56)	5.89 (1.69)
	TRF-15	7.60 (5.23)	5.07 (4.56)	7.27 (5.57)	5.18 (4.81)	7.49 (6.01)	5.54 (5.05)
	*AVG**	*7.87 (5.11)*	*5.40 (4.56)*	*7.86 (5.38)*	*5.30 (4.33)*	*8.04 (5.55)*	*5.95 (4.85)*

**AVG: Average RMSE excluding TRF-1 and TRF-3*

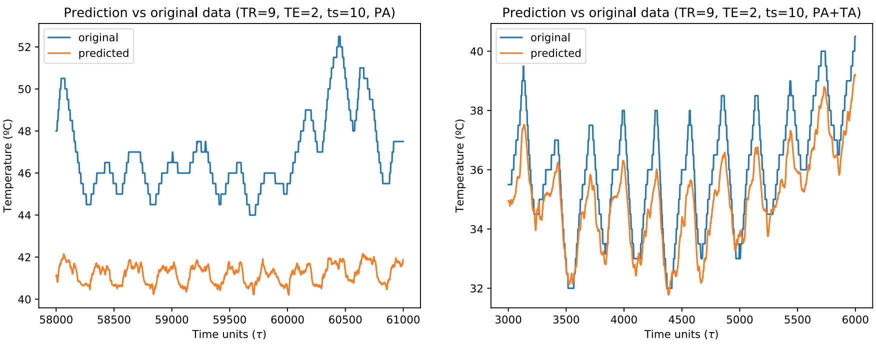


Fig. 4 Prediction of the time series of transformer 0 test set and extrapolation of the model to TRF-2, when using a time-step prediction of 10 (50 min)

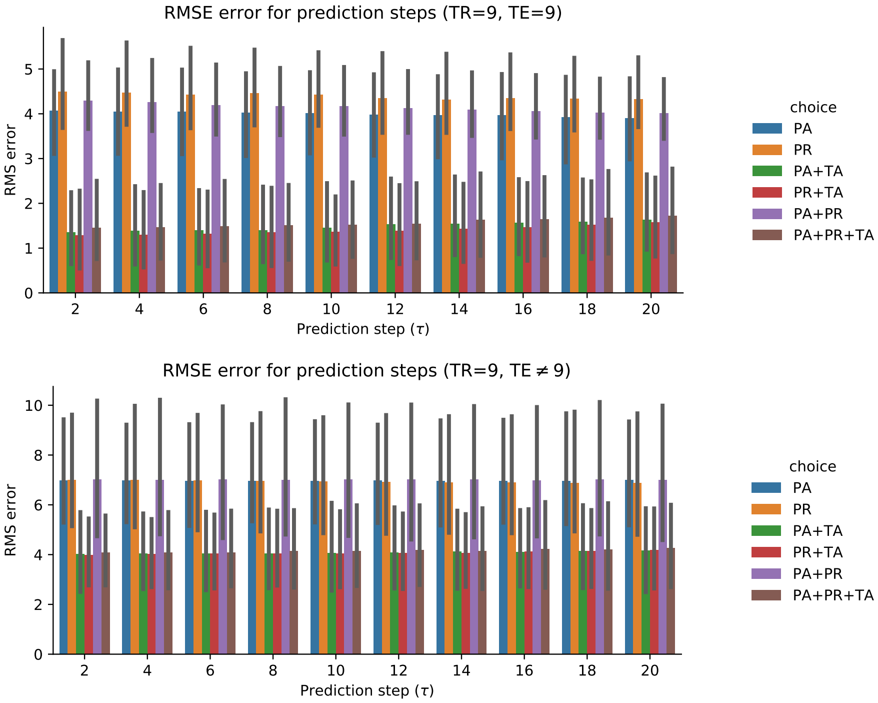


Fig. 5 RMSE error for the different prediction steps when trained and tested in transformer 9 (above, using TS-CV) and when extrapolating the model fitted with TRF = 9 to all remaining transformers (below)

Furthermore, the differences between the estimation error between more and less τ are far smaller than the differences between the choice of variables, especially when taking into account the TA.

From Fig. 5 and Table 1, and once we have used TA to correct general temperature trends related to climate conditions, PA and PR are both useful for modelling high-frequency components. However, there are evidence that these two variables can be correlated, as it could also be observed in Fig. 3. It is also supported by the fact that the PA + TA seems to provide the best results. PR + TA for its part, and PR + PA + TA perform similarly, however, but there seems to be no further advantage in including these variables. It could therefore be concluded that it would be indistinct to use PR or PA in order to model the high-frequency components, and it would be enough to use one of these two variables without any loss of accuracy in the predictions.

Regarding the prediction interval, we can observe that those performed in an interval smaller than 10τ are accurate enough, which is equivalent to 50 min. This confirms that in the case of an abnormal temperature rise caused by abnormal power functioning, our system could provide early warnings that could help reconfigure the energy distribution network in order to avoid a transformer overload and subsequent problems.

This model makes it possible to predict with a low error the temperature value using only the PA variables of the three lines, together with the ambient temperature at this time, at least 50 min in advance. This would make it possible to discover trends towards anomalous temperature values in different transformers and act accordingly by planning a redistribution of the workload, avoiding possible incidents or service interruptions.

4 Conclusions

The near future of energy is confirmed by a plethora of heterogeneous sources as well as an increasing demand, with the main focus being on the possibility of solar energy and renewable energy generation with new technologies, for example, electric cars, smart grids, etc. This poses a challenge for the supply, transmission and utilization of energy in a flexible and competitive environment. Energy efficiency, environmental sustainability and economic viability are some of the considerations when building energy efficiency technologies and services, and in that context, it will be essential that Medium Voltage/Low Voltage (MV/LV) become smart distribution grids. This work tackles the problem of predicting the transformer temperature at each node of these energy networks, a fundamental tool in the development of reliable and sustainable energy systems. In this work, we have taken advantage of LSTM networks, a very recent advance in the neural network field, from which we could predict with a low error the temperature value using only the active power of the three lines at each transformer, together with the ambient temperature at every instant. A prediction up to 100 min—using the last 100 min—was possible with a small RMSE, proving the ability of this architecture that can discover trends towards anomalous temperature values in different transformers and act accordingly by planning a redistribution of the workload, avoiding possible incidents or service interruptions.

Acknowledgements This work was supported by the PASTORA: PREVENTIVE ANALYSIS OF INTELLIGENT NETWORKS IN REAL TIME AND INTEGRATION OF RENEWAL RESOURCES (ITC-20181102) funded by the Centre for Industrial Technological Development (CDTI), supported by the Ministry of Science, Innovation and Universities and co-financed by the European Union with ERDF funds through the "Plurirregional Operational Programme of Spain 2014–2020". It was also partially supported by the MINECO/ FEDER under the TEC2015-64718-R, RTI2018-098913-B-I00 and PGC2018-098813-B-C32 projects. Work by F.J.M.M. was supported by the MICINN "Juan de la Cierva" Fellowship.

References

1. Stetz, T., Marten, F., Braun, M.: Improved low voltage grid-integration of photovoltaic systems in Germany. IEEE Trans. Sustain. Energy **4**(2), 534–542 (2013)
2. Kahrobaeian, A., Mohamed, Y.A.I.: Analysis and mitigation of low-frequency instabilities in autonomous medium-voltage converter-based microgrids with dynamic loads. IEEE Trans. Ind. Electron. **61**(4), 1643–1658 (2014)
3. Wiener, N.: The theory of prediction. In: Beckenbach, E. F. (ed.), pp. 165–190. McGraw Hill, New York, NY, USA (1956)
4. Barrett, A.B., Barnett, L., Seth, A.K.: Multivariate granger causality and generalized variance. Phys. Rev. E **81**(4) (2010)
5. Rodriguez, J., Ramirez, J., Martinez, F.J., Segovia, F., Ortiz, A., Salas, D., Castillo, D., Puntonet, C.G., Leiva, F.J., Carillo, S., Consortium, P., Gorriz, J.M.: Granger causality-based information fusion applied to electrical measurements from power transformers. Inf. Fusion (2019)
6. Patil, S.L., Tantau, H.J., Salokhe, V.M.: Modelling of tropical greenhouse temperature by auto regressive and neural network models. Biosyst. Eng. **99**(3), 423–431 (2008)
7. Chang, B.R., Tsai, H.F.: Forecast approach using neural network adaptation to support vector regression grey model and generalized auto-regressive conditional heteroscedasticity. Expert Syst. Appl. **34**(2), 925–934 (2008)
8. Sharma, N., Sharma, P., Irwin, D., Shenoy, P.: Predicting solar generation from weather forecasts using machine learning. In: 2011 IEEE International Conference on Smart Grid Communications (SmartGridComm), pp. 528–533, Oct 2011
9. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)
10. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, vol. 25, pp. 1097–1105. Curran Associates, Inc. (2012)
11. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, vol. 27, pp. 2672–2680. Curran Associates, Inc. (2014)
12. Martinez-Murcia, F.J., Ortiz, A., Gorriz, J.M., Ramirez, J., Castillo-Barnes, D.: Studying the manifold structure of Alzheimer's disease: a deep learning approach using convolutional autoencoders. IEEE J. Biomed. Health Inform
13. Lawhern, V.J., Solon, A.J., Waytowich, N.R., Gordon, S.M., Hung, C.P., Lance, B.J.: EEGnet: a compact convolutional neural network for EEG-based brain–computer interfaces. J. Neural Eng. **15**(5), 056013 (2018)

14. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
15. Chen, K., Zhou, Y., Dai, F.: A LSTM-based method for stock returns prediction: a case study of China stock market. In: 2015 IEEE International Conference on Big Data (Big Data), pp. 2823–2824, Oct 2015
16. Graves, A., Mohamed, A., Hinton, G.: Speech recognition with deep recurrent neural networks. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6645–6649, May 2013
17. Sak, H., Senior, A., Beaufays, F.: Long short-term memory recurrent neural network architectures for large scale acoustic modeling. 5
18. Eck, D., Schmidhuber, J.: Learning the long-term structure of the blues. In: Dorronsoro, J.R. (ed.) Artificial Neural Networks—ICANN 2002. Lecture Notes in Computer Science, pp. 284–289. Springer, Berlin, Heidelberg (2002)
19. Seth, A.K.: A matlab toolbox for granger causal connectivity analysis. J. Neurosci. Methods **186**(2), 262–273 (2010)
20. Längkvist, M., Karlsson, L., Loutfi, A.: A review of unsupervised feature learning and deep learning for time-series modeling. Pattern Recognit. Lett. **42**, 11–24 (2014)
21. Martinez-Murcia, F.J., Gorriz, J.M., Ramirez, J., Ortiz, A.: Convolutional neural networks for neuroimaging in Parkinson’s disease: is preprocessing needed? Int. J. Neural Syst. 1850035 (2018)
22. Hart, J.D.: Automated kernel smoothing of dependent data by using time series cross-validation. J. R. Stat. Soc. Ser. B (Methodological) **56**(3), 529–542 (1994)
23. Arlot, S., Celisse, A.: A survey of cross-validation procedures for model selection. Stat. Surv. **4**, 40–79 (2010)

Deep Multilayer Perceptron for Knowledge Extraction: Understanding the *Gardon de Mialet* Flash Floods Modeling

Bob E. Saint Fleur, Guillaume Artigue, Anne Johannet, and Séverin Pistre

Abstract Flash floods frequently hit Southern France and cause heavy damages and fatalities. To enhance persons and goods safety, official flood forecasting services in France need accurate information and efficient models to optimize their decisions and policy in crisis management. Their forecasting is a serious challenge as heavy rainfalls that cause such floods are very heterogeneous in time and space. Such phenomena are typically nonlinear and more complex than classical flood events. This analysis had led to consider complementary alternatives to enhance the management of such situations. For decades, artificial neural networks have been proved very efficient to model nonlinear phenomena, particularly rainfall-discharge relations in various types of basins. They are applied in this study with two main goals: first, modeling flash floods on the *Gardon de Mialet* basin (Southern France); second, extract internal information from the model by using the KnoX: knowledge extraction method to provide new ways to improve models. The first analysis shows that the kind of nonlinear predictor strongly influences the representation of information, e.g., the main influent variable (rainfall) is more important in the recurrent and static models than in the feed-forward one. For understanding "long-term" flash floods genesis, recurrent and static models appear thus as better candidates, despite their lower performance. Besides, the distribution of weights linking the exogenous variables to the first layer of neurons is consistent with the physical considerations about spatial distribution of rainfall and response time of the hydrological system.

Keywords Neural networks · Flash floods · Knowledge extraction · Deep learning

B. E. Saint Fleur · G. Artigue (✉) · A. Johannet
LGEI, IMT Mines Alès, Alès, France
e-mail: guillaume.artigue@mines-ales.fr

B. E. Saint Fleur · S. Pistre
Hydrosciences Montpellier, University of Montpellier, CNRS, IRD, 34090 Montpellier, France

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_22

1 Introduction

In the Mediterranean regions, flash floods due to heavy rainfalls frequently occur and cause numerous fatalities and costly damages. During the last few years, Southern France has been particularly exposed to these catastrophic events. In such cases, in only one event, there can be more than 20 fatalities and damages that can reach more than one billion euros, in only one event [1]. Facing these issues, authorities need reliable forecasts for early warning purposes. Unfortunately, both the short-term rainfall forecasts and the processes leading to the discharge response remain poorly known at the space and time scales required. It is thus difficult to provide forecasts using the traditional coupling between a meteorological model and a physically based hydrological model.

Artificial neural networks therefore appear as an alternative paradigm as they are able to provide forecasts of an output (discharge) without making any other hypothesis on the system than the causality between rainfall and discharge. Artificial neural networks have been applied in a wide variety of domains, as they are essentially based on data and training [2]. They appear as particularly suitable for identifying the generating processes in hydrological time series because of their ability to model nonlinear dynamic systems [3, 4]. However, due to their statistical origin, it is difficult to associate meaning to their internal parameters, and they are rightly considered as black-box models. For this reason and to enhance the understanding of the behavior of both the model and the physical processes, several works have been done to bring more transparency in the operating mode and introduced concepts of gray-box and transparent-box models [5, 6]. Some other works have been conducted to make neural network models more hydrologically meaningful [6–8].

2 Materials and Methods

2.1 Study Area: Location and General Description

The *Gardon de Mialet* basin covers 220 km^2 in Southern France. It is part of the *Cévennes* range, which is known as a preferential location for the well-known meteorological phenomenon named “cevenols episodes” (Fig. 1). These episodes consist in short duration (less than 2 days) very heavy rainfall events.

The elevation of the *Gardon de Mialet* basin ranges from 150 to 1170 m.a.s.l., and its mean slope is about 33%. As for the most of basins of the *Cévennes*, these characteristics lead to limited deep infiltration or deep underground flow, and thus to a high drainage density. Its response time is relatively short: between 2 and 4 h [4]. The area is dominated by a metamorphic formation with 95% of mica-schist and gneiss, which leads to a poorly porous and impermeable rocky sub-soil. The land use is almost homogeneous while covered by natural vegetation (chestnut trees, conifers, mixed forest, and bush) for 92%. The rest is shared between rocks and urban areas.

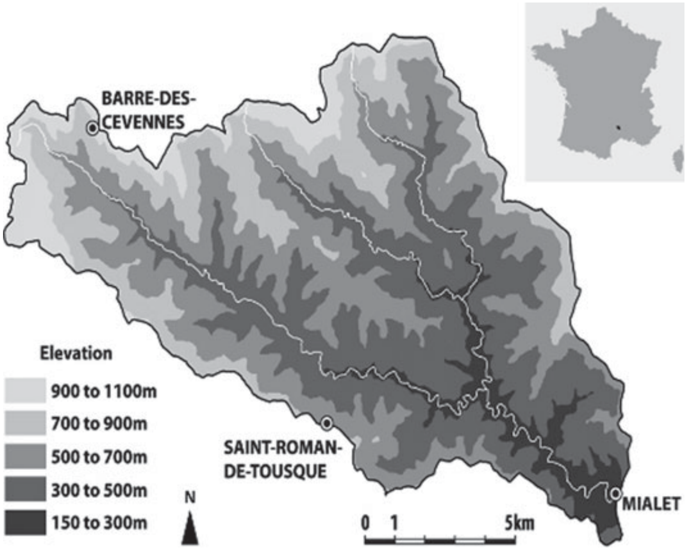


Fig. 1 The study area (Artigue 2012)

Typically, in Mediterranean regions, heavy rainfalls sometimes exceed 500 mm in only 24 h, to be compared to the 600 mm that falls on Paris annually. They are mainly produced by convective events, triggered either by relief, by a wind convergence, or by both. For example, in September 2002, the Gard (France) department has registered 687 mm of rainfall in 24 h with 137 mm in only one hour at *Anduze* (a few kilometers distant from *Mialet*).

2.2 Database

The database used in this study is essentially compounded with hourly observations from 1992 to 2002, and 5 min time-step observations from 2002 to 2008, on three rain gauges and one hydrometric station at the outlet at Mialet (Fig. 1). From upstream to downstream, these stations are BDC (Barre-des-Cévennes), SRDT (Saint-Roman de Tousque), and Mialet which coincides with the discharge station. They are all managed by the local Flood Forecasting Service (SPC Grand Delta). 58 events were extracted at 30 min time step (based on rainfall events having at least 100 mm accumulation in 48 h on any of the rain gauges). Data description is synthesized in Tables 1 and 2.

Table 1 Data description

	Rainfall (mm)			Discharge	
	BDC	SRDT	Mialet	(m^3 s^{-1})	(m^3 s^{-1} km^{-2})
Maximum (30 min)	33.3	41.8	62.0	819.3	3.72
Median (30 min)	0.3	0.3	0.2	29.3	0.13
Moy	1.0	1.3	1.2	43.4	0.20
Min	0	0	0	2.13	0.010

Table 2 Test event description

Event	Date	Duration	Maximum of discharge (m^3 s^{-1})	Mean discharge (m^3 s^{-1})	Cumulative rainfall (mm)	Intensity (mm h^{-1})
13	Sept. 00	26 h	454	70	230	40

2.3 Artificial Neural Networks

As widely explained in [4, 9], three kinds of neural network models have been used in this study: a static model, a recurrent model, and a feed-forward model. The same references should provide the reader guidance about the implemented methods for the control of the bias-variance dilemma and of overtraining (early stopping, cross-validation, ensemble model) and about the performance criteria used (R^2 criterion and peak analysis). Only the part about knowledge extraction is reminded here, due to its important role in the study.

2.4 Extracting Information: KnoX Method

First, the KnoX method is applied to a specific architecture, based on multilayer perceptron, which represents the behavior of the physical process, in order to constrain the model to represent this physical behavior [7]. As the rain is essentially added in the first step of the rainfall-runoff transformation, we have introduced one layer of linear neurons implementing the addition of rains fallen at different time steps (delayed rains). This supplementary layer is called "*i*" (linear hidden neurons) as shown in Fig. 2. The second hidden layer (nonlinear hidden layer) calculates a nonlinear combination of the "locally added" rains.

The KnoX method [7–9] allows calculating a "simplified" contribution of each input to the model output. This method is described for the general deep model (two hidden layers) shown in Fig. 2. The principle of the method is that a contribution of an individual input variable can be quantified, after training, by the product of the parameter's chain linking this input to the output. The considered parameters are (i) "normalized" by the sum of the parameters linked to the same targeted neuron

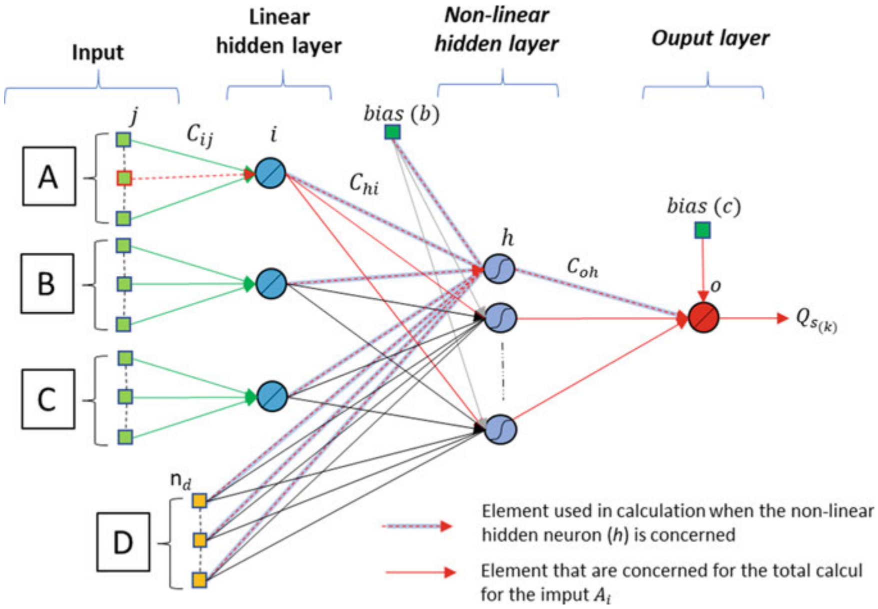

Input
Linear hidden layer
Non-linear hidden layer
Ouput layer
j
C_{ij}
i
bias (b)
C_{hi}
A
B
C
h
bias (c)
C_{oh}
o
$Q_{s(k)}$
n_d
D
Element used in calculation when the non-linear hidden neuron (h) is concerned
Element that are concerned for the total calcul for the imput A_i

Fig. 2 Application of the KnoX method on the deep multilayer perceptron

and (ii) made independent from the model initialization by calculating the median of absolute values of their values for 20 different random initializations. This regularized value is noted as ${}^{M}\left|C_{ij}\right|$ for the parameter C_{ij} linking the neuron (or input) *j* to the neuron *i*.

As the value of the sigmoid is not taken into account in Eq. 2, this contribution can be seen as the contribution of the "linearized" model. Nevertheless, the model is really a nonlinear model.

Regarding the model shown in Fig. 2, it appears that inputs are applied in several groups, for example, *A*, *B*, etc. Each group corresponds to a variable, for example, the rain gauge of *Mialet*, or the previous discharge (*D*). As the output depends dynamically on these inputs, following a complex and unknown multi-scale relation, these inputs are applied at several time steps in order to allow the model to estimate these multi-scale relations. Thus, the contribution (P_A) of the grouped inputs *A* (including several delayed inputs) is the sum of the contributions of each individual delayed input of the group *A*. The equation calculating the contribution for just one element (the value for the delay *j*) of the input *A* is provided in Eq. (1). Unhopefully, it is not possible to explain more comprehensively the method in the short present paper, so we suggest to the reader to refer to [7, 8].

$$P_{A(j)} = \frac{{}^{M}|C_{ij}|}{\sum_{i=1}^{n_A} {}^{M}|C_{ij}|} \sum_{h=1}^{H} \left(\frac{{}^{M}|C_{hi}|}{\sum_{i=1}^{n_i} {}^{M}|C_{hi}| + \sum_{d=1}^{n_d} {}^{M}|C_{hd}| + b_h} \right) \left(\frac{{}^{M}|C_{oh}|}{\sum_{h=1}^{H} {}^{M}|C_{oh}| + c_o} \right), \tag{1}$$

and

$$P_A = \sum\nolimits_{j=1}^{n_A} \left(P_{A(j)}\right), \qquad (2)$$

where the categories of parameters C_{ij}, C_{hi}, C_{oh}, and C_{hd} are shown in Fig. 2; n_A is the number of inputs in the group A; H is the number of hidden nonlinear neurons; n_j is the number of hidden linear neurons (first hidden layer); n_d is the number of delayed inputs of the group and D; b_{h} is the bias inputted to the nonlinear hidden input; and c_{o} is the bias parameter inputted to the output neuron.

3 Results

3.1 Choice of Variables

Starting from previous works of [4], we chose the following exogenous variables: (i) *Barre-des-Cevennes* rain gauge, *Saint-Roman de Tousque* rain gauge, and *Mialet* rain gauge, each one with a sliding window length $\{k, \ldots k - n_{\mathrm{r}} + 1\}$; and (ii) the sum of the mean rain (mean calculated over the three gauges) fallen from the beginning of the event. Of course, a bias input is used; several values were tried in order to evaluate the sensitivity of the KnoX method to its value.

Depending on the kind of considered model, state variables can be added: previously observed discharges for the feed-forward model, and previously estimated discharges for the recurrent one. The static model only takes rains and mean rains into account [9].

3.2 Model Selection

Model selection is a key issue of machine learning. The goal is to define accurately the architecture of the model managing the bias-variance tradeoff. This was done in this following work [10] using cross-correlation, cross-validation, and early stopping using the following rules.

- Hyper-parameters are adjusted for each one of the three kinds of model (static, feed-forward, and recurrent), input sliding windows width (n_{A}, n_{B}, n_{C}, n_{D}), and number of nonlinear hidden neurons (h).
- Widths of the rainfall windows applied to the model, $\{n_{\mathrm{A}}, n_{\mathrm{B}}, n_{\mathrm{C}}\}$, are selected, thanks to cross-correlation [11]. Initially proposed by [12, 13] generalizes the application of cross-correlation in hydrology. The used equation in this study is presented in Eq. (3).

Table 3 Correlation analysis of the data

Rain gauge		Mialet (h)	SRDT (h)	BDC (h)
Average response time		2	3	4.5
Response time range		1–3.5	2.5–4.5	4–5.5
Rainfall-discharge average cross-correlation		0.40	0.455	0.44
Rain gauge cross-correlation	Mialet	–	0.58	0.45
	SRDT	–	–	0.61

$$C_{xy}(k) = \frac{Cov(x_i, y_{i+k})}{\sigma_x \sigma_y} = \frac{\frac{1}{n}\sum_{i=1}^{n-k}(x_i - \bar{x})(y_{i+k} - \bar{y})}{\sigma_x \sigma_y} \tag{3}$$

With $k \in \mathbb{N}^+$; the truncation m, which is the maximum value of k, is recommended to be $m = n/3$. [12] indicated that two hydrological variables can be considered as statistically independent if their cross-correlation is superior to 0.2. Starting from this work, we selected three possible lengths for the sliding windows of rain gauges inputs: (i) the number of time step between $C_{xy} = 0$ and $C_{xy} = 0.2$ that defines the memory effect (called *memory window*); (ii) the sliding window between $C_{xy} = 0.2$ (positive slope) and $C_{xy} = 0.2$ (negative slope) (called *strong correlation* window); and (iii) all the m positive values of C_{xy} (called *full correlation* window). Based on [12], the correlations between gauges and response times are indicated in Table 3.

- The partial cross-validation score was operated on a subset of K *events*, the 17 most intense events in the database [3].
- The number of hidden neurons was increased from 1 to 10. The best model was chosen according to the highest cross-validation score S_v estimated as follows:

$$S_v = \frac{1}{K}\sqrt{\sum_{i=1}^{K}|E_i|^2} \tag{4}$$

where E_i is the validation error of the subset i used in partial cross-validation.

- An ensemble model is used in order to regularize on the initialization of parameters; moreover, the output values are the result of the median of the outputs of an ensemble of 20 members differing only by their initialization before training [8].
- Three bias values were considered (0.01; 0.1; 1), three depths of sliding windows, and three kinds of models (see Sect. 2.3), i.e., 27 different models have been designed following the procedure indicated in [9]. The best one in each kind of models has been chosen, regarding the test event, in order to have the most efficient models to analyze.

Architectures presented in Table 4 were thus selected.

Table 4 Selected models

Input variables	Static	Recurrent	Feed-Forward
Rain gauge window width (n_r) (BDC/SRDT/Mialet)	32/32/23	27/28/20	32/32/23
Cumulative rainfall window width	3	3	3
Order (r)/Previously observed outputs	x	3	3
Number of hidden nonlinear neurons	2	10	5
Bias value	1	0.01	0.01

3.3 Discharge Estimation

As shown in [4, 9], the best results are provided by the feed-forward model. This is usual because the feed-forward model uses the previously observed output as a state variable. The recurrent model is not as efficient but exhibits better dynamics, which is also frequently observed [4]. The static model presents an acceptable performance, being able to generate 63% of the peak discharge (Table 5; Fig. 3).

3.4 Contributions of Input Variables

After having verified that the models are convenient, it is possible to apply the KnoX method. The extracted contributions are presented in Table 6 [9].

It is interesting to compare the relative weights of the three rain gauges with a classic method dedicated to distribute rainfall on a watershed and widely used in hydrology: the Thiessen polygons method (or Voronoï polygons). This comparison is presented in Fig. 4.

As *Mialet* (MIA) is at the outlet of the basin and *Barre-des-Cévennes* (BDC) at the top of the basin, they are both represented with less contribution than *Saint-Roman de Tousque* (SRDT, near the middle of the basin) by the Thiessen polygon method. It is more or less also the case for the neural network models, with a very similar distribution to Thiessen distribution for the static model, being a little more different for the recurrent model and even more for the feed-forward model (providing the best results).

Table 5 Models performances on the test set

Model	R^2	SPPD %	P_D (0.5 h)
Static	0.83	63.3	1
Recurrent	0.89	78.5	0
Feed-forward	0.99	99.3	1

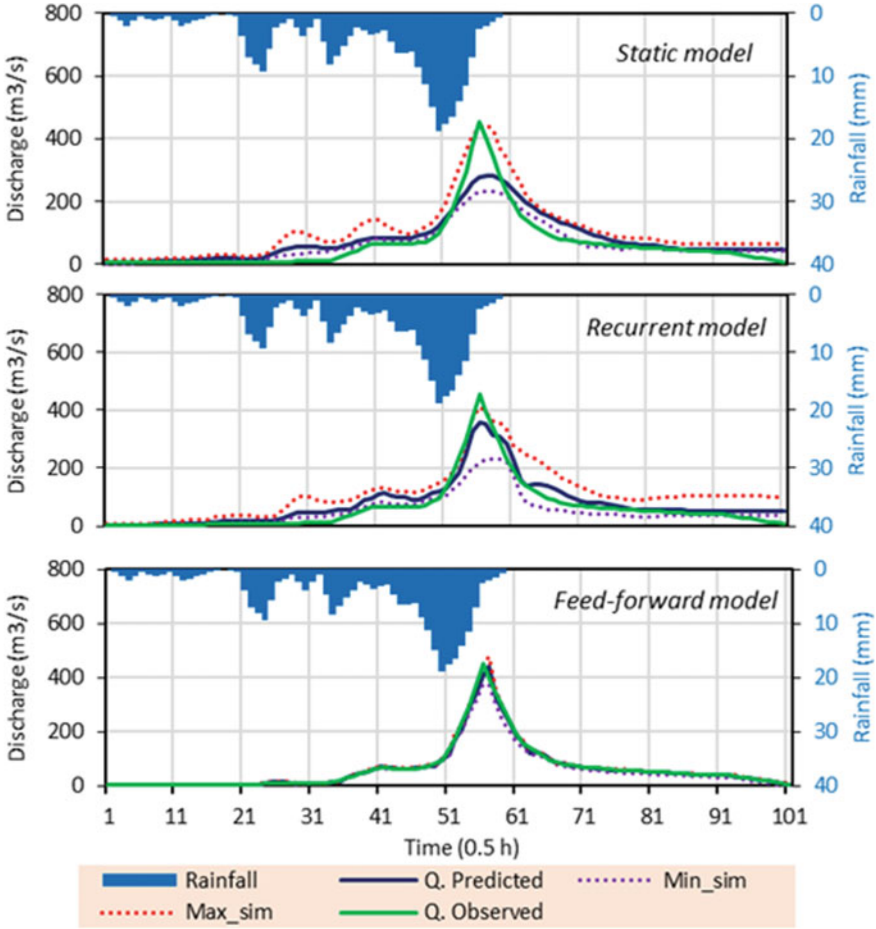


Fig. 3 Hydrographs for the test set. Min_sim and Max_sim correspond to the minimum and maximum values of the ensemble model. Q is the median of the 20 members of the ensemble

Table 6 Contributions (P_{x}) for the variables, from each model, expressed in %

Name of variable X	Static	Recurrent	Feed-forward
BDC	11%	10%	9%
SRDT	31%	17%	22%
Mialet	13%	12%	5%
Cumulated rainfall	31%	20%	12%
Previous Q. obs	–	–	45%
Previous Q. calc	–	25%	–
bias	14%	16%	7%
Total	100%	100%	100%

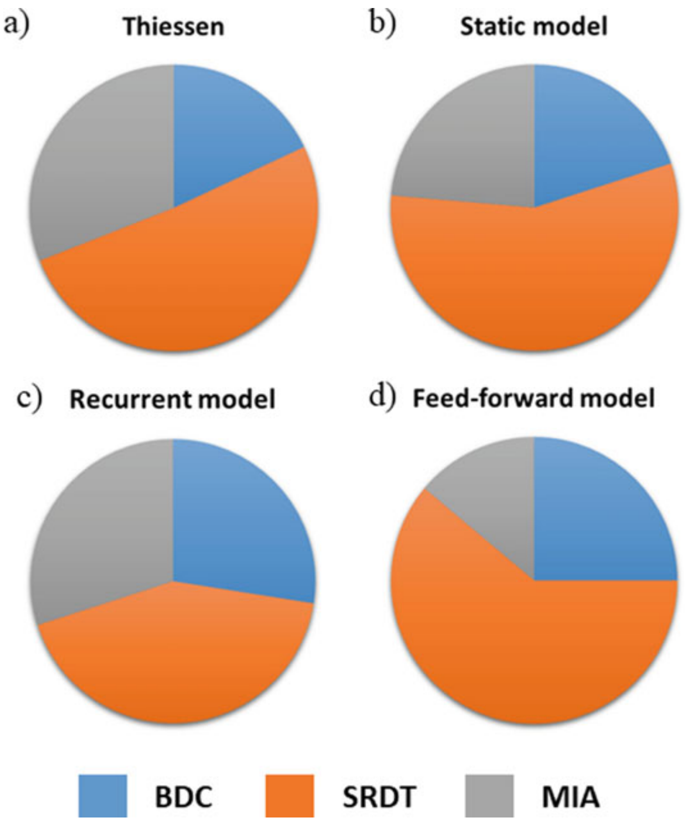


Fig. 4 Thiessen method weights (**a**) and relative weights form the models of the three rain gauges (**b**, **c**, **d**)

3.5 Results: Contributions as a Function of Time Windows

Here, we have considered the distribution of contributions among the time delay in the first layer of parameters (arriving at the linear neurons in Fig. 2). We compare the sum of these contributions (for the three rain gauges) to the cross-correlogram of the average rainfall (average of the three rain gauges) and the discharge. This comparison involves the three selected models presented in Sect. 3.1 (Fig. 5).

The static model shows the greatest similarity with the cross-correlogram, for the total contributions and for the relative contributions of each rain gauge. Regarding the response time (time corresponding to the peak of the cross-correlogram), the static model seems also to be the best. This result is logical because the variables taken into account by the static models are similar to those considered by the cross-correlogram: only rains.

For the three models, the SRDT rain gauge is the most represented in most of the time lags considered and not only in general, as shown in Sect. 3.4.

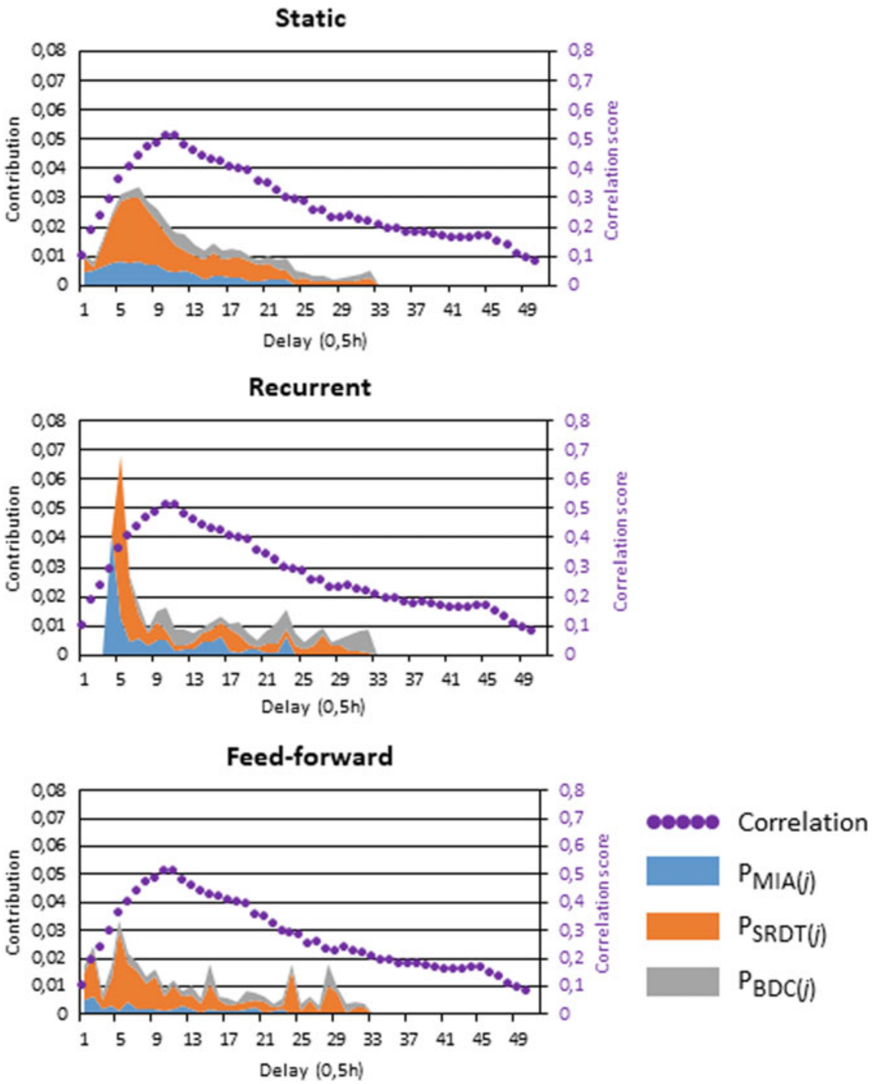


Fig. 5 Cross-correlogram (mean rainfall-Mialet discharge) and distribution of contributions calculated as indicated in Sect. 2.4

3.6 Results: Effects of the Bias

Before obtaining the selected models, many different combinations were tried during optimization. Among these combinations, three values of bias have been experimented, each separated by an order of magnitude: 0.01, 0.1, and 1. Figure 6 shows the contributions of *Saint-Roman de Tousque* among the delays of the input time

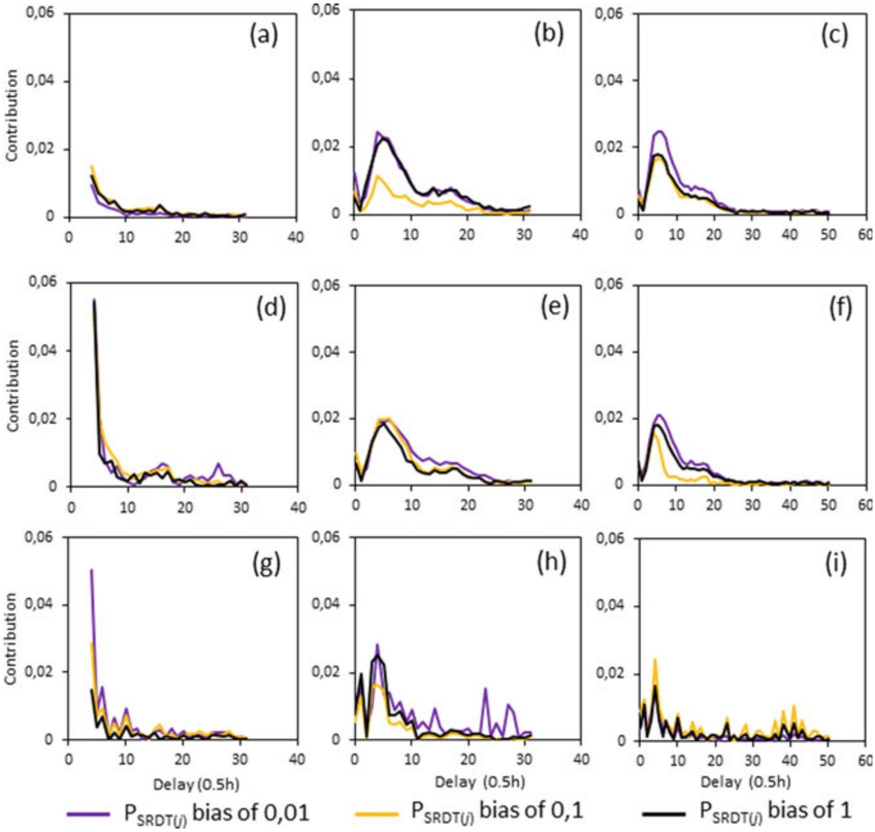


Fig. 6 *Saint-Roman de Tousque* contributions calculated as indicated in Sect. 2.4 with different bias and in different modeling configurations: **a**, **b,** and **c** are for static models; **d**, **e,** and **f** are for recurrent models; **g**, **h**, and **i** are for feed-forward models, whereas (**a**), (**d**), and (**g**) are for memory windows; (**b**), (**e**), and (**h**) are for strong correlation windows and (**c**), (**f**), and (**i**) are for full correlation windows

window, for the three types of models (static, recurrent, and feed-forward) and for the three time windows defined in Sect. 3.2 (memory, strong correlation, and full correlation windows). The other rain gauges have not been presented here due to the large number of figures it would have produced, but the *Saint-Roman de Tousque* station is representative of the three rain gauges from this point of view.

It can be noticed that the bias value does not deeply impact the contributions of the input variables. In particular, it does not change the general shape of these contributions even if in some cases, moderate amplitude differences appear.

4 Discussion

These results show how the kind of model takes into account explanatory variables on an observed phenomenon. Even if they use the same exogenous variables in the same context, their performances and behaviors are different due to their configuration and architecture.

4.1 Selecting a Model Type for Physical Knowledge Extraction

Analyzing the contributions assigned to each input variable (Table 6), it appears that

- The static model strongly uses exogenous variables (total contribution of 55%) and uses an important contribution (31%) to the cumulated rainfall that is useful to represent the soil saturation and could thus be considered as a substitute to a state variable.
- The recurrent model uses mostly previously estimated discharge (25%), whereas the total contribution of cumulated rainfall (20%) and of exogenous variables (40%) is lower than for the static model.
- The feed-forward model uses a smaller contribution for rains (12% for cumulated rainfall and 36% for exogenous variables), whereas previously observed values of discharge contribution are predominant (45%).

As foreseen by [14], the optimal type of model is strongly linked to the quality of explanatory information that is given to the model during the training phase. Here, we show that, despite its low performance, the static model is forced to represent the physical relationship between exogenous variables and the output, whereas the recurrent model and the feed-forward model are helped in this task respectively by the previously estimated or observed discharge. Consequently, the total use of the exogenous variables decreases when state variable information increases. If we compare the relative contributions of the three rain gauges with the Thiessen polygons, we observe a decrease of the similarity while state variables are added.

Finally, in this study, the best tradeoff between model performance and knowledge extraction capacities seems to be provided by the recurrent model. Nevertheless, this conclusion is based on one test set, and it should be confirmed by further studies.

4.2 Response Time and Contributions

The cross-correlation provides a simple linear representation of the behavior of the modeled system and allows estimating the response time. Here again, while the contributions of state variables appear, the similarity with the correlogram decreases.

This could be interpreted as a confirmation that recurrent and feed-forward models represent well the behavior that takes profit of the rich information provided by the previous discharge input (estimated or simulated): the accumulation of previously fallen rains. The less the model is helped by the previous discharge input, the more it is forced to represent well the role of recent and ancient rainfalls. This appears in Fig. 6d, g with great values of recent rains contributions; on the equivalent contributions in Fig. 6e and the "noisy" contribution in Fig. 6i (feed-forward with the maximum window width).

4.3 Bias Input Importance

The bias input plays a role that is usually interpreted in hydrology as the base flow (remaining discharge when there is no rainfall). In this case, its contribution is consistent: it is significantly less involved in the calculation of the output when the previously observed discharges are used as input (the previous base flow is thus applied by the inputs). In other cases, it seems to guide the models to acceptably approximate the discharge information when necessary.

If the bias input seems necessary to guide the model, its value does not deeply change the distribution of the contribution of the rain gauges as a function of the instant of the time window. One could suppose that changing an order of magnitude in the bias input value can easily be counterbalanced during the training step by applying a proportional modification to the weights applied to this input.

5 Conclusions and Perspectives

Flash flood forecasting is a very challenging task, especially in the *Cévennes* range. Several examples of robust forecasts using neural networks have been published but the results did not always allow understanding how close the model was to the physical behavior of the basin, in addition of being close to the observed output. The obtained results prove again that when using relevant and properly combined variables on any of the networks used here, an efficient model can be implemented.

Nevertheless, enhancing these models and applying them to an increasing number of basins, in a context of climate change, and with various characteristics, require a better understanding of the processes involved in their operation as well as in such flood events. For this purpose, the KnoX method, developed to extract information from a neural network model, was applied to the *Gardon de Mialet* basin. This method allows an understanding of how the variables are handled by the model to approximate the modeled phenomenon. First, it appears that the bias input was consistently used to model the base flow. Then, interestingly, there has been evidence that the variables do not express themselves in the same way depending on the different models used. It was known that the choice of a model must be driven by the modeling goal (for example a recurrent model for a long-term prediction). Besides being driven by the modeling goal, it appears that the choice for a model might be guided by the situation: availability (real-time and historical) of data, quality and explanatory nature of the data. In this study, this results in three kinds of model: static, recurrent, and feedforward, showing increasing performances while taking into account more realistic state variables. On the other hand, if assess the performance of a kind of model by the ability to extract physical information from it, the ranking is reversed and the less the model considers state variables, the more the design of its estimator will adopt behaviors that mimic the physical processes.

Finally, it appears that the KnoX method shows very interesting capabilities; the next steps will consist in generalizing this method to other sites and other rainfall events in the *Cévennes* range, with an increasing complexity in the physical processes to extract (dams and/or karst systems for example).

Acknowledgements The authors thank IMT Mines Alès that funded the present study; the SPGD flood-forecasting agency for providing datasets. Our gratitude is extended to B. Janet for the stimulating collaboration shared with the SCHAPI and to R. Moussa and P. Roussel-Ragot for the helpful discussions and support. The constant effort made by D. Bertin and the Geonosis Company to enhance and develop the neural network software RNF Pro are thereby acknowledged as well.

References

1. Rouzeau, M., Martin, X., Pauc, J.C.: Retour d'expérience des inondations survenues dans le departement du Var les 15 et 16 juin 2010. http://cgedd.documentation.developpement-durable.gouv.fr/documents/cgedd/007394-01_rapport.pdf
2. Roberts, S.J., Penny, W.: Neural networks: friends or foes? Sens. Rev. **17**(1), 64–70 (1997)
3. Toukourou, M., Johannet, A., Dreyfus, G., et al.: Rainfall-runoff modeling of flash floods in the absence of rainfall forecasts: the case of "Cévenol Flash Floods". App. Intell. **35**(2), 178–189 (2011)
4. Artigue, G., Johannet, A., Borrell, V., et al.: Flash flood forecasting in poorly gauged basins using neural networks: case study of the Gardon de Mialet Basin (Southern France). NHESS **12**(11), 3307–3324 (2012)
5. Oussar, Y., Dreyfus, G.: How to be a gray box: dynamic semi-physical modeling. Neural Netw. **14**(9), 1161–1172 (2001)
6. Johannet, A., Vayssade, B., Bertin, D.: Neural networks: from black box towards transparent box—application to ETP modelling. Int. J. Comp. Intell. **24**(1), 162 (2007)
7. Kong-A-Siou, L., Cros, K., Johannet, A., et al.: KnoX method, or Knowledge eXtraction from neural network model. Case study on the Lez karst aquifer (southern France). J. Hydrol. **507**, 19–32 (2013)
8. Darras, T., Borrel-Estupina, V., Kong-A-Siou, L., et al.: Identification of spatial and temporal contributions of rainfalls to flash floods using neural network modelling: case study on the Lez basin (southern France). Hydrol. Earth Syst. Sci. **19**, 4397–4410 (2015)
9. Saint-Fleur, B., Artigue, G., Johannet, A., et al.: Knowledge Extraction (KnoX) in deep learning: application to the Gardon de Mialet flash floods modelling. In: Proceedings ITISE-2019, pp. 178–189, Granada, 25th–27th September (2019)
10. Kong-A-Siou, L., Johannet, A., Borrell, V., et al.: Optimization of the generalization capability for rainfall–runoff modeling by neural networks: the case of the Lez aquifer (southern France). Environ. Earth Sci. **65**, 2365–2375 (2012)

11. Kong-A-Siou, L., Johannet, A., Borrell, V., et al.: Complexity selection of a neural network model for karst flood forecasting: the case of the Lez Basin (southern France). J. Hydrol. **403**, 367–380 (2011)
12. Jenkins, G.M., Watts, D.G.: Spectral Analysis and Its Applications. Holden-Day (1969)
13. Mangin, A.: Pour Une Meilleure Connaissance Des Systèmes Hydrologiques à Partir Des Analyses Corrélatoire et Spectrale. J. Hydrol. **67**(1–4), 25–43 (1984)
14. Nerrand, O., Roussel-Ragot, P., Personnaz, L., et al.: Neural networks and nonlinear adaptive filtering: unifying concepts and new algorithms. Neural Comput. **5**(2), 165–199 (1993)

Forecasting Short-Term and Medium-Term Time Series: A Comparison of Artificial Neural Networks and Fuzzy Models

T. V. Afanasieva and P. V. Platov

Abstract The study is focused on experimental comparison of time series models of two classes, namely, artificial neural networks and fuzzy time series models. In each class, three basic models of the time series were selected to compare their predictive abilities, which were evaluated by the MAPE criterion. To the class of models using artificial neural networks, multilayer perceptron networks, as well as RNNs, containing LSTM or GRU blocks in the hidden layer, were investigated. In this study, three basic fuzzy time series models were used: the model with fuzzified time series values, the model with fuzzified first differences of time series values, and the model based on the elementary fuzzy tendency. A comparative study was conducted based on dataset of time series, which were divided into two groups relative to the length of time series. An experimental study showed that for medium-term time series on the test interval, the RNNs based on LSTM showed the smallest error on average (MAPE = 3.0013%), and for the short-term time series, the fuzzy models showed the smallest error on average (MAPE = 5.7313%), while models of the ANN class predicted the short-term time series with MAPE > 9.8% in average.

Keywords Artificial neural networks · RNN · LSTM · GRU · Fuzzy models · Time series forecasting · Comparison · Predicting

1 Introduction

To our days, a lot of time series (TS) with different features are stored in databases. These time series could have different lengths and behaviors. As a kind of efficient nonlinear function approximators, artificial neural networks (ANN) have been popularly applied to time series forecasting. TS models based on artificial neural networks are becoming frequently used due to their opportunity to learn data dependencies. One of the problems of ANN models is how to determine the structure of the network, i.e., the number of layers, the model prediction order, the number, and

T. V. Afanasieva (✉) · P. V. Platov
Ulyanovsk State Technical University, Ulyanovsk, Russia
e-mail: tv.afanasjeva@gmail.com

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*, Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_23

types of neurons in each layer. The amount of related works demonstrates the interest in the analysis of recurrent neural networks (RNNs) for time series forecasting from a different point of view. RNNs are defined as a class of supervised machine learning models, made of artificial neurons with one or more feedback loops [1]. In general, these networks include nonlinear but simple units, enabled to store, remember, and process past complex signals for long time periods. In this way, RNNs can learn the temporal context of input sequences, map an input sequence to the output sequence at the current time step, and predict the sequence in the next time step. It could be said that one of the most popular units in hidden layers in RNNs is Long Short-Term Memory (LSTM) [2]. Since ANN-based models are difficult to interpret, the other way to solve the problem of TS forecasting came from the theory of fuzzy models [3]. Unlike traditional TS, the values of fuzzy TS are fuzzy sets, not real numbers of observations. Fuzzy TS models could be presented in the form of rules that are easy to interpret, and the forecast results are expressed in both linguistic and numerical forms. A positive property of fuzzy models is their tolerance concerning random fluctuations and the length of the TS. Analysis of studies in the field of TS forecasting based on ANN models shows that they are usually carried out in comparison with different types of ANNs or in comparison with statistical models, which are considered as benchmarks. Note that the confirmation of the adequacy and accuracy of the developed fuzzy TS models is implemented according to the same scenario. At the same time, the question of comparing the forecasting accuracy of models of these two different classes (ANNs and fuzzy TS models) over the set of real time series is still open. Therefore, this article is focused on filling this gap and is aimed at experimental research and comparison of forecasting accuracy of models based on ANNs and fuzzy TS, in particular, with respect to short-term and medium-term TS.

Since a large number of such models have been developed at present, the following ANN models have been selected for experimental research: multilayer perceptron network (MLP) [4], RNN based on LSTM [2], and RNN based on GRU [5]. The MLP was chosen as the simplest ANN model, and the choice of RNN based on LSTM was determined due to the successful application of LSTM in forecasting: in the TS forecasting competition (CIF-2016) [6], the techniques with LSTMs showed very good results [7]. Compared to LSTM, the GRU is characterized by some simplifications leading to calculate a smaller number of weights. While the LSTM is commonly used, the RNN based on GRU is more novel. The accuracy of TS forecasting of these ANN models will be compared with the predictive accuracy of fuzzy TS models. We focused on testing accuracy of three basic fuzzy TS models: the model with fuzzified TS values [8], the model with fuzzified first differences of TS values [9] and model, based on the elementary fuzzy tendency [10]. We choose these fuzzy TS models as they could be considered to our opinion as basic fuzzy TS models. Comparison study of the accuracy in TS forecasting between these classes of models will be held at the dataset given at competition on Computational Intelligence in Forecasting (CIF-2016) [6], which was organized within the IEEE World Congress on Computational Intelligence (IEEE WCCI-2016).

The structure of the paper includes six sections. In the second section, related works are considered. The TS models based on RNNs used in prediction and comparison are presented in the third section. Section 4 includes the description of basic fuzzy TS. The results of the comparison of the predicting accuracy of two classes of TS models, based on ANNs and fuzzy models, are described in Sect. 5. The conclusions are given in Sect. 6.

2 Related Works

ANNs are data-driven, self-adaptive models with few prior assumptions and could be very efficient in solving nonlinear problems. This feature is in contrast to many traditional models for TS predictions, such as ARIMA, which assume that the series are generated from linear processes and as a result might be inappropriate for most real-world problems that are nonlinear [11].

In the article [12], the comparison of ANN and AR models was derived in the simulation experiment and was shown that for nonlinear TS models the usage ANN model outperforms the AR model in terms of both mean and variability of the observed coverage. For linear and weak nonlinear TS models, the two approaches seem to be equivalent. Multilayer perceptron network (MLP), FIR neural network, and Elman neural network were compared in the study [13]. As follows from the paper MLP network performed well in one-step predictions for four TS and output more accurate forecasts for 75% of TS. Currently, various configurations of RNNs are proposed, e.g., BRNN (a bidirectional NN), LSTM (a long short-term memory), and GRU (a gated recurrent unit) [2, 14, 15]. The successful implementation of RNNs and LSTMs as a component of forecasting methods for TS analysis leads to increasing interest in them [7, 16–18]. In [15], ten LSTM architectures were considered, and their main disadvantages were mentioned as follows: a higher memory demand and computational complexity than a simple RNN due to the many memory cells. GRU [2] uses the same ideas as LSTM, but there are differences between these networks. First, the GRU does not contain an output gate: there are only reset and update gates that are similar to the input and forget gates in the LSTM. Second, the network state does not depend on the state in the previous step as in LSTM that allows us to consider GRU as simplification of LSTM. While the LSTM is commonly used for time series forecasting, the RNN based on GRU is more novel. Therefore, the comparative study of their effectiveness in TS predictions attracts more and more attention.

In the work [19], Laptev et al. studied RNNs in event forecasting and found that neural networks might be a better choice in comparison with classical TS methods when the number, the length, and the correlation of the TS are high. Che et al. in the report [20] described a GRU-based model with a decay mechanism to capture informative missingness in multivariate TS. A methodology DeepAR for probabilistic forecasting, based on training an auto-regressive RNN on a large number of related TS, was proposed in [21]. Instead of forecasting raw TS, the authors focused on estimating the probability distribution of a time series' future given its past. The accuracy improvements of around 15% compared to state-of-the-art models, in particular, ETS model [22] with automatic model selection, were shown through empirical evaluation on several real-world forecasting datasets.

The comparison of LSTM with ARIMA in the forecasting of financial and economic TS was provided in [23]. As follows from the work, the LSTM showed significantly better prediction accuracy than ARIMA according to the RMSE criterion. In the work [24], it was carried out an empirical study in TS forecasting using both LSTM and GRU networks. To compare LSTM and GRU, the TS set referred to bike sharing was used. In this work, the prediction technique was proposed, including bootstrap samples of sequences and preliminary presentation of TS properties such as cyclicality, seasonality, the beginning of the month, holidays or working days, and some others. Although the author used many performance indicators, the conclusion was that two networks produce very similar forecasts.

In the paper [25], the study of prediction accuracy of the LSTM and GRU units in RNN configurations was held in dependence of TS behavior and regardless of their behavior. In this study for the 30 simulated TS, the prediction accuracy of RNNs on average was as follows: MAPE (GRU) = 5.729% compared with LSTM (MAPE (LSTM) = 5.072%). Each of TS had the same length, which was equal to 200; the splitting on training and testing parts were established as 90:10. According to this comparative study, it was difficult to determine the winner in forecasting regardless of TS behavior because the RNNs results were about the same. In dependence of time series behavior, the study showed that the forecasts of LSTM were more accurate in comparison with GRU for TS that included trend with random fluctuations and did not have the seasonal component.

In parallel with the development of RNN-based models, forecasting techniques, based on fuzzy TS with different structures, were developed. However, as follows from numerous works, an accuracy of fuzzy models was tested on one time series only and frequently for the training part of time series. First, the concept of fuzzy time series has been put forward by Song и Chissom in 1993 [8]. They proposed fuzzy models of stationary and non-stationary (time-invariant and time-variant) first-order TS and used the developed models to predict the number of enrolling students at the University of Alabama by fuzzifying a numerical TS. Chen [26] believing that Song и Chissom ‘s method is too complex to apply, proposed some simplification using arithmetic operations instead of logical Maximin composition. In the work [9], Hwang, J. R., Chen, S. M., and Lee, C. H. proposed a modification of the Song's method, in which instead of fuzzified values of numeric TS, their first differences were fuzzified and used in fuzzy modeling.

To our days, the fuzzy TS is widely used in forecasting and analysis of real TS [26–31]. According to the study [32], the fuzzy TS models showed comparable with ARIMA-model accuracy on average for 53% of given 91 TS of the dataset [6] and can obtain high accuracy forecasts together (SMAPE < 0.06). The study of the prediction errors in dependence of the TS length demonstrated that the fuzzy TS model produced good forecasting accuracy on average for medium-term TS [32].

In the work [33], a comparative study of accuracy of ANN and fuzzy TS for the prediction of one short-term TS, that is, wheat production, in dependence of metrological parameters (average weekly temperature, sunshine, and rainfall) was carried out. The length of TS was equal to 15. In this study, ANN seems to be MLP and has three layers. The hidden layer has 5 neurons and ANN is trained for 4000 epochs. In the comparison, fuzzy TS model proposed by S. M. Chen was used [27]. The authors described in detail the technique for constructing the ANN architecture, but do not provide any information about fuzzy models, the membership functions used, and the fuzzy inference method. Therefore, the conclusions drawn in this article about a more accurate prediction made by the ANN in comparison with fuzzy TS need additional explanation. In the work [34], the similar study was presented with respect to prediction of one short-term time series, that is, marine fish production in India. However, the comparison of forecasting accuracy of fuzzy TS models was derived according to the proposed method that combined ANN and FTS.

The analysis of research works shows that there are not enough studies in accuracy comparison of ANNs and fuzzy TS carried out on the set of short-term and medium-term TS.

3 Concepts of LSTM and GRU

A recurrent neural network (RNN) is an extension of a conventional feedforward neural network which is one of the best tools for solving image recognition problems. In TS analysis, the RNN is focused on learning the function of the input x_t and previous output y_{t-1}. The simplified form of this function can be expressed as follows:

$$y_t = \mathrm{f}(x_t, y_{t-1})$$

Today, there are many architectures for RNNs. The simplest of them is SimpleRNN, the main disadvantage of which is the inability to store information about the previous elements of the sequence for a long time because the gradients tend to vanish. To solve this problem, Hochreiter developed the Long Short-Term Memory (LSTM) architecture [2], which was subsequently studied and improved by other researchers and is now used in solving a wide range of problems with very good results.

3.1 Long Short-Term Memory

The LSTM unit contains three gates: an input gate, a forget gate, and an output gate. The gates are implemented as a logistic function for calculating the value on the interval [0; 1]. Multiplication by this value is used to partially allow or prohibit information from getting into memory and disappearing from memory. For example, the input gate controls the degree to which the element being processed is stored in memory, and the forgetting gate controls the degree to which the value is stored in memory. The output gate controls the degree to which the value in memory is used in calculating the output value. The calculation of the gate vectors, as well as the state vector and output vector, is as follows:

$$\begin{aligned}
f_t &= \sigma\left(W_{xf}x_t + W_{fy}y_{t-1} + b_f\right)\\
i_t &= \sigma\left(W_{xi}x_t + W_{iy}y_{t-1} + b_i\right)\\
o_t &= \sigma\left(W_{xo}x_t + W_{oy}y_{t-1} + b_o\right)\\
h_t &= f_t \circ h_{t-1} + i_t \circ \tanh\left(W_{xh}x_t + W_{hy}y_{t-1} + b_h\right)\\
y_t &= o_t \circ \tanh(h_t)
\end{aligned}$$

where f_t—vector of the forgetting gate at time t, showing the degree of memorization of old information, i_t—vector of the input gate at time t, showing the degree of receipt of new information, o_t—vector of the output gate—"candidate for the output value" at time t, W, b—corresponding weight matrices and displacement vectors, and $\circ$ is the Hadamard product (elementwise multiplication).

So, at each time LSTM updates two variables, the output y_t, and the state h_t, using controlled by gates' composition of functions on previous values of these variables and learned weights. Thus, the output at a given time is calculated not only based on the previous output and input, but also using the previous state. Including the additional variable and function to modify it makes the architecture of RNN with LSTM more complex for understanding in comparison to simple RNN. The architectures of LSTM and comparison of their advantages are described in [15], which was noted that for the identical size of hidden layers, a typical LSTM has about four times more parameters than a simple RNN but can model long-term sequential dependencies and is more robustness to vanishing gradients.

3.2 Gated Recurrent Unit

The Gated Recurrent Unit (GRU) architecture is another solution to the vanishing gradient problem critical to SimpleRNN [35]. GRU was developed by Cho [5] and used the same ideas as LSTM, but there are differences between these networks. First, the GRU does not contain an output gate: there are only reset and update gates that are similar to the input and forget gates in the LSTM. Second, the network state does not depend on the state in the previous step and does not store explicitly, as in LSTM. Like to major unit of RNN, the GRU computes the output as a function of the input x_t and previous output y_{t-1}, but using gates that modulate the flow of information inside the unit:

$$\begin{aligned}
z_t &= \sigma\left(W_{xz}x_t + W_{zy}y_{t-1} + b_z\right)\\
r_t &= \sigma\left(W_{xr}x_t + W_{ry}y_{t-1} + b_r\right)\\
h_t &= \tanh\left(W_{xy}x_t + W_{ry}(r_t \circ y_{t-1}) + b_h\right)
\end{aligned}$$

$$y_t = z_t \circ y_{t-1} + (1 - z_t) \circ (h_t)$$

where z_t—vector of the update gate at time t, r_t—vector of the reset gate at time t. Thus, this type of network has fewer parameters and performs fewer operations compared to LSTM; therefore, the implementation of this network will consume less memory compared to LSTM, and forward and reverse distribution will be faster.

4 Fuzzy Time Series Models

There are many modifications of fuzzy TS models (FTS); in this study, three FTS models were used: model with fuzzified TS values [8], model with fuzzified first differences of TS values [9], and model, based on the elementary fuzzy tendency [10]. We choose these FTS models as they could be considered as the basic FTS models.

4.1 Fuzzy TS Model, Based on Fuzzified TS Values

Let X $= \{x_t, \forall x_t \in \mathbb{R} | t = 1, 2, \ldots\}$ be a numerical TS and $\mathbb{R}$ is the set of all real numbers. Let for TS X a fuzzy TS $\tilde{X} = \{\tilde{X}_t | t = 1, 2, \ldots\}$ is defined, such that each $\tilde{X}_t$ is a fuzzy set in $\mathbb{R}$ for corresponding x_t. We assume that each $\tilde{X}_t$ satisfies the properties of normality and convexity. The FTS model, based on fuzzified TS values, according to [8, 32], is defined as a time-invariant model:

$$\tilde{X}_t = \left(\tilde{X}_{t-1} \times \tilde{X}_{t-2} \times \ldots \times \tilde{X}_{t-p}\right) \circ R(t, \ldots, t-p),$$

where "×" is the Cartesian product, $R(t, \ldots, t-p)$ is the FTS model as fuzzy relation, which can be calculated by Mamdani's algorithm, p is the order of the model (usually, $p = 1, 2, 3, 4, 5, 6$), "$\circ$" is the max-min composition. Further, for simplicity, this FTS model will be designated as an S-model.

4.2 Fuzzy TS Model, Based on Fuzzified First Differences of TS Values

In this TS model instead of TS values x_t their first differences $\Delta x_t = (x_t - x_{t-1})$ are fuzzified to obtain fuzzy sets $\Delta\tilde{X}_t$ [9], and fuzzy TS $\Delta\tilde{X} = \{\Delta\tilde{X}_t | t = 2, 3, \ldots\}$ has to be forecasted. It is necessary to notice that each $\Delta\tilde{X}_t$ is a fuzzy set in the set $\mathbb{R}$ of all real numbers and satisfies the same properties as $\tilde{X}_t$: normality and convexity. So, the FTS model based on fuzzified first differences of TS values in the form of the p-th order time-invariant TS forecasting model in accordance with [32] is presented as

$$\Delta\tilde{X}_t = \left(\Delta\tilde{X}_{t-1} \times \Delta\tilde{X}_{t-2} \times \ldots \times \Delta\tilde{X}_{t-p}\right) \circ R(t, \ldots, t-p).$$

For simplicity, the fuzzy TS model, based on fuzzified first differences of TS values, will be indicated in the study as a D-model.

4.3 Fuzzy TS Model, Based on Elementary Fuzzy Tendencies

The notion of fuzzy tendency was introduced in [30], as a linguistic characteristic of a TS behavior (Increase, Decrease, Fluctuation), expressed in fuzzy terms, which can be obtained for any numerical TS X $= \{x_t, \forall x_t \in \mathbb{R} | t = 1, 2, \ldots\}$. Further, the notion of elementary fuzzy tendency TS for a numerical TS was done and successfully used in fuzzy forecasting techniques [36]. FTS model, based on elementary fuzzy tendencies, is presented as two fuzzy TS: $V = \left\{\tilde{V}_t | t = 2, 3, \ldots\right\}$, which characterizes the type changing of elementary fuzzy tendencies, and TS $A = \left\{\tilde{A}_t | t = 2, 3, \ldots\right\}$, representing their intensities. Here each $\tilde{V}_t$ and $\tilde{A}_t$ are the fuzzy sets, which are defined in the set of real numbers $\mathbb{R}$ and satisfy the normality and convexity properties. The FTS model, based on elementary fuzzy tendencies, is represented in accordance with [32] as

$$\tilde{V}_t = \left(\tilde{V}_{t-1} \times \tilde{V}_{t-2} \times \ldots \times \tilde{V}_{t-p}\right) \circ R_{\tilde{\vartheta}}(t, \ldots, t-p),$$
$$\tilde{A}_t = \left(\tilde{A}_{t-1} \times \tilde{A}_{t-2} \times \ldots \times \tilde{A}_{t-q}\right) \circ R_{\tilde{a}}(t, \ldots, t-q).$$

Further for simplicity the FTS model, based on the elementary fuzzy tendencies, will be called briefly as a T-model.

5 Experiments and Results

To conduct the comparative study of the accuracy in TS forecasting for two classes of models (namely, ANN and FTS), the software tools were developed.

Each ANN model in the experimental study had three-layer architecture. In ANNs the output layer included 1 neuron, the input layer contained from 1 to 20 neurons, and the hidden layer had 10 ones in each type of configurations: RNN based on LSTM, RNN based on GRU and MLP. For each ANN, backpropagation training algorithm, the loss function MSE, optimizer Adam, activation function ReLU for RNNs, and

Sigmoid function for MLP were used. A maximum of 200 epochs has been established for training each neural network on TS dataset, and the parameter lookback was varied from one to ten. In software for TS forecasting by ANNs, the Python 2.7 was used with the Keras library for building a neural network under the TensorFlow library for automatic differentiation [37]. The Pandas library for data processing and the library scikit-learn mainly for time series normalization/denormalization were implemented as well. The developed software includes the set of modules grouped by features: TS pre-processing and normalization, building and training models for TS prediction, TS post-processing and denormalization, calculating the accuracy of prediction by criterion MAPE, and output the results of prediction in graph and text forms. For more details, please see the paper [25].

The software for TS forecasting using FTS models includes S-model, D-model, and T-model, considered in the previous section. For all FTS, the linguistic variables were constructed on TS universe and the orders of the FTS models varied from one to five to choose the best model for each time series. To obtain a fuzzy representation of each numerical TS, ten linguistic terms were built on intervals of equal length of universe, and corresponding membership functions having the same symmetrical and triangular shape were used for automatic generation of fuzzy models. There are six major components that were developed in the software: fuzzification tools, tool for FTS models creation, fuzzy inference system based on Mamdani technique, library of FTS models, component for selection of the best FTS among three basic fuzzy models, calculating the accuracy criterion MAPE and outputs component in graph and text forms. All the components of the library of FTS models are developed by C# using .Net Framework 4.5.1.

The developed software tools were applied to perform real forecasts on the TS set of the competition CIF-2016 [6]. The CIF-2016 competition dataset consists of monthly time series, composed of TS related to the banking industry. A sample of 20 time series was used from this dataset of different lengths and different behaviors, divided into two groups, short-term time series (from 32 to 90 points) and medium-term ones (from 90 to 200 points). We focused on the study of two groups of time series according to their length not to their behavior for two reasons. First, often time series are monthly or weekly observations. Such time series have different lengths and can be considered as short-term time series and medium-term ones. Second, in the studies [25, 32], the comparison of forecasting accuracy was carried out depending on the behavior of time series. That is why in this study, we are interested in answering the question of whether there is a difference in the forecasting accuracy of the considered TS models depending on their length. The obtained information would be useful in the rational choice of a suitable TS model in the forecasting problem. It is necessary to note that before prediction each time series was divided into two parts: a train part and a test part. The train part was used to construct and to identify time series model. The test part of a time series was used to test the identified time series model and to calculate the errors for the comparison using criterion MAPE:

$$MAPE = \frac{100\%}{n} \sum\nolimits_{t=1}^{n} \frac{|F_t - x_t|}{|x_t|}$$

where F_t is the predicted values, produced by the model for the given TS; n is the number of predicted points in the test part of TS; x_t is the real values of test part of the TS unknown for model. Pre-defined length of a test part for each TS was determined as 10% of a time series length as for ANN models.

For each TS, the adequate and accurate model from the set of described TS model (ANNs and FTSs) was identified. The identification means the process, where the order, parameters and the type of the best forecasting model for each TS are defined on the train part of time series. The errors for investigated models, estimated by MAPE, obtained on the test part of the medium-term time series are presented in Table 1, and for short-term time series they are shown in Table 2.

In these tables, three columns for ANNs are presented and in one column depicts the errors for one FTS model that showed the minimal error for corresponding TS. The last column contains the designation of model winner. When comparing the results of TS forecasting by the considered models, we used the concept of a minimum difference in accuracy equal to 0.05% of MAPE. If for the compared models, for instance, Model1 and Model2, the errors calculated by criterion MAPE differ by no more than 0.05%, then these models will be considered as models predicting the TS with approximately the same accuracy. In the last columns of Tables 1 and 2, these models will be designated as Model1 = Model2. A model with a minimum MAPE value is considered to be the winner if its MAPE error is less than the MAPE value of other models by more than 0.05%, then the designation of this model will be shown in the last column of Tables 1 and 2. If the MAPE value of the winner model is more than double the MAPE values of any other compared models, then in the last column of Tables 1 and 2 the model winner will be indicated with the plus sign.

As follows from Table 1 for the ten medium-term TS on average, RNN based on LSTM is the winner model, and for one TS, this model showed the error less than twice the error of any other models in terms of MAPE. A comparison of the average

Table 1 Prediction errors estimating by criterion MAPE in percentage for medium-term TS

Num. of TS	RNN with GRU	RNN with LSTM	MLP	Best FTS	Model winner
1	3.1562	2.9594	4.9185	2.401	FTS
2	1.3317	1.2125	3.1733	1.9027	LSTM
4	3.2957	3.1674	6.5034	4.0993	LSTM
7	2.2713	2.2537	4.2778	2.4879	LSTM = GRU
12	2.5570	2.6014	4.4378	3.2748	LSTM = GRU
13	4.6007	4.0631	5.9157	11.5820	LSTM
14	2.4874	2.4293	5.6953	5.7363	LSTM
17	3.3280	2.5986	4.9044	5.0004	LSTM
19	5.4248	5.9477	11.3940	9.3373	GRU
24	6.2833	2.7805	10.2709	12.7698	LSTM+
Mean	3.4736	3.0013	6.1491	5.8592	LSTM
Std. dev.	1.5265	1.2603	2.6507	3.9737	LSTM

Table 2 Prediction errors estimating by criterion MAPE in percentage for short-term TS

Num. of TS	RNN with GRU	RNN with LSTM	MLP	Best FTS	Model winner
49	1.9988	2.7382	2.3802	2.3647	GRU
50	1.3078	1.3201	1.7499	2.0406	LSTM = GRU
55	0.0389	0.0151	0.0317	0.0406	LSTM = GRU = FTS = MLP
58	0.0403	8.0309	22.2175	11.5488	GRU+
63	11.5221	11.4313	9.2555	6.2155	FTS
67	43.8490	33.7121	27.1040	13.2705	FTS+
68	23.0603	25.8134	12.2089	4.2096	FTS+
70	7.7692	8.3351	8.6787	6.4127	FTS
71	24.7122	21.9804	7.2232	5.1433	FTS
72	13.9993	14.5189	7.5981	6.0668	FTS
Mean	12.8298	12.7896	9.8448	5.7313	FTS
Std.dev.	14.1969	11.2296	8.7415	4.1028	FTS

values in the forecast errors for ten medium-term TS shows that behind the RNN based on LSTM, the RNN based on GRU follows by a small margin, then by a large margin MLP. The largest error, but in principle acceptable in practical forecasting of the time series, was shown by the class of fuzzy TS models (MAPE = 3.974%). If we consider the percentage of cases when RNN based on LSTM predicted more accurately, then they were the winners only for 60% of the considered medium-term TS. For the rest medium-term TS, the competing models (that is, FTS and RNN based on GRU) were more accurate or showed approximately the same accuracy. Figure 1 shows the forecast graph obtained by the winner model, RNN based on LSTM for the test part of TS 4. Figure 2 shows the forecast graph obtained on the

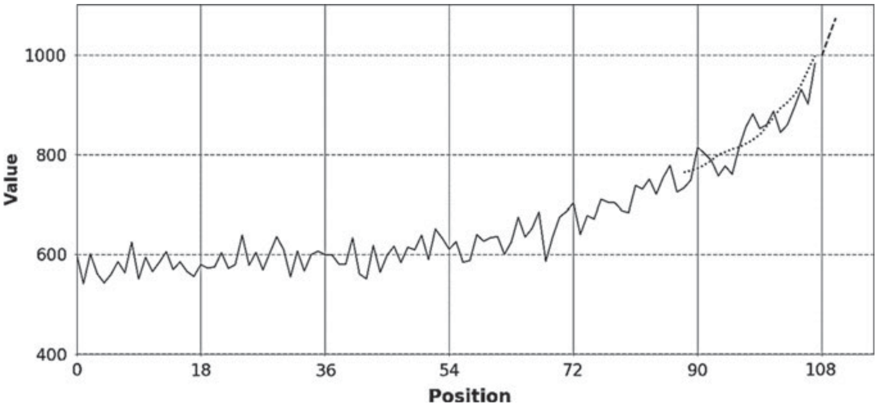


Fig. 1 TS 4 prediction based on RNN with LSTM model

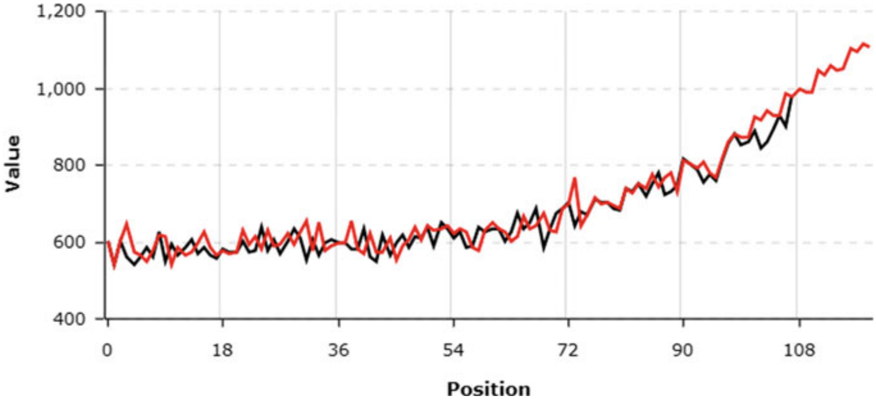


Fig. 2 TS 4 prediction based on two order D-model

basis of the fuzzy D-model [9], the accuracy of which was not the best for the TS 4. Nevertheless, as can be seen from Fig. 2, the predicted values obtained by the fuzzy D-model adequately reflect the behavior of TS 4.

In accordance with Table 2, for the short-term time series, fuzzy TS models won in the average accuracy on the test interval and they showed the best accuracy for 60% of short-term TS. The fuzzy TS models are followed by MLP; the last place is shared by models of the RNN class. In general, the accuracy for the short-term TS for all models under consideration is lower than for the medium-term time series, due to the insufficient number of observations. Note that the greatest decrease in accuracy (almost four times) was shown by the RNN class models; their error in the MAPE criterion averaged 12%. Interestingly, for one short-term time series, the RNN based on the GRU showed a prediction error lower by 2 orders of magnitude compared to other models, and the FTS prediction errors for two TS were half that of competing models. It should be noted that the MLP showed forecasting accuracy better for 40% short-term TS in comparison with the RNNs as can be seen from Table 2.

6 Conclusion

Based on the CIF-2016 TS set, the accuracy of the forecasting models of two classes, based on ANNs and FTS, using the MAPE criterion, was investigated. Models of the ANN class were represented by RNN based on LSTM, RNN based on GRU, and MLP. Models of the fuzzy time series class were presented by model with fuzzified TS values, model with fuzzified first differences of TS values and model, based on the elementary fuzzy tendency. The study was conducted separately for short-term time series and for medium-term time series. For medium-term TS, the smallest error on average was shown by the RNN based on LSTM models, presented by MAPE

= 3.0013% and for the short-term TS, the smallest error, MAPE = 5.7313%, on average was shown by FTS.

The further work will be focused on investigation TS models of ANNs and FTS with respect to time consumption and on their comparative study on forecasting accuracy for more wide variating of TS length.

Acknowledgements The reported study was funded by RFBR and Ulyanovsk region, project number 19-47-730001 and was funded by RFBR, project number 20-07-00672.

References

1. Haykin, S.: Neural Networks: A Comprehensive Foundation. Prentice Hall PTR (1994)
2. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
3. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)
4. Rosenblatt, F.: Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Spartan Books, Washington DC (1961)
5. Cho, K., Merrienboer, B., van Bahdanau, D., Bengio, Y.: On the properties of neural machine translation: encoder-decoder approaches (2014). https://arxiv.org/pdf/1409.1259.pdf. Accessed 10 Nov 2019
6. Dataset of CIF-2016 competition. http://irafm.osu.cz/cif/main.php. Accessed 10 Nov 2019
7. Smyl, S., Ranganathan, J., Pasquam, A.: M4 Forecasting competition: introducing a new hybrid ES-RNN model (2018). https://eng.uber.com/m4-forecasting-competition/. Accessed 10 Nov 2019
8. Song, Q.: Fuzzy time series and its models. Fuzzy Sets Syst. **54**, 269–277 (1993)
9. Hwang, J.R., Chen, S.M., Lee, C.H.: Handling forecasting problem using fuzzy time series. Fuzzy Sets Syst. **100**, 217–228 (1998)
10. Perfilieva, I., Yarushkina, N., Afanasieva, T., Romanov, A.: Time series analysis using soft computing methods. Int. J. Gen. Syst. **42**(6), 687–705 (2013)
11. Khashei, M., Bijari, M.: An artificial neural network (p, d, q) model for time series forecasting. Expert Syst. Appl. **37**(1), 479–489 (2010)
12. Giordano, F., Rocca, M.L., Perna, C.: Forecasting nonlinear time series with neural network sieve bootstrap. Comput. Stat. Data Anal. **51**(8), 3871–3884 (2007)
13. Koskela, T., Lehtokangas, M., Saarinen, J., Kaski, K.: Time series prediction with multilayer perceptron, FIR and Elman neural networks. https://pdfs.semanticscholar.org/82c8/e5d0cd4a7467f7f54ad823b2136b973eeb6e.pdf. Accessed 10 Nov 2019
14. Lipton, Z.C., Berkowitz, J., Elkan, C.: A critical review of recurrent neural networks for sequence learning (2015). https://arxiv.org/abs/1506.00019. Accessed 10 Nov 2019
15. Salehinejad, H., Sankar, S., Barfett, J., Colak, E., Valaee, S.: Recent advances in recurrent neural networks (2018). https://arxiv.org/pdf/1801.01078.pdf. Accessed 10 Nov 2019
16. Bandara, K., Bergmeir, C., Smyl, S.: Forecasting across time series databases using recurrent neural networks on groups of similar series: a clustering approach (2018). https://arxiv.org/pdf/1710.03222.pdf. Accessed 10 Nov 2019
17. Khargharia, H.S., Shakya, S., Ainslie, R., AlShizawi, S., Owusu, G.: Predicting demand in IoT enabled service stations. In: Proceedings of 2019 IEEE Conference on Cognitive and Computational Aspects of Situation Management (CogSIMA 2019) (2019). https://doi.org/10.1109/cogsima.2019.8724239. Accessed 10 Nov 2019
18. Rivero, C.R., et al.: Time series forecasting using recurrent neural networks modified by Bayesian inference in the learning process. In: Proceedings of IEEE Colombian Conference on

Applications of Computational Intelligence (ColCACI 2019), Barranquilla, Colombia (2019). https://doi.org/10.1109/colcaci.2019.8781984. Accessed 10 Nov 2019

19. Laptev, N., Yosinski, J., Li, E., Smyl, S.: Time-series extreme event forecasting with neural networks at Uber. In: Proceedings of International Conference on Machine Learning, no. 34, pp. 1–5 (2017)
20. Che, Z., Purushotham, S., Cho, K., Sontag, D., Yan, L.: Recurrent neural networks for multivariate time series with missing values (2018). https://arxiv.org/pdf/1606.01865.pdf. Accessed 10 Nov 2019
21. Salinas, D., Flunkert, V., Gasthaus, J., Deep, A.R.: Probabilistic forecasting with autoregressive recurrent networks (2019). https://arxiv.org/pdf/1704.04110.pdf. Accessed 10 Nov 2019
22. Hyndman, R., Koehler, A.B., Ord, J.K., Snyder, R.D.: Forecasting with Exponential Smoothing: The State Space Approach. Springer Series in Statistics. Springer (2008). ISBN 9783540719182
23. Namini, K.S.S., Tavakoli, N., Namin, A.S.: A comparison of ARIMA and LSTM in forecasting time series. In: Proceedings of 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA) (2018). https://doi.org/10.1109/icmla.2018.00227
24. G´abor Petneh´azi. Recurrent neural networks for time series forecasting (2019). https://arxiv.org/pdf/1901.00069. Accessed 10 Nov 2019
25. Afanasieva, T., Platov, P.: The study of recurrent neuron networks based on GRU and LSTM in time series forecasting. In: Proceedings of International Conference on Time Series and Forecasting (ITISE 2019), 25–27 Sept., Granada (Spain), Godel Impresions Digitales, vol. 1, pp. 190–201 (2019). ISBN 978-84-17970-79-6
26. Afanasieva, T., Yarushkina, N., Sibirev, I.: Time series clustering using numerical and fuzzy representations. In: Proceedings of Joint 17th World Congress of International Fuzzy Systems Association and 9th International Conference on Soft Computing and Intelligent Systems (IFSA-SCIS 2017) (2017). https://doi.org/10.1109/ifsa-scis.2017.8023356
27. Chen, S.M.: Forecasting enrollments based on fuzzy time series. Fuzzy Sets Syst. **81**, 311–319 (1996)
28. Chen, S.M., Hwang, J.R.: Temperature prediction using fuzzy time series. Trans. Syst. Man Cybern. Part B: Cybern. **30**(2), 263–275 (2000)
29. Tsai, C.C., Wu, S.J.: Forecasting enrollments with high-order fuzzy time series. In: Proceeding of 19th International Conference of the North American Fuzzy Information Processing Society, pp. 196–200 (2000)
30. Yarushkina, N., et al.: Time series processing and forecasting using soft computing tools. In: Lecture Notes in Computer Science, Vol. 6743, Proceedings of 13-th International Conference RSFDGrC-2011, vol. XIII, pp. 155–163. Springer (2011)
31. Sahin, A., Furkan Dodurka, M., Kumbasar, T., Yesil, E., Siradag, S.: Review study on fuzzy time series and their applications in the last fifteen years. In: Proceedings of International Fuzzy Systems Symposium (FUZZYSS'15), At İstanbul, vol. 4, pp. 166–170 (2015)
32. Afanasieva, T., Yarushkina, N., Gyskov, G.: The study of basic fuzzy time series Forecasting models. In: World Scientific Proceedings on Computer Engineering and Information Science, vol. 10. Uncertainty Modelling in Knowledge Engineering and Decision Making. Proceedings of the 12th International FLINS Conference ENSAIT (FLINS 2016), pp. 295–300 (2016). https://doi.org/10.1142/9789813146976_0049
33. Pandey, A.K., Sinha, A.K., Srivastava, V.K.: A comparative study of neural-network & fuzzy time series forecasting techniques—case study: wheat production forecasting. IJCSNS Int. J. Comput. Sci. Netw. Secur. **8**(9), 382–387 (2008). https://www.researchgate.net/publication/254027822_A_Comparative_Study_of_Neural-Network_Fuzzy_Time_Series_Forecasting_Techniques_-_Case_Study_Wheat_Production_Forecasting. Accessed 10 Nov 2019
34. Yadov, V.K., et al.: A comparative study of neural-network & fuzzy time series forecasting techniques—case study: marine fish production forecasting. Indian J. Geo-marine Sci. **42**(6), 707–716 (2013). https://pdfs.semanticscholar.org/2a13/49e68f1b3208bf7997c71748f053a082ac28.pdf. Accessed 10 Nov 2019

35. Chung, J., Gulcehre, C., Cho, K.H., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling (2014). https://arxiv.org/pdf/1412.3555.pdf. Accessed 10 Nov 2019
36. Afanasieva, T., Yarushkina, N., Toneryan, M., Zavarzin, D., Sapunkov, A., Sibirev, I.: Time series forecasting using fuzzy techniques. In: Proceeding of International Joint Conference IFSA-EUSFLAT (16th World Congress of the International Fuzzy Systems Association (IFSA), 9th Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT)), June 30th–July 3rd, Gijon (Asturias) Spain), 2015, pp. 1068–1075 (2015)
37. Time Series Prediction with LSTM Recurrent Neural Networks in Python with Keras: (2016). https://machinelearningmastery.com/time-series-prediction-lstm-recurrent-neural-networks-python-keras/. Accessed 10 Nov 2019

Inflation Rate Forecasting: Extreme Learning Machine as a Model Combination Method

Jeronymo Marcondes Pinto and Emerson Fernandes Marçal

Abstract Inflation rate forecasting is one most discussed topics on time-series analysis due to its importance on macroeconomic policy. The majority of these papers' findings point out that forecasting combination methods usually outperform individual models. In this sense, we evaluate a novel method to combine forecasts based on Extreme Learning Machine Method [15], which is becoming very popular but, to the best of our knowledge, has not been used to this purpose. We test Inflation Rate forecasting for a set of American countries, for one, two, three, ten, eleven and twelve steps ahead. The models to be combined are automatically estimated by *R forecast* package, as SARIMA, Exponential Smoothing, ARFIMA, Spline Regression, and Artificial Neural Networks. Another goal of our paper is to test our model against classical combination methods such Granger Bates, Linear Regression, and Average Mean of models as benchmarks, but also test it against basic forms of new models in the literature, like [8, 10, 26]. Therefore, our paper also contributes to the discussion of forecast combination by comparing versions of some methods that have not been tested against each other. Our results indicate that none of these methods have an indisputable superiority against the others, however, the Extreme Learning Method proved to be the most efficient of all, with the smaller Mean Absolute Error and Mean Squared Error for its predictions.

J. M. Pinto (✉)
Brazilian Ministry of Economics, Sao Paulo, Brazil
e-mail: jeronymomp@gmail.com

E. F. Marçal
Sao Paulo School of Economics, Sao Paulo, Brazil
e-mail: emerson.marcal@fgv.br

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_24

1 Introduction

The inflation rate is a core indicator of economic activity. This indicator is closely monitored by policy-makers, practitioners, portfolio managers, and economic researchers owing to its importance in macroeconomic policy. Therefore, inflation rate time-series forecasting is a trending topic in forecasting literature.

For example, inflation rate forecasting has been discussed in several classical papers such as [6, 21, 23, 27], which remain relevant even today.

One of the main discussions in inflation forecasting literature centers on the role of forecasting combinations in improving the predictions of models. Most papers indicate that a combination of models usually increases forecasting performance.

The seminal work of [1] suggested that a simple forecast combination, such as simple or rolling weighted averages, can outperform individual models. The importance of model combinations has been highlighted in recent papers such as [4, 14].

Reference [24] reviews classical methods to combine forecasts, such as by generating prediction weights based on a linear regression or by giving equal weights to all methods, similar to taking the average mean of all models. Reference [1] proposed combining forecasts based on a weighted average of each model's mean squared errors.

However, the discussion of which is the best way to combine forecasts is still open to debate, with several papers suggesting new methods to obtain higher forecast accuracy. Reference [8] proposes a forecast ensemble based on LASSO (*Lasso*) that selects and shrinks toward equal combining weights. Reference [10] develops a method based on the model confidence set (*MCS*) of [12], which allows the user to equally combine forecasts selected by the MCS. Reference [26] evaluates the performance of a forecast combination with weights calculated by an Artificial Neural Network (ANN), using a multilayer perceptron architecture (*Mlp*). All these papers develop their proposed methods against some classical benchmarks. This is increasingly linked to the actual research on machine learning literature and its possibilities to improve forecasts.

Based on the work of [15], we propose a new way to combine forecasts, with weights estimated using Extreme Learning Machine method (*Elm*). This method has proved to be a very efficient machine learning approach to forecasting, with good accuracy results, as discussed in [2, 25], and excellent algorithm performance. In this sense, this paper contributes to the forecasting literature by evaluating a new method to combine forecasts.

The following time-series models are used to generate forecasts to be combined or selected: exponential smoothing, SARIMA, artificial neural networks (ANNs), ARFIMA, and Spline Regression. All of these models' functional specifications are automatically provided by *forecast R* package.

Through this work, we extend our previous work presented at the 6th International Conference on Time Series and Forecasting [19]. We run a pseudo-real-time forecast exercise to evaluate the forecasting performance of our strategy by applying it to the inflation growth rates of a set of American countries: Brazil, Mexico, Chile, Peru, Canada, and the United States. The data are monthly and were obtained from the Bank of International Settlements (BIS) (https://www.bis.org/). We forecast this series for one, two, three, ten, eleven, and twelve steps ahead.

We compare our forecast results to forecasts produced using classical benchmarks, such as random walk (*RandomWalk*), average mean (*AverageMean*), and linear regression (*LinearRegression*), as discussed in [24] as well as [1] (*GB*).

Another goal of this paper is to test some of the new combination methods that have been published in the forecasting literature, focusing on the machine learning aggregating models. Our method is compared to versions of the recent models proposed by [8, 10, 26]. To the best of our knowledge, no one has tried to compare the accuracy of these approaches. In addition to these models, we test a combination method based on ridge regression (*Ridge*), as an extension of [8].

The reader must be attentive over the approach in this study regarding the use of these methods. We do not necessarily use the algorithm used by the original author, but rather, one that is based on their central idea. For example, in the case of [26], the author uses a network architecture and a backpropagation schema specific to his problem, which we do not replicate here; instead, we only borrow the central idea of using an *Mlp* to estimate the weights in the combination.

Therefore, our paper contributes to the discussion of new forecasting methods combined with machine learning techniques, by proposing a new method based on Extreme Learning Machine. Our paper also contributes to the evaluation of some new methods that have not previously been tested against each other.

This paper is organized as follows. Section 1 discusses our proposed strategy to generate forecasts. Section 2 reports our forecasting strategy. Section 3 shows our results and discusses the merits and pitfalls of our strategy. Finally, some concluding remarks are drawn.

2 Extreme Learning Machine Method

The *Elm* algorithm was proposed by [15] and is based on a single hidden layer feedforward neural network (SLNN); it is designed to address the usual problems in the artificial neural networks literature, such as the method's speed.

For M arbitrary samples (x_i, t_i), with $x \in \mathbf{R}^n$ being the input and $t \in \mathbf{R}^m$ the output of a given econometric problem, a standard way to model an SLNN with an activation function given by $g(x)$ is

$$\sum_{i=1}^{M} \beta_i g_i(x_i) = \sum_{i=1}^{M} \beta_i g_i(w_i x_i + b_i) = o_j, \, j = 1, \ldots, N, \tag{1}$$

where $w_i = [w_{i1}, w_{i2}, \ldots, w_{in}]^T$ is the vector of weights that connects the input layer to a hidden layer, $\beta_i = [\beta_{i1}, \beta_{i2}, \ldots, \beta_{im}]^T$ is the set of weights between the output and hidden nodes, o_j is the tested output, and b_i is the threshold of the ith hidden neuron.

The SLNN with N hidden neurons and $g(x)$ activation function can approximate these N samples with zero error, as $\sum_{j=1}^{n} ||o_j - t_j|| = 0$, and there exist β_i, w_i, and b_i such that

$$\sum_{i=1}^{M} \beta_i g_i(w_i x_i + b_i) = t_j, j = 1, \ldots, N. \tag{2}$$

This equation can be written as follows:

$$H\beta = T, \tag{3}$$

where H = $\begin{pmatrix} g(w_1 x_1 + b_1) & \ldots & g(w_N x_1 + b_N) \\ \vdots & \ldots & \vdots \\ g(w_1 x_N + b_1) & \ldots & g(w_N x_N + b_N) \end{pmatrix}$,

β = $\begin{pmatrix} \beta_1^T \\ \vdots \\ \beta_N^T \end{pmatrix}$ and T = $\begin{pmatrix} t1^T \\ \vdots \\ tN^T \end{pmatrix}$.

According to [15], contrary to the common understanding that all parameters of SLNN must be tuned, experiments show that they can be arbitrarily given. Reference [15] indicates that for small parameter values in the activation function, training an SLNN is simply equivalent to finding the least-squares solution β of the linear system $H\beta = T$:

$$min_{\beta}||H(w_1, \ldots, w_n, b_1, \ldots, b_n)\beta - T||. \tag{4}$$

Based on this method, our paper proposes solving the problem given by (4) to obtain the weights for combining different forecast models. Therefore, our inputs are the forecasts of different models while the output is the actual value of the predicted variable. To the best of our knowledge, no paper has used this approach.

The *Elm* architecture is defined by a process of cross-validation applied to different sets of networks. Our method chooses the number of hidden nodes through an analysis of the least mean absolute error generated on the training set. For our purposes, we tested five, ten, fifteen, twenty, twenty-five, and thirty possible hidden nodes.

3 Proposed Benchmarks and Forecasting Strategy

3.1 Benchmarks

In this subsection, we expose new methods that use the machine learning framework, which were used to combine forecasts in recent studies. Essentially, in all of these methods, the explanatory variable is given by each forecast to be combined while the dependent variable is the series value to be predicted.

LASSO and Ridge Regression

Reference [8] proposed using a LASSO-based procedure that selects and shrinks toward equal combining weights (*Lasso*). They aim to find a method that can select the best predictors to combine forecasts. In this study, we use a basic form of LASSO regression to develop a forecast ensemble.

LASSO estimates are given by

$$\beta^{Lasso} = argmin_{\beta} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2. \tag{5}$$

subject to $\sum_{j=1}^{p} |\beta_j| \leq t$.

Based on this maximization problem, it is possible to establish the weights for each forecast (x_{ij}), even zero.

In the same way, the ridge regression estimate (*Ridge*) is given by

$$\beta^{Lasso} = argmin_{\beta} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2. \tag{6}$$

subject to $\sum_{j=1}^{p} \beta_j^2 \leq t$. Here, x_{ij} refers to each of the forecasts that will be combined in our experiment.

It is possible to infer that the basic difference between *Lasso* and *Ridge* is the maximization restriction, which is given by the absolute value of β in the former case and it's square in the case of the ridge regression. In our study, we use LASSO and the ridge regression with a basic framework, without some of the specifications discussed in [8].

Artificial Neural Network

Reference [26] used artificial neural network as part of a framework of forecast combination. Basically, they use a multilayer perceptron artificial neural network (*Mlp*) given by

$$y_i = \sum_{j=1}^{m} f(w_{ij}x_j + b_i). \tag{7}$$

In this study, we use a feedforward neural network, where each input x_j feeds its value to a hidden neuron, known as hidden layers, until the final output is obtained from the neural network. During its passage by each neuron, the input value is multiplied for its respective weight w_{ij}. For more details about this method and *Mlp* architecture, see [9].

In this study, we use a simple *Mlp* architecture with three layers and a logistic activation function, which was defined by experimentation based on the best results.

MCS
Reference [12] introduced the concept of a model confidence set (*MCS*). *MCS* is a set of models that is constructed such that it contains the best model with a given level of confidence. The *MCS* is analogous to a confidence interval for a parameter. Reference [10] evaluated the use of *MCS* for selecting the best combination of models to generate a forecast ensemble.

Classical Methods to Forecast Combination as Benchmarks
To evaluate any strategy, it is important to choose proper benchmarks. If a strategy is unable to outperform forecasts obtained from simple benchmarks, it should be abandoned. Simple benchmarks serve as a lower bound to assess any strategy. For example, if the analyst wants to forecast an exchange rate, random walk is a difficult benchmark to be beaten [18, 20]. An autoregressive model of order 1 is a difficult benchmark to surpass when forecasting a consumer price index [3, 22]. A forecast obtained from a double difference model can be difficult to outperform for data where the data generation process faces structural breaks [5].

We use the following classic benchmarks:

- Average forecast combination [24];
- Linear regression of forecasts [24];
- Granger Bates method [1];
- Random walk [24].

3.2 *Forecasting Strategy*

We test our models on inflation growth rate data for a set of American countries: Brazil, Peru, Mexico, Chile, Canada, and the United States.

The strategy in our forecasting exercise is based on the following schema:

- Training set equals 50% of data;
- Validation set equals 40%—"number of steps ahead to forecast" + 1 of data;
- Test set equals the total data minus (Training Set + Validation Set).

All series were tested with the Augmented Dickey-Fuller test and the results indicated that there is a unit root with 1% of confidence. Therefore, all of our experiments are based on the inflation growth rate, which is stationary at 1% confidence.

The experiment was developed by applying all cited models and methods to the inflation growth rate series. All of the tested combination methods aim to select the best ensemble of models from a set of possible choices. Specifications for each model are selected using algorithms from *forecast R* package:

- SARIMA,
- Exponential Smoothing (Ets),
- Artificial Neural Network (ANN),
- ARFIMA,

- Spline Regression (Spline).

All models generated are univariate, based on the use of only the lags of the inflation rate as the information set, and no other variables.

The *forecast* package is described in detail in [16]. For more details regarding the aforementioned methodologies, see [11, 17]. This package chooses a particular specification based on the information available. Model performance may vary throughout the sample.

4 Findings

4.1 Pseudo-Real-Time Experiment

The data gathered for the countries is used to create many variations of models to forecast the inflation growth rate. The sample is split into three parts. The first part of the sample is used to estimate the individual models, the second is used to train and combine the estimates, while the third is used to evaluate the forecast performance of the various methods over various horizons. In our exercise, we attempt to simulate a real-time operation. We use an information set that reflects, as closely as possible, the one available to agents at the time of the forecast.

For each model, forecasts are generated for one, two, three, ten, eleven, and twelve steps ahead. Therefore, our initial training set includes the first 228 observations. The values we use to run our projections are not the same as those that were available to agents at that time. We run projections in our pseudo-real-time experiment with a slightly better information set. This may result in better forecasting accuracy compared to the projections generated in real-time. To assess the predictive performance of the proposed models, a comparison of Mean Absolute Error (*MAE*) and Mean Squared Error (*MSQE*) is generated for each method. Tables 1 and 2 present a summary of the best models in terms of *MAE* and *MSQE* from one up to twelve steps ahead forecasts. All tables show the first and second models in terms of forecasting performance and if the first models have statistical dominance over the second.

It is possible to infer that *Elm* mechanisms show good performance in comparison to the others achieving, approximately, 72% of the best results in terms of *MAE* and *MSQE* in all experiments. Additionally, even in the cases where the *Elm* method was not the best model, it had good performance being one of the two best models in almost all cases. The tables with detailed results are shown in the Appendix section.

To compute the statistical significance of these results, we use the method in [7]. We apply the Diebold-Mariano method to the pair of all models tested against our *Elm* strategy. We intend to analyze whether a model has statistically lesser *MAE* and/or *MSQE* than the other. Our statistical test analyzes the null hypothesis that the second model is less accurate than the first. Detailed results are available by request to the author.

Table 1 Models with the lowest mean absolute error for 1, 2, 3, 10, 11, and 12 steps ahead forecasts

	Brazil	Chile	Peru	Mexico	USA	Canada
1 Step Ahead	1° Elm*	1° Lasso	1° Lasso	1° Elm	1° Elm	1° Elm
	2° Lasso	2° Ridge	2° Ridge	2° Ridge	2° Lasso	2° Lasso
2 Steps Ahead	1° Elm*	1° Ridge	1° Ridge	1° Lasso	1° Elm	1° Elm
	2° Gb	2° Lasso	2° Elm	2° Elm	2° Lasso	2° Lasso
3 Steps Ahead	1° Elm	1° Ridge	1° Elm**	1° Lasso	1° Elm	1° Elm
	2° Gb	2° Elm	2° Ridge	2° Ridge	2° Lasso	2° Lasso
10 Steps Ahead	1° Gb	1° Elm	1° Elm**	1° Elm***	1° Elm	1° Elm**
	2° Elm***	2° Ridge	2° Gb	2° Lasso	2° Lasso	2° Lasso
11 Steps Ahead	1° Gb	1° Elm	1° Elm***	1° Elm**	1° Elm	1° Elm
	2° Elm**	2° Ridge	2° Gb	2° Lasso	2° Lasso	2° Lasso
12 Steps Ahead	1° Gb	1° Elm	1° Elm***	1° Elm**	1° Elm	1° Elm
	2° Elm**	2° Gb	2° Gb	2° Lasso	2° Lasso	2° Lasso

Significances: $^{*}p < 0.1$; $^{**}p < 0.05$; $^{***}p < 0.01$

Table 2 Models with the lowest mean squared error for 1, 2, 3, 10, 11, and 12 steps ahead forecasts

	Brazil	Chile	Peru	Mexico	USA	Canada
1 Step Ahead	1° Elm*	1° Lasso	1° Lasso	1° Elm	1° Elm	1° Ridge
	2° Lasso	2° Ridge	2° Ridge	2° Ridge	2° Ridge	2° Lasso
2 Steps Ahead	1° Elm*	1° Ridge	1° Ridge	1° Lasso	1° Elm	1° Elm
	2° Lasso	2° Gb	2° Elm	2° Elm	2° Lasso	2° Lasso
3 Steps Ahead	1° Elm	1° Ridge	1° Elm**	1° Lasso	1° Elm	1° Elm
	2° Gb	2° Gb	2° Ridge	2° Ridge	2° Lasso	2° Lasso
10 Steps Ahead	1° Elm***	1° Elm	1° Elm**	1° Elm***	1° Elm	1° Elm**
	2° Gb	2° Ridge	2° Gb	2° Lasso	2° Lasso	2° Ridge
11 Steps Ahead	1° Elm**	1° Elm	1° Elm***	1° Elm**	1° Elm	1° Elm
	2° Gb	2° Ridge	2° Gb	2° Lasso	2° Lasso	2° Ridge
12 Steps Ahead	1° Elm**	1° Elm	1° Elm***	1° Elm**	1° Elm	1° Elm
	2° Gb	2° Ridge	2° Gb	2° Lasso	2° Lasso	2° Ridge

Significances: $^{*}p < 0.1$; $^{**}p < 0.05$; $^{***}p < 0.01$

To compare the forecast accuracy of the two different methods, we use the alternative version of the Diebold-Mariano test as proposed in [13]. We test the alternative hypothesis that a second method is less accurate than the *Elm* strategy.

Exercises performed with the Brazilian data show outstanding results for Extreme Learning Machine combination. Based on one up to the twelve ahead forecasts, the *Elm* strategy does not only statistically outperformed *Gb* model with three steps ahead forecast. This performance can be seen in Fig. 1 in our Appendix.

The *Elm* combination had the worst forecast performance in the case of the Chilean data. We can only reject the null that *Gb* performed better than *Elm* at 15% for the twelve steps ahead forecast. However, *Elm* outperformed the *AverageMean* for all steps from ten to twelve at 7%, 12%, and 14%, respectively. The MAE and MSQE dynamic can be seen in Fig. 2 in the appendix section.

Our results on the Mexican data show that the *Elm* method outperformed all models for two, three, ten, eleven, and twelve steps ahead forecasts with statistical dominance dictated by the test results of [13]. However, the model did not show satisfactory performance in the short run, as evident from the results of the one step ahead forecast.

In the Peruvian case, *Elm* presented a result very similar to that of the Mexican case. Our model statistically outperformed all models, for three, ten, eleven, and twelve steps ahead forecasts. Detailed results are in the Table. The results for the Mexican and Peruvian cases can be inferred by a graphical analysis in Figs. 3 and 4, respectively.

North American countries, the USA and Canada, showed very similar results. In both cases, *Elm* was the model with the lowest *MAE* and *MSQE* in all experiments. However, Diebold-Mariano's test was only able to reject the null of equal performances for ten steps ahead forecasts in the Canadian case. A graphical analysis of those results allows us to infer that besides Diebold-Mariano's results, it seems that the *Elm* model has dominance over all other models during the training and test phase. It is possible that this inability to reject the null was driven by a very stable series, with low variance, which is typical of North American countries' inflation rates, but further experiments should be conducted for a more conclusive inference. These graphics can be seen in our Appendix.

It is also possible to evaluate the adjustment during the training and testing period. All graphical adjustment analyses are described in the Appendix. It is very useful to evaluate the dynamics of the method; however, for this exercise, we opted to show only the one and twelve steps ahead forecasts, focusing on the short and long runs.

Our results follow the same pattern of the findings of [15]. In our tests, *Elm* showed satisfactory performance in terms of computation time. All experiments took less than three seconds to compute. This result shows Extreme Learning Machine as a fast algorithm without losing in terms of performance. It is worth noting that all of our proposed *Elm* architectures statistically surpassed the *Mlp* combination approach.

5 Concluding Remarks

Inflation rate is one of the most important economic indicators to forecast. As a result, inflation rate forecasting was and still is one of the most discussed topics in time-series forecasting.

Our work analyzed a new forecast combination framework based on Extreme Learning Machine framework proposed by [15] to forecast the inflation rate. We tested our model against some recent proposed combination methods and classical benchmarks. To perform this exercise, we used a time series of the price index growth rate for a set of American countries: Brazil, Chile, Mexico, Peru, Canada, and the United States.

It is possible to infer that the *Elm* mechanisms show good performance in comparison to the others, achieving, approximately, 72% of the best results in terms of *MAE* and *MSQE* in all experiments. Additionally, even in the cases where the *Elm* method was not the best model, it had good performance, being one out of the two best models in almost all cases. All of these results were statistically validated by the use of the testing method in [13] to test the null whether the compared benchmark has better performance than *Elm*.

It is also important to consider the algorithm speed, one of the main advantages of the *Elm* over other artificial neural network architectures. All of our experiments were performed in less than 10 seconds.

Our results proved that Extreme Learning Machine combination method has great potential, which raises the research question of what kind of different architectures could be applied to this model to obtain even better performance. In this sense, future work on this machine learning technique can improve our actual forecasting combination methods.

It is worth noting that our conclusions are not a general theory, with results only applicable to the cases analyzed here. In this sense, it is possible to overcome some of this work's limitations by extending the analysis to more and different countries, as by using different benchmarks.

Appendix

See Figs. 5, 6, 7, 8, 9, 10, 11, 12.

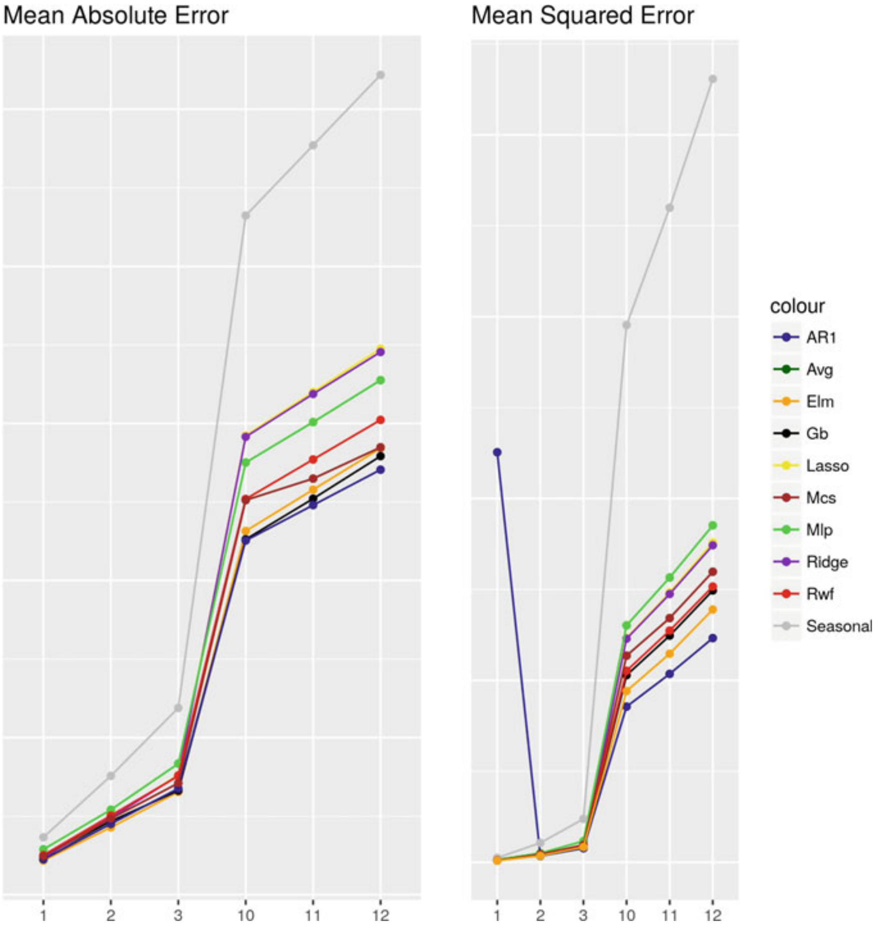


Fig. 1 Mean absolute error and mean squared error for Brazilian forecasting exercise

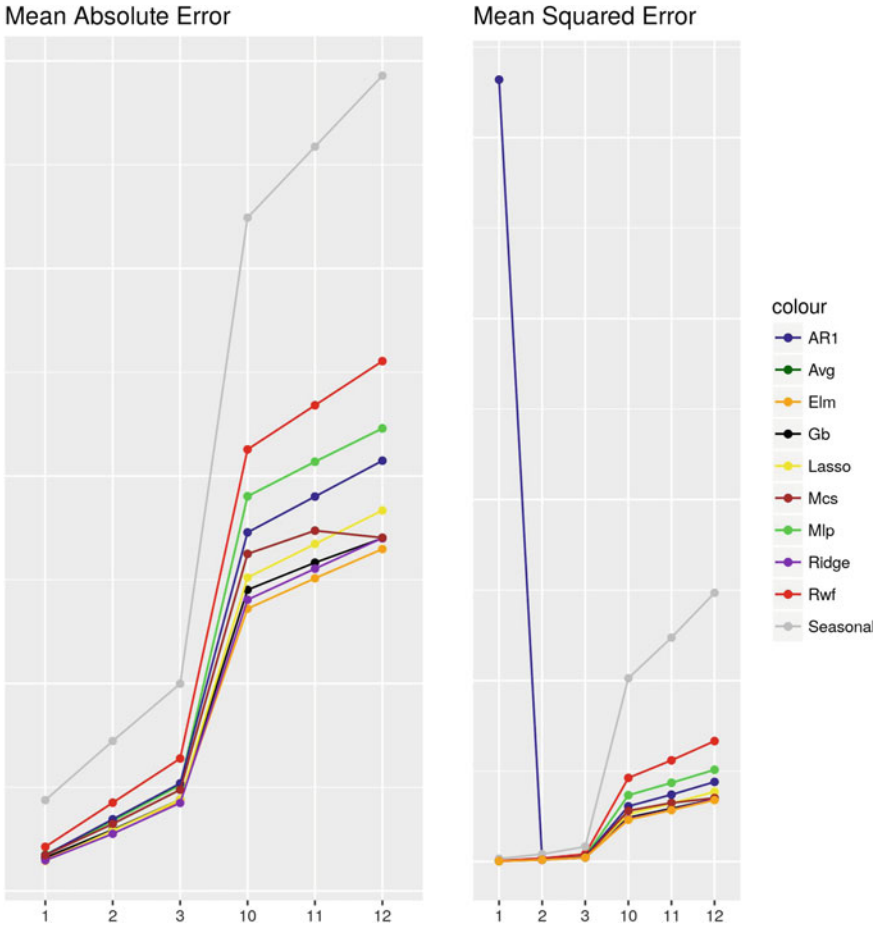


Fig. 2 Mean absolute error and mean squared error for Chilean forecasting exercise

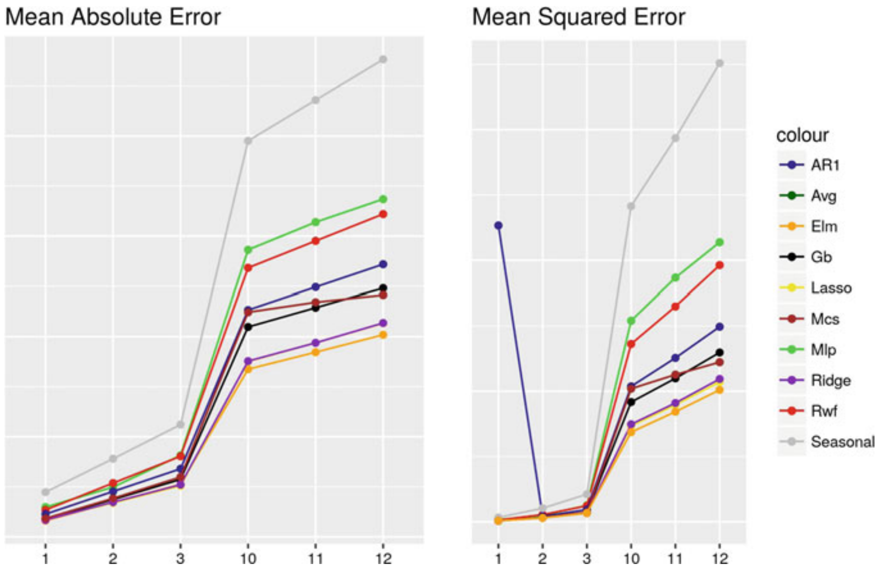


Fig. 3 Mean absolute error and mean squared error for Mexican forecasting exercise

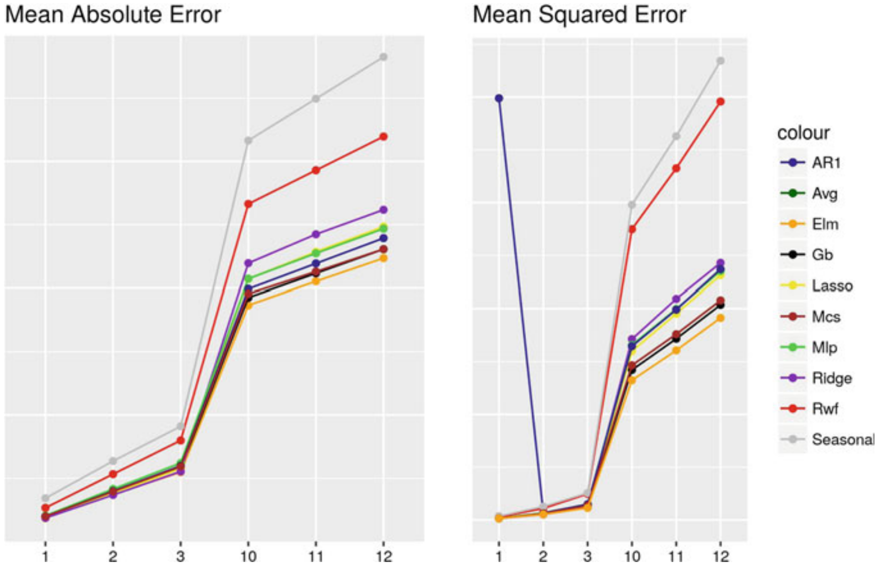


Fig. 4 Mean absolute error and mean squared error for Peruvian forecasting exercise

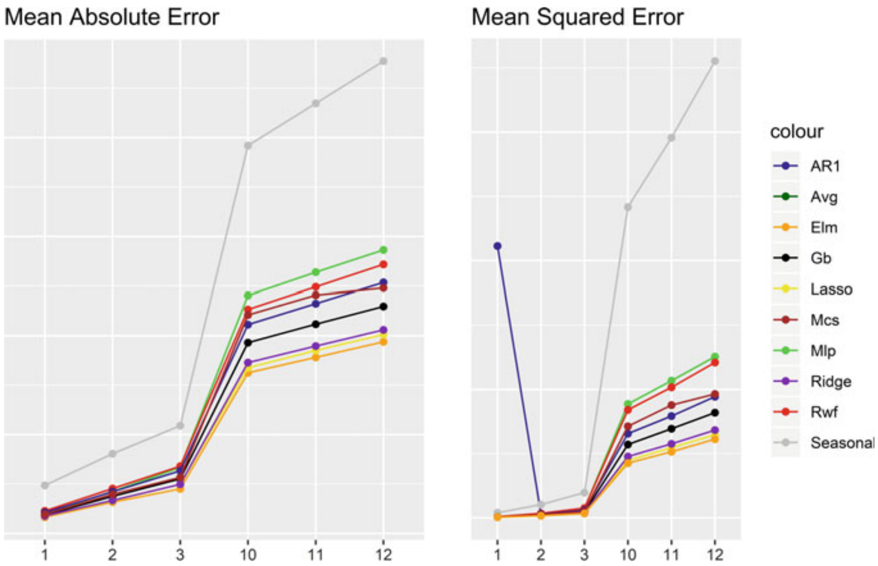


Fig. 5 Mean absolute error and mean squared error for USA forecasting exercise

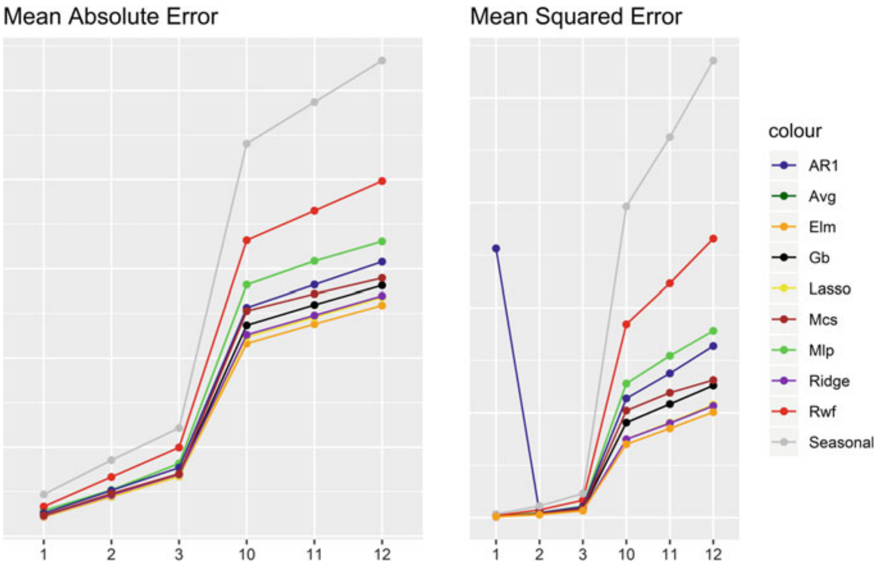


Fig. 6 Mean absolute error and mean squared error for Canada forecasting exercise

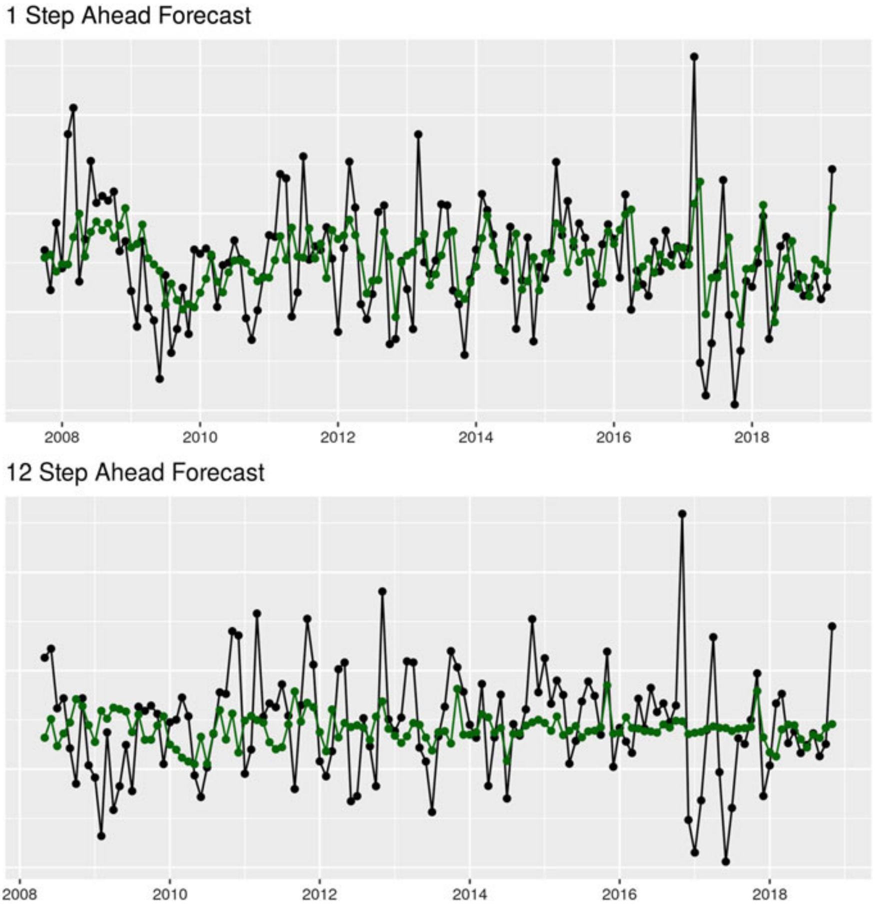


Fig. 7 Statistical adjustment of *Elm* method for Brazil, from one up to twelve steps ahead forecasts

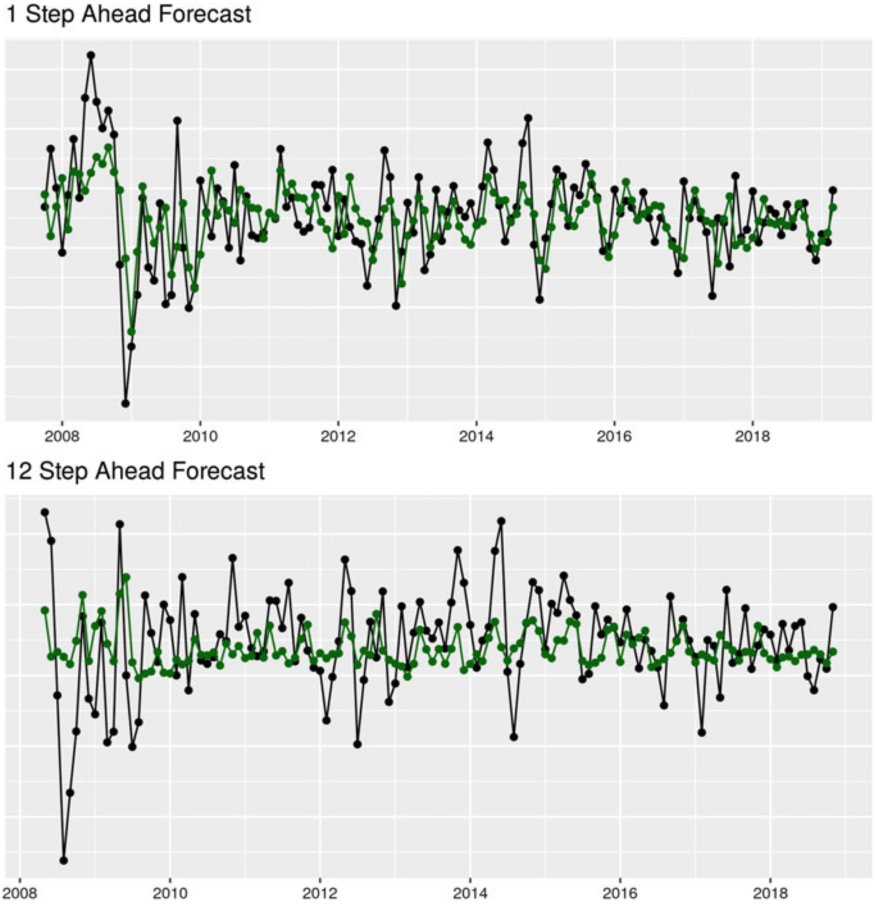
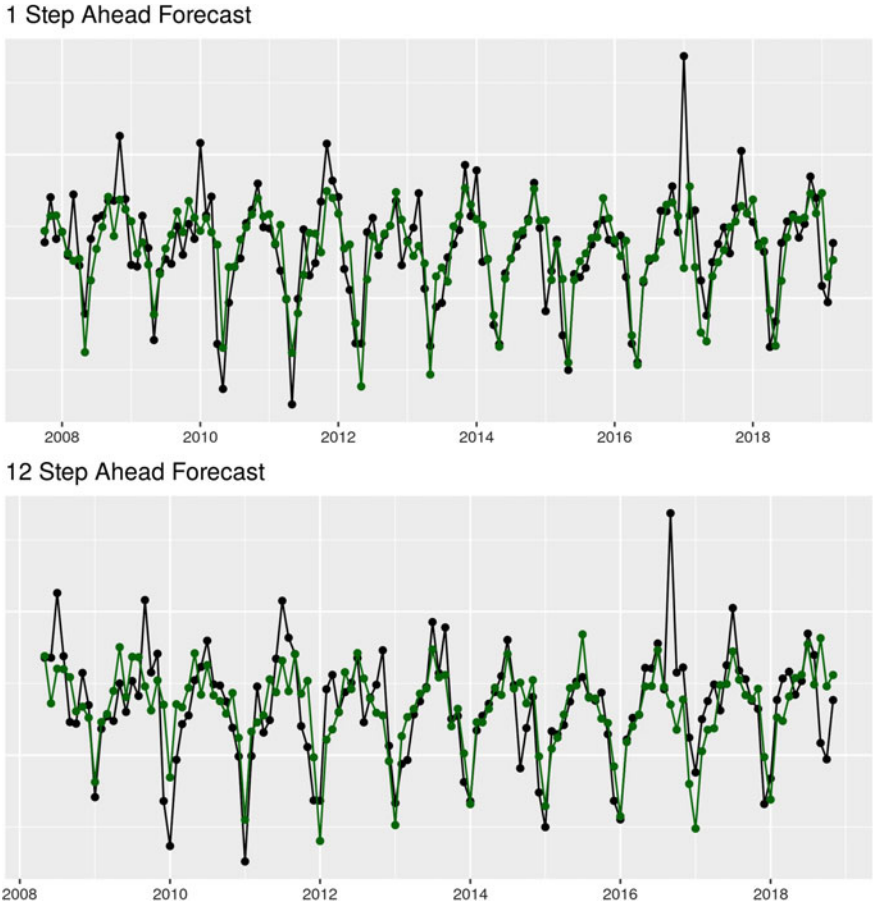


Fig. 8 Statistical adjustment of *Elm* method for Chile, from one up to twelve steps ahead forecasts


1 Step Ahead Forecast

12 Step Ahead Forecast

Fig. 9 Statistical adjustment of *Elm* method for Mexico, from one up to twelve steps ahead forecasts

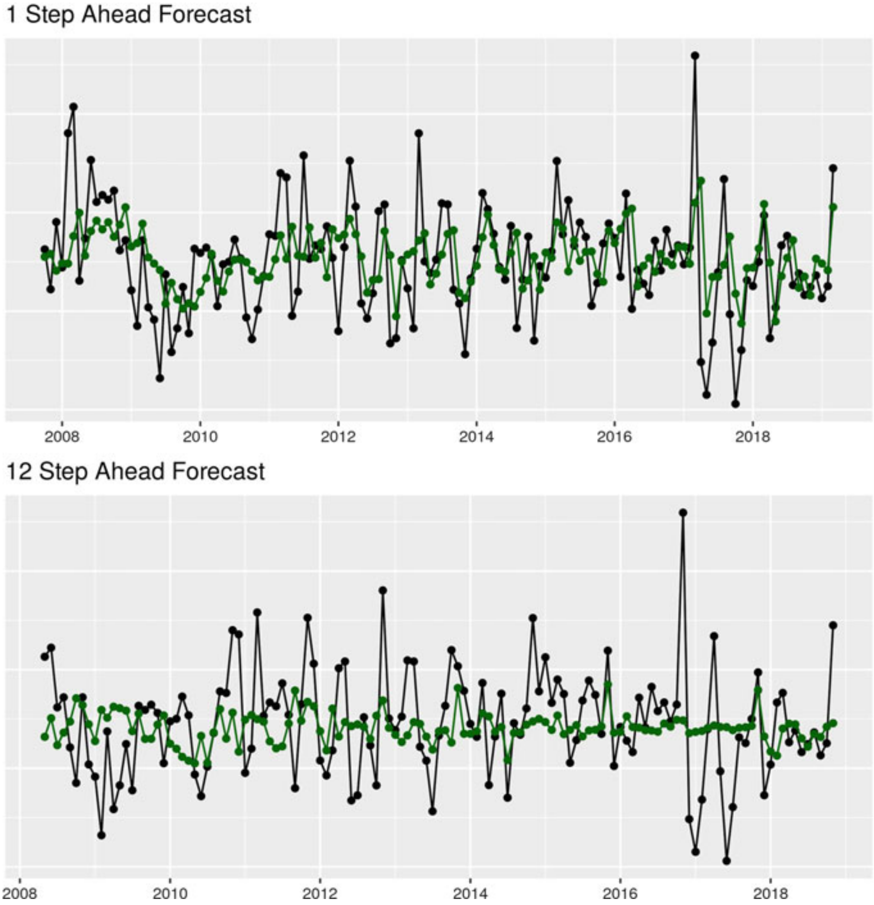


Fig. 10 Statistical adjustment of *Elm* method for Peru, from one up to twelve steps ahead forecasts

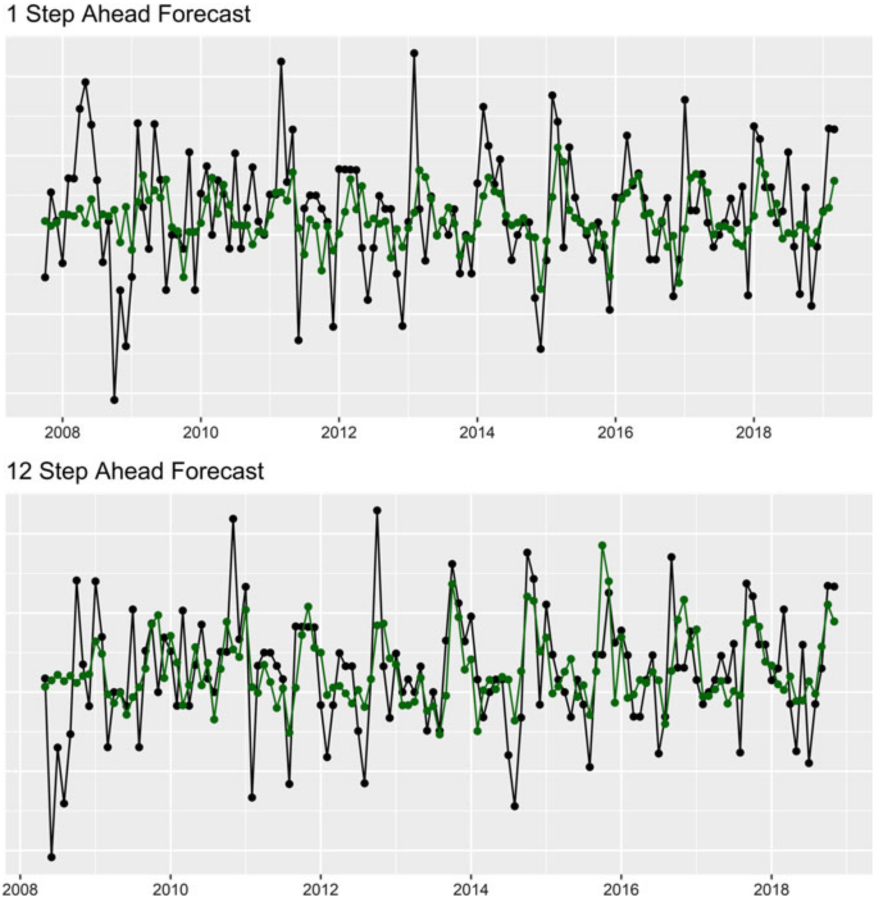
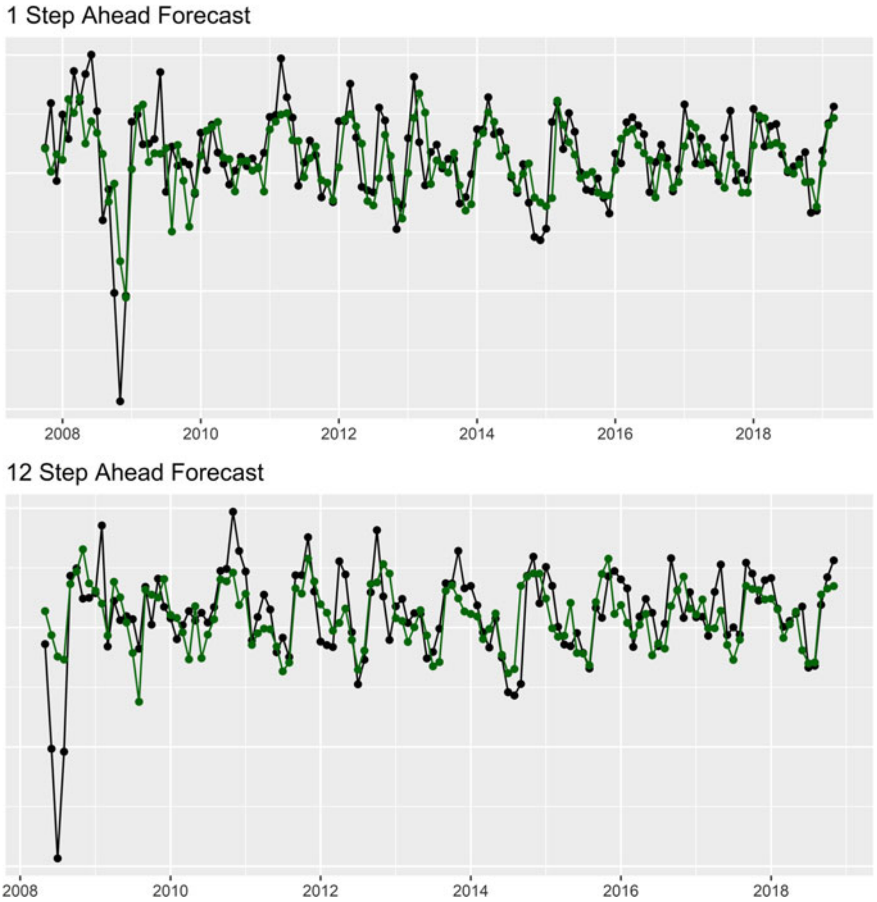


Fig. 11 Statistical adjustment of *Elm* method for Canada, from one up to twelve steps ahead forecasts


1 Step Ahead Forecast

12 Step Ahead Forecast

Fig. 12 Statistical adjustment of *Elm* method for USA, from one up to twelve steps ahead forecasts

References

1. Bates, J.M., Granger, C.W.J.: The combination of forecasts. J. Oper. Res. Soc. **20**(4), 451–468 (1969)
2. Behbahani, H., Amiri, A.M., Imaninasab, R., Alizamir, M.: Forecasting accident frequency of an urban road network: a comparison of four artificial neural network techniques. J. Forecast. **37**(7), 767–780 (2018)
3. Castle, J.L., Clements, M.P., Hendry, D.F.: Forecasting by factors, by variables, by both or neither? J. Econ. **177**(2), 305–319 (2013)
4. Chan, F., Pauwels, L.L.: Some theoretical results on forecast combinations. Int. J. Forecast. **34**(1), 64–74 (2018)
5. Clements, M.P., Hendry, D.F.: Forecasting Non-Stationary Economic Time Series. Mit Press (2001)
6. Deutsch, M., Granger, C.W., Teräsvirta, T.: The combination of forecasts using changing weights. Int. J. Forecast. **10**(1), 47–57 (1994)
7. Diebold, F.X., Mariano, R.S.: Comparing predictive accuracy. J. Bus. Econ. Stat. **20**(1), 134–144 (2002)
8. Diebold, F.X., Shin, M.: Machine learning for regularized survey forecast combination: partially-egalitarian lasso and its derivatives. Int. J. Forecast. (2018)
9. Friedman, J., Hastie, T., Tibshirani, R.: The Elements of Statistical Learning, vol. 1. Springer series in statistics New York (2001)
10. Garcia, M.G.P., Medeiros, M.C., Vasconcelos, G.F.R.: Real-time inflation forecasting with high-dimensional models: the case of Brazil. Int. J. Forecast. **33**(3), 679–693 (2017)
11. Hamilton, J.D.: Time Series Analysis, vol. 2. Princeton University Press Princeton, NJ (1994)
12. Hansen, P.R., Lunde, A., Nason, J.M.: The model confidence set. Econometrica **79**(2), 453–497 (2011)
13. Harvey, D., Leybourne, S., Newbold, P.: Testing the equality of prediction mean squared errors. Int. J. Forecast. **13**(2), 281–291 (1997)
14. Hsiao, C., Wan, S.K.: Is there an optimal forecast combination? J. Econ. **178**, 294–309 (2014)
15. Huang, G.-B., Zhu, Q.-Y., Siew, C.-K., et al.: Extreme learning machine: a new learning scheme of feedforward neural networks. Neural Netw. **2**, 985–990 (2004)
16. Hyndman, R., Khandakar, Y.: Automatic time series forecasting: the forecast package for R. J. Stat. Softw., Artic. **27**(3), 1–22 (2008)
17. James, G., Witten, D., Hastie, T., Tibshirani, R.: An Introduction to Statistical Learning, vol. 112. Springer, Berlin (2013)
18. Meese, R.A., Rogoff, K.: Do they fit out of sample? J. Int. Econ. **14**, 3–24 (1983)
19. Pinto, J.M., Marçal, E.F.: Inflation rate forecasting: extreme learning machine as a model combination method. In: ITISE 2019 (6th International Conference on Time Series and Forecasting) (2019)
20. Rossi, B.: Exchange rate predictability. J. Econ. Lit. **51**(4), 1063–1119 (2013)
21. Stock, J.H., Watson, M.W.: Forecasting inflation. J. Monet. Econ. **44**(2), 293–335 (1999)
22. Stock, J.H., Watson, M.W.: Macroeconomic forecasting using diffusion indexes. J. Bus. Econ. Stat. **20**(2), 147–162 (2002)
23. Tallman, E.W., Zaman, S.: Forecasting inflation: phillips curve effects on services price measures. Int. J. Forecast. **33**(2), 442–457 (2017)
24. Timmermann, A.: Forecast combinations. Handb. Econ. Forecast. **1**, 135–196 (2006)
25. Wang, J., Athanasopoulos, G., Hyndman, R.J., Wang, S.: Crude oil price forecasting based on internet concern using an extreme learning machine. Int. J. Forecast. **34**(4), 665–677 (2018)
26. Wang, L., Wang, Z., Hui, Q., Liu, S.: Optimal forecast combination based on neural networks for time series forecasting. Appl. Soft Comput. **66**, 1–17 (2018)
27. Zhang, B.: Real-time inflation forecast combination for time-varying coefficient models. J. Forecast. **38**(3), 175–191 (2019)

Time Series Analysis and Prediction in Other Real Problems

Load Forecast by Multi-Task Learning Models: Designed for a New Collaborative World

Leontina Pinto, Jacques Szczupak, and Robinson Semolini

Abstract This paper proposes a forecasting model designed for lack of data problems based on Multi-Task Learning techniques (MTL). It is especially useful for evolutionary markets and systems, where new paradigms (like renewable penetration or prosumers) significantly impact behavior and dynamics, creating unforeseen responses that would be unpredictable from past (possibly obsolete) historical data. A case study targeting the recent Brazilian load changes illustrates the approach performance: it was possible to combine data from three different distribution companies, creating a learning network, yielding reliable results where all other models failed.

Keywords Load forecast · Lack of data · Multi-tasking learning · Collaborative learning

1 Introduction

Energy demand is perhaps the market's most important pillar: all institutions, agents, and processes—from planning and operation to marketing and management—are essentially organized to serve it. However, although projecting load future evolution is crucial for an economical and secure supply, it is still one of our major challenges. The behavior of the consumer changes continuously, offering unpredictable reactions to various stimuli, as prices, economic indicators, expectations, and perceptions not always based on reality.

Brazilian load offers an interesting case study. The year 2018 experienced an anomalous increase in consumption throughout Brazil, almost always without connection to any of the classical explaining triggers: GDP experienced a sharp fall, as did income and all economic activities' indicators. We currently face a major

L. Pinto (✉) · J. Szczupak
ENGENHO, Rio de Janeiro, RJ 22793-312, Brazil
e-mail: leontina@engenho.com

R. Semolini
ELEKTRO–NEOENERGIA, Campinas, SP 13053-024, Brazil

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_25

challenge: consumer behavior has changed, old dynamics no longer represent the present and we *must predict the future without any past basis*. In fact, in this context, the longer the history, the worse is the prediction.

This behavior almost lies within the concepts proposed by [1, 2], where income raise yields a sensible behavior change, breaking the previous classical correlations between consumption and economy indicators.

However, the Brazilian case steps further: even without a significant income raise, popular expectations lead to new apparel acquisitions (specially climatization) and thus to consumption increase. Correlations are broken, and only behavioral economics can explain this anomaly.

It is necessary to develop mathematical models and computational tools as agile as the consumer, able to understand, follow and maybe anticipate its behavior, with the speed of our new times.

2 Objective

This paper describes a model able to accommodate more than just lack of data: we deal with *extreme* scarcity, where forecast needs to be performed from very few observations—for example, one year (twelve months). In this case, historical records are not even enough to allow a backtracking test (identification/prediction): it will be necessary to start from scratch.

It is necessary to "populate" the load history with valid information—and it is important to distinguish information from numbers: it would be possible to create synthetic samples from the available data, but they would contain the same poor information—anything else could even lead us to distorted results.

However, although it is not possible to extract more information from a history beyond the availability limits, it is feasible to *combine* similar experiences: observations from different agents that exhibit similar behaviors. For example, it is possible that distributors in neighboring regions share the same dynamics of consumption. In this case, it might be interesting to "blend the knowledge" of each company into a single richer, more complete history.

This is the proposal of collaborative learning (MTL) [3–5]. By joining forces, information is shared without losing individuality. The model should select the common dynamics and point specificities, leading to a more consistent and reliable projection.

The advantages of the proposed model are highlighted through a comparison between the new model and a Hilbert Space approach, previously used in many Brazilian companies, also designed for lack of data forecast problems.

3 Multi-Task Learning Approach

Considering space limitations, this article summarizes the applied collaborative learning model. More details, including alternative implementations, may be found in [3].

The proposed approach establishes a set of outputs or tasks $\boldsymbol{t}$ (in our case, the target variables, loads, or consumption). Each of these tasks is associated to a set of explanatory variables (inputs) $\boldsymbol{x}$ (in our case, economic, climatic, behavioral activities, etc.). The successful collaborative learning model requires that outputs $\boldsymbol{t}$ react similarly to inputs $\boldsymbol{x}$.

The function that "maps" the input $\boldsymbol{x}$ to the output $\boldsymbol{t}$ is written as

$$\boldsymbol{f}_{\boldsymbol{t}}(\boldsymbol{x}) = \sum\nolimits_{i=1}^{d} \boldsymbol{a}_{it}\boldsymbol{u}_{i}(\boldsymbol{x}) : \forall \boldsymbol{t} \in \boldsymbol{T}; \boldsymbol{a}_{it} \in \mathbb{R};\ \boldsymbol{x} \in \mathbb{R}^{\boldsymbol{d}} \tag{1}$$

where

$\boldsymbol{x}$ is the vector of input variables

$\boldsymbol{f}_{\boldsymbol{t}}(\boldsymbol{x})$ is the output associated to task $\boldsymbol{t}$.

function $\boldsymbol{u}_{i}(\boldsymbol{x})$ expresses the shared responses of all inputs $\boldsymbol{x}$ and different tasks $\boldsymbol{t}$.

coefficients $\boldsymbol{a}_{it}$ measure the "coupling" between different tasks.

For the sake of simplicity, this work assumes linear functions (non-linear extensions are possible and relatively straightforward). In this case, function ***f(t)*** corresponds to a vector product which may be written as

$$\boldsymbol{w}_{t} = \sum\nolimits_{i=1}^{d} \boldsymbol{a}_{it}\boldsymbol{u}_{i} \tag{2}$$

and therefore

$$\boldsymbol{f}_{\boldsymbol{t}}(\boldsymbol{x}) = \boldsymbol{w}_{t}(\boldsymbol{x}) : \forall \boldsymbol{t} \in \boldsymbol{T}; \boldsymbol{x} \in \mathbb{R}^{\boldsymbol{d}} \tag{3}$$

where $\boldsymbol{w}_{t}(\boldsymbol{x})$ combines the individual task coefficients a to the shared $\boldsymbol{u}$.

Finally, for concision

$$\boldsymbol{W} = \boldsymbol{UA} : \boldsymbol{W} \in \mathbb{R}^{\boldsymbol{d}\times\boldsymbol{T}} \tag{4}$$

These coefficients are obtained from the historical observations among all agents (even if scarce). Among other methods, the most intuitive is the well-known technique of function fitting to the available history

$$\boldsymbol{min}\left\{\sum\nolimits_{i=1}^{m} \boldsymbol{L}\left(\boldsymbol{y}_{ti}, \left\langle \boldsymbol{a}_{ti}, \boldsymbol{U}^{\boldsymbol{T}}\boldsymbol{x}_{ti}\right\rangle\right)\right\} : \boldsymbol{a}_{t} \in \mathbb{R}^{\boldsymbol{d}} \tag{5}$$

where ***L(.,.)*** measures the empirical deviation between the model outputs and the available data.

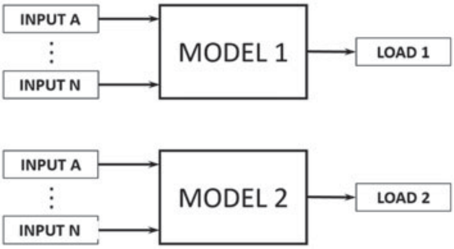


Fig. 1 Classic, individual approach

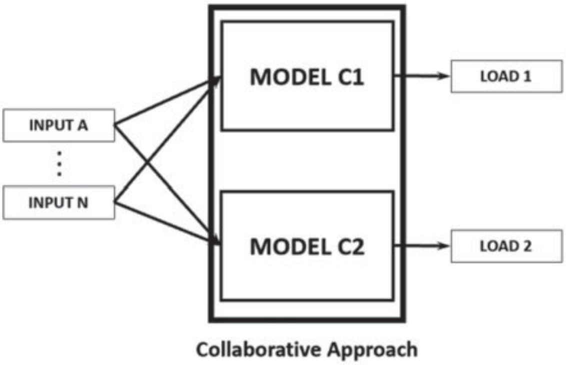


Fig. 2 Collaborative approach

4 Architecture Differences

Figures 1 and 2 illustrate the conceptual difference between the classical and collaborative approaches. While the classical approaches use each set of observations independently, collaborative approach combines all observations, creating a common pattern without losing each agent's uniqueness.

5 The Classical Hilbert Space Approach

The classical Hilbert approach was previously designed to handle the lack of data, aiming to adapt to the ever-changing Brazilian consumer's behavior is described in [6, 7] and will be summarized here.

5.1 Projection Theorem

Functional Analysis has been extensively applied to optimization processes [8]. It might be used on a statistical basis, as it is often found in communications, or on a deterministic point of view, the latter usually associated to Hilbert Spaces.

Hilbert Space elements may be seen as vectors, or, in our computerized world, data sequences representing loads, temperatures, economy index, etc. The Hilbert Space is a complete metric space [9], being able to approximate any given vector, always satisfying the Projection Theorem and the Orthogonality Condition [10].

This is shown in Fig. 3, where a given load vector is approximated by the vector sum of three "explaining variable" vectors, V_{e1}, V_{e2}, and V_{e3} (for instance, GDP, income, and temperature).

Figure 4 illustrates the decomposition process for just one "explaining variable". The original vector is projected (using the Projection Theorem) over the "explaining variable" (say, V_{e1}), yielding the "explained component". The remaining orthogonal vector corresponds to the unexplained component, or the error vector.

The unexplained component (error) will then be projected over the second explaining vector (say, V_{e2}) and the process will continue until the final error is considered negligible.

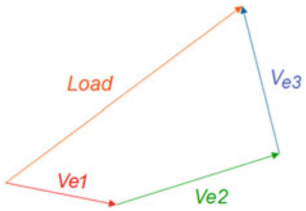


Fig. 3 Hilbert space decompositsion

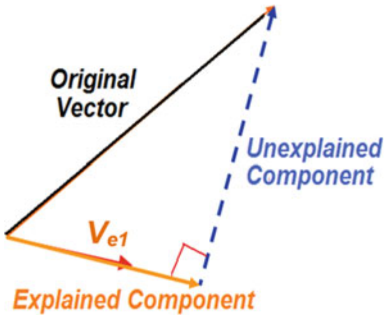


Fig. 4 Original vector decomposition over a first "explaining vector"

5.2 Parallel Processing Implementation

Let C be the desired vector to be decomposed by the set of “explaining variables-vectors” $\underline{S}$, $\underline{S_2}, \ldots, \underline{S_N}$. Therefore, one should look for the optimum combination of these “basis” vectors

$$\underline{C} \cong \underline{\underline{S}}\,\underline{\alpha} = \left[\underline{S_1}, \underline{S_2}, \ldots, \underline{S_N}\right]\underline{\alpha} \tag{6}$$

such as to minimize the error norm

$$min \underbrace{\left\| \underline{C} - \underline{\underline{S}}\,\underline{\alpha} \right\|}_{\underline{\alpha}\|\underline{\varepsilon}\|} \tag{7}$$

The Projection Theorem states the optimum approximation error is orthogonal to the space of “explaining vectors” and, therefore, to any of its elements, such as

$$\underline{\varepsilon}^t \underline{S_i} = \underline{C}^t \underline{S_i} - \underline{\alpha}^t \underline{\underline{S}}^t \underline{S_i} = 0 \text{ for } i = 1, 2, \ldots, N \tag{8}$$

or, for all “explaining vectors”

$$\underline{C}^t \underbrace{\left[\underline{S_1}, \underline{S_2}, \ldots, \underline{S_N}\right]}_{\underline{\underline{S}}} = \underline{\alpha}^t \left[\underline{\underline{S}}^t\right] \underbrace{\left[\underline{S_1}, \underline{S_2}, \ldots, \underline{S_N}\right]}_{\underline{\underline{S}}} \tag{9}$$

leading finally to the unique [9] optimum set of coefficients

$$\underline{\alpha} = \left(\underline{\underline{S}}^t \underline{\underline{S}}\right)^{-1} \underline{\underline{S}}^t \underline{C} \tag{10}$$

The method is now able to work with large sets of “explaining vectors” in a very efficient way. Moreover, it solves the “co-integration” problem, automatically accommodating inter-correlated explaining variables, finding the best fit while eliminating possible “double counting” effects due to the interdependencies.

Finally, Hilbert Decomposition does not require a large historical period. Although, of course, more reliable information yields a more precise result, it will work at its best within a constrained history, and it suited to a lack of data framework. It has been successfully used in many Brazilian companies, and was able—until now—to yield a reliable forecast based on a mere 5-year history (60 monthly observations).

6 Case Study

6.1 *The Challenge*

The necessity of a new model, able to deal with lack of data, is shown in Fig. 5. After three years of stagnation, the load finally experienced a steep—and unexpected—rise.

The explanation to this phenomenon, however, was unclear. Figures 6, 7, and 8 show the classical model forecast results for a backtracking process (identification and projection) applied to three neighboring distributors (COELBA, CELPE, COSERN), based on usual explaining variables (GDP, Income, Temperature). There is a sensible, abnormal step associated to 2019 summer in all companies (in fact, all Brazilian distributors exhibited the same behavior, and many different statistical models led to similar results). No available model was able to predict—even to explain this response.

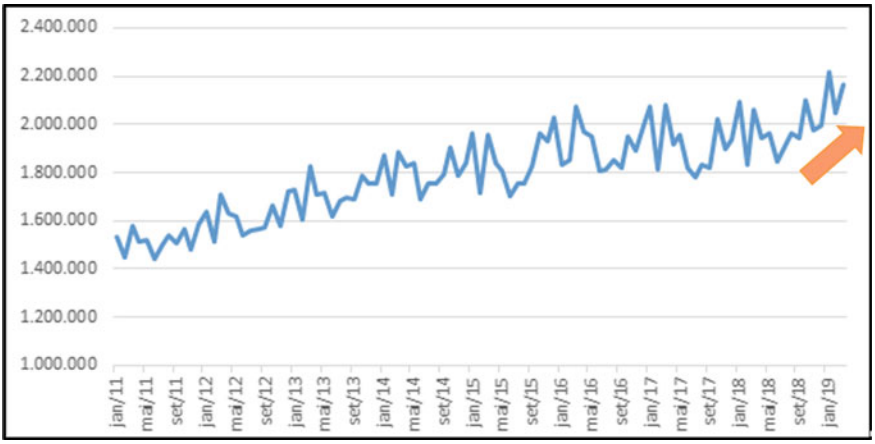


Fig. 5 Bahia (COELBA) load growth

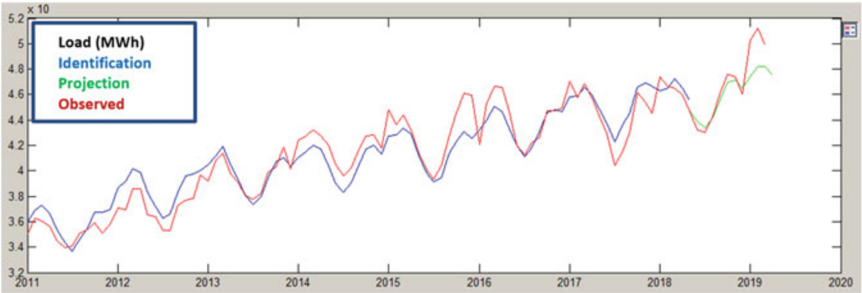


Fig. 6 Bahia (COELBA) load dynamics

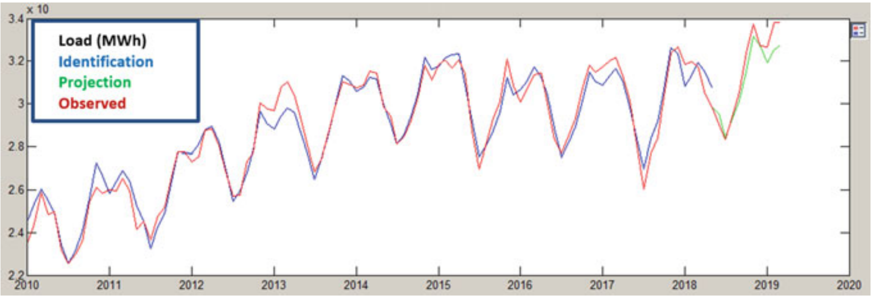


Fig. 7 Pernambuco (CELPE) load dynamics

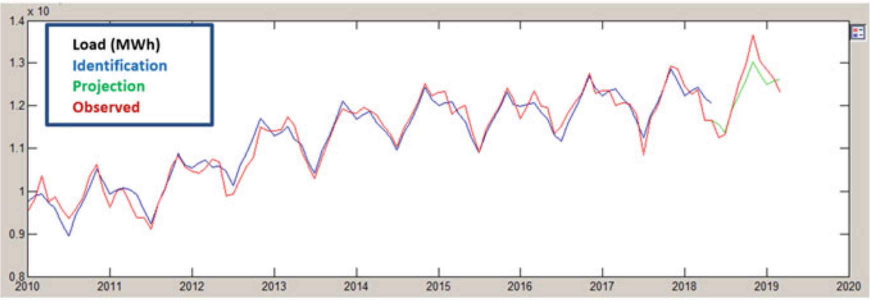


Fig. 8 Rio Grande do Norte (COSERN) load dynamics

More than absorbing the deviations, the main question is should that step be an anomaly, or should it be a change in consumer's behavior—in other words, is this a new permanent pattern? This question is, of course, related to the consumer's reactions and the answer requires a deeper—non-statistical—understanding.

Extensive field research [11], based on behavioral economics [12, 13], uncovered an interesting fact: a disputed election restored the consumer's belief on a stronger economy and a change for the better. This faith in the future, associated to an unusual warm summer, leads to the highest level of refrigeration equipment purchase observed in a decade.

It must be noticed that no economy or income growth backed up this trend: it was a matter of hope and belief. Therefore, no model based on past correlations would be able to account for this change.

As a consequence, consumers possess a new basis of installed demand, and will use it from now on. There is indeed a new standard, which will induce a new response, that must be predicted based on a few observations.

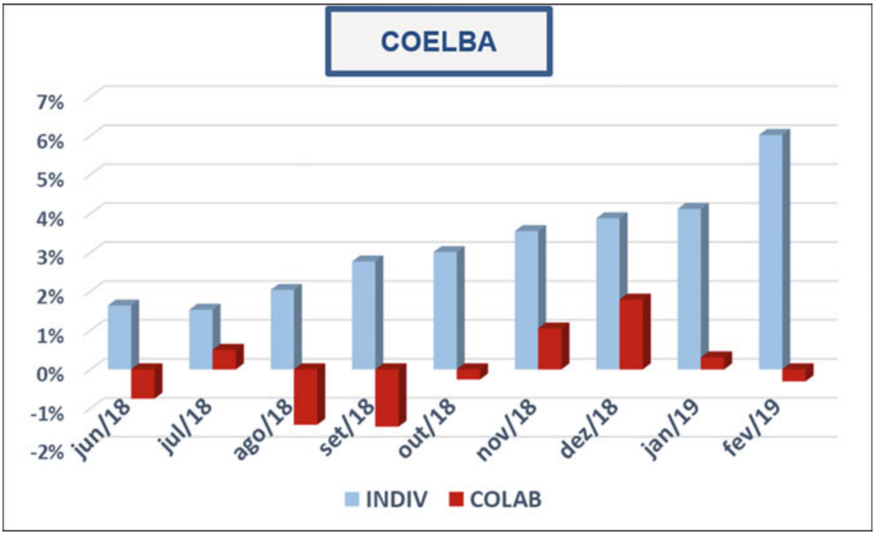


Fig. 9 Individual x collaborative learning, Bahia (COELBA)

6.2 *The Proposed Solution*

The anomalous behavior was detected from May 2018. It would be very difficult, if not impossible, to apply existing models to as few as 12–18 months for model identification/validation.

We proceeded to try the collaborative learning technique. As our goal was predicting 2019 summer, we based our identification phase on the period from October 2017 to May 2018—where the behavior was still establishing. Of course, more observations will improve the results and will be used as they become available.

Figure 9, 10, and 11 compare the results obtained from our best classical Hilbert Space model (individual learning) and from the collaborative learning. It is interesting to notice that (as expected) the results show slightly higher errors during springtime (as consumers were still adapting, taking decisions, buying equipment). However, projection for summer months is much better.

In any case, the proposed approach offered a clear enhancement on the overall forecast quality. All deviations are significantly lower, despite the almost non-existing information. Moreover, the "deviation trend" is broken, offering a more stable and reliable insight of the future.

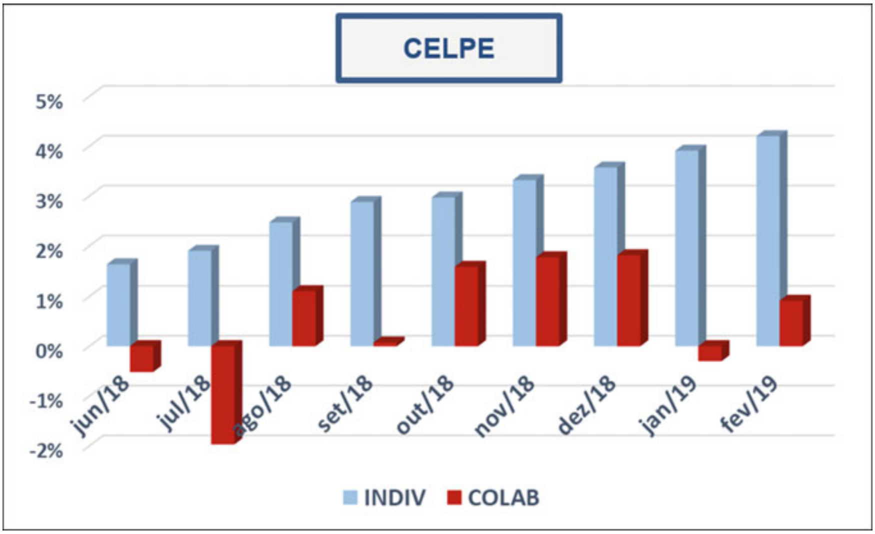


Fig. 10 Individual x collaborative learning, Pernambuco (CELPE)

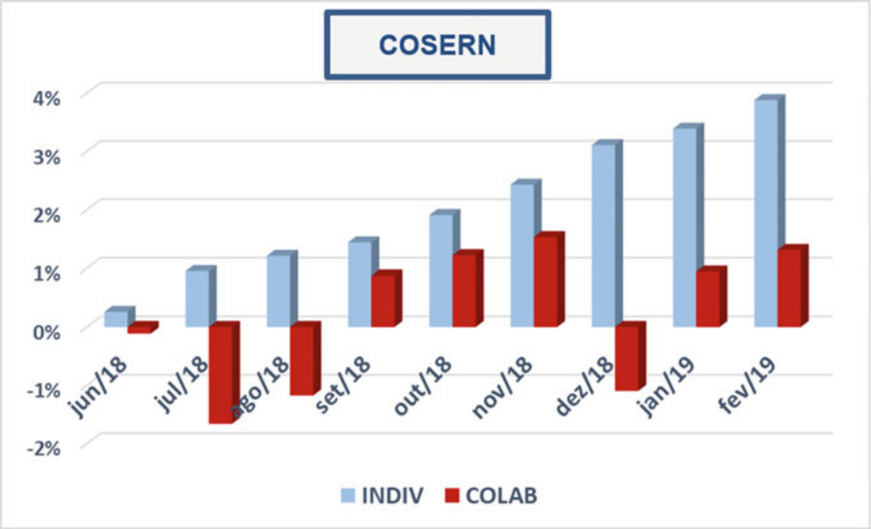


Fig. 11 Individual x collaborative learning, Rio G. do Norte (COSERN)

7 Conclusions

We live in a changing world, and consumption dynamics is not an exception. Preparedness for the future requires the forecast of the unknown. It is crucial to build models that are able to quickly detect modifications—and know the difference from anomalies. It will be necessary to adapt, adjust, absorb novelties.

In the context, classical models, that try to repeat the past, will not be able to foresee the future. The ability to collect and store a huge history may not ensure the quality of information. Number of observations will not necessarily yield precision.

We propose a model designed for this new reality: a collaborative learning technique, able to combine information from different agents, identify common and individual characteristics and build a rich history without traveling back to a distant past.

The described approach was applied to a hard challenge: the projection of the summer load for three Brazilian distributors which broke any known record. A mere 8-month observed data was able to provide much better results for all companies, paving the path to explain the (previously) unexplainable behavior.

These promising results suggest an interesting way, which will be pursued and reported in the near future.

References

1. Fuchs, A., Gertler, P., Shelef, O., Wolfram, C.: The demand for energy-using assets among the world's rising middle classes. Am. Econ. Rev. (2016)
2. Auffhammer, M., Wolfram, C.D.: Powering up China: income distributions and residential electricity consumption. (2014)
3. Caruana, R.: Multitask learning. Mach. Learn. **28**(1), 41–75 (1997)
4. Argyriou, A., Evgeniou, T., Pontil, M.: Multi-task feature learning. Adv. Neural. Inf. Process. Syst. **19**, 41–48 (2006)
5. Zhang, Y., Yang, Q.: A survey on multi-task learning. arxiv pre-print (2017)
6. Szczupak, J., Pinto, L., Macedo, L.H., Pascon, J., Semolini, R., Inoue, M., Almeida, C., Almeida, F.R.: Load modeling and forecast based on a Hilbert space decomposition. In: 2007 IEEE Power Engineering Society General Meeting, disponível na base de dados do repositório IEEEXPLORE. https://ieeexplore.ieee.org/document/4275991
7. Pinto, L., Szczupak, J., Almeida, C., Macedo, L., Inoue, M., Massaro, R., Semolini, R., Pascon, J., Albarelli, E., Tortelli, D.: Load forecast under uncertainty: accounting for the economic crisis impact. In: 2009 IEEE Bucharest PowerTech, pp. 1–5 (2009)
8. Haykin, S.: Adaptive Filter Theory, 4th edn, Prentice Hall (2001)
9. Debnath, L., Mikusinski, P.: Introduction to Hilbert Spaces with Application. Academic Press (1999)
10. Akhiezer, N.I., Glazman, I.M.: Theory of Linear Operators in Hilbert Space. Dover (1988)
11. ENGENHO Brazilian Load Growth Diagnostics, report, available from www.engenho.com
12. Eia, US energy information administration, Behavioral economics applied to energy demand analysis: a foundation (2014)
13. Thaler, R.H.: Misbehaving: The Making of Behavioral Ecsonomics. W. W. Norton & Company (2016)

Power Transformer Forecasting in Smart Grids Using NARX Neural Networks

J. Ramírez, F. J. Martínez-Murcia, F. Segovia, A. Ortiz, D. Salas-González, S. Carrillo, J. Leiva, J. Rodríguez-Rivero, and J. M. Górriz

Abstract In the next years, with a growing presence of electric vehicles and a massive penetration of renewable sources and low levels of voltage for self-consumption, it will be essential that medium- and low-voltage distribution networks be planned, operated, and supervised as transportation networks have been managed for decades, from the distributor to be a simple agent of distribution assets to be the operator of the network. This paper shows a non-linear autoregressive neural network with exogenous inputs (NARX) for time-series forecasting and power transformers monitoring. The NARX network model provides a description of the system by means of a non-linear function of lagged inputs, outputs, and prediction errors that can be interpreted as a recurrent dynamic network, with feedback connections enclosing several layers of the network. The prediction model consists of a multilayer perceptron (MLP) in the hidden layer that takes as input a window of past independent (exogenous) inputs and past outputs followed by an output layer that finally forecast the target time series. A previous study was carried out in order to select the most important electrical measurements enabling the prediction of the safe operation of the power transformer. The selection of the electrical measurements that have more influence on the transformer temperature was based on the computation of the pairwise Pearson's correlation coefficient, the Kendall's rank correlation coefficient as well as the cumulative conditional Granger causalities. The proposed NARX network was trained and evaluated in open-loop and closed-loop modes showing a high accuracy when predicting and monitoring the operation of power transformers.

Keywords Power transformer monitoring · Fault diagnosis · Time-series forecasting · NARX neural networks

J. Ramírez (✉) · F. Segovia · D. Salas-González · J. M. Górriz
Departmant Signal Theory, Telematics and Communications, University of Granada, Granada, Spain
e-mail: javierrp@ugr.es

F. J. Martínez-Murcia · A. Ortiz
Departmant of Communications Engineering, University of Málaga, Málaga, Spain

S. Carrillo · J. Leiva · J. Rodríguez-Rivero
Endesa Distribución, Madrid, Spain

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_26

1 Introduction

Systems identification is a well-defined methodology to identify and evaluate the response of a dynamic system by defining models based on measurements of the inputs and outputs thereof [1]. The applications of the system identification problem extend to any system in which the inputs and outputs are known and include industrial process control, social data analytics, control systems, mechanical and aerospace engineering, biomedical systems, economical data, financial systems, etc.

Non-linear system modeling techniques have significantly evolved during the last decades. Among them, the non-linear autoregressive moving average with exogenous inputs (NARMAX) [2–4] model represents a wide class of discrete-time non-linear systems. The NARMAX model provides a description of the system by means of a non-linear function of lagged inputs, outputs, and prediction errors. Since the definition of the NARMAX model is independent of the non-linear functional, multilayered neural networks offer a powerful alternative in this context for modeling complex non-linear systems and time-series forecasting [5]. Among these networks, the non-linear autoregressive neural network with exogenous inputs (NARX) can be interpreted as a recurrent dynamic network, with feedback connections enclosing several layers of the network and has found application in many real scenarios for time-series forecasting [6–11] https://doi.org/10.1016/j.neucom.2020.05.078.

The next generation of low- and medium-voltage distribution networks will demand to be better planned, operated, and supervised as transportation networks have been managed for decades. The adaptation of these networks will require to incorporate much more intelligence, sensorization, broadband communications, optimal control, and intelligent reporting among other emerging technologies. In order to accomplish it, the systems have to incorporate much more intelligence than before, which involves a whole spectrum of digital technologies: sensorization, smart meters, broadband communications, local electronic device controllers, IoT (Internet of Things), SCADAs (Supervision, Control, and Acquisition of Data), energy management centers, advanced data processing software (data analytics), optimal control, intelligent reporting, etc [12, 13].

A large collection of predictive techniques and methods to diagnose the health of power transformers are available in the literature [14–17]. These techniques are classified as off-line or on-line methods depending on if the monitoring process requires to disconnect the transformer or not. Expert knowledge and experienced engineers are needed to correctly interpret the results of the monitoring process. This paper shows a power transformer monitoring approach based on a non-linear autoregressive discrete-time model with exogenous inputs and neural networks.

2 System Identification Modeling

Systems identification is a methodology to characterize a dynamic system through a mathematical model whose definition is based on its inputs and outputs [1]. The models used in this scenario can be linear and non-linear system identification models. Linear systems are defined as those that satisfy the superposition principle and can be broadly classified into nonparametric and parametric methods. The development of linear system identification systems started in the early 70s and is still an open research problem. On the contrary, non-linear system identification systems are those that do not satisfy with the superposition principle.

Among the non-linear systems' identification methods, non-linear autoregressive moving average with exogenous inputs modeling (NARMAX) [2, 18–20], first introduced in 1981, represent a broad class of non-linear system modeling techniques. The discrete-time NARMAX model is defined by means of

$$
\begin{aligned}
y(t) = & F[y(t-1), y(t-2), \ldots, y(t-n_y), \\
& u(t-d), u(t-d-1), \ldots, u(t-d-n_u), \\
& e(t-1), e(t-2), \ldots, e(t-n_e)] + e(t)
\end{aligned}
\tag{1}
$$

where $y(t)$, $u(t)$, and $e(t)$ denote the input, output, and noise sequences of the system, respectively, and n_y, n_u, and n_e are the maximum lags for these sequences. $F[\cdot]$ represents a non-linear function and d a delay that is usually set to 1 [1]. The noise term $e(t)$ in the model is normally defined as the prediction error. It is included in the definition in the model to deal with the effects of measurement noise, modeling errors, etc. There are numerous forms to approximate the non-linear function $F[\cdot]$ that is considered in the definition of the NARMAX model. Among them, the most frequently used are power-form polynomial models, rational models, neural networks, fuzzy logic-based models, wavelet expansions, and radial basis function (RBF) networks. In this paper, we will adopt the multilayer perceptron (MLP) neural network to model the non-linear $F[\cdot]$ function of the NARMAX model for predicting the temperature of the power transformer temperature as a function of a number of exogenous input variables and past values of the output.

The architecture of a typical dynamically driven recurrent single-layer NARX network for a single input, single output systems is shown in Fig. 1, where $\phi_i(\cdot)$ and w_i ($i = 1, 2, \ldots, m$) in the hidden layer are predetermined non-linear scalar functions, referred to as activation functions, and the networks weights, respectively. Mathematically, the operation of the recurrent NARX neural network can be described by

$$
y(t) = F[\mathbf{x}(t)] = w_0 + \sum_{i=1}^{m} w_i \phi_i(\mathbf{x}(t)) \tag{2}
$$

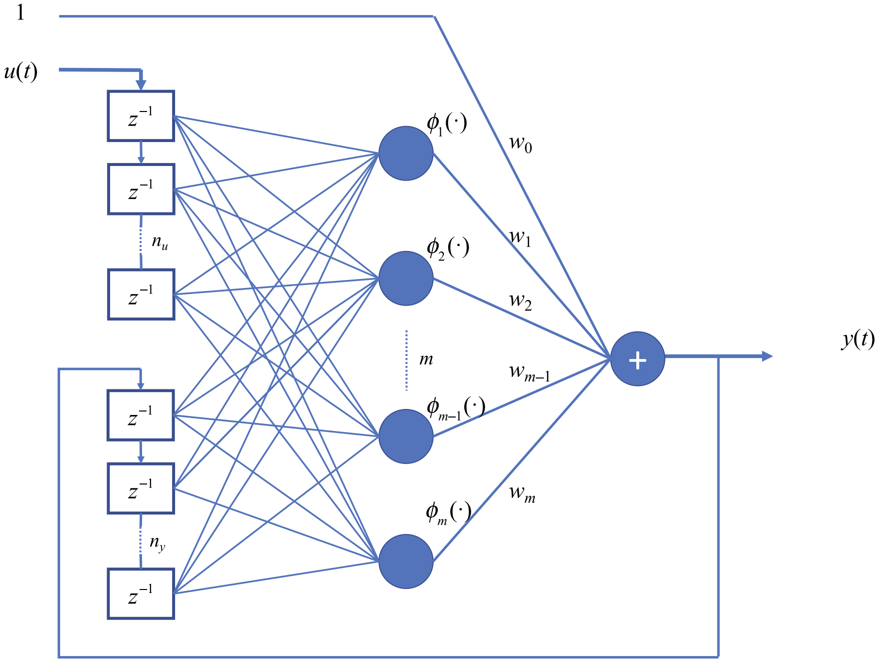


Fig. 1 A recurrent NARX neural network for a single input, single output system

where the vector $\mathbf{x}(t) = [x_1(t), x_2(t), \ldots, x_n(t)]$ is defined to be

$$x_m(t) = \begin{cases} y(t-m)) & 1 \le m \le n_y \\ u(t-m+n_y)) & n_y+1 \le m \le n = n_y + n_u \end{cases} \tag{3}$$

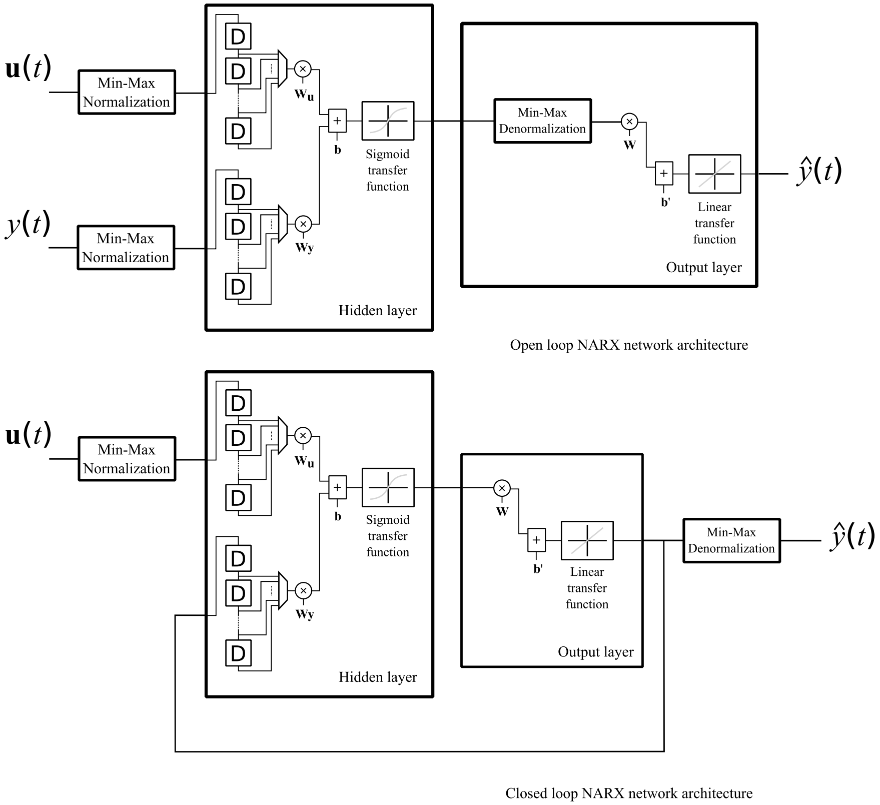
3 NARX Neural Network for Power Transformer Temperature Prediction

The main objective of this paper is to show an efficient and accurate method for monitoring the temperature of the power transformer based on time-series forecasting and sensorized transformer data. Dynamically driven recurrent NARX neural networks enabling time-series forecasting are a potentially attractive technology for this challenging problem since these models can be generalized to incorporate multiple exogenous inputs. The NARMAX model previously introduced [2–4] is a discrete-time non-linear system model that can be generalized in order to provide an estimation of the next value of the dependent output signal $y(t)$ in terms of a non-linear function of past values of the output signal and previous values of independent (exogenous) input signals. A special case of the NARMAX model is the NARX model which does not include any noise-dependent model terms such as $e(t-1)$ and $e(t-2)$, and can be implicitly formulated as

$$\begin{aligned} y(t) =& F[y(t-1), y(t-2), \ldots, y(t-n_y), \\ & \mathbf{u}(t-1), \mathbf{u}(t-2), \ldots, \mathbf{u}(t-n_u)] + e(t) \end{aligned} \tag{4}$$

where the $\mathbf{u}$ vector represents the set of exogenous signals that are used to predict the value of the output signal $y(t)$ while the noise term $e(t)$ is an independent sequence.

Neural networks are widely used for the implementation of the non-linear $F[\cdot]$ function in Eq. 4 leading to an elegant solution for time-series forecasting based on multiple exogenous inputs. Figure 2 shows the architecture of open-loop and closed-loop non-linear autoregressive neural networks with exogenous inputs. These consist of a multilayer perceptron (MLP) in the hidden layer that takes as input a window of past independent (exogenous) inputs $\{\mathbf{u}(t-1), \mathbf{u}(t-2), \ldots, \mathbf{u}(t-n_u)\}$ and past



Open loop NARX network architecture

Closed loop NARX network architecture

Fig. 2 Architecture of open-loop and closed-loop non-linear autoregressive neural networks with exogenous inputs

outputs $\{y(t-1), y(t-2), \ldots, y(t-n_y)\}$, and calculates the current output $y(t)$, followed by an output layer. These networks are usually trained in open-loop mode based on past values of the exogenous signals and the output to predict the next sample. Once the network is trained, the time-series forecasting problem can be accomplished without using past values of the output signal and based only on the exogenous inputs. This step requires to introduce a feedback from the output layer to the input of the hidden layer so that the past values of $y(t)$ are replaced by their predictions as shown in Fig. 2.

4 Power Transformation Center Datasets

Data used in this study was provided by Endesa Distribución company through the Open Innovation Living Lab Smartcity Málaga (Málaga, Spain). The objective of the project MONICA is the deployment of sensors and measurement, automation and control equipment in medium and low-voltage transformation centers. The challenge is to implement an electrical network integrating artificial intelligence technology not only to the new monitoring and tracking elements, but also to the built-in network analysis techniques that deal with all the input information, including its uncertainty.

The Open Innovation Living Lab Smartcity Málaga includes 17 power transformation centers. For each of them, a total of 20 variables were recorded at a sample rate of 12 samples/hour for the whole 2018 year. These signals include the 'Phase Imbalance (PI)', 'Active Energy Exported (AEE)', 'Active Energy Imported (AEI)', 'Capacitive Reactive Energy Exported (CREE)', 'Capacitive Reactive Energy Imported (CREI)', 'Inductive Reactive Energy Exported (IREE)', 'Inductive Reactive Energy Imported (IREI)', 'Intensity R (IR)', 'Intensity S (IS)', 'Intensity T (IT)', 'Active Power R (APR)', 'Active Power S (APS)', 'Active Power T (APT)', 'Active Power (AP)', 'Reactive Power R (RPR)', 'Reactive Power S (RPS)', 'Reactive Power T (RPT)', 'Reactive Power (RP)', 'Room Temperature (RTemp)', 'Transformer Temperature (TTemp)', 'Tension R (TR)', 'Tension S (TS)', and 'Tension T (TT)'.

In this paper, a new method to monitor the operation of the power transformer, fault diagnosis, and rapid intervention based on a NARX neural network is shown. In this scenario, the increase in the temperature of the power transformer is the main consequence of electrical failure and then, the target of the time-series prediction problem. Time-series forecasting assumes the use of an accurate autoregressive model of the power transformer that is able to predict future values of the target signal based on previously observed values of electrical magnitudes recorded at the power transformation center.

5 Electrical Measurement Analysis from Power Transformers

Prior to the implementation and evaluation of the NARX neural network, a study was carried out to select the most relevant electrical measurements to predict the operation of the power transformer. The analysis was based on the computation of the pairwise Pearson's correlation coefficient, the Kendall's rank correlation coefficient as well as the cumulative conditional Granger causalities.

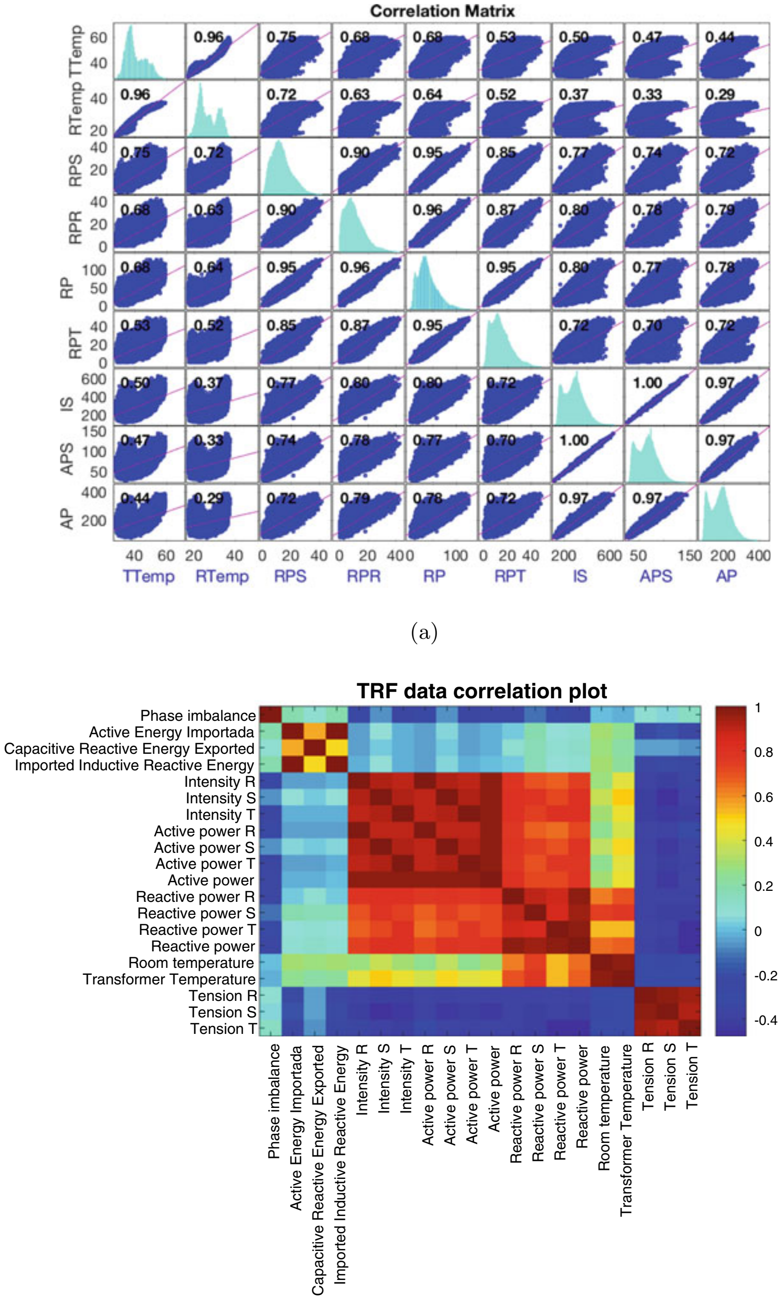
5.1 Correlation Analysis

A correlation analysis was conducted in order to determine the Pearson's correlations among pairs of variables for each of the power transformer electrical measurements in the dataset. This information will be used to identify the exogenous inputs enabling forecasting the temperature of the power transformers. Figure 3a provides a matrix of plots showing the correlations among pairs of variables that are highly correlated with the TTemp target. Note that, histograms of the variables are shown along the diagonal of the matrix plot while scatter plots of variable pairs appear in the off-diagonal. The slopes of the least-squares linear regression problem are equal to the displayed correlation coefficients. It can be concluded that several electrical measurements are strongly correlated. Finally, the correlation coefficients are summarized in Fig. 3b for a given power transformer in the dataset.

In addition to Pearson's correlation analysis, Kendall's rank correlation analysis [21] between the power transformer time series was carried out. Kendall's Tau coefficient is a nonparametric measure of relationships between time series. The Tau correlation coefficient returns a value between 0 and 1, where 0 means no relationship and 1 a perfect relationship. Figure 4 shows Kendall's correlation coefficients between pairs of highly correlated electrical measurement. A hypothesis test was also performed in order to determine which correlations are significantly different from zero. The correlation coefficients highlighted in red in Fig. 4 indicate which pairs of variables have correlations significantly different from zero. It can be concluded that all the pairs of variables shown in Fig. 4 have correlations significantly different from zero.

5.2 Granger Causality-Based Analysis

In order to justify the selection of the aforementioned exogenous variables for improving the prediction of TTemp variable, we conducted a similar analysis as in [12] to assess the ensemble connectivity maps. In this sense, we computed the cumulative



(a)

(b)

Fig. 3 Correlation analysis. **a** Correlation plots between multiple electrical measurements of the power transformer, **b** Correlation matrix between transformer data time series

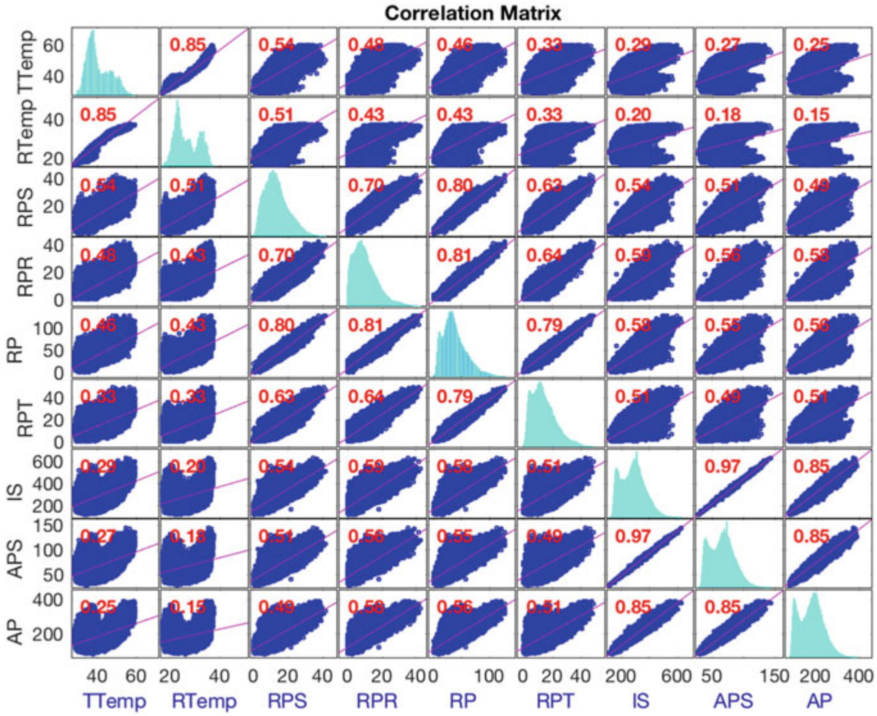


Fig. 4 Kendall's rank correlation coefficients. Hypothesis test to determine electrical measurements that have correlations significantly different from zero

conditional pairwise Granger causalities and generated the circular graphs accordingly. In addition, we calculated the occurrences of the corresponding (weak or strong) connections focusing on the ones causing the target variable (TTemp). As shown in Fig. 5, there are persistent weak connections which suggest the possibility of improving the prediction of some variables by including exogenous information, i.e., RTemp, AR, APR, APS, and APT variables could help in the prediction of TTemp. Indeed, the current explains the power transformer (PT) temperature except for a particular lag that depends on the thermal inertia of the PT, crucial for detecting PT failures. This is shown in Fig. 5, where the connections between the variables under assessment were highlighted. From this figure, we see how these variables are weighting factors, in the sense of causal notion, of the target variable with the same relevance.

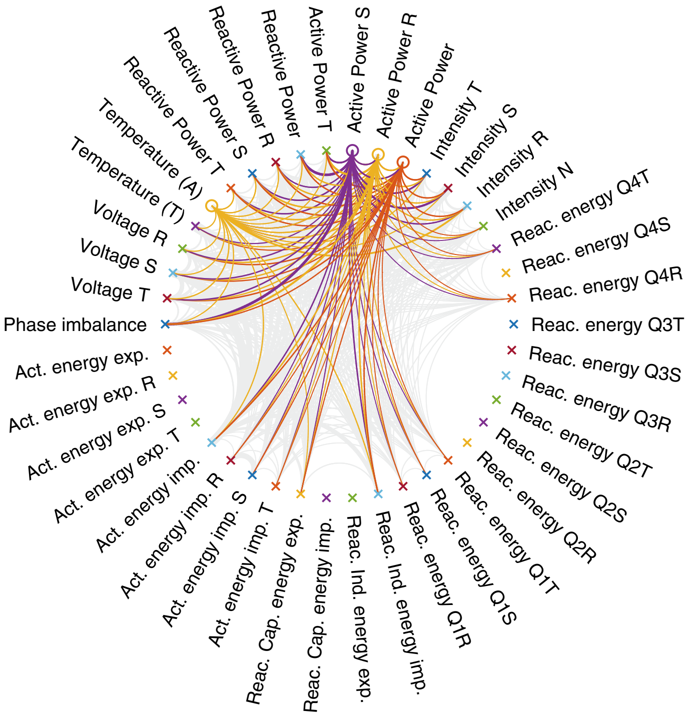


Fig. 5 Ensemble connectivity maps obtained after computing the cumulative conditional pairwise Granger causalities and generated the circular graphs accordingly

6 NARX Implementation and Evaluation Experiments

Before carrying out the training process of the NARX network, the transformed signals that were digitized were resampled with a uniform sampling rate of 12 samples/hour. This process is necessary since the signals stored in the cloud are not registered with a uniform period in all the cases and the NARX networks are based on a uniform sample rate discrete-time model. The corresponding resampling process was based on a simple linear interpolation between consecutive samples.

Once the resampling procedure was carried out, an analysis of the variables recorded in the transformer that could have influenced the variation of the transformer temperature was carried out. In this way, an experiment was conducted in which the TTemp time series was predicted based on other transformer variables such as RTemp, AP, APR, APS, and APT, through a two-delay element NARX network consisting of a 10-neuron single MLP-based hidden layer. In the first phase, 75,000 data samples were used for cross-validation (70% for training, 15% for validation, and 15% for testing). Figure 6 shows the training and evaluation experiments that were carried out in order to analyze the performance of the NARX network for

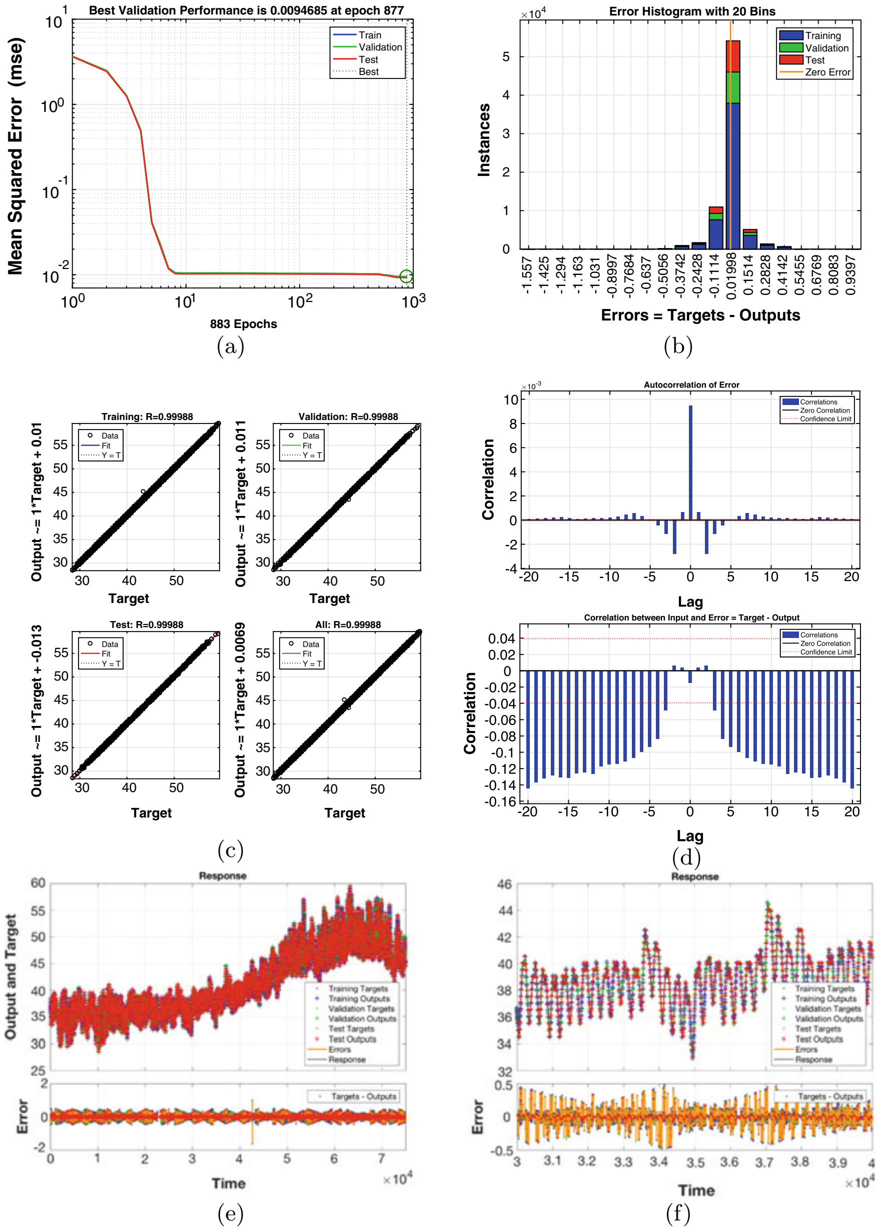


Fig. 6 NARX training: **a** MSE as a function of the number of epochs, **b** Histogram of the time-series forecast error, **c** Output-target linear regression, **d** Autocorrelation of the time-series prediction error and correlation among the input and the prediction error, **e** Response on unseen data (output, target and error) of the NARX network, and **f** Detail (zoom) of the response on unseen data of the NARX network

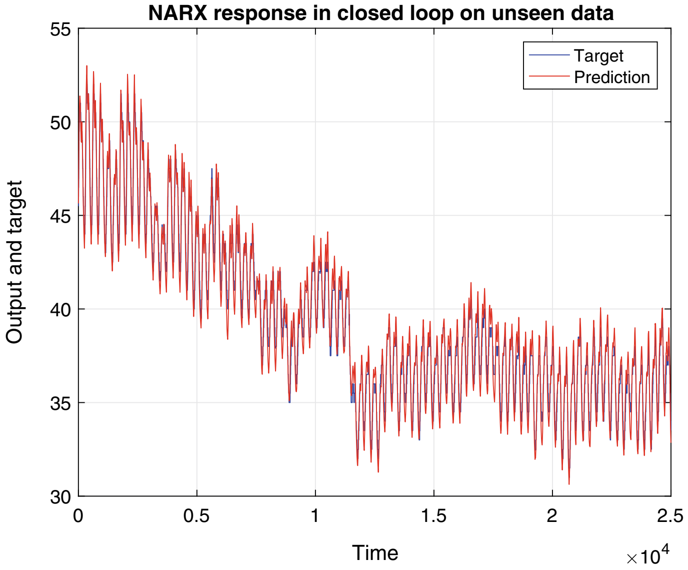


Fig. 7 Closed-loop NARX response (output and target) on unseen data

time-series forecasting and power transformer monitoring. Figure 6a analyzes the learning curve and the convergence of the network in terms of the mean squared error (MSE) as a function of the number of training epochs. It is concluded that the MSE converges to 0.01 in about 10 training epochs. Figure 6b shows the histogram of the prediction error. It is centered at zero mean and has a reduced variance. Figure 6c shows the linear output-objective regression analysis where a good fit between both variables is obtained with a regression coefficient close to unity. Figure 6d analyzes the autocorrelation of the error and the correlation between the input and the prediction error. Finally, Fig. 6e plots the response of the model and the prediction of the temperature of the transformer (TTemp) for training, validation, and test targets showing high accuracy in cross-validation experiments.

Once the open-loop NARX network was trained, the closed-loop network model shown in Fig. 2 was built in order to formulate a prediction of the temperature of the transformer (TTemp) without using previous values of the target. Thus, the closed-loop NARX network only uses values of the exogenous variables RTemp, AP, APR, APS, and APT to predict the temperature of the transformer. To evaluate this second model, 25,000 samples not used previously for training were used. Figure 7 shows the output of the NARX network in closed loop and the target. It can be concluded that the NARX network model used for monitoring the temperature of the transformer effectively tracking its variation in time yielding a high prediction accuracy.

7 Conclusion

This paper explores machine learning technologies for the next generation of power distribution networks that will demand to be better planned, operated, and supervised in a similar way as transportation networks have been managed for decades. The adaptation of these networks will require to incorporate much more intelligence, sensorization, broadband communications, optimal control, and intelligent reporting among other emerging technologies.

A power transformer monitoring approach based on a non-linear autoregressive discrete-time neural network with exogenous inputs was proposed in this paper. The proposed NARX network predicted the temperature of the transformer as a function of past values of outputs and exogenous inputs. The system was then described by a non-linear function of lagged inputs, outputs, and prediction errors that can be interpreted as a recurrent dynamic network, with feedback connections enclosing several layers of the network.

Data used in this study was provided by Endesa Distribución company through the Open Innovation Living Lab Smartcity Málaga (Málaga, Spain). The objective of the project PASTORA is the deployment of sensors and measurement, automation and control equipment in medium and low-voltage transformation centers. The challenge is to implement an electrical network integrating artificial intelligence technology not only to the new monitoring and tracking elements, but also to the built-in network analysis techniques that deal with all the input information, including its uncertainty. The Open Innovation Living Lab Smartcity Málaga includes seventeen power transformation centers. For each of them, a total of 20 variables were recorded at a sample rate of 12 samples/hour for the whole 2018 year.

Prior to the implementation and evaluation of the NARX neural network, a study was carried out in order to select the most relevant electrical measurements to predict the operation of the power transformer. The analysis was based on the computation of the pairwise Pearson's correlation coefficient, Kendall's rank correlation coefficient as well as the cumulative conditional Granger causalities. Finally, the NARX networks were trained and evaluated by cross-validation showing high accuracy when operated in open-loop and closed-loop modes.

Acknowledgements The authors would like to acknowledge the support of CDTI (Centro para el Desarrollo Tecnológico Industrial, Ministerio de Ciencia, Innovación y Universidades and FEDER) under the PASTORA (Preventive analysis of intelligent networks in real time and integration of renewal resources) project (Ref.: ITC-20181102).

References

1. Billings, S.A.: Nonlinear System Identification: NARMAX Methods in the Time, Frequency, and Spatio-Temporal Domains. Wiley (2013)
2. Chen, S., Billings, S.A., Grant, P.M.: Non-linear system identification using neural networks. Int. J. Control **51**(6), 1191–1214 (1990)

3. Lin, T., Horne, B.G., Tino, P., Giles, C.L.: Learning long-term dependencies in NARX recurrent neural networks. IEEE Trans. Neural Netw. **7**(6), 1329–1338 (1996)
4. Siegelmann, H.T., Horne, B.G., Giles, C.L.: Computational capabilities of recurrent NARX neural networks. IEEE Trans. Syst., Man, Cybern., Part B (Cybern.) **27**(2), 208–215 (1997)
5. Ramírez, J., Martínez-Murcia, F.J., Segovia, F., Carrillo, S., Leiva, J., Rodríguez-Rivero, J., Górriz, J.M.: Power transformer monitoring based on a non-linear autorregressive neural network model with exogenous inputs. In: Proceedings of the International Conference on Time Series and Forecasting, vol. 2, pp. 835–843. Granada, Spain (2019)
6. Guzman, S.M., Paz, J.O., Tagert, M.L.M.: The use of NARX neural networks to forecast daily groundwater levels. Water Resour. Manag. **31**(5), 1591–1603 (2017). Mar
7. Marcjasz, G., Uniejewski, B., Weron, R.: On the importance of the long-term seasonal component in day-ahead electricity price forecasting with NARX neural networks. Int. J. Forecast. (2018)
8. Cadenas, E., Rivera, W., Campos-Amezcua, R., Heard, C.: Wind speed prediction using a univariate arima model and a multivariate NARX model. Energies **9**(2) (2016)
9. Vaz, A., Elsinga, B., van Sark, W., Brito, M.: An artificial neural network to assess the impact of neighbouring photovoltaic systems in power forecasting in Utrecht, the Netherlands. Renew. Energy **85**, 631–641 (2016)
10. Basso, M., Giarre, L., Groppi, S., Zappa, G.: Narx models of an industrial power plant gas turbine. IEEE Trans. Control Syst. Technol. **13**(4), 599–604 (2005). July
11. Villacci, D., Bontempi, G., Vaccaro, A., Birattari, M.: The role of learning methods in the dynamic assessment of power components loading capability. IEEE Trans. Ind. Electron. **52**(1), 280–290 (2005). Feb
12. Rodríguez-Rivero, J., Ramírez, J., Martínez-Murcia, F.J., Segovia, F., Ortiz, A., Salas-Rodríguez, D., Castillo, D., Puntonet, C.G., Leiva, J., Carrillo, S., Rodríguez-Rivero, J., Consortium, P., Górriz, J.M.: Granger causality-based information fusion applied to electrical measurements from power transformers. Inf. Fusion (2020)
13. Martínez-Murcia, F.J., Ramírez, J., Segovia, F., Ortiz, A., Carrillo, S., Leiva, J., Rodríguez-Rivero, J., Górriz, J.M.: Prediction of transformer temperature for energy distribution smart grids using recursive neural networks. In: Proceedings of the International Conference on Time Series and Forecasting, vol. 1, pp. 167–177. Granada, Spain (2019)
14. Booth, C., McDonald, J.: The use of artificial neural networks for condition monitoring of electrical power transformers. Neurocomputing **23**(1), 97–109 (1998)
15. de Faria, H., Costa, J.G.S., Olivas, J.L.M.: A review of monitoring methods for predictive maintenance of electric power transformers based on dissolved gas analysis. Renew. Sustain. Energy Rev. **46**, 201–209 (2015)
16. AJ, C., Salam, M., Rahman, Q., Wen, F., Ang, S., Voon, W.: Causes of transformer failures and diagnostic methods–a review. Renew. Sustain. Energy Rev. **82**, 1442–1456 (2018)
17. Catterson, V.M., McArthur, S.D.J., Moss, G.: Online conditional anomaly detection in multivariate data for transformer monitoring. IEEE Trans. Power Deliv. **25**(4), 2556–2564 (2010). Oct
18. Billings, S.A., Leontaritis, I.J.: Identification of nonlinear systems using parametric estimation techniques. In: Proceedings of the IEE Conference on Control and its Application, pp. 183–187. Warwick, UK
19. Leontaritis, I.J., Billings, S.A.: Input-output parametric models for non-linear systems part i: deterministic non-linear systems. Int. J. Control **41**(2), 303–328 (1985)
20. Leontaritis, I.J., Billings, S.A.: Input-output parametric models for non-linear systems part ii: stochastic non-linear systems. Int. J. Control **41**(2), 329–344 (1985)
21. Kendall, M.G.: A new measure of rank correlation. Biometrika **30**(1–2), 81–93 (06 1938)

Short-Term Forecast of Emergency Departments Visits Through Calendar Selection

Cosimo Lovecchio, Mauro Tucci, Sami Barmada, Andrea Serafini, Luigi Bechi, Mauro Breggia, Simona Dei, and Daniela Matarrese

Abstract Emergency Departments (ED) overcrowding is a common and well-known problem, associated with decreased patient safety, increased mortality rates, and which leads to staff burning out. The ability to predict the ED hourly visits can then relieve overcrowding's consequences. In this paper (The content of this paper is a contribution to the International Conference on Time Series and Forecasting 2019 (ITISE2019), held in Granada [1].), we present a method which takes into account calendar effects for short-term forecasting of ED visits is presented. Our approach combines a calendar selection rule with a well-known machine learning algorithm belonging to the class of similar shape algorithms, to predict the incoming visit volume for a tunable number of days ahead.

Keywords Emergency department · Hospital · Forecasting · Time-series prediction

1 Introduction

Overcrowding of Emergency Department (ED) is defined as "the situation in which ED function is in a difficult situation primarily because of the excessive number of patients waiting to be taken in charge, undergoing assessment and treatment, or waiting for departure compared to the capacity of the ED" [2]. This must not be confused with major emergencies that are due to clearly different causes, and require different solutions. Overcrowding is a condition that is strongly associated with the

C. Lovecchio · M. Tucci (✉) · S. Barmada
Department of Energy, Systems, Territory and Constructions Engineering, University of Pisa, Pisa, Italy
e-mail: mauro.tucci@unipi.it

A. Serafini · L. Bechi · M. Breggia · S. Dei
AUSL Tuscany South-East, Grosseto, Italy

D. Matarrese
AUSL Tuscany Center, Florence, Italy

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*, Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_27

risk of impairment of the quality of care provided: latency in taking charge, delay in carrying out diagnostic tests and in starting treatment, increase in errors and adverse events [3]. According to the Joint Commission on the Accreditation of Healthcare Organizations, one third of sentinel events in the EDs are caused by an overcome of the ED capacities. Overcrowding in the EDs leads to many negative consequences, such as an increase in mortality [4, 5], negative perception by patients [6–8] often resulting from prolonged stay on stretchers without privacy or adequate responses to basic needs, and a higher probability of ED staff "burn-out", that causes a further loss of efficiency and a worsening of the shelters filter function with an increase in overall hospitalization times. It is a widespread problem that has been addressed in recent years with targeted interventions in several countries with universal access health systems, such as United Kingdom, Canada, Australia, and New Zealand [9–13]. Trends of ED visits are quite predictable throughout the year and during the different moments of the day, based on seasonal epidemiology and circadian distribution of accesses. The correct management of these trends allows to avoid critical situations, in particular during periods of influenza epidemics [14]. Several factors have been recognized, often acting simultaneously, whether at the presenting of the patient at the ED ("input" factors), along the internal path to the PS ("throughput") or at the patient discharge/transfer ("output" factors). Input factors refer to the numerous ED visits mainly due to seasonal epidemiology, while throughput factors indicate the length of the patient's stay in ED. Finally, output factors are influenced by the difficulty of hospitalization, due to lack of available beds and the difficulty of discharge, especially for patients with social problems. It has been widely demonstrated that throughput and output factors contribute the most to the system overload and, unlike input factors, can be significantly modified by adopting appropriate organizing strategies [15]. The overcrowding of ED depends on two factors: – Crowding: the critical increase in both the admissions and permanence within ED of patients who are completing the diagnostic-therapeutic process; – Boarding: the accumulation in ED of patients who have already completed the care process but who, for various reasons, cannot be discharged from ED [16].

1.1 Crowding and Boarding

The analysis of the level of ED crowding is mainly addressed to 2 areas: the access phase (how many patients arrive, how, by whom, at what time of day, etc.) and the "process" phase, i.e., the whole clinical and therapeutic path within ED. In Tuscany, the analysis of data on the trend of time bands, especially with reference to color codes, confirms an inappropriate use of ED instead of other settings (70–75% 8–20 vs. 30% 20–8, usually <10% of the admissions of 24–8) [17]. In the population, there is the belief that ED is the starting point of many of the diagnostic-therapeutic pathways 'subjectively' considered urgent, while family doctors are considered for the continuation of the pathway and follow-up. It is necessary to redistribute the inappropriate share of demand through an intervention strategy that crosses several treatment processes. Another contribution to crowding is represented by people affected by chronic diseases, already followed by other services both at the local and hospital level, which experience a high percentage of repeated admissions for the same disease (heart failure, complicated diabetes, etc.). Investigation of these patients involves repetitions of laboratory and instrumental tests that unnecessarily absorb a large number of resources, and which would not be necessary if the patients had addressed the doctors who treat them. Countermeasures to the phenomenon of crowding include the redistribution of tasks within the assigned staff, the activation of available staff, and the detention in service of "disassembly" staff. This also applies when crowding has been largely generated by boarding, which absorbs time and staff work, contributing to the progressive increase in waiting time. In this case, the actions must be supplemented by those necessary for the proper management of boarding [17, 18].

The accumulation in ED of patients who have already completed the care process is largely due to the waiting for the bed, mainly in the medical area. These are mainly elderly people with comorbidities with high absorption of resources who remain for a long time in unsuitable environments. In many cases, the demand for hospitalization is generated by the hospital facilities themselves, where these patients are already being treated, reaching, in some cases, about 10–15% of hospitalizations. In addition, there are chronic patients with repeated hospitalizations for the same disease (heart failure, COPD, complicated diabetes, etc.), mainly intended for the medical area. The clinical evolution of these patients is in many cases gradual and progressive and this could have allowed the organization of hospitalization, when appropriate, without the need to access the ED that, in fact, becomes only the place of waiting for the bed. The number of these admissions can also represent 20–25% of admissions in the medical area and often involves more admissions during the year, always through the ED. This "avoidable boarding" is about 30 and 40% of the phenomenon. To solve the problem of boarding, the whole hospital must work together to ensure the balance between supply and demand at various stages of the treatment process. For this reason, it is necessary to effectively manage the flows of incoming and outgoing patients, to optimize the emergency and planned routes and to make more efficient use of the hospital beds [17].

1.2 Forecast Motivation

If on one hand the internal queues and patient flow management is a crucial aspect to consider in order to reduce overcrowding, improve the quality of service, and reduce operating costs, on the other by an accurate forecast of the ED services demand enable proper planning of the clinical resources amount to activate. Nevertheless, the identification of a feasible forecast tool rises some challenges. A first aspect to consider is the quality and quantity of historical informations about a specific scenario. One might be tempted to claim that collecting a large amount of data describing the present, we could be able to predict the immediate future visits volume. Actually, it often turns out that the most accurate source of information which can be used to predict the future behavior of a physical quantity is the past behavior of the quantity itself. Another important aspect to undertake is the selection of a predictive model. Once the ED patient volumes in a sufficiently long time window have been collected, this type of forecast can be enclosed in the time-series prediction framework, one of the most transversal research topic. In fact, a plethora of analytical tools are available to describe temporal dynamics, ranging from classical statistical models [19–26], to more recent artificial intelligence based algorithm. Each of these tools has features which make it more suitable or reliable than the others in a particular application. In many time-series forecasting problems, where human and social activities are predicted, environmental factors affect the resulting collective behavior to a different extent, but among all the external sources of influence, calendar patterns play a crucial role [27]. It is well known how, for different calendar day types (working day, holidays, special holidays), different human dynamics (e.g., shopping behavior of buyers, traffic patterns, crowding effects in places of entertainment, etc.) can be observed. In this work, we use a variation of a popular and well-established time-series forecasting model belonging to the class of "similar shape" algorithms (K-nearest neighbors, or knn), to predict several days ahead hourly patient volumes in 13 ED facilities of a local health center of Tuscany. Our model (C-knn in the following) includes a control mechanism on the calendar condition for the prediction provided. We evaluate the forecast accuracy by means of two performance indicators, the mean absolute percentage error, MAPE, and variance of absolute percentage error, VAPE, estimators.

2 Methodology

We apply our model to a dataset of aggregated informations, extracted from the accesses records in the EDs facilities. The facilities analyzed have different characteristics, such as size, services provided (depending on the hospital equipments), or dimension of areas served. Each record of the databases extracted from the servers contain all the data related to a single ED access, such as date and time of admission, priority code (color code), ED infrastructure, age, sex, and other specific informations. Among these, we focus on date, time, and color code, calculating the aggregated time series (Fig. 1). The data cover a time window starting on 2014-01-01, and ending on 2018-11-14. Performance analysis of our algorithm was carried out by exploiting the last available year of information (test set), while the remaining data (train set) were used to fit the model metaparameters.

The C-knn algorithm belongs to the class of "similar day-based" or "similar shape" methods [27]. The key idea consists of the research, between the available data, for historical days that are characterized by intraday dynamics similar to the recent past (e.g., similar average, maximum values, or peaks positions) in order to predict the near future. For a more reliable prediction, the set of days in which the search is performed can be bounded by constraints. In particular, our algorithm automatically finds similar profiles in the available database, selecting only those whose weekday sequence exactly matches the actual one. Let us show in detail how the algorithm works. Suppose we are in the day $d_0 \in \mathbb{R}^{24}$, and our goal is the prediction of the hourly accesses in the future N days $f_N = \{d_1, ..., d_N\} \in \mathbb{R}^{24\times N}$. To calculate the prediction we exploit the historical database, assuming that the data covers the accesses history until d0. The steps performed by the C-knn algorithm are

1. Pick out from the database the accesses profile of the consecutive most recent M days $a_M = \{d_{-M+1}, ..., d_0\} \in \mathbb{R}^{24\times M}$, and subtract from it its mean value $a_M(0) = a_M - \langle a_M \rangle$. The number of days M to select is a metaparameter of the model.
2. Take all the possible sub-series of M+N consecutive days in the historical database $p(i) = \{d_{i-M+1}, ..., d_i, ..., d_{i+N}\} \in \mathbb{R}^{24\times(M+N)}, i = -N, -N-1, ...$
   For each series $p(i)$, the first M days portion will be denoted as $p_M(i) \in \mathbb{R}^{24\times M}$, the last N days as $p_N(i) \in \mathbb{R}^{24\times N}$.
3. Discard from the set $p(i)|_{i=-N,...}$ those elements whose calendar condition on the $p_N(i)$ part is different from the f_N one. We will clarify in the following the "calendar condition" meaning. The remaining "bounded" set will be used for the reconstruction.
4. Calculate the zero mean profiles $p_M^{(0)}(i) = p_M(i) - \langle p_M(i) \rangle$.
5. Calculate the weighted distances
   $d(i) = ||p_M^{(0)}(i) - a_M^{(0)}||_{W^2} = ||W \cdot (p_M^{(0)}(i) - a_M^{(0)})||$
   where $W \in \mathbb{R}^{24\times M\times M}$ is a weight vector giving different importance to different hours. The coefficient in W are also metaparameters of the model.
6. Select the k most similar $p_M(i)$ (i.e., those whose corresponding $d(i)$ is minimum) and the related $p_N(i)$. The value of k is another metaparameter of the model. We will denote the chosen $p_M(i)|_{i=i_1,...,i_k}$ as the "best profiles" $b_M(j)|_{j=1,..,k}$, and the related $p_N(i)|_{i=i1,...,ik}$ as "best candidates" $c_N(j)|_{j=1,..,k}$.
7. Compute the similarity scores sj by the Gaussian kernel
   $s_j = e^{-d^2(j)/\sigma^2}, j = 1, ..., k.$
   The kernel width value σ in the equation above is defined as proportional to the smallest distance $d(j)$, $\sigma = \lambda\, min\{d_j\}$, where λ is a positive constant to be optimized.
8. Reconstruct the recent past a_M by using the similarity scores and the best profiles $\overline{a_M} = \sum_j s_j b_M(j)$ and look for the best scaling factor α^* which minimizes the distance between the recent past and the reconstructed one, i.e.,
   $\alpha^* = argmin_\alpha ||\alpha\, \overline{a_M} - a_M||$.
9. The final forecast is finally given by the scaled weighted sum of the best candidates $f_N^* = \alpha^* \sum_j s_j c_N(j)$.

2.1 Calendar Conditions

ED patient volume strongly depends on calendar variables. In addition to seasonal trends, special days or events occurring during the year appear as anomalies compared to other days, as shown in Fig. 1. Volume is in average lower on national holidays

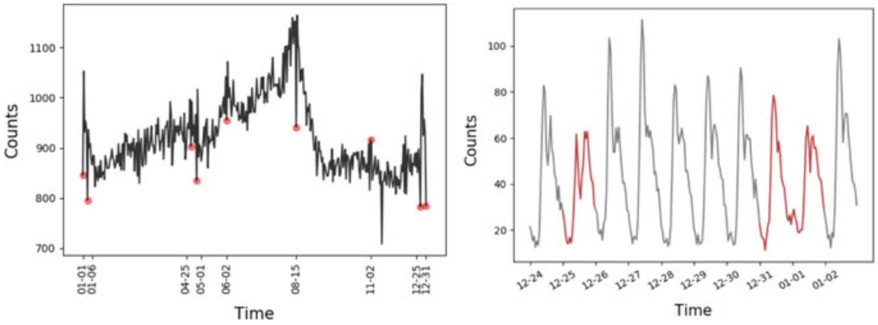


Fig. 1 Weekday dependence of the total patient visits in the 13 ED facilities under study. On the left side: mean daily patient volume during the year (The average is computed on 4 years). Red points highlight some of the national holidays, namely New Year's day, Epiphany, Liberation day, May Day, Italian Republic Holiday, Assumption day, All Souls' Day, Christmas, New Year's Eve. On the right side: mean hourly visits amount during Christmas holidays. Red line highlight Christmas day, New Year's Eve and New Year's day

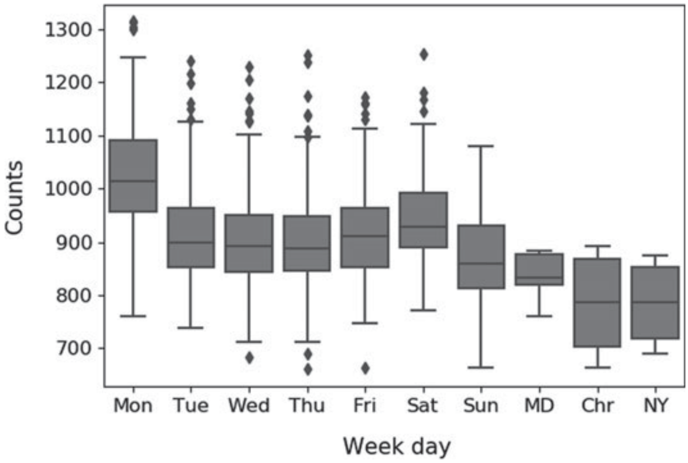


Fig. 2 Boxplot of the daily total accesses during weekdays and some of the national holidays (MD May day, Chr Christmas, NY New Year's day)

and on Sunday, while appear higher on Monday, as can be noted in Fig. 2. Therefore, a day parametrization mapping the calendar pattern turns out to improve the quality of the forecast. In particular, we divide the weekdays into the three following classes:

1. Working Days: days from Monday to Friday, excluding special holidays.
2. Saturdays: all Saturdays excluding holidays.
3. Holidays: all Sundays and special holidays (Easter Monday, Christmas, New Year's Day, etc.).

The third step of the algorithm listed above consists of the elimination of those sequences whose future calendar condition does not match with the actual future we aim to predict.

3 Model Evaluation

The goodness of the model was assessed evaluating two performance indexes: mean absolute error (MAE) and variance of the absolute Error (VAE). Given a time-series $y(n), n \in [1, ..., N]$ and its reconstruction $\overline{y(n)}$, MAE is defined as

$$\text{MAE} = \frac{1}{N}\sum_{n=1}^{N}\left|y(n) - \overline{y(n)}\right| \tag{1}$$

and VAE as

$$\text{VAE} = \frac{1}{N}\sum_{n=1}^{N}\left(\left|y(n) - \overline{y(n)}\right| - \text{MAE}\right)^2 \tag{2}$$

The first index reflect the model accuracy, while the second one is a measure of the model stability. We also report the Mean Bias Error (MBE), defined as

$$\text{MBE} = \frac{1}{N}\sum_{n=1}^{N} y(n) - \overline{y(n)}. \tag{3}$$

The MBE quantifies how the model is biased compared to the true time series.

The model parameters which can be optimized are

1. The number of days in the past M to compare with the historical database.
2. The weight vector W filtering the time sequences.
3. The number of most similar patterns k.
4. The size of the kernel function.

To downsize the computational effort in the parameters tuning, we assume W to be diagonal with linearly increasing coefficients, reducing its degree of freedom to only the initial and final values, and $\lambda = 1$. The performance indexes landscape was then obtained by Grid Search over a suitable parameter space, uniformly sampled. In particular, for every parameter combination $\overline{p} = \{M, k\}$, we simulated a true forecast using an incrementally expanding historical set, which was performed iteratively on the last available year accesses. After the forecasts production, $\text{MAE}_i(\overline{p})$ and $\text{VAE}_i(\overline{p}), i \in [1, ...13]$, were calculated for each ED facility, for a prediction horizon of 1 day-ahead.

To set a convenient metaparameters combination, we finally calculated the total MAE and VAE as $\text{MAE}_T(\overline{p}) = \sum_i \text{MAE}_i(\overline{p})$, $\text{VAE}_T(\overline{p}) = \sum_i \text{VAE}_i(\overline{p})$. A density

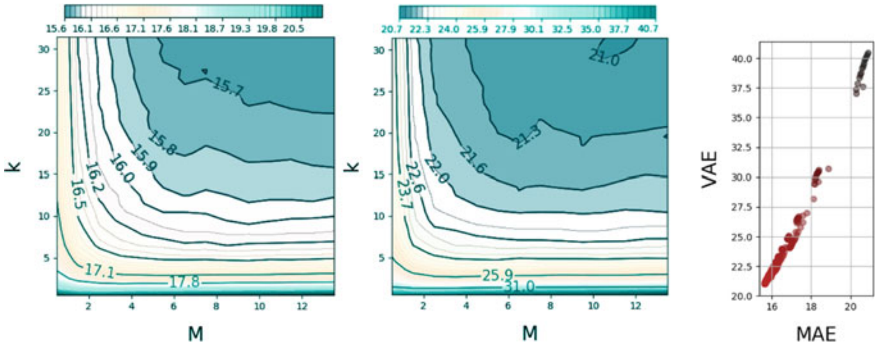


Fig. 3 Density plot of MAE_T (left side) and VAE_T (center) as a function of the optimization parameters M (number of comparison days in the past), and k (number of nearest neighbors). As can be noted, no improvements of the total scores are appreciable for $M \gtrapprox 6$, $k \gtrapprox 25$. MAE dependence of VAE (left side) in the parameters grid explored. As can be noted, there is an almost linear dependence between the two quantities

plot of these two quantities against the M and k is shown in Fig. 3. As can be noted the algorithm performances monotonically increases for both increasing M and k, reaching a plateau region for approximately $M \gtrapprox 25$, $k \gtrapprox 6$. Moreover, MAE and VAE are linearly correlated for the set of parameter explored.

The final settings we adopted to perform the forecast thus are $M = 6$, $k = 32$.

3.1 Results

The resulting scores calculated in correspondence of the selected metaparameters M and k, and for all the single ED structures, are summarized in Table 1. In particular, we report MAE, VAE, and MBE on a daily and hourly aggregated timescale.

In Fig. 4, we show a forecast example of the accesses for the ED 3 in Table 1, in correspondence of the New Year's Eve week. As can be observed, the predicted data (solid red line) adequately resembled the actual data (solid gray line) in the test set.

As can be noted from Table 1, the algorithm performs as better as the average hourly and daily accesses are higher, since for small volumes of patient income the daily dynamics are closer to a random process. For the predicted data the hourly MAE(h) ranges from 87% of the mean hourly accesses in the smaller facility (ED 6), to the 27% in the bigger one (ED 3). On a daily timescale, the prediction quality appears to improve, since for the same ED (ED 6 and ED 3) MAE(d) are 29% and 8.5%.

Table 1 Table of resulting forecast MAE, VAE, and MBE, calculated along the last one year of data, aggregated on daily (MAE(d), VAE(d), and MBE(d)), and hourly (MAE(h), VAE(h) and MBE(h)) timescale

ED	Mean daily accesses	MAE(d)	MBE(d)	VAE(d)	Mean hourly accesses	MAE(h)	MBE(h)	VAE(h)
1	38.65	8.3	1.02	44.27	1.61	1.03	0.04	1.06
2	39.72	7.43	−1.45	36.06	1.65	0.95	−0.06	1.06
3	203.8	17.35	1.23	214.68	8.49	2.32	0.05	4.49
4	104.08	13.09	−2.23	111.4	4.34	1.62	−0.09	2.15
5	36.05	7.12	−0.07	30.78	1.5	0.91	0	0.88
6	17.42	5.03	−0.64	19.8	0.73	0.56	−0.03	0.49
7	75.83	11.47	−3.18	157.41	3.16	1.35	−0.13	1.66
8	75.1	11.05	−2.12	141.3	3.13	1.23	−0.09	1.42
9	39.51	7.55	−2.97	39.6	1.65	0.93	−0.12	0.95
10	67.71	10.5	−2.13	82.3	2.82	1.31	−0.09	1.73
11	14.82	4.55	−0.26	14.4	0.62	0.54	−0.01	0.44
12	21.69	5.53	−1.18	22.82	0.9	0.68	−0.05	0.58
13	186.71	19.88	−0.77	251.57	7.78	2.25	−0.03	4.14

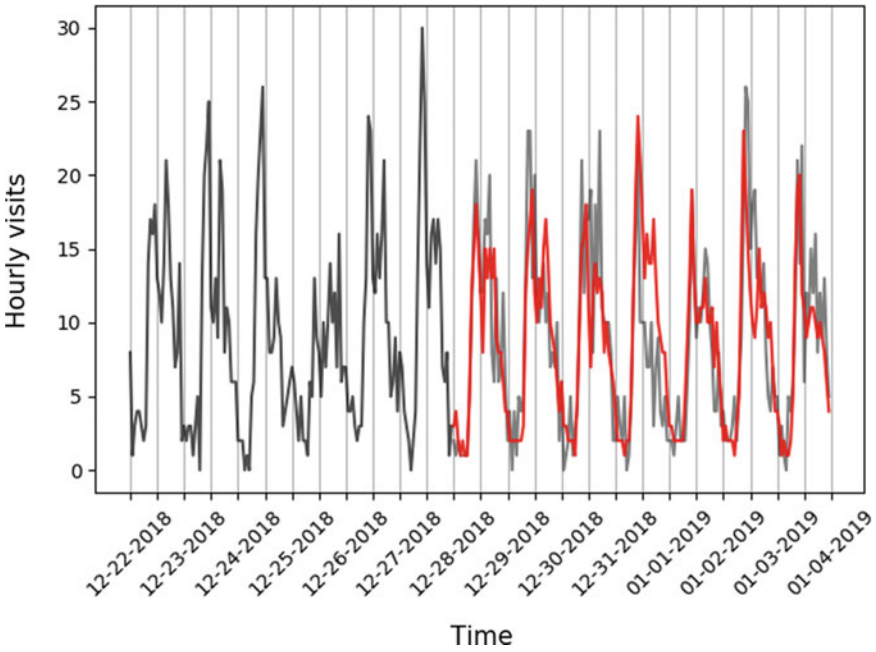


Fig. 4 Incoming visits forecast in correspondence of the New Year's Eve week. Black solid line represents the data portion used to look for similar patterns in the historical dataset, light gray the true data and red line their forecast

3.2 Comparison to Other Models

We compared the performance of our forecasting scheme to two alternative prediction systems: a REplication model (RE), and an Artificial Neural Network model (ANN). In the RE model, the N days ahead prediction is obtained by replicating the most recent sequence in the historical dataset sharing the same calendar pattern. In this way, the prediction will always mimic the most recent matching past. The ANN model consists of a single layer feedforward network composed of 130 hidden neurons (as obtained by metaparameter optimization). The hidden and output neurons activation functions are the rectified linear (ReLU) activation. This model was trained to reconstruct the $p_N(i)$ vectors in the training set in two different ways, based on the input provided:

- Only the $p_M(i)$ vectors (see Model section), thus with no information about any calendar pattern.
- The concatenation between $p_M(i)$ and the calendar condition of the days to reconstruct, the latter encoded in a vector in $\mathbb{R}^N$.

The models' performances are listed in Table 2 where we show, for each algorithm, the average scores for all the ED facilities.

As can be noted, the C-knn outperforms all the competing models, while the RE algorithm is less biased and second best. On the contrary, both the ANNs do not perform well, probably because they encode a representation of the full training set, thus their predictions tend to be an average of the whole past dynamic which lose some short scale details instead captured by the C-knn, which restrict the set from which the forecast is built only to the more similar temporal patterns.

Table 2 Performances of the tested models. MAE, MBE, and VAE for a single model are calculated averaging the scores of all EDs, and has to be compared with the average mean daily accesses 70.85, and the average mean hourly accesses 2.95 in all EDs

Model	MAE(d)	MBE(d)	VAE(d)	MAE(h)	MBE(h)	VAE(h)
Replica	10.17	0.06	88.06	1.59	0.00	2.73
ANN (no calendar)	15.54	13.22	123.52	1.39	0.55	2.10
ANN (calendar)	14.11	11.56	117.47	1.38	0.48	2.08
C-knn	9.91	−1.13	89.72	1.21	−0.05	1.62

4 Conclusion

The adverse consequences of ED crowding can be as much severe as clinical staff and ED administrators are unaware of the incoming situation. The ability to predict future input demand can relieve the negative effects of a possible disadvantageous situation, and support structural intervention to maintain performances and help service improvement. The unsupervised algorithm presented here, belonging to the "similar shape" algorithm category, is able to automatically provide a short-term hourly forecast based on calendar condition, without the need of a training phase, but only exploiting a historical dataset in which similar patterns are picked up. Thanks to these aspects, it is suitable to be directly applied to any specific situation, providing accurate and reliable predictions.

References

1. Lovecchio, C., Tucci, M., Barmada, S., Serafini, A., Bechi, L., Breggia, M., Dei, S., Matarrese, D.: Calendar based forecast of emergency department visits. In: International Conference on Time Series and Forecasting 2019 (ITISE2019), pp. 869–880 (2019)
2. Statement on Emergency Department overcrowding. Australasian College for Emergency Medicine. Jul 16; S57 (2011)
3. Liu, S.W., Thomas, S.H., Gordon, J.A., Hamedani, A.G., Weissman, J.S.: A pilot study examining undesirable events among emergency department-boarded patients awaiting inpatient beds. Ann. Emerg. Med. **54**(3), 381–385 (2009)
4. Richardson, D.B.: Increase in patient mortality at 10 days associated with emergency department overcrowding. Med. J. Aust. **184**(5), 213–216 (2006)
5. Sprivulis, P.C., Da Silva, J.A., Jacobs, I.G., Frazer, A.R., Jelinek, G.A.: The association between hospital overcrowding and mortality among patients admitted via Western Australian emergency departments. Med. J. Aust. **184**(5), 208–212 (2006)
6. Pines, J.M., Iyer, S., Disbot, M., Hollander, J.E., Shofer, F.S., Datner, E.M.: The effect of emergency department crowding on patient satisfaction for admitted patients. Acad. Emerg. Med. **15**(9), 825–831 (2008)
7. Di Somma, S., Paladino, L., Vaughan, L., Lalle, I., Magrini, L., Magnanti, M.: Overcrowding in emergency department: an international issue. Int. Emerg. Med. **10**(2), 171–175 (2015)
8. Chen-Mei, H., Li-Lin, L., Yun-Te, C., Wang-Chuan, J.: Emergency department overcrowding: Quality improvement in a Taiwan Medical Center. J. Formosan Med. Assoc. **118**(1), 186–193 (2019)
9. Forero, R., Hillman, K.M., McCarthy, S., Fatovich, D.M., Joseph A.P., Richardson D.B.: Access block and ED overcrowding. Emerg. Med. Australas. **22**, 119–135 (2010)
10. Gilligan, P., Winder, S., Ramphul, N., O'Kelly, P.: The referral and complete evaluation time study. Eur. J. Emerg. Med. **17**, 349–353 (2010)
11. Bullard, M.J., Villa-Roel, C., Bond, K., Vester, M., Holroyd, B., Rowe, B.: Tracking emergency department overcrowding in a tertiary care academic institution. Healthc. Q. **12**, 99–106 (2009)
12. Richardson, D.: Access Block Point Prevalence Survey. The Australasian College for Emergency Medicine (2008)
13. Sun, B.C., Hsia, R.Y., Weiss, R.E., et al.: Effect of emergency department crowding on outcomes of admitted patients. Ann. Emerg. Med. **61**, 605–11.e6 (2013)
14. Amodio, E., Cavalieri, d'Oro L., Chiarazzo, E., Picco, C., Migliori, M., Trezzi, I., Lopez, S., Rinaldi, O., Giupponi, M.: Emergency department performances during overcrowding: the

experience of the health protection agency of Brianza AIMS. Public Health **5**(3), 217–224 (2018)

15. Asplin, B.R., Magid, D.J., Rhodes, K.V., Solberg, L.I., Lurie, N., Camargo Jr., C.A.: A conceptual model of emergency department crowding. Ann. Emerg. Med. **42**(2), 173–180 (2003)
16. Higginson, I.: Emergency department crowding. Emerg. Med. J. **29**, 437–443 (2012)
17. Piani Aziendali per la gestione del sovraffollamento in Pronto Soccorso (PGSA)-Linee di indirizzo, regione Toscana. https://www.frgeditore.it/images/cop/pdf/titolo-1/razionalizzazione/toscana/all_dgr_974
18. McCarthy, M.L., Ding, R., Pines, J.M., Zeger, S.L.: Comparison of methods for measuring crowding and its effects on length of stay in the emergency department. Acad. Emerg. Med. **18**(12), 1269–1277 (2011)
19. Jones, S.S., Thomas, A., Evans, R.S., Welch, S.J., Haug, P.J., Snow, G.L.: Forecasting daily patient volumes in the emergency department. Acad. Emerg. Med. **15**(2), 159–170 (2008)
20. Schweigler, L.M., Desmond, J.S., McCarthy, M.L., Bukowski, K.J., Ionides, E.L., Younger, J.G.: Forecasting models of emergency department crowding. Acad. Emerg. Med. **16**(4), 301–308 (2009)
21. Peck, J.S., Benneyan, J.C., Nightingale, D.J., Gaehde, S.A.: Predicting emergency department inpatient admissions to improve same-day patient flow. Acad. Emerg. Med. **19**(9), E1045–E1054 (2012)
22. Boyle, J., Jessup, M., Crilly, J., Green, D., Lind, J., Wallis, M., Fitzgerald, G.: Predicting emergency department admissions. Emerg. Med. J. **29**(5), 358–365 (2012)
23. Kadri, F., Harrou, F., Chaabane, S., Tahon, C.: Time series modelling and forecasting of emergency department overcrowding. J. Med. Syst. **38**(9), 107 (2014)
24. Afilal, M., Yalaoui, F., Dugardin, F., Amodeo, L., Laplanche, D., Blua, P.: Forecasting the emergency department patients flow. J. Med. Syst. **40**(7), 175 (2016)
25. Zor, C., Çebi, F.: Demand prediction in health sector using fuzzy grey forecasting. J. Enterp. Inf. Manag. **31**(6), 937–949 (2018)
26. Zhang, Y., Luo, L., Yang, J., Liu, D., Kong, R., Feng, Y.: A hybrid ARIMA-SVR approach for forecasting emergency patient flow. J. Ambient Intell. Humanized Comput. 1–9 (2018)
27. Barmada, S., Raugi, M., Tucci, M.: A multi-objective optimization algorithm based on self-organizing maps applied to wireless power transfer systems. Int. J. Numer. Model.: Electron. Netw. Dev. Fields **30**(3–4), e2145 (2017)

Discordant Observation Modelling

Sonya Leech and Bojan Bozic

Abstract Time-series modelling on discordant observations or volatile data needs careful consideration when choosing the right model. Our approach is to identify model performance based on time-varying volatility in application log files. A comparison will be done on two different modelling techniques, ARIMA and GARCH whilst extracting a limited understanding of the types of messages sent to the log files. Different model parameter settings will also aid in model performance analysis. Being able to predict volatile data whilst understanding the context of the data can help Dev-Ops support teams be more sagacious in their support and control of business processes that can help narrow the bandwidth of future occurrences. This paper presents a comparative analysis into time-series predictions of log events using both ARIMA and a hybrid model of ARIMA-GARCH that will aid in anomaly detection. It also takes a simple approach to the classification of the textual messages of the log data to understand the type of messages being recorded. The findings of the study conclude that ARIMA is not suitable for modelling volatile data whilst ARMA-GARCH is a more performant model.

1 Introduction

Anomaly detection has been heavily researched in many domains. It has been written about as early as 1887 by [1] in which he refers to a discordant observation as an anomaly. These discordant observations are patterns in data that do not conform to expected behaviours over a function of time. Anomaly detection has been implemented in many domains including but not limited to cellular cloning [2], credit card fraud [3], network intrusions [4] and network traffic monitoring [5].

S. Leech (✉)
IBM, Dublin, Ireland
e-mail: leechsy@ie.ibm.com

B. Bozic
Tu Dublin, Dublin, Ireland
e-mail: bojan.bozic@tudublin.ie

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_28

Anomaly detection is a critical feature in application domains as they can often identify critical actions that need to occur before a major action has caused a significant impact on a system [6]. Such is the case that a technical glitch in Amazon's landing page cost them a loss of $99 million in revenue based on 1P sales from 2017 [7]. $1.6 million was lost in Zappos due to a pricing error [8].

This research is focused on modelling volatile datasets that aid in supporting anomaly detection. Our paper explores different models and approaches to address this problem and is structured as follows: Sect. 2 lists the related work in the field of anomaly detection and volatility modelling. Section 3 describes the approach we took to anomaly detection on volatile data while Sect. 4 shows the evaluation of our approach. Section 5 gives an overview of future work and conclusions.

2 Related Work

A comparative analysis of collective anomalous events was implemented on non-volatile datasets [9]. Hodge and Austin [10] classified anomalies into three different types: modelling unseen, normal and abnormal data. Markou and Singh [11] researched novel techniques using statistical methods alongside neural networks. Singh and Upadhyaya [12] researched outlier detection techniques in multifarious ways and highlighting the complexities within each application domain. A comprehensive review of volatility models was conducted by [13].

3 Volatility Modelling

To detect anomalies some time-series modelling needs to be implemented. These models are known as Stochastic models which observe continuous data over discrete points in time [14]. For a time-series model to be effective in anomaly detection, it needs a fully articulated, well-defined pre- and post-analysis checklist. This will aid in the support and understanding of the data being modelled which can only then become a fully calibrated, highly functional, simulated model which results in a more efficient anomaly detection tool [14].

In time-series modelling, a common approach is the classical white Box-Jenkins Auto-Regressive Integrated Moving Average (ARIMA) model [15]. ARIMA modelling is suited for time-series stationary data. Aggregation of application log data normally makes them non-Gaussian and non-stationary. Unless transformed they can sometimes be quite volatile. If the data displays volatility, one can transform the data using a log or Box-Cox transform. The Generalized Auto-Regressive Conditional Heteroscedasticity (GARCH) model is heavily used in volatile non-stationary datasets and widely used in financial data [16].

An ARCH time-series model would be a more appropriate model for volatile datasets as it can model changes in variance. ARCH understands the difference between the conditional and unconditional variances in the data, letting the condi-

Table 1 Collective Event Counts

Events	Value
Total	3 m
Info	2.7 m
Warn	183 k
Error	179 k

tional variance allow for changes over time as a function of residual errors from a zero-mean process [17]. ARCH relies on previous squared observations and previous variances to help model current variation. ARCH models have a mean of zero, and the time-series data is uncorrelated and contains non-constant variances [17]. A GARCH model is an extension to the ARCH model. GARCH reduces its forecasting errors by accounting for errors in prior forecasts which should enhance its model accuracy for future predictions. GARCH estimates the variance of today as the sum of the alpha, beta and omega components. As stated previously, the aim of our research is to model volatile data using ARIMA and hybrid ARIMA-GARCH models. ARIMA modelling brings back a predicted value whilst a hybrid ARIMA-GARCH model predicts the mean of the time series and the predicted variance. These combined models have been very successful in forecasting volatile financial datasets.

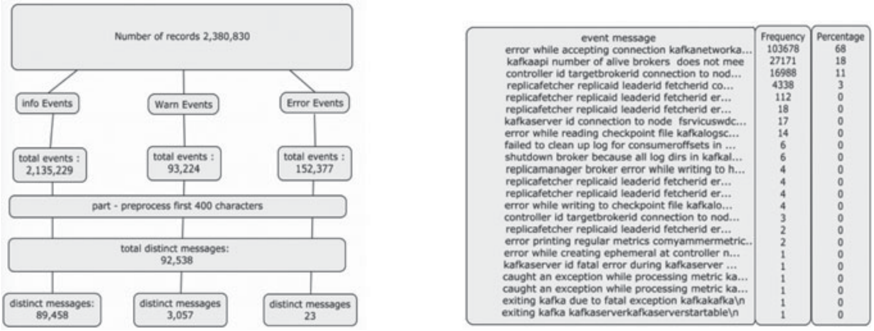
4 Evaluation

On initial observation of the log data, it was identified that some data points were missing completely at random (MCAR) observations. A total of twenty-five individual hours over seven months of data was missing. Different methods can be implemented to support missing data on non-normal data like Robust analysis, Bayesian estimations and multiple imputations [18]. An imputed mean value was used for missing observations on the hourly data for the warn type events. Informational and Error type events were not fully analyzed for time-series modelling.

Table 1 displays a breakdown of the events logged for each severity type. It is noted that a total of just over three million log events were recorded of which 88% were informational, 6% were warn and 6% were error type events.

4.1 Text Analysis

Topic modelling is the discovery of topics in a collection of documents. This is useful when you want to shrink and group the text into different clusters for further analysis. A text corpus is a structured set of texts in a document. To support topic modelling, a Latent Dirichlet Allocation (LDA) method can be used to classify the text in a document [19]. A subset of the log data was analyzed for topic modelling to understand what types of messages are being sent to the log files. A simple grep and



event message	Frequency	Percentage
error while accepting connection kafkanetworka...	103678	68
kafkaapi number of alive brokers does not mee	27171	18
controller id targetbrokerid connection to nod...	16988	11
replicafetcher replicaid leaderid fetcherid co...	4338	3
replicafetcher replicaid leaderid fetcherid er...	112	0
replicafetcher replicaid leaderid fetcherid er...	18	0
kafkaserver id connection to node fsrvicuswdc...	17	0
error while reading checkpoint file kafkalogsc...	14	0
failed to clean up log for consumeroffsets in ...	6	0
shutdown broker because all log dirs in kafkal...	6	0
replicamanager broker error while writing to h...	4	0
replicafetcher replicaid leaderid fetcherid er...	4	0
replicafetcher replicaid leaderid fetcherid er...	4	0
error while writing to checkpoint file kafkalo...	4	0
controller id targetbrokerid connection to nod...	3	0
replicafetcher replicaid leaderid fetcherid er...	2	0
error printing regular metrics comyammermetric..	2	0
error while creating ephemeral at controller n...	1	0
kafkaserver id fatal error during kafkaserver ...	1	0
caught an exception while processing metric ka...	1	0
caught an exception while processing metric ka...	1	0
exiting kafka due to fatal exception kafkakafka\n	1	0
exiting kafka kafkaserverkafkaserverstartable\n	1	0

Fig. 1 Text Classification Frequency

data pre-processing approach was used to parse each of the different types of message events. 2.3 million messages were classified into ninety-two thousand distinct messages.

Figure 1 shows that eighty-nine thousand distinct messages were classified from the 2.1 million informational events. Ninety-three thousand of the warn events were classified into three thousand distinct messages and of the one hundred and fifty-two thousand error events, these could be broken down into twenty-three distinct messages. It is quite interesting to identity the minute amount of distinct messages for the error events. As error events are something of interest to dev-ops people, Fig. 1 also displays the twenty-three distinct messages for the one hundred fifty-two thousand error events. 68% of the messages were caused due to an error accepting a connection to the Kafka network while 18% was related to the Kafka API alive broker message.

4.2 Volatility

Volatile data can be represented as a series of low values followed by a short sharp burst in data that then returns to normal. Figure 2 shows the warn types events over the course of seven months from late September 2018 to late April 2019. The chart to the right displays the daily data and the chart to the left displays the hourly data. We observe from the time-series charts that the data appears volatile. The dataset with the highest degree of volatility ideally should be used for further analysis and anomaly detection.

To identify volatility, we use statistical measures of variance and standard deviations (STD). A STD is a measure of the variability, dispersion in the data. Variability is a measure of volatility. The smaller the standard deviation lends towards the data being centred around the mean and less volatile. The larger the standard deviation the more spread out the data is from the mean and the more volatile it becomes. As

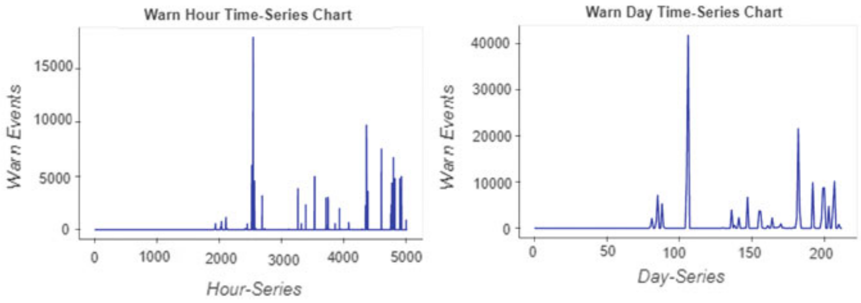


Fig. 2 Time-Series Volatile Warn Type Events

Table 2 Event Analysis: Split Into Groups

Aggregation type	Test	Group 1	Group 2	Difference
Hour	Variance	4033	415603	411570
Day	Variance	1975188	24324812	22349624
Hour	STD	63	644	581
Day	STD	1405	4932	3526

Table 3 Variance And Standard Deviation

Test	Hour	Day
Mean	36	877
Variance	210684	13444351
Standard Deviation	459	3666

per Table 2, the data was split into two different groups for each type of aggregation. Looking at the variance and STD values from the table it shows that the daily warn dataset has a higher degree of volatility than that of the hourly dataset.

Table 3 shows the data over hourly and daily periods, not split into different subgroups. The table shows that the mean, variance and standard deviation of the daily data are quite significant in comparison to the hourly data. The table also shows that the standard deviation is quite larger compared to the mean value. A standard deviation greater than the mean can imply that there is a significant distance between the high and low values and that there may be outliers in the data. It is also an indication that the data is skewed or that there is a wide range of variation amongst the data. A standard deviation that is greater than the mean fails the null hypothesis at the 95% confidence level. It is also observed that the variance is also significantly greater than the mean value, which is quite common in over-dispersed count data.

These results indicate that the dataset for both hourly and daily data is heterogeneous. On first glance of our analysis, these datasets may need to be transformed for ARIMA time-series modelling, and this may not be the case for the hybrid model.

The results of the analysis indicate that the daily warn dataset will be used for time-series modelling and anomaly detection. No further analysis will be done on the hourly dataset.

4.3 Normality

Normality tests were conducted on the data. The "Gaussian" name for the normality tests is derived from the mathematician Johann Karl Gauss [20]. Normality checks can be conducted using visual aids or statistical tests. Some of the most common visual aids to check for normality are histograms, boxplots, probability and quantile plots. These visual approaches are an aid in an assumption about the data being Gaussian but more robust normality tests would need to be conducted. Statistical tests for normality can be implemented using parametric significance tests, which will compare a sample distribution to that of a normal distribution. Some of these tests are regression tests, analysis of variance and t-tests [20].

Quantile and probability graphs were created on the daily warn dataset. As we can see from Fig. 3, outliers are observed. It was noted earlier that the standard deviation was significantly higher than the mean and a possible cause was outliers in the data. Although it is quite common to use the mean plus or minus a three standard deviation, these statistical measures are sensitive to outliers and can be problematic. Another approach would be to use the median absolute deviation for outlier detection [21]. The quantile and probability plots show that the data is not normally distributed as the data does not fit along the regression line.

A skewness of 0 and a kurtosis of 3 would be an indication of a Gaussian distribution. Skewness measures the uniformity in a distribution and kurtosis measures the combined size of the two tails. Shapiro–Wilks (SW) and Anderson–Darling (AD) are statistical tests that measure normality using test statistics and p values. The density graph in Fig. 4 is an indication that the data is not Gaussian and has a heavy right-tailed distribution. We can also see from Tables 4 and 5 that the data is not Gaussian.

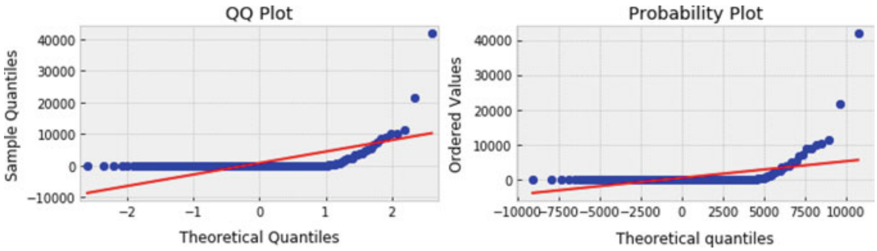


Fig. 3 Quantile and Probability Graphs

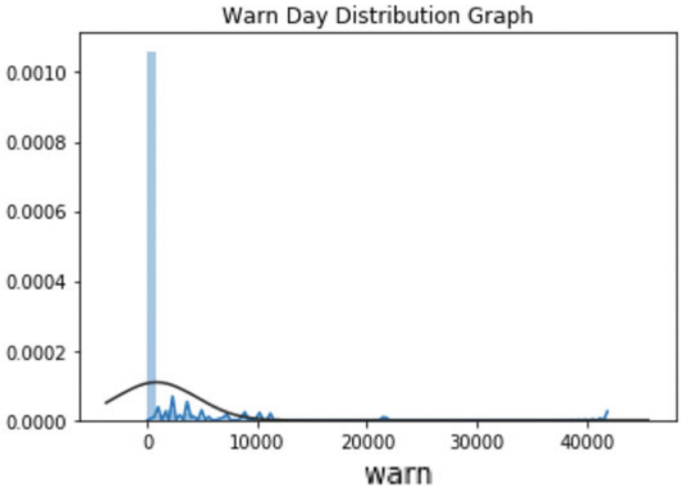


Fig. 4 Warn Type Event Distribution

Table 4 Skewness And Kurtosis

Test	Result
Skewness	7.9
Kurtosis	77.1

Table 5 AD, SW Tests

Test	Statistic
SW	0.25
AD	58.02

To understand if skewness and kurtosis are affected by sample size, Table 6 shows the results of different sample size tests. We can see that the best result for skewness and kurtosis was with a sample size of twenty with the worst sample size being one hundred and twenty observations There appears to be a lot of variation in the results based on the sample sizes specified.

4.4 *Unit Root Test*

Unit root tests were conducted on the data. If a unit root exists, it indicates that a time-series orderly pattern is unpredictable and would need to be transformed. An Augmented Dickey Fueller (ADF) and Kwiatkowski–Phillips–Schmidt–Shin (KPSS) test was implemented. For an ADF test with a P value (0.00) and test statistic (-9.63), we reject the null hypothesis (H0), and there is evidence to suggest that the data is stationary. For a KPSS test which tests for trend stationarity with a p-value (0.07) and

Table 6 Sample Size Skewness And Kurtosis

Sample size	Skewness	Kurtosis
20	2.0	2.8
40	4.7	22.5
60	6.8	47.9
80	7.9	63.8
100	5.0	29.4
120	8.4	79.3
140	6.6	53.1
160	6.9	58.8
180	7.5	68.0
200	7.7	73.3
212	7.9	77.1

test statistic (0.40) at 5% critical value (0.46), we fail to reject the null hypothesis, and the data is trend stationarity. These unit root tests provide evidence to suggest that the time-series data does not need to be transformed.

4.5 *Trend and Seasonality*

For trend, one can look for a serial correlation of the time-series data. This can be done using Auto-Correlation plots. This autocorrelation is also known as an AutoCorrelation function or a correlogram. The blue cone represents the confidence intervals which by default is set to 95%. Any values outside of the cone are considered correlated and not a statistical fluke. A value of 0 indicates no correlation and a value close to 1 indicates a very high correlation. Figure 5 shows the ACF and PACF plots. The results of the ACF and PACF plot show that p and q values should be (1,1). As the data implies stationarity, the d value is 0. No trend is detected in the correlograms.

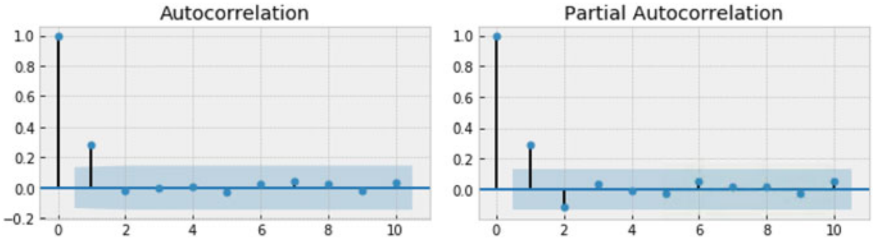


Fig. 5 Auto Correlation, Partial Auto Correlation Graph

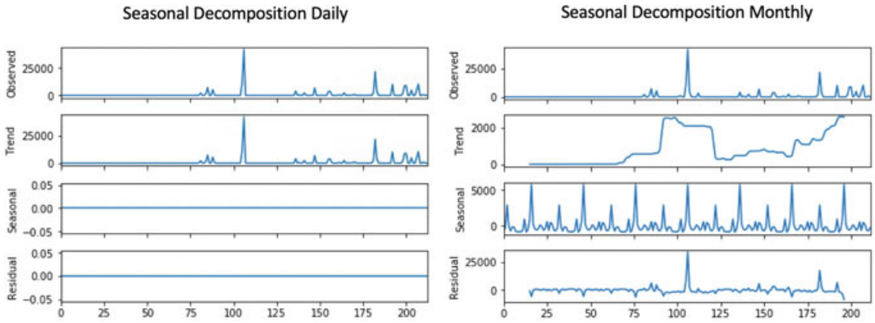


Fig. 6 Seasonal And Trend Observerations

For seasonal identification, a seasonal decomposition test was implemented. Seasonal decomposition decomposes the data into seasonal, trend and random patterns. An example of seasonality is weekdays versus weekends. An example of a trend is a linear increase in data over a period of time. A random pattern is also known as white noise which is the remaining data with the removal of the seasonal and trend data points. The results of the test as per Fig. 6 indicates evidence that no trend or seasonality exists on the daily data but on the higher level of monthly dimensional data, and it does show trend and seasonality. A statistical method to detect seasonality would be a Canova and Hanson test. There are limitations in this test as the data first needs to be transformed and the higher levels of seasonality are not detected by the test [9].

4.6 Goodness of Fit

Goodness of fit tests needs to be conducted on the models. Different goodness of fit tests exist. Those include a portmanteau Ljungbox test. It tests for serial correlation in the residuals. Engle's La Grange Multiplier ARCH test assesses the significance of ARCH effects. It tests for no conditional heteroskedasticity. Statistical significant scores like root mean squared error (RMSE), mean squared error (MSE), Maximum likelihood estimation (MLE) and Akaike's Information Criterion (AIC) are also indicators of goodness of fit tests. AIC is known to asymptotically select a model that results in a set of forecasts with the lowest mean squared error, although AIC can suffer from over-fitting.

Both ARIMA and GARCH models need to define either the best p, d, q or best p, q parameters. An approach to identifying these parameters is to use a best-fit autoregressive (AR) model. The next step is to identify the autocorrelations of the error term. The third step is to test for significance. When choosing the AR model one needs to specify how many prior residual error lags to include in the model. For seasonal data, the recommended lag parameter is twenty and ten for non-seasonal data [22].

Table 7 AR, MA Check

(AR,MA)	(AR,MA)	(AR,MA)	(AR,MA)	AIC
(1,1)	(1,2)	(1,3)	(1,4)	4068
(2,1)	(2,2)	(2,3)	(2,4)	4067
(3,1)	(3,2)	(3,3)	(3,4)	4069
(4,1)	(4,2)	(4,3)	(4,4)	4071

Table 8 ARIMA Model Analysis

Type	Result	AIC	RMSE	MSE
Auto Arima	(0,0,1)	2840	3672	13487675
Grid search (p,d,q)	(2,1,1)	4052	3361	11298511
Correlogram	(1,0,1)	4049	3391	11499774

Different approaches were taken to identify the ARIMA (p,d,q) parameters. The first approach was to run an AR, MA checker on the data to identify the best p and q values. The second approach was to use a statistical auto ARIMA method. The third approach was to use a grid search method with the fourth approach using significant lags from the correlograms.

For our first approach an AR, MA check was implemented. The results from Table 7 show that AR(2), MA(1) are the best parameters that have the lowest AIC score. It appears that when only the AR parameters change, so then does the AIC score. The simplest model with the lowest AIC score should be chosen.

From Table 8, we observe the results of the rest of our statistical approaches. All approaches resulted in different (p,d,q) parameters but similar RMSE scores. Auto Arima had the lowest AIC score but a higher RMSE score than that of the grid search approach. The manual approach using the correlogram resulted in the least performant AIC and RSME scores. When choosing models a simpler model is preferred than that of a more complex model with greater p, d, q parameters.

For GARCH, different approaches were used for detecting the best model. The first approach was the AR, MA checker to determine the p and q values. The second approach was the correlograms. The third approach was running a solver test that identifies the best solver method with the lowest AIC score.

MLE and AIC can be used for goodness of fit tests for GARCH. Different AIC values were extracted from multiple GARCH parameters. As per Table 9 GARCH (1,0) and AR (0,0) had the best AIC score although different measures will be analyzed to avoid over-fitting.

**Best AIC score

As our data was univariate, an ruGARCH model was chosen. If the data was multivariate an rmGARCH would have been the preferred model. Other criteria for the goodness of fit tests were selecting the largest log-likelihood value from the

Table 9 GARCH Model Analysis

GARCH	ARCH	AIC
**(1,0)	(0,0)	18.16
(1,1)	(0,0)	19.47
(1,1)	(1,0)	19.48
(1,1)	(1,1)	19.06

models whilst defining the best solver name with different omega, alpha and beta parameter values and analyzing the fitted and forecasted results. With a GARCH order set to (1,0) and an ARMA order set to (0,0), the default standard GARCH (sGARCH) model was selected. A test was implemented to identify the best solver for fitting the models. The test was based on single and combined solvers. Figure 10 shows the log-likelihood (LLH) value with the solver choices. The results of the table show that nloptr and PRAXIS had the highest LLH score.

**Best Solver Engine

For simplicity, the single solver nloptr was chosen instead of the more complex solvers of nloptr and PRAXIS. Nloptr has then been compared against the default fitted solver. The results of the tests are listed in Tables 11 and 12.

The second row in each table identifies the ARMA and GARCH orders. The third row runs the model with no solver or out of sample values defined and returns the LLH score. The out of sample size was based on a 70–30 split in the data with 30% having 64 data points. A value of − indicates that no parameter was specified and

Table 10 GARCH Solver AIC Scores

Solver	LLH
**nloptr+PRAXIS	−2088
nloptr+AUGLAG+PRAXIS	−2088
nloptr+BOBYQA	−2074
nloptr+AUGLAG+BOBYQA	−2074
nloptr+COBYLA	−2049
nloptr+AUGLAG+COBYLA	−2049
hybrid	−2014
solnp	−2014
nloptr+NELDERMEAD	−1930
nloptr+AUGLAG+NELDERMEAD	−1930
nlminb	−1917
nloptr+SBPLX	−1917
nloptr+AUGLAG+SBPLX	−1917
gosolnp	−1871

Table 11 ARMA(1,0)–GARCH(0,0)

Spec	Out of sample	Log likehood	Solver
ARMA(1,0)–GARCH(0,0)			
Spec 1–	–	−2170	–
Spec 2–	64	−1332	–
**Spec 3–	–	−3207	nloptr
Spec 4–	64	−2200	nloptr

** Represents the best solver engine

Table 12 ARMA(1,1)–GARCH(1,1)

Spec	Out of sample	Log likehood	Solver
ARMA(1,1)–GARCH(1,1)			
Spec 1–	–	−2024	–
Spec 2–	64	−1285	–
Spec 3–	–	−1918	nloptr
Spec 4–	64	−1228	nloptr

default values were used. It is observed that no alpha, beta or omega parameters were defined, as when defined the model reported errors on fitting, so they were removed from the analysis. It is worth noting that using another solver method like solnp did not result in fitting errors and the models performed better when the default omega, alpha and beta parameters were not used.

The result of Table 11 indicates that spec 3 was the model with the highest LLH score and was significantly different from the best score in Table 12. These results favour with the AR, MA checker for GARCH which also identified the same model order.

It is important to note that both visual and statistical comparisons were done on the models. Not one visualization or statistical test was the deciding factor in the deciding model. A collaborative view of the results was used to decide the best model. Spec 1 of ARMA(1,1)–GARCH(1,1) did not perform well against that of spec 3 of ARMA(1,0)–GARCH(0,0). Spec 3 was the accepted model and was the best model for handling the variance in the data.

A Ljungbox (LB) goodness of fit test on the warn data before modelling with a p-value (0.04), we reject the null hypothesis, and there is serial correlation in the data up to lag 10.

The LB test for no autocorrelation on the residuals of the GARCH model with a p-value(1), we accept that the null hypothesis serial correlation does not exist up to lag 10. The same result is achieved in the ARIMA residual LB test.

The goodness of fit Engle LaGrange Multiplier test for conditional heteroscedasticity checks to see if coefficients in the regression are zero. With a p-value(0.99) and a test statistic (0.32), we reject the null hypothesis and ARCH element does exist in the data. One or more coefficients are non-zero and is suited for GARCH time-series modelling.

4.7 Models

ARIMA

Figure 7 returns the predictions for both models (0,0,1) and (2,1,1) on the train and test dataset. We can see from the figures that neither model is a great fit with model (2,1,1) being the worst performant model for predictions. Neither model appears to be fully able to capture the volatility in the data but as a visual aid, the most performant model out of the two models is model (0,0,1). To recap these models were not transformed as they initially passed unit root and trend tests.

GARCH

Figure 8 shows the observed original time-series data on the blue line with the predicted 2 STD's on the red line also known as the sigma. The result of the model does indicate that it can model the variance in the data very well, and there are some under-predictions but overall the STD is quite close to the observed values.

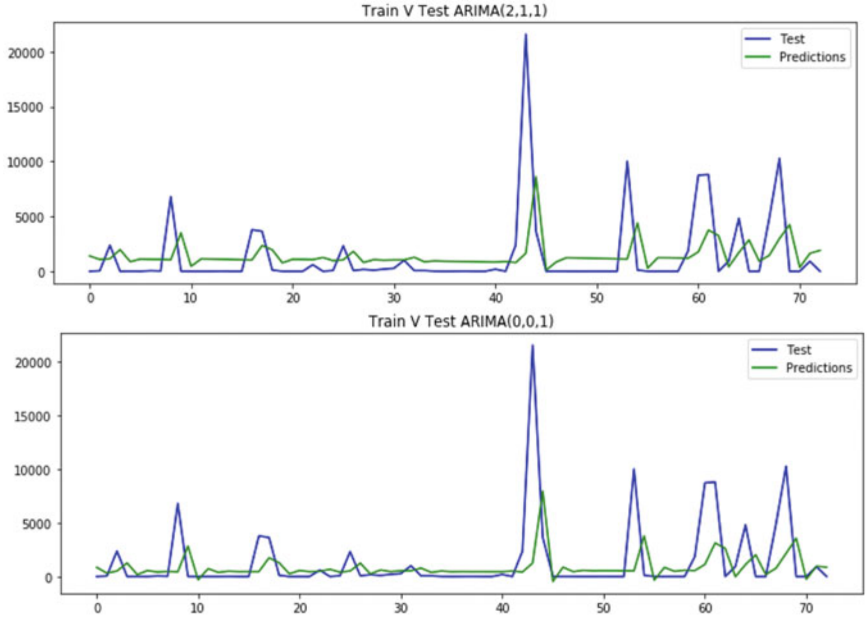


Fig. 7 ARIMA Prediction Results

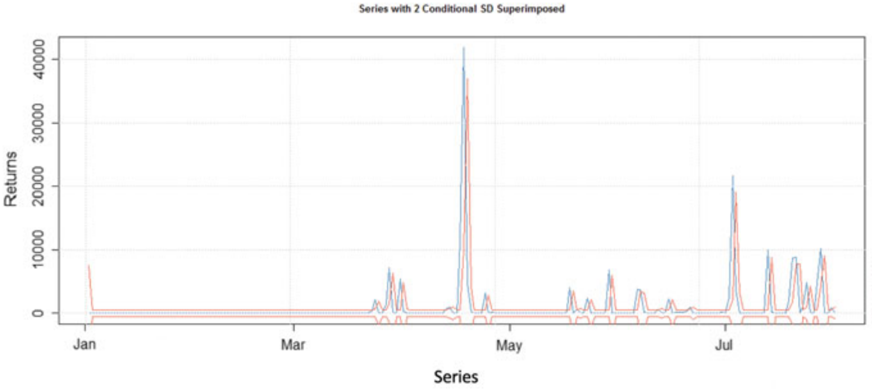


Fig. 8 GARCH Prediction Results

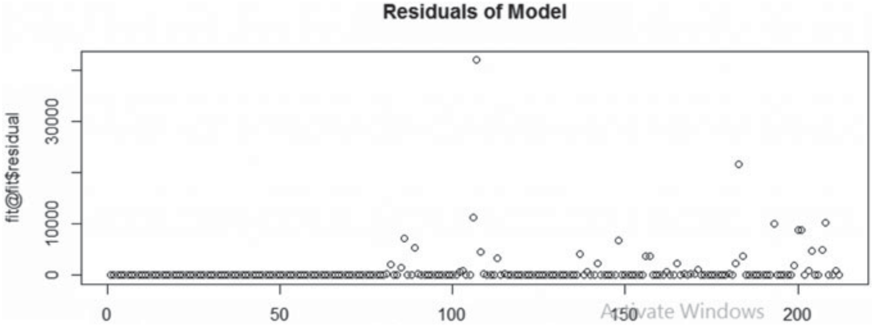


Fig. 9 GARCH Residuals

Figure 9 shows the residuals of the fitted GARCH model. There does appear to be some fluctuations above zero and a transformation of the model would be required to smooth out the errors.

5 Conclusion

Modelling volatile data can be quite complex when not using default parameter specifications. Using default parameters does not always result in the best performant model. It is important to understand the difference between automated modelling tools and the choice of measure used for model selection. As seen in Auto ARIMA, using AIC for model selection with a Grid Search approach using an MSE value, the results of both models confirmed that the methods should not be taken at face value. It has been seen that the grid search approach using the MSE measure was more fitting than that of the statistical auto ARIMA method. The same can be said for GARCH with default parameters set. It was seen that adding the solver method to the fitted model resulted in a better score than when removed. It was also noted that when alpha and beta parameters were defined the model also performed better. Although it was seen that for the solver method the identified model would not fit with alpha and beta parameters set but would fit for other solver names.

It is evident from the results that ARIMA is not a good fit for modelling volatile datasets and GARCH is a more appropriate model. As the data was heteroskedastic, the data for ARIMA needs to be transformed to support better prediction accuracy. As the residuals of the GARCH model show white noise, a log transform of the data would be required.

For missing data, a mean imputation implemented would not have been the best recommendation especially in a volatile dataset but was implemented due to time constraints. More consideration should be taken for the MCAR observations using the other methods identified, which were Robust analysis, Bayesian estimations and multiple imputations.

For topic modelling, a simple approach was used to understand and classify the data within the log messages. A more hardened modelling approach should be considered for future analysis using different classification techniques, although it was quite interesting to see the low amount of distinct messages for the error type events and to understand the highest consumer of those messages.

References

1. Edgeworth, F.Y.: Xli. on discordant observations. London, Edinburgh, Dublin Philos. Mag. J. Sci. **23**(143), 364–375 (1887)
2. Fawcett, T., Provost, F.: Adaptive fraud detection. Data Mining Knowl. Discov. **1**(3), 291–316 (1997)
3. Chan, P.K., Fan, W., Prodromidis, A.L., Stolfo, S.J.: Distributed data mining in credit card fraud detection. IEEE Intell. Syst. **6**, 67–74 (1999)
4. Garcia-Teodoro, P., Diaz-Verdejo, J., Maciá-Fernández, G., Vázquez, E.: Anomaly-based network intrusion detection: techniques, systems and challenges. Comput. Secur. **28**(1–2), 18–28 (2009)
5. Barford, P., Kline, J., Plonka, D., Ron, A.: A signal analysis of network traffic anomalies. In: Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment, pp. 71–82. ACM (2002)
6. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. ACM Comput. Surv. (CSUR) **41**(3), 15 (2009)
7. Korosec, K.: What amazon lost (and made) on amazon prime day (2018). [Online]. Available: https://techcrunch.com/2018/07/18/amazon-prime-day-outage-cost
8. Smith, J.: Pricing error costs zappos $1.6 million (2010). [Online]. Available: https://www.aol.com/2010/05/24/pricing-error-costs-zappos-1-6-million
9. Leech, S. et al.: Forecasting anomalous events and performance correlation analysis in event data (2019)
10. Hodge, V., Austin, J.: A survey of outlier detection methodologies. Artif. Intell. Rev. **22**(2), 85–126 (2004)
11. Markou, M., Singh, S.: Novelty detection: a review-part 1: statistical approaches. Signal Process. **83**(12), 2481–2497 (2003)

12. Singh, K., Upadhyaya, S.: Outlier detection: applications and techniques. Int. J. Comput. Sci. Issues (IJCSI) **9**(1), 307 (2012)
13. Poon, S.-H., Granger, C.W.: Forecasting volatility in financial markets: a review. J. Econ. Literature **41**(2), 478–539 (2003)
14. Hipel, K.W., McLeod, A.I.: Time Series Modelling of Water Resources and Environmental Systems, vol. 45. Elsevier (1994)
15. Box, G.E., Jenkins, G.M., Reinsel, G.C., Ljung, G.M.: Time Series Analysis: Forecasting and Control. Wiley (2015)
16. Engle, R.: Garch 101: The use of arch/garch models in applied econometrics. J. Econ. Perspect. **15**(4), 157–168 (2001)
17. Engle, R.F.: Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. Econometrica: J. Econometric Soc. 987–1007 (1982)
18. Sterne, J.A., White, I.R., Carlin, J.B., Spratt, M., Royston, P., Kenward, M.G., Wood, A.M., Carpenter, J.R.: Multiple imputation for missing data in epidemiological and clinical research: potential and pitfalls. BMJ **338**, b2393 (2009)
19. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
20. Ghasemi, A., Zahediasl, S.: Normality tests for statistical analysis: a guide for non-statisticians. Int. J. Endocrinol. Metab. **10**(2), 486 (2012)
21. Leys, C., Ley, C., Klein, O., Bernard, P., Licata, L.: Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. J. Exp. Soc. Psychol. **49**(4), 764–766 (2013)
22. Hyndman, R.J., Athanasopoulos, G.L.: Forecasting: principles and practice. OTexts (2018)

Applying Diebold–Mariano Test for Performance Evaluation Between Individual and Hybrid Time-Series Models for Modeling Bivariate Time-Series Data and Forecasting the Unemployment Rate in the USA

Firas Ahmmed Mohammed and Moamen Abbas Mousa

Abstract Unemployment rate forecasting has become a particularly promising domain of comparative studies in recent years because it is a major issue facing the economic forecasting process. Since the time-series data are rarely pure linear or nonlinear, obviously, sometimes contain both components jointly. Therefore, this study introduces a hybrid model that combines two commonly used models, namely, the Linear Autoregressive Moving Average with exogenous variable (**ARMAX**) model and nonlinear Generalized Autoregressive Conditional Heteroskedasticity with exogenous variable (**GARCHX**) model whose conditional variance follows a General error distribution (GED). That is, build a hybrid (**ARMAX-GARCHX-GED**) model employed in modeling bivariate time-series data of the unemployment rate and exchange rate. Usually, the forecasting performance evaluation based on the common classical forecast accuracy criteria such as Root Mean Squared Error (**RMSE**), Mean Absolute Error (**MAE**), and Mean Absolute Percent Error (**MAPE**) have some specific limitations in application to choosing the optimal forecasting model. Therefore, in this paper, we employed a modern evaluation criterion based on the methodology advocated by Diebold–Mariano (**DM)** known as (DM test) as a new criterion for evaluation based on statistical hypothesis tests. This (**DM test**) has been applied in this study to distinguish the significant differences in forecasting accuracy between hybrid (ARMAX-GARCHX-GED) and individual ARMAX models. From the case study results and according to DM-test it is observed that the differences between the forecasting performances of models are significant and the hybrid model (ARMAX-GARCHX-GED) is more efficient than the individual competitive ARMAX model for the unemployment rate forecasting.

F. A. Mohammed · M. A. Mousa (✉)
Department of Statistics, College of Administration and Economics, Baghdad University, Baghdad, Iraq
e-mail: saidmoamen@gmail.com

F. A. Mohammed
e-mail: firasmohana@coadec.uobaghdad.edu.iq

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*,
Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9_29

Keywords ARMAX-GARCHX · General error distribution (GED) · Hybrid model · Unemployment rate · Exchange rate Diebold–Mariano (DM) test

1 Introduction and Motivation

Time-series forecasting is an important statistical analysis technique used as a basis for manual and automatic planning in many application domains [13]. The econometric analysis of economic and business time series is a major field of research and application. The last few decades have witnessed an increasing interest in both theoretical and empirical developments in constructing time-series models and in their important application in forecasting [11]. Time-series forecasting is an important area of forecasting in which the observations of the same variable appear as time series: a monthly sequence, daily sequence, hourly sequence, and so on, which are collected and analyzed to develop and build a model describing the fundamental relationship [10]. Forecasting rules can play an important role in many areas such as business, industry, and intergovernmental organizations.

Apart from several factors such as Gross Domestic Product (GDP) and inflation, the exchange rate and unemployment are some of the major factors that are important in economic growth advancement. For modeling these two factors, the time-series literature provided one of the major and commonly used approaches for analysis of the **bivariate time-series** data, which is the Autoregressive Moving Average with exogenous variable (**ARMAX**) model. Compared with the ARX, the ARMAX model is probably the second most popular linear model after the **ARX** and more flexible class because it possesses an extended noise model and due to its statistical properties [19]. **ARMAX** is a flexible class of models including mixed pure autoregressive **(AR)** and moving average **(MA)** models with additional external input called exogenous variable. But one of the main constraints of ARMAX models is the linearity structure of the models. This assumption of linearity restricts the application of the ARMAX model to real time-series data. There are many studies that have discussed the application of this model such as [1, 25].

Linear models have no possibility to describe any volatility in the actual conditional variance in the real time-series data or in the residuals of ARMAX linear model. To overcome this problem, Engle [8] proposes the Autoregressive Conditional Heteroskedasticity (**ARCH**) statistical model, for the purpose of capturing the volatility in time-series data that describes the variance of the current error term or innovation as a function of the actual sizes of the previous time periods error terms. The ARCH model is appropriate when the error variance in a time series follows an autoregressive (**AR**) model and has a disadvantage with a large number of parameters required in building the forecast model. Therefore, Bollerslev [3] proposes a more parsimonious technique that is the Generalized ARCH (**GARCH**) model appropriate when the mixed autoregressive moving average (**ARMA**) model is assumed for the error variance. There are many empirical studies which confirm that the nonlinear models have a good performance for long-term forecasting whereas the linear models are appropriate for short-term forecasting, as well as the real time-series data often composed of linear and nonlinearities compound [17, 18]. So there is a necessity of hybridization of the linear and nonlinear models in one hybrid model in order to obtain a more efficient forecast. And for the process of building a hybrid model, we use the nonlinear GARCHX model whose conditional variance follows a General error distribution (GED), i.e., (GARCHX-GED), which the GED assumes for capturing heavy-tailed properties in residuals of linear ARMAX model. A number of hybrid models are tested and the optimum model is chosen based on model selection criteria such as Akaike information criteria (AIC) and Bayesian information criteria (BIC). Nevertheless, to deal with the problem of no individual model guaranteed to give the ideal forecast, different forecasting models can be mixed together in one hybrid model to increase the chance of capturing different structures and then yield a more efficient forecasting model and get the optimum structure of the final hybridized forecast. Many researchers [15, 21, 27] and others have studied the combined ARMAX model with GARCH and showed the improvement of the hybrid model in forecasting accuracy. In the present study, the linear ARMAX model and hybrid ARMAX-GARCHX-GED model according to [26] methodology have been applied to the real dataset. Subsequently, these two competing models are evaluated for the forecasting accuracy first by using classical statistical evaluation measures, namely, Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). Though these classical evaluation criteria are simple and easily understandable, they have some limitations. On the one hand, they do not provide a statistical test of the significant difference between the two forecasting models. On the other hand, forecasting value given by competing forecasting models can be interfered by stochastic differences. So, for these reasons these measures of forecast accuracy are useful only for comparing different models [5–7]. In this paper, the second forecast criteria based on a modern evaluation criterion which has been advocated by Diebold and Mariano [7] is introduced to evaluate forecasting performance between ARMAX and ARMAX-GARCH-GED competing models, and then choosing the optimal unemployment rate forecasting model. MATLAB, R, and EViews software packages have been used for the data analysis.

2 Materials and Methods

In this section, we will describe forecasting time-series models, Statistical test, Zhang hybrid methodology, and Performance Evaluation.

2.1 *The Hybrid ARMAX-GARCH-GED Forecasting Model*

The hybrid ARMAX-GARCH model combines two time-series models represented as the following.

2.1.1 Conditional Mean Model

In modeling the **mean equation** of the hybrid model, one of the most important linear models for modeling bivariate time series is the single-input–single-output Autoregressive Moving Average with exogenous variable (**SISO-ARMAX**). In the **SISO-ARMAX** model structure specification, the endogenous variable is modeled as difference equation in Eqs. (1 and 2) [23, 24]:

$$\mathbf{y_t} = \boldsymbol{\mu} + \sum_{\mathbf{i=1}}^{\mathbf{n_p}} \varphi_{\mathbf{i}} \mathbf{y_{t-i}} - \sum_{\mathbf{j=1}}^{\mathbf{n_q}} \boldsymbol{\theta}_{\mathbf{j}} \boldsymbol{\varepsilon}_{\mathbf{t-j}} + \sum_{\boldsymbol{\kappa}=1}^{\mathbf{n_b}} \boldsymbol{\phi}_{\boldsymbol{\kappa}} \mathbf{x}_{\mathbf{t-n_k}} + \boldsymbol{\varepsilon}_{\mathbf{t}}, \tag{1}$$

Or in the compact form:

$$\boldsymbol{\Phi}_{\mathbf{np}}(\mathbf{L})\mathbf{y_t} = \boldsymbol{\phi}_{\mathbf{n_b}}(\mathbf{L})\mathbf{x}_{t-nk} + \boldsymbol{\Theta}_{\mathbf{nq}}(\mathbf{L})\boldsymbol{\varepsilon}_{\mathbf{t}}, \tag{2}$$

where

$\mathbf{y_t}$: endogenous variable (Model output at time t).

$\mathbf{x}_{t-n_k}$: exogenous variable, previous and delayed inputs on which the current output depends.

The parameters $\mathbf{n_p}$, $\mathbf{n_b}$, and $\mathbf{n_q}$ are the orders of the ARMAX model (the order of autoregressive, exogenous variable and moving average, respectively), and $\boldsymbol{n_k}$ is the delay time. $\boldsymbol{\Phi}_{\mathbf{n_p}}(\mathbf{L})$, $\boldsymbol{\Theta}_{\mathbf{n_q}}(\mathbf{L})$, $\boldsymbol{\phi}_{\mathbf{n_d}}(\mathbf{L})$ are the polynomial of lag operator L of order $\mathbf{n_p}$, $\mathbf{n_q}$, and $\mathbf{n_d}$, respectively, with root outside the unit circle such that

$$\boldsymbol{\Phi}_{\mathbf{n_p}}(\mathbf{L}) = 1 + \varphi_1\mathbf{L} + \varphi_2\mathbf{L}^2 + \cdots + \varphi_{\mathbf{n_p}}\mathbf{L}^{\mathbf{n_p}} = 1 + \sum\nolimits_{\mathbf{i=1}}^{\mathbf{n_p}} \varphi_{\mathbf{i}}\mathbf{L}^{\mathbf{i}}, \tag{2a}$$

$$\boldsymbol{\Theta}_{\mathbf{n_q}}(\mathbf{L}) = 1 - \boldsymbol{\theta}_1\mathbf{L}^1 - \boldsymbol{\theta}_2\mathbf{L}^2 - \cdots - \boldsymbol{\theta}_{\mathbf{n_q}}\mathbf{L}^{\mathbf{n_q}} = 1 - \sum\nolimits_{\mathbf{j=1}}^{\mathbf{n_q}} \boldsymbol{\theta}_{\mathbf{j}}\mathbf{L}^{\mathbf{j}}, \tag{2b}$$

$$\boldsymbol{\phi}_{\mathbf{n_d}}(\mathbf{L}) = \boldsymbol{\phi}_1\mathbf{L}^{-1} + \boldsymbol{\phi}_2\mathbf{L}^{-2} + \cdots + \boldsymbol{\phi}_{\mathbf{n_d}}\mathbf{L}^{\mathbf{n_d}} = \sum\nolimits_{\mathbf{k=1}}^{\mathbf{n_d}} \boldsymbol{\phi}_{\mathbf{k}}\mathbf{L}^{\mathbf{k}}, \tag{2c}$$

The parameters $\varphi_{\mathbf{i}}$, $\boldsymbol{\theta}_{\mathbf{j}}$ and $\boldsymbol{\phi}_{\boldsymbol{\kappa}}$ are estimated by Recursive Least Square Method with Exponential Forgetting Factor (RLS-EF). The detail computational procedure for this method can be found in [23].

2.1.2 Conditional Variance Model

In Eq. (1), the conditional variance of ARMAX residuals ($\boldsymbol{\varepsilon}_{\mathbf{t}}$) is analyzed by using the **GARCH (r, s)** with **exogenous variable**, and the **GARCH (r, s)** model is represented in the following equations [3, 12]

$$\boldsymbol{\varepsilon}_{\mathbf{t}} = \boldsymbol{h_t \eta_t}, \boldsymbol{\eta}_{\mathbf{t}} \sim \mathbf{i.i.dN}(0, 1), \tag{3}$$

where

$\boldsymbol{\varepsilon}_{\mathbf{t}}$: represents GARCH innovations.

$\boldsymbol{\eta}_t$: The random variable $\boldsymbol{\eta}_t$ is an innovation term which is typically assumed to be independent and identically distributed (i.i.d) with mean zero and unit variance.

$\boldsymbol{h}_t$: Conditional variance $\boldsymbol{h}_t$ is modeling as GARCH (r, s), as in Eq. (4):

$$\mathbf{h_t} = \propto_{\mathbf{o}} + \sum\nolimits_{\mathbf{i}=1}^{\mathbf{r}} \propto_{\mathbf{i}} \boldsymbol{\varepsilon}_{\mathbf{t-i}}^2 + \sum\nolimits_{\mathbf{j}=1}^{\mathbf{s}} \boldsymbol{\vartheta}_{\mathbf{j}} \mathbf{h}_{\mathbf{t}-\mathbf{j}}, \tag{4}$$

where r, s are the orders of GARCH model, and parameters $\propto_{\mathbf{i}}$ and $\boldsymbol{\vartheta}_{\mathbf{j}}$ estimate for the ARCH and GARCH effects of **i*th*** and **j*th*** orders, respectively. The parameters ($\propto_{\mathbf{o}}$, $\propto_1$, …, $\propto_r$, $\boldsymbol{\vartheta}_1$, …, $\boldsymbol{\vartheta}_{\mathbf{s}}$) are estimated by Maximum Likelihood Estimation according to the BHHH optimization algorithm and restricted such that ($\mathbf{h_t} > 0$) for all $\boldsymbol{t}$, which is ensured when:

$$\propto_{\mathbf{o}} > 0, \propto_{\mathbf{i}} \geq 0, \textbf{for } \mathbf{i} = 1, 2, 3, \ldots \mathbf{r}, \boldsymbol{\vartheta}_{\mathbf{j}} \geq 0, \textbf{for } \mathbf{j} = 1, 2, 3, \ldots, \mathbf{s}$$

From Eq. (4) the simple **GARCH (1, 1)** model is the most popular for modeling volatility. We write this model as

$$\begin{aligned} \boldsymbol{\varepsilon}_{\mathbf{t}} &= \boldsymbol{h}_t \boldsymbol{\eta}_t, \boldsymbol{\eta}_{\mathbf{t}} \sim \mathbf{i.i.dN}\,(0, 1) \\ \mathbf{h_t} &= \propto_{\mathbf{o}} + \propto_1 \boldsymbol{\varepsilon}_{\mathbf{t}-1}^2 + \boldsymbol{\vartheta}_1 \mathbf{h}_{\mathbf{t}-1}, \end{aligned} \tag{5}$$

The conditional variance in Eq. (5) is modeled by the past shock $\boldsymbol{\varepsilon}_{\mathbf{t}-1}^2$ and its own lagged value $\mathbf{h}_{\mathbf{t}-1}$. For $\propto_{\mathbf{o}} \geq 0$, $\propto_1 > 0$, $\boldsymbol{\vartheta}_1 > 0$, and $\propto_1 + \boldsymbol{\vartheta}_1 < 1$ [9].

Expression (5) of GARCH (1, 1) model is typically extended to be more complex and involves exogenous variable $\mathbf{x}_t$ in volatility equation. The volatility Eq. (5) can be extended and rewritten for the GARCHX (1, 1, 1) model and represented as in Eq. (6) [14, 22]:

$$\mathbf{h_t} = \propto_{\mathbf{o}} + \propto_{\mathbf{1}} \boldsymbol{\varepsilon}^{\mathbf{2}}{}_{\mathbf{t-1}} + \boldsymbol{\vartheta}_{\mathbf{1}} \mathbf{h}_{\mathbf{t-1}} + \mathrm{w}_{\mathbf{1}} \boldsymbol{x}^{\mathbf{2}}{}_{\mathbf{t-1}} \tag{6}$$

for exogenous variable $\boldsymbol{x}_t$ which is squared to ensure that ($\mathbf{h_t} > 0$). The including of the additional exogenous variable $\mathbf{x}_t$ helps to explain the volatilities of exchange rate series and tend to lead to improve in-sample fit and out-of-sample forecasting perform.

2.1.3 GARCHX Model Specified Under Heavy-Tailed Distribution

The **mixture** of **mean Eq.** (1) and **volatility Eq.** (6) will give us a **hybrid ARMAX-GARCHX** model whose conditional variance follows a **Gaussian distribution**; Normal distribution assumption was found to be not useful in capturing the heavy-tailed behavior of the series. Therefore, Nelson [20] proposed GED distribution to capture the heavy-tailed (leptokurtic) behavior of the process. Thus, to obtain more forecast efficiency of hybrid **ARMAX-GARCH** model, the hybrid model has been applied based on Generalized Error Distribution (GED) proposed by Nelson [20] represented in the following equations [2, 10].

From Eq. (4) the error term can be rewritten as $\boldsymbol{\eta_t} = \frac{\boldsymbol{\varepsilon_t}}{\sqrt{\boldsymbol{h_t}}} \sim \boldsymbol{N}(0, 1)$, when applied Generalized Error Distribution to the GARCH model, the corresponding density functions of $\boldsymbol{\varepsilon_t}$ are described below:

$$\boldsymbol{f}(\boldsymbol{\varepsilon_t}) = \frac{\boldsymbol{vexp}\left[-\frac{1}{2}|z/\boldsymbol{\lambda}|^{\boldsymbol{v}}\right]}{\boldsymbol{\lambda}\left[2^{(1+\frac{1}{v})}\boldsymbol{\Gamma}\left(\frac{1}{\boldsymbol{v}}\right)\right]}, (-\infty < z < \infty),$$

where $\boldsymbol{\lambda}$ is defined as

$$\boldsymbol{\lambda} = \left[2^{(-2/\boldsymbol{v})}\boldsymbol{\Gamma}\left(\frac{1}{\boldsymbol{v}}\right)\boldsymbol{\Gamma}\left(\frac{3}{\boldsymbol{v}}\right)\right]^{1/2}$$

$\boldsymbol{v}$: Tail thickness parameter $(0 < \boldsymbol{v} \leq \infty)$, $\boldsymbol{\Gamma}$: Gamma function.

2.2 *The Zhang Hybrid Methodology*

The hybrid method supposes that the time-series process decomposes as a mixture of both linear and nonlinear components. This follows the Zhang [26] hybrid approach. Consequently, the relationship between linear and nonlinear components can be expressed as follows:

$$\mathbf{y_t} = \underbrace{\mathbf{L_t}}_{\mathbf{ARMAX}} + \underbrace{\mathbf{N_t}}_{\mathbf{GARCHX}} \rightarrow \textbf{for ARMAX}-\textbf{GARCHX}, \tag{7}$$

where $\mathbf{L_t}$ and $\mathbf{N_t}$ represent the linear and nonlinear components present in the time-series data, and these two components are to be estimated for ARMAX and GARCHX models, respectively. This hybrid approach of combining forecasting values for the hybrid (**ARMAX-GARCHX**) model has the following steps [17, 26]

1. First, fit a linear ARMAX time-series model for the data.
2. In the second step, the residuals time series are extracted from the fitted ARMAX linear model. The residuals will contain only the nonlinear components. Let $\boldsymbol{e_t}$ denotes the residual at the time t from the linear model, then

$$\boldsymbol{e_t} = \mathbf{y_t} - \hat{\boldsymbol{L}}_t, \tag{8}$$

where $\hat{\boldsymbol{L}}_t$: The optimal forecast of the mean Eq. (1) for the ARMAX model.

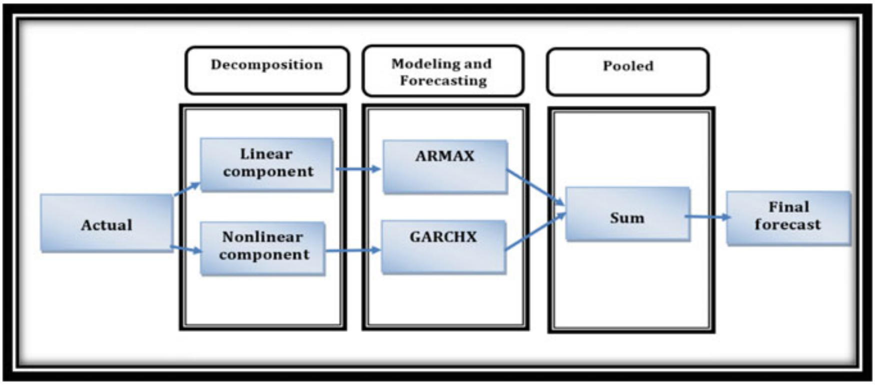


Fig. 1 Schematic representation of ARMAX-GARCHX hybrid methodology

3. Tests for nonlinearity: Brock Dechert–Scheinkman (**BDS**) test: The presence of nonlinearity pattern in the extracted residuals of the fitted ARMAX model can be tested using BDS test; the BDS test is used to test the null hypothesis that the residuals are specific as linearity pattern against the alternative hypothesis that there exists a nonlinearity pattern in the residuals. The detail computational procedure and test statistic can be found in Brock et al. [4].
4. If residuals confirm the nonlinearity, then the residuals are modeled using nonlinear GARCHX models. Subsequently, obtain the optimal forecast $\hat{\mathbf{N}}_{\mathbf{t}}$ for the residual series using the optimal GARCHX models.
5. In the final step, the linear $\hat{L}_t(\ell)$ and nonlinear $\hat{\mathbf{N}}_{\mathbf{t}}(\ell)$ forecasted components are combined to obtain the pooled forecast values for the hybrid model (**ARMAX-GARCHX**) as in Eq. (12):

$$\hat{\mathbf{y}}_{\mathbf{t}}(\ell) = \hat{\mathbf{L}}_{\mathbf{t}}(\ell)_{\mathbf{ARMAX}} + \hat{\mathbf{N}}_{\mathbf{t}}(\ell)_{\mathbf{GARCHX}}, \tag{9}$$

The hybrid approach for (**ARMAX-GARCHX**) model can be graphically represented as in Fig. 1.

2.3 *Forecasts Evaluation*

The forecasting performance of competing models is evaluated using two different procedures.

2.3.1 Loss Functions Criteria

The common classical forecast accuracy criteria, namely, Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) are presented in the following equations [12, 24]:

- **Mean Square Error (MSE)**:

$$\mathbf{RMSE} = \sqrt{\frac{1}{\mathbf{n}} \sum\nolimits_{\mathbf{t}=1}^{\mathbf{n}} \left(y_t - \hat{y}_t\right)^2}, \tag{10}$$

- **Mean Absolute Error (MAE)**:

$$\mathbf{MAE} = \frac{1}{\mathbf{n}} \sum\nolimits_{\mathbf{t}=1}^{\mathbf{n}} |\mathbf{e_t}| = \frac{1}{\mathbf{n}} \sum\nolimits_{\mathbf{t}=1}^{\mathbf{n}} \left|y_t - \hat{y}_t\right|, \tag{11}$$

- **Mean Absolute Percentage Error (MAPE)**:

$$\mathbf{MAPE} = \frac{1}{\mathbf{n}} \sum\nolimits_{\mathbf{t}=1}^{\mathbf{n}} \left|\frac{\mathbf{e_t}}{y_t}\right| * 100\%, \tag{12}$$

2.3.2 Diebold–Mariano Test

The DM test was originally proposed by Diebold and Mariano as a test of forecast accuracy between two sets of forecasts using the MSE. To determine whether one forecasting model (say, the first model, model A (ARMAX-GARCHX-GED)) predicts more accurately than another (say, the second model, model B (ARMAX)), we may test the null hypothesis: no difference in the accuracy (equal predictive ability) of the two competing models is given as [5–7, 16]:

$$\mathbf{H}_0 : \mathbf{E}\left(\mathbf{d}_{\mathbf{A},\mathbf{t}}\right) = \mathbf{E}\left(\mathbf{d}_{\mathbf{B},\mathbf{t}}\right)$$

Vs The alternative hypothesis that one is better than the other is given as

$$\mathbf{H}_1 : \mathbf{E}\left(\mathbf{d}_{\mathbf{A},\mathbf{t}}\right) \neq \mathbf{E}\left(\mathbf{d}_{\mathbf{B},\mathbf{t}}\right), \tag{13}$$

the DM test is based on the loss differentials $\mathbf{d_t}$:

$$\mathbf{d_t} = \mathbf{e}_{\mathbf{A},\mathbf{t}}^2 - \mathbf{e}_{\mathbf{B},\mathbf{t}}^2, \text{ or } \mathbf{d_t} = \left|\mathbf{e}_{\mathbf{A},\mathbf{t}}\right| - \left|\mathbf{e}_{\mathbf{B},\mathbf{t}}\right|$$

and the DM test statistics is represented in the following equations:

$$\mathbf{DM} = \frac{1}{\sqrt{\left[\gamma_0 + 2\sum_{k=1}^{h-1}\gamma_k\right]\Big/\boldsymbol{n}}} \cong \mathbf{N}(0,1), \tag{14}$$

the variable $\boldsymbol{\gamma_k}$ denotes the k-th auto-covariance of $\mathbf{d_t}$ which is given as

$$\hat{\gamma}_{\mathbf{k}} = \frac{1}{\mathrm{n}}\sum_{\mathrm{t=k+1}}^{\mathrm{n}}\left(\mathbf{d_t} - \bar{\mathbf{d}}_{\mathbf{t}}\right)\left(\mathbf{d_t} - \bar{\mathbf{d}}_{\mathbf{t-k}}\right)$$

And $\bar{\mathbf{d}}_{\mathbf{t}}$ the sample mean loss differential defined as

$$\bar{\mathbf{d}}_{\mathbf{t}} = \frac{1}{\mathbf{n}}\sum_{\mathbf{t=1}}^{\mathbf{n}}\mathbf{d_t} \quad t = 1, 2 \ldots, \boldsymbol{n}$$

and n, h step forecasts are computed from models A and B. We can reject the null hypothesis if p-value is less than 0.05, and since the DM statistics converge to a normal distribution, also we can reject the null hypothesis at the 5% level if (|DM| > 1.96).

3 Case Study (Data Sets in the Experiment)

The data for this study represent a bivariate time-series data of the unemployment rate and the exchange rate in the USA, as monthly measurements for the period from **January 2000 to December 2017** as a training dataset for parameter estimation and the last 18 months (observations) (**from January 2018 to June 2019**) considered as a testing set is used for obtaining the out-of-sample forecast and also for validation using classical evaluation criteria and the modern DM test statistics collected (Source; fred.stlouisfed.org). Figure 2a and b shows the plot of bivariate time series of the unemployment rate and the exchange rate.

3.1 Fitting the Hybrid ARMAX−GARCHX−GED *Model*

The suitable hybrid **ARMAX−GARCHX−GED** model was selected based on minimum AIC and BIC criteria, and it is found that **ARMAX (2, 1, 1, 0)-GARCHX (1, 1, 1)-GED** is the best model for modeling the dynamic relation between the unemployment rate and the exchange rate. The parameter estimates of fitted

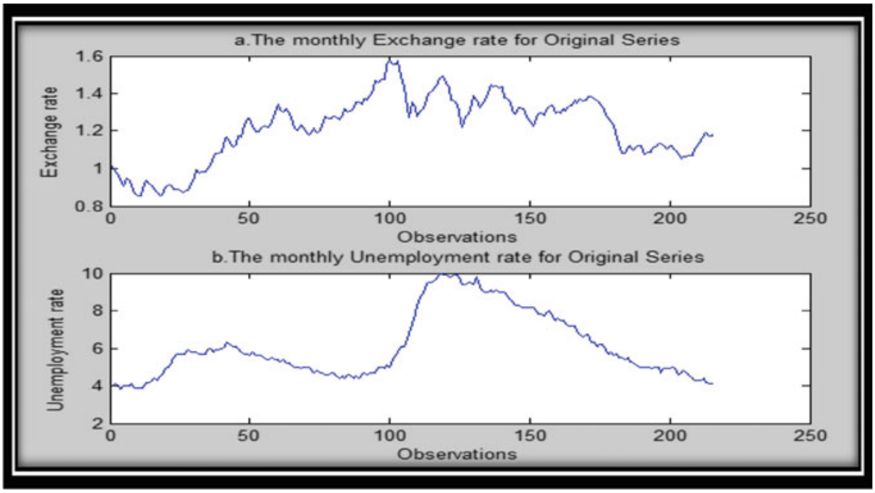


Fig. 2 The monthly **a Unemployment rate** and **b** The **exchange rate** for original series from (January 2000 to December 2017)

Table 1 Significance of ARMAX (2, 1, 1, 0) model parameters using RLS-EF method

Model		Par.	Values	Standard error	t-value	p-value
Mean equation ARMAX (2,1,1,0)	AR(1)	$\hat{\varphi}_1$	−0.5391	0.00793	67.9823	0.00001
	AR(2)	$\hat{\varphi}_2$	−0.4329	0.00791	54.7282	0.00001
	MA(1)	$\hat{\theta}_1$	−0.1341	0.01030	13.0194	0.00001
	x(1)	$\hat{\phi}_1$	0.1182	0.00945	12.5079	0.0001

ARMAX(2, 1, 1, 0)[1] model using **RLS-EF** method are furnished in Table 1 along with their Standard error, t-value, significance level (p-value).

$\hat{\phi}_1$ denotes the parameter of exogenous variable. And according to the ($\mathbf{p-value} < 0.05$), all parameters are significant and have an effect in the model. The ARMAX (2, 1, 1, 0) model can be written from above table as

$$\mathbf{y_t} = 0.5391\mathbf{y_{t-1}} + 0.4329\mathbf{y_{t-2}} + 0.1341\boldsymbol{\varepsilon}_{\mathbf{t}-1} + 0.1182\mathbf{x_t} + \boldsymbol{\varepsilon}_\mathbf{t}\ \mathbf{meanequation}$$

[1] ARMAX model orders were selected based on minimum AIC and BIC criteria and observing the significance of autocorrelation (ACF), partial autocorrelation (PACF), extended autocorrelation function (EACF) and cross-correlation (CCF) functions to identify the model. From Cross-correlation functions (CCF), it is found that the delay time equals to zero. Results implemented using MATLAB (2018a).

Table 2 BDS test for nonlinearities of ARMAX (2, 1, 1, 0) residuals

$H_0 = \text{linearity in } \hat{\varepsilon}_{\mathbf{t}}$			
Test	Dimension	BDS-statistic	p-value
BDS	$m = 2$	0.181994	0.0000
BDS	$m = 3$	0.305099	0.0000
BDS	$m = 4$	0.386709	0.0000

Table 3 Maximum likelihood estimation for GARCHX (1, 1, 1) models with GED distribution

Models	Par.	Values	Stand.error	t-value	p-value
Variance equation for GARCHX(1, 1, 1) $\mathbf{h_t} = \hat{\propto}_{\mathbf{o}} + \hat{\propto}_{\mathbf{1}}\, \boldsymbol{\varepsilon}_{\mathbf{t-1}}{}^{\mathbf{2}} + \hat{\boldsymbol{\vartheta}}_{\mathbf{1}}\mathbf{h}_{\mathbf{t-1}} + \hat{\omega}_{\mathbf{1}}\, \boldsymbol{x}^{\mathbf{2}}_{\mathbf{t-1}}$	$\hat{\propto}_{\mathbf{o}}$	0.002267	0.001152	1.967975	0.0491
	$\hat{\propto}_1$	0.622778	0.129305	4.816344	0.0000
	$\hat{\vartheta}_1$	0.241299	0.036489	6.613002	0.0000
	$\hat{\omega}_{\mathbf{1}}$	0.002686	0.001226	2.190962	0.0285
	$\hat{\boldsymbol{v}}$	4.971380	1.466313	3.390394	0.0007

3.1.1 Testing for Nonlinearity (BDS Test)

BDS test has been used to test the presence of any remaining nonlinearities structure in the residuals **ARMAX**(2, 1, 1, 0) model to test the null hypothesis (Ho: linearity in $\hat{\varepsilon}_{\mathbf{t}}$ exist). And the result of this test is expressed in Table 2.

According to (p-value), the results of the BDS test indicate to exist a nonlinear pattern in the residuals of **ARMAX**(2, 1, 1, 0) model for different dimensions.

3.1.2 Estimation of Hybridization GARCHX (1, 1, 1)-GED Model

For building hybrid models, we use the two-step procedure of estimation. The estimation of the parameters of mean equation for ARMAX (2, 1, 1, 0) model is given in Sect. 3.1 (Table 1), and the estimation of the parameters of variance equation for the **GARCHX (1, 1, 1)-GED** models as mentioned in Eq. (6) using Maximum Likelihood Estimation according to the BHHH optimization algorithm are furnished in Table 3 along with their Standard error, t-value, significance level (p-value)[2]:

All parameters are significant and satisfy the conditions of GARCHX model as discussed in Sect. 2.2.2, $\hat{\boldsymbol{v}}$ parameter of GED. Therefore, the hybrid model (**ARMAX-GARCHX-GED)** can be written as in the following equations:

The **mean equation** follows ARMAX (2, 1, 1, 0) model from the Table 2 as:

$$\mathbf{y}_t = 0.5391\mathbf{y}_{\mathbf{t}-1} + 0.4329\mathbf{y}_{\mathbf{t}-2} + 0.1341\boldsymbol{\varepsilon}_{\mathbf{t}-1} + 0.1182\boldsymbol{x}_t + \boldsymbol{\varepsilon}_{\mathbf{t}}$$

[2] Results are implemented using EViews 9.

Table 4 Comparison of forecasting performance

Models	RMSE	MAE	MAPE (%)
ARMAX	0.2039	0.1493	3.8680
ARMAX-GARCHX-GED	*0.1919*	*0.1452*	*3.7681*

$$\boldsymbol{\varepsilon}_{\mathbf{t}} = \boldsymbol{\eta}_{\mathbf{t}}\sqrt{\mathbf{h}_{\mathbf{t}}},\, \boldsymbol{\eta}_{\mathbf{t}} \sim \mathbf{GED}(\hat{\mathbf{v}} = 4.971)$$

The **variance equations $\mathbf{h}_{\mathbf{t}}$** follow **GARCHX (1, 1, 1)-GED** models from Table 3 as:

$$\mathbf{h}_{\mathbf{t}} = 0.002267 + 0.622778\boldsymbol{\varepsilon}^2_{\mathbf{t}-1} + 0.241299\mathbf{h}_{\mathbf{t}-1} + 0.002686\boldsymbol{x}^2_{\mathbf{t}-1}$$

The above equations represent the hybrid models and employ the forecasting of the unemployment rate for the period (**from January 2018 to June 2019**).

4 Evaluation of Forecasting Performance

The prediction abilities of the individual ARMAX model and the hybrid model **ARMAX-GARCHX-GED** for the last 18 months (observations) (**from January 2018 to June 2019**) are compared based on two procedures.

4.1 Loss Function Criteria

Three classical statistical criteria, **RMSE, MAE, and MAPE** mentioned in Sect. 2.3.1 are furnished in Table 4 for ARMAX and ARMAX-GARCHX-GED models.

From Table 2, according to the minimum values of RMSE, MAE, and MAPE, we can conclude that the forecasting results of the hybrid **ARMAX-GARCH-GED** model look more efficient than the individual ARMAX model. And the following graph shows the **actual** and **forecast** values extracted by the hybrid **ARMAX (2, 1, 1, 0)-GARCHX (1, 1, 1)** model for the last 18 observations for the unemployment rate (from January 2018 to June 2019) (Fig. 3).[3]

However, this forecast accuracy measures are reported in Table 4 based on mean square error (MSE), and the MSE is invalid to determine whether the difference is due to chance, and is unable to diagnose significant differences. So, it is necessary to evaluate the forecast accuracy by employing the DM test and then distinguishing the forecasting performance between the competing forecasting models.

[3] Results of Table 4 and graph are implemented using MATLAB (2018a).

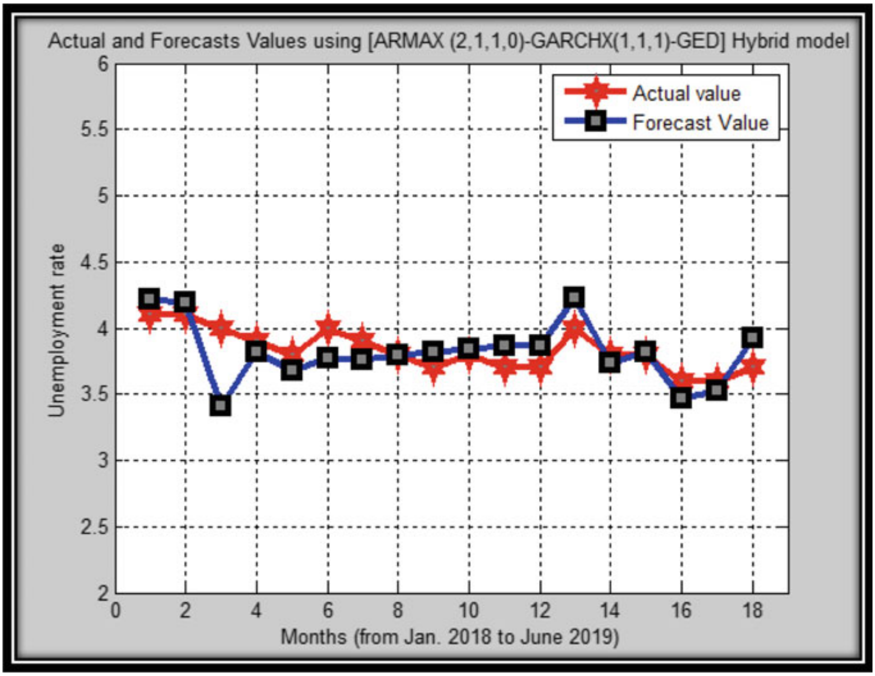


Fig. 3 Actual and forecast values using **ARMAX (2, 1, 1, 0)-GARCHX (1, 1, 1)-GED** for the last 18 observations for the unemployment rate

Table 5 DM test for forecasting performance based on (model A, ARMAX) and (model B, ARMAX-GARCHX-GED)

$\mathbf{H}_0 : \mathbf{E}(\mathbf{d}_{\mathbf{A},\mathbf{t}}) = \mathbf{E}(\mathbf{d}_{\mathbf{B},\mathbf{t}})$ **vs** $\mathbf{H}_1 : \mathbf{E}(\mathbf{d}_{\mathbf{A},\mathbf{t}}) \neq \mathbf{E}(\mathbf{d}_{\mathbf{B},\mathbf{t}})$			
Absolute-error loss function, $|\mathbf{d}_\mathbf{t}|$		Squared-error loss function, $\mathbf{d}_\mathbf{t}^2$	
DM-test	p-value	DM-test	DM-test
4.0398	8.039e−05	4.0398	2.0341

4.2 *Forecasting Evaluation Based on DM Test*

In this section, the forecasting performance of the two competing ARMAX and ARMAX-GARCH-GED models has been compared by employing the DM test. The DM test employs testing the hypothesis given in Eq. (13) by DM test statistics given in Eq. (14), The results of the forecasting comparison of two competing models are furnished in Table 5[4]:

From Table 3, according to the DM test based on the absolute-error loss and squared-error loss, respectively, since the p-value of DM statistic is less than 0.05,

[4] Results implemented using R 3.5.2.

the null hypothesis is rejected, i.e., we accepted the alternative hypothesis, that is to say, the observed differences are significant and the forecasting accuracy of the hybrid **ARMAX-GARCHX-GED** model is **more efficient** than that of the **ARMAX** model.

5 Conclusions

The main purpose of this study is to employ the Diebold–Mariano test for performance evaluation between individual and hybrid time-series models for unemployment rate forecasting. A hybrid method decomposes bivariate time series into its linearity and nonlinearity pattern and then modeling each part individually before they are aggregate for getting final forecast using Zhang hybrid methodology. Based on the practical dataset results, one can conclude the following outcomes:

1. The classical linear time-series models such as ARMAX are not always sufficient for modeling bivariate time series that consists of linear and nonlinear structures. And when the assumption of linearity is violated, the hybrid approach which combines linear and nonlinear models performs better as compared to classical time-series models under Heteroscedasticity problem.
2. The residual time series that are extracted from the fitted ARMAX linear model was tested by BDS test which reveals that nonlinearity pattern exists in the residual time series. And then based on this residual series, the nonlinear GARCHX model is build and employed to forecast the volatility and in a hybridization method with ARMAX model.
3. The individual and hybrid models have been applied in forecasting the unemployment rate in the USA. And the comparison of forecasting performance has been checked based on minimum values of classical forecast criteria (RMSE, MAE, and MAPE) as mentioned in Table 4, and it was observed that the hybrid model looks more efficient than the individual ARMAX model.
4. Apart from classical forecast accuracy criteria, this study adopts the [7] known as DM test to further evaluate the statistical significance of the two competing models. The findings from the DM test statistics show that the observed differences of forecasting values between the ARMAX and ARMAX-GARCHX-GED models are significant at the 5% level of significance as mentioned in Table 5, and indicate that the hybrid model has better forecasting efficiency than individual ARMAX model.

Acknowledgements The authors would like to thank **ITISE 2019** Conference organizers to receive this contribution. The authors would like to thank the editors and all the reviewers for their advice and suggestions on improving this paper.

References

1. Aldemir, A., Hapoğlu, H.: Comparison of ARMAX model identification results based on least squares method. IJMTER **02**, 27–35 (2015)
2. Bollerslev, T.: A conditional heteroscedastic time series model for speculative prices and rates of return. Rev. Econ. Stat. **69**, 542–547 (1987)
3. Bollerslev, T.: Generalized autoregressive conditional heteroscedasticity. J. Econometrics **31**, 307–327 (1986)
4. Brock, W.A., Dechert, W., Scheinkman, J., Lebaron, B.: A test for independence based on the correlation dimension. Econ. Rev. **15**, 197–235 (1996)
5. Chen, H., Wan, Q., Wang, Y.: Refined Diebold-Mariano test methods for the evaluation of wind power forecasting models. Energies **7**, 4185–4198 (2014)
6. Diebold, F.X.: Element of Forecasting, 4th edn. Cincinnati, OH, USA, Thomson South-Western (2007)
7. Diebold, F.X., Mariano, R.S.: Comparing predictive accuracy. J. Bus. Econ. Stat. **13**, 253–263 (1995)
8. Engle, R.: Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. Econometrica **50**, 987–1007 (1982)
9. Engle, R.: GARCH 101: the use of ARCH/GARCH models in applied econometrics. J. Econ. Perspective **15**, 157–168 (2001)
10. Feng, L., Shi, Y.: A simulation study on the distributions of disturbances in the GARCH model. Coge. Econ. Finance **5**, 1–19 (2017)
11. Franses, P.H., van Dijk, D.J.C., Opschoor, A.: Time Series Models for Business and Economic Forecasting, 2nd edn. Cambridge University Press (2014)
12. George, E.P.B., Gwilym, M.J., Gregory, C.R., Greta, M.L.: Time Series Analysis Forecasting and Control, 5th edn. Hoboken, New Jersey (2015)
13. Gooijer, J.G.D., Hyndman, R.J.: 25 years of time-series forecasting. Int. J. Forecasting **22**, 443–473 (2006)
14. Han, H., Kristensen, D.: Asymptotic theory for the QMLE in GARCH-X models with stationary and nonstationary covariates. J. Bus. Econ. Stat. **32**, 416–429 (2014)
15. Hickey, E., Loomis, D.G., Mohammadi, H.: Forecasting hourly electricity prices using ARMAX–GARCH models: an application to MISO hubs. Energy Econ. **34**, 307–315 (2012)
16. Hung-Chung, L., Yen-Hsien, L., Ming-Chih, L.: Forecasting China stock markets volatility via GARCH models under skewed-GED distribution. J. Mon. Inv. Ban **7**, 542–547 (2009)
17. Mitra, D., Paul, R.K.: Hybrid time-series models for forecasting agricultural commodity prices. Model Ass. Stat. Appl. **12**, 255–264 (2017)
18. Moeeni, H., Bonakdari, H.: Impact of normalization and input on ARMAX-ANN model performance in suspended sediment load prediction. Water Resour. Manage. **32**, 845–863 (2017)
19. Nelles, O.: Nonlinear System Identification from Classical Approaches to Neural Networks and Fuzzy Models. Springer, New York (2001)
20. Nelson, D.B.: Conditional heteroscedasticity in asset returns: a new approach. Econometrica **59**, 347–370 (1991)
21. Porshnev, A., Valeria, L., Ilya, R.: Could Emotional Markers in Twitter Posts Add Information to the Stock Market ARMAX-GARCH Model. Higher School of Economics Research Paper No. WP BRP 54/FE (2016)
22. Rachev, S., Mittnik, S., Fabozzi, F., Focardi, S., Jasic, T.: Financial Econometrics: From Basics to Advanced Modeling Techniques, Inc. New York (2007)
23. Soderstrom, T., Stoica, P.: System Identification. Prentice-Hall International, Hemel Hempstead, U.K. (2001)
24. Tsay, R.S.: An Introduction to Analysis of Financial Data with R. Hoboken (2013)
25. Yiu, J., Wang, S.: Multiple ARMAX modeling scheme for forecasting air conditioning system performance. Energy Convers. Manag. **48**, 2276–2285 (2007)

26. Zhang, G.P.: Time series forecasting using a hybrid ARIMA and neural network model. Neurocomputing. Comput. **50**, 159–175 (2003)
27. Zhao, J.H., Dong, Z.Y., Zhao, M.L.: A statistical model for flood forecasting. Aus. J. Water Res. **13**, 43–52 (2009)

Author Index

O. Valenzuela et al. (eds.), *Theory and Applications of Time Series Analysis*, Contributions to Statistics, https://doi.org/10.1007/978-3-030-56219-9