A Radial Basis Function Neural Network-Based Coevolutionary Algorithm for Short-Term to Long-Term Time Series Forecasting

E. Parras-Gutierrez, V.M. Rivas and J.J. Merelo

Abstract This work analyzes the behavior and effectiveness of the L-Co-R method using a growing horizon to predict. This algorithm performs a double goal, on the one hand, it builds the architecture of the net with a set of RBFNs, and on the other hand, it sets a group of time lags in order to forecast future values of a time series given. For that, it has been used a set of 20 time series, 6 different methods found in the literature, 4 distinct forecast horizons, and 3 distinct quality measures have been utilized for checking the results. In addition, a statistical study has been done to confirms the good results of the method L-Co-R.

Keywords Time series forecasting · Co-evolutionary algorithms · Neural networks · Significant lags

1 Introduction

Formally defined, a time series is a set of observed values from a variable along time in regular periods (for instance, every day, every month or every year) [25]. Accordingly, the work of forecasting in a time series can be defined as the task of predicting successive values of the variable in time spaced based on past and present observations.

E. Parras-Gutierrez · V.M. Rivas
Department of Computer Sciences, University of Jaen, Campus Las Lagunillas s/n, 23071 Jaen, Spain
e-mail: eparrasg@vrivas.es

V.M. Rivas
e-mail: vrivas@vrivas.es

J.J. Merelo (✉)
Department of Computers, Architecture and Technology, University of Granada, C/ Periodista Daniel Saucedo s/n, 18071 Granada, Spain
e-mail: jmerelo@geneura.ugr.es

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 613,
DOI 10.1007/978-3-319-23392-5_7

For many decades, different approaches have been used for to modelling and forecasting time series. These techniques can be classified into three different areas: descriptive traditional technologies, linear and nonlinear modern models, and soft computing techniques. From all developed method, ARIMA, proposed by Box and Jenkins [3], is possibly the most widely known and used. Nevertheless, it yields simplistic linear models, being unable to find subtle patterns in the time series data.

New methods based on artificial neural networks, such as the one used in this paper, on the other hand, can generate more complex models that are able to grasp those subtle variations.

The L-Co-R method [24], developed inside the field of ANNs, makes jointly use of Radial Basis Function Networks (RBFNs) and EAs to automatically forecast any given time series. Moreover, L-Co-R designs adequate neural networks and selects the time lags that will be used in the prediction, in a coevolutive [7] approach that allows to separate the main problem in two dependent subproblems. The algorithm evolves two subpopulations based on a cooperative scheme in which every individual of a subpopulation collaborates with individuals from the other subpopulation in order to obtain good solutions.

While previously work [24] was focused on 1-step ahead prediction, the main goal of this one is to analyze the effectiveness of the L-Co-R method in the medium and long-term horizon, using the own previously predicted values to perform next predictions. Thus, 6 different methods used in time series forecasting have been selected in order to test the behavior of the method.

The rest of the paper is organized as follows: Sect. 2 introduces some preliminary topics related to this research; Sect. 3 describes the method L-Co-R; and finally Sect. 4 presents the experimentation and the statistical study carried out.

2 Preliminaries

Approaches proposed in time series forecasting can be mainly grouped as linear and nonlinear models. Methods like exponential smoothing methods [34], simple exponential smoothing, Holt's linear methods, some variations of the Holt-Winter's methods, State space models [29], and ARIMA models [3], have stand out from linear methods, used chiefly for modelling time series. Nonlinear models arose because linear models were insufficient in many real applications; between nonlinear methods it can be found regime-switching models, which comprise the wide variety of existing threshold autoregressive models [31] as: self-exciting models [32], smooth transition models [8], and continuous-time models [4], among others. Nevertheless, soft computing approaches were developed in order to save disadvantages of nonlinear models like the lack of robustness in complex model and the difficulty to use [9].

ANNs have also been applied successfully [17] and recognized as an important tool for time-series forecasting. Within ANNs, the utilization of RBFs as activation functions were considered by works as [5] and [27], and applied to time series by Carse and Fogarty [6], and Whitehead and Choate [33]. Later works like the ones by Harpham and Dawson [13] or Du [10] focused on RBFNs for time series forecasting.

On the other hand, an issue that must be taken into account when working with time series is the correct choice of the time lags for representing the series. Takens' theorem [30] establishes that if d, a d-dimensional space where d is the minimum dimension capable of representing such a relationship, is sufficiently large is possible to build a state space using the correct time lags and if this space is correctly rebuilt also guarantees that the dynamics of this space is topologically identical to the dynamics of the real systems state space.

Many methods are based in Takens' theorem (like [19]) but, in general, the approaches found in the literature consider the lags selection as a pre or post-processing or as a part of the learning process [1, 23]. In the L-Co-R method the selection of the time lags is jointly faced along with the design process, thus it employs co-evolution to simultaneously solve these problems.

Cooperative co-evolution [26] has also been used in order to train ANNs to design neural network ensembles [12] and RBFNs [18]. But in addition, cooperative co-evolution is utilized in time series forecasting in works as the one by Xin [20].

3 Description of the Method

This section describes L-Co-R [24], a co-evolutionary algorithm developed to minimize the error obtained for automatically time series forecasting. The algorithm works building at the same time RBFNs and sets of lags that will be used to predict future values. For this task, L-Co-R is able to simultaneously evolve two populations of different individual species, in which any member of each population can cooperate with individuals from the other one in order to generate good solutions, that is, each individual represents itself a possible solution to the subproblem. Therefore, the algorithm is composed of the following two populations:

- Population of RBFNs: it consists of a set of RBFNs which evolves to design a suitable architecture of the network. This population employs real codification so every individual represent a set of neurons (RBFs) that composes the net. During the evolutionary process neurons can grow or decrease since the number of neurons is variable. Each neuron of the net is defined by a center (a vector with the same dimension as the inputs) and a radius. The exact dimension of the input space is given by an individual of the population of lags (the one chosen to evaluate the net).
- Population of lags: it is composed of sets of lags evolves to forecast future values of the time series. The population uses a binary codification scheme thus each gene indicates if that specific lag in the time series will be utilized in the forecasting

Fig. 1 General scheme of method L-Co-R

```
Trend preprocessing
t = 0;
initialize P_lags(t);
initialize P_RBFNs(t);
evaluate individuals in P_lags(t);
evaluate individuals in P_RBFNs(t);
while termination condition not satisfied do
begin
  t = t+1;
  /* Evolve population of lags */
  for i=0 to max_gen_lags do
  begin
    set threshold;
    select P_lags'(t) from P_lags(t);
    apply genetic operators in P_lags'(t);
    /* Evaluate P_lags'(t) */
      choose collaborators from P_RBFNs(t);
      evaluate individuals in P_lags'(t);
    replace individuals P_lags(t) with P_lags'(t);
    if threshold < 0
    begin
      diverge P_lags(t);
    end
  end
  /* Evolve population of RBFNs */
  for i=0 to max_gen_RBFNs do
  begin
    select P_RBFNs'(t) from P_RBFNs(t);
    apply genetic operators in P_RBFNs'(t);
    /* Evaluate P_RBFNs'(t) */
      choose collaborators from P_lags(t);
      evaluate individuals in P_RBFNs'(t);
    replace individuals with P_RBFNs'(t);
  end
end
train models and select the best one
forecast test values with the final model
Trend post-processing
```

process. The length of the chromosome is set at the beginning corresponding with the specific parameter, so that it cannot vary its size during the execution of the algorithm.

As the fundamental objective, L-Co-R forecasts any time series for any horizon and builds appropriate RBFNs designed with suitable sets of lags, reducing any hand made preprocessing step. Figure 1 describes the general scheme of the algorithm L-Co-R.

L-Co-R performs a process to automatically remove the trend of the times series to work with, if necessary. This procedure is divided into two main phases: preprocessing, which takes places at the beginning of the algorithm, and post-processing, at the end of co-evolutionary process. Basically, the algorithm checks if the time series includes trend and, in affirmative case, the trend is removed.

The performance of L-Co-R starts with the creation of the two initial populations, randomly generated for the first generation; then, each individual of the populations is evaluated. The L-Co-R algorithm uses a sequential scheme in which only one population is active, so the two population take turns in evolving. Firstly, the evolutionary process of the population of lags occurs: the individuals which will belong to the subpopulation are selected; following the CHC scheme [11], genetic operators are applied; the collaborator for every individual is chosen from the population of RBFNs; and the individuals are evaluated again and assigned the result as fitness. After that, the best individuals from the subpopulation will replace the worst individuals of the population. During the evolution, the population of lags checks that al least one gene of the chromosome must be set to one because necessarily the net needs one input to obtained the forecasted value.

In the second place, the population of RBFNs starts the evolutionary process. For the first generation, every net in the population has a number of neurons randomly chosen which may not exceed a maximum number previously fixed. As in population of lags, the individuals for the subpopulation are selected, the genetic operators are applied, every individual chooses the collaborator from the population of lags, and then, the individuals are evaluated and the result is assigned as fitness. Fitness function is defined by the inverse of the root mean squared error At the end of the co-evolutionary process, two models formed by a set of lags (from the first population) and a neural network (from the second population) are obtained. On the one hand, a model is composed of the best set of lags and its best collaborator, and on the other hand, the other model is composed of the best net found and its best collaborator. Then, the two models are trained again and the final model chosen is the one that obtains the best fitness. This final model obtains the future values of the time series used for the prediction, and then, forecasted data will be used to find next values.

The collaboration scheme used in L-Co-R is the best collaboration scheme [26]. Thus, every individual in any population chooses the best collaborator from the other population. Only at the beginning of the co-evolutionary process, the collaborator is selected randomly because the population has not been evaluated yet.

The method has a set of specific operators specially developed to work with individuals from every population. The operators used by L-Co-R are the followings:

- Population of RBFNs: tournament selection, x_fix crossover, four operators to mutate randomly chosen (C_random, R_random, Adder, and Deleter) and replacement of the worst individuals by the best ones of the subpopulation.
- Population of lags: elitist selection, HUX crossover operator, replacement of the worst individuals, and diverge (the population is restarted when it is blocked).

4 Experimentation and Statistical Study

The main goal of the experiments is to study the behavior of the algorithm L-Co-R using 4 different and growing horizons, and to compare the results with other 6 methods found in the literature and for 3 different quality measures.

4.1 Experimental Methodology

The experimentation has been carried out using 20 data bases taken from the INE.[1] The data represent observations from different activities and have different nature, size, and characteristics. The data bases have been labeled as: Airline, WmFrancfort, WmLondon, WmMadrid, WmMilan, WmNewYork, WmTokyo, Deceases, SpaMovSpec, Exchange, Gasoline, MortCanc, MortMade, Books, FreeHouPrize, Prisoners, TurIn, TurOut, TUrban, and HouseFin.

To compare the effectiveness of L-Co-R it has used, on the one hand, 6 methods found within the field of time series forecasting: Exponential smoothing method (ETS), Croston, Theta, Random Walk (RW), Mean, and ARIMA [16], and on the other hand, 4 different horizons in order to test the effectiveness when the horizon rises: 1, 6, 12, and 24.

An open question when dealing with time series is the measure to be used in order to calculate the accuracy of the obtained predictions. Mean Absolute Percentage Error (MAPE) [2] was the first measure employed in the M-competition [21] and most textbooks recommended it. Later, many other measures as Geometric Mean Relative Absolute Error, Median Relative Absolute Error, Symmetric Median and Median Absolute Percentage Error (MdAPE), and Symmetric Mean Absolute Percentage Error, among others, were proposed [22]. However, a disadvantage was found in these measures, they were not generally applicable and can be infinite, undefined or can produce misleading results, as Hyndman and Koehler explained in their work [15]. Thus, they proposed Mean Absolute Scaled Error (MASE) that is less sensitive to outliers, less variable on small samples, and more easily interpreted.

In this work, the measures used are MAPE (i.e., $mean(\mid p_t \mid)$), MASE (defined as $mean(\mid q_t \mid)$), and MdAPE (as $median(\mid p_t \mid)$), taking into account that Y_t is the observation at time $t = 1, ..., n$; F_t is the forecast of Y_t; e_t is the forecast error (i.e. $e_t = Y_t - F_t$); $p_t = 100 e_t / Y_t$ is the percentage error, and q_t is determined as:

$$q_t = \frac{e_t}{\frac{1}{n-1} \sum_{i=2}^{n} \mid Y_i - Y_{i-1} \mid}$$

[1]National Statistics Institute (http://www.ine.es/).

Table 1 Results of the methods L-Co-R, ETS, Croston, Theta, RW, Mean, and ARIMA, with respect to horizon 1 and MAPE

Time series	L-Co-R	ETS	ARIMA	CROSTON	THETA	MEAN	RW
Airline	30.380*	274.770	53.636	72.606	141.452	49.965	137.986
WmFrancfort	16.423	17.393	12.136*	40.544	22.745	64.632	25.169
WmLondres	2.860*	5.383	5.212	27.682	10.136	51.852	13.397
WmMadrid	20.101	27.035	12.930*	44.285	25.505	64.326	27.034
WmMilan	30.529*	34.858	34.823	49.750	34.078	59.840	34.823
WmNuevayork	8.259	7.182*	7.536	30.297	14.669	60.812	18.073
WmTokio	4.764*	12.807	12.591	20.556	10.575	42.627	12.591
Deceases	5.981*	8.002	8.040	7.472	7.264	9.663	8.040
SpaMovSpec	53.788*	217.978	88.197	78.648	70.500	63.288	78.935
Exchange	43.044	46.025	45.254	31.121	39.138	24.217*	33.631
Gasoline	1.654*	7.986	9.359	9.587	6.701	18.460	7.974
MortCanc	1.137*	12.979	5.440	32.489	5.889	46.655	6.256
MortMade	3.931*	13.526	31.000	46.362	40.272	42.120	12.800
Books	13.787*	23.588	23.476	23.122	22.360	24.895	22.640
FreeHouPrize	3.424*	8.540	10.227	29.271	5.215	48.746	9.220
Prisoners	8.392	3.103*	3.150	14.220	6.888	35.839	9.474
TurIn	1.357*	7.074	6.377	11.234	7.084	30.424	7.110
TurOut	8.133*	13.261	9.634	12.159	15.238	34.781	13.226
TUrban	2.734*	11.957	9.291	9.067	8.949	16.884	10.116
HouseFin	16.452*	22.296	19.555	21.548	19.947	42.314	22.887

Due to its stochastic nature, the results yielded by L-Co-R have been calculated as the average errors over 30 executions with every time series. For each execution, the following parameters are used in the L-Co-R algorithm: lags population size = 50, lags population generations=5, lags chromosome size = 10 %, RBFNs population size = 50, RBFNs population generations=10, validation rate=0.25, maximum number of neurons of first generation=0.05, tournament size = 3, replacement rate=0.5, crossover rate=0.8, mutation rate=0.2, and total number of generations=20.

Tables 1, 2, 3, 4, 5, and 6, show the results of the L-Co-R and the utilized methods to compare (ETS, Croston, Theta, RW, Mean, and ARIMA), for measures MAPE, MASE, and MdAPE, for horizons 1, and 6, respectively. Due to space limitations, this paper only shows results of the horizons 1 and 6, the results of the rest horizons, 12 and 24, can be accessed at https://goo.gl/frHK7z.

Table 2 Results of the methods L-Co-R, ETS, Croston, Theta, RW, Mean, and ARIMA, with respect to horizon 1 and MASE

Time series	L-Co-R	ETS	ARIMA	CROSTON	THETA	MEAN	RW
Airline	1.913	12.707	1.441*	2.738	5.853	2.045	5.664
WmFrancfort	3.578*	3.608	7.988	7.984	4.673	12.341	5.159
WmLondres	1.648	1.603*	3.484	8.410	3.099	15.566	4.119
WmMadrid	4.442*	5.686	8.625	9.126	5.362	13.050	5.685
WmMilan	5.967*	6.684	19.327	9.263	6.534	10.986	6.678
WmNuevayork	2.667	1.837*	6.228	7.982	3.942	15.620	4.879
WmTokio	2.791	2.443	1.628*	3.935	2.129	8.364	2.402
Deceases	1.059	1.059	1.144	0.952*	0.955	1.274	1.064
SpaMovSpec	1.027	2.027	1.933	1.009	1.023	0.997*	1.010
Exchange	41.181	44.039	70.734	30.448	37.807	23.911*	32.825
Gasoline	1.198*	1.543	1.698	1.864	1.274	3.533	1.541
MortCanc	0.646	1.618	0.277*	4.098	0.725	5.917	0.796
MortMade	1.314	1.303*	1.712	4.500	3.869	4.068	1.315
Books	0.762	0.965	1.147	0.936	0.894	1.040	0.759*
FreeHouPrize	3.339*	5.642	6.805	19.468	3.487	32.371	6.183
Prisoners	14.482	5.485	4.031*	23.979	11.934	58.935	16.305
TurIn	1.903	1.902	1.950	3.151	1.824*	8.328	1.916
TurOut	2.005	2.000	2.241	2.088	2.239	5.826	1.996*
TUrban	0.886	0.978	0.897	0.772	0.744*	1.576	0.887
HouseFin	1.319	1.283	1.502	1.234	1.095*	2.426	1.322

As mentioned before, every result indicated in the tables represent the average of 30 executions for each time series. Best result per database is marked with character '*'. Considering every horizon tested:

- Horizon 1: the L-Co-R algorithm obtains the best results in most of the time series. With respect to MAPE, the L-Co-R algorithm obtains the best results in 15 of 20 time series used, as can be seen in Table 1. Regarding MASE, L-Co-R stands out yielding the best results for 5 time series as can be observed in Table 2. And concerning MdAPE, L-Co-R acquires better results than the other methods in 12 of 20 time series, as Table 3 shows.
- Horizon 6: the L-Co-R has better results than all the other methods using MAPE and MdAPE, as can be seen in Tables 4 and 6, and the best results in 15 o the 20 time series for MASE, as can be observed in Table 5.
- Horizon 12: the L-Co-R yields the best results in 19, 17, and 18 of the 19 time series (MortCanc has not enough values to use with this horizon) respecting MAPE, MASE, and MdAPE, respectively.

Table 3 Results of the methods L-Co-R, ETS, Croston, Theta, RW, Mean, and ARIMA, with respect to horizon 1 and MdAPE

Time series	L-Co-R	ETS	ARIMA	CROSTON	THETA	MEAN	RW
Airline	15.057*	233.934	15.212	54.657	119.754	31.012	118.090
WmFrancfort	14.610	14.603	11.026*	39.259	19.960	63.868	22.750
WmLondres	3.498*	5.430	5.099	30.550	10.474	53.761	15.722
WmMadrid	22.718	28.116	11.446*	45.817	26.787	65.307	28.116
WmMilan	30.476*	34.685	34.643	50.040	33.872	60.072	34.643
WmNuevayork	9.114	4.598*	5.712	35.253	16.505	63.598	23.137
WmTokio	5.517*	9.864	9.556	18.782	9.075	40.967	9.556
Deceases	4.267*	5.464	5.458	6.121	4.440	7.144	5.458
SpaMovSpec	17.669*	107.283	54.033	51.653	53.104	54.045	51.568
Exchange	44.368	46.597	45.961	34.121	38.832	27.517	36.521
Gasoline	1.792*	7.587	8.923	9.045	6.429	18.825	7.563
MortCanc	11.25	9.694	5.116	30.568	4.047*	44.528	5.339
MortMade	3.459*	12.111	28.374	45.704	41.989	41.482	15.629
Books	4.868*	18.111	18.093	17.230	16.566	20.509	11.567
FreeHouPrize	1.803*	5.222	6.572	29.683	5.201	49.044	9.748
Prisoners	6.766	1.512*	1.621	12.651	5.287	34.665	7.817
TurIn	2.945*	6.627	4.605	11.696	4.779	31.502	6.669
TurOut	5.289*	11.331	7.689	11.518	10.873	36.500	11.392
TUrban	5.290	8.262	6.374	6.822	4.922*	17.828	8.900
HouseFin	18.286	22.623	17.297*	21.279	18.845	43.533	23.684

- Horizon 24: the L-Co-R algorithm obtains better results than the other methods in 17, 16, and 16 of the 17 time series (MortCanc, MortMade, and FreeHouPrize have not enough values to use with this horizon) with regard to MAPE, MASE, and MdAPE, respectively.

Thus, the L-Co-R algorithm is able to achieve a more accurate forecast in the most time series for any of the horizons and quality measures considered.

4.2 Analysis of the Results

To analyze in more detail the results and check whether the observed differences are significant, two main steps are performed: firstly, identifying whether exist differences in general between the methods used in the comparison; and secondly,

Table 4 Results of the methods L-Co-R, ETS, Croston, Theta, RW, Mean, and ARIMA, with respect to horizon 6 and MAPE

Time series	L-Co-R	ETS	ARIMA	CROSTON	THETA	MEAN	RW
Airline	28.740*	277.892	48.025	63.178	128.199	44.240	123.817
WmFrancfort	0.531*	19.056	13.004	43.264	25.102	66.250	27.844
WmLondres	0.113*	5.281	5.074	29.310	10.699	52.935	14.427
WmMadrid	0.312*	29.678	13.565	46.994	27.928	66.061	29.678
WmMilan	1.203*	38.440	38.403	52.914	37.562	62.369	38.403
WmNuevayork	0.140*	7.553	7.961	32.490	16.318	62.045	20.251
WmTokio	0.232*	13.255	13.052	20.777	10.908	42.825	13.052
Deceases	0.508*	8.266	8.309	7.385	7.440	10.085	8.309
SpaMovSpec	24.791*	235.399	93.095	82.501	72.432	64.599	82.821
Exchange	0.320*	46.431	33.296	30.949	39.226	24.028	33.465
Gasoline	0.205*	7.985	9.439	9.709	6.656	18.833	7.972
MortCanc	0.135*	12.562	5.963	36.563	5.829	51.164	6.334
MortMade	0.008*	15.078	34.276	55.378	49.472	50.875	12.375
Books	5.831*	23.590	21.059	23.026	22.118	25.159	21.274
FreeHouPrize	1.863*	12.678	15.393	30.282	5.416	49.478	10.517
Prisoners	0.204*	3.357	3.423	15.034	7.516	36.448	10.333
TurIn	0.042*	7.110	6.758	11.858	7.076	30.956	7.170
TurOut	0.603*	39.240	10.230	12.386	14.984	35.319	12.836
TUrban	2.052*	11.764	8.811	8.591	8.408	17.084	9.832
HouseFin	6.729*	21.571	18.953	20.797	19.092	42.674	22.177

determining if the best method is significant better than the rest of the methods. To do this, first of all it has to be decided if is possible to use parametric o non-parametric statistical techniques. An adequate use of parametric statistical techniques reaching three necessary conditions: independency, normality and homoscedasticity [28].

Owing to the former conditions are not fulfilled, the Friedman and Iman-Davenport non-parametric tests have been used. Tables with results of these tests are available at https://goo.gl/frHK7z. They show, from left to right, the Friedman and Iman-Davenport values (χ^2 and F_F, respectively), the corresponding critical values for each distribution by using a level of significance $\alpha = 0.05$, and the *p-value* obtained for the measures utilized. Finally, the critical values of Friedman and Iman-Davenport are smaller than the statistic, it means that there are significant differences among the methods in all cases.

Table 5 Results of the methods L-Co-R, ETS, Croston, Theta, RW, Mean, and ARIMA, with respect to horizon 6 and MASE

Time series	L-Co-R	ETS	ARIMA	CROSTON	THETA	MEAN	RW
Airline	1.595	12.290	1.585*	2.278	5.133	1.772	4.921
WmFrancfort	1.247*	3.946	2.360	8.548	5.133	12.831	5.675
WmLondres	1.317*	1.637	1.570	9.141	3.369	16.408	4.553
WmMadrid	1.302*	6.827	3.025	10.680	6.427	14.922	6.827
WmMilan	1.181*	7.915	7.908	10.710	7.735	12.537	7.908
WmNuevayork	1.235*	1.884	1.968	8.317	4.231	15.633	5.265
WmTokio	1.531*	2.459	2.423	3.862	2.150	8.182	2.423
Deceases	0.956*	1.113	1.119	0.963	0.997	1.348	1.119
SpaMovSpec	0.958	2.114	1.037	0.983	0.966	0.939*	0.984
Exchange	1.147*	44.047	32.240	30.039	37.574	23.546	32.399
Gasoline	0.051*	1.567	1.860	1.913	1.286	3.647	1.565
MortCanc	0.918	1.077	0.527	3.202	0.483*	4.497	0.533
MortMade	1.077*	1.689	3.876	6.335	5.641	5.810	1.370
Books	1.020	0.979	0.838	0.948	0.900	V1.062	0.730*
FreeHouPrize	1.214*	8.940	10.874	21.782	3.917	35.550	7.606
Prisoners	0.484*	5.684	5.795	24.350	12.457	57.773	17.012
TurIn	1.047*	1.863	1.728	3.225	1.769	8.237	1.882
TurOut	0.966*	5.986	1.556	2.131	2.200	5.912	1.943
TUrban	0.951	1.028	0.806	0.788	0.751*	1.705	0.928
HouseFin	1.026*	1.328	1.035	1.275	1.121	2.565	1.369

In addition, Friedman provides a ranking of the algorithms, so that the method with a lowest result is taken as the control algorithm. For this reason, and according to Tables 7, 8, 9, and 10, the L-Co-R algorithm results to be the control algorithm for all horizons considered and the three quality measures used.

In order to check if the control algorithm has statistical differences regarding the other methods used, the Holm procedure [14] is used. Tables 11, 12, 13, and 14 presents the results of the Holm's procedure since shows the adjusted p values from each comparison between the algorithm control and the rest of the methods for MAPE, MASE, and MdAPE, and for horizons 1, 6, 12, and 24 considering a level of significance of $alpha = 0.05$.

As can be seen in Tables 11, 12, 13, and 14, there are significant differences among L-Co-R and all the rest of the methods in the most of the cases. Analyzing more specifically for every horizon:

Table 6 Results of the methods L-Co-R, ETS, Croston, Theta, RW, Mean, and ARIMA, with respect to horizon 6 and MdAPE

Time series	L-Co-R	ETS	ARIMA	CROSTON	THETA	MEAN	RW
Airline	10.574*	223.150	13.222	31.476	89.154	26.266	85.401
WmFrancfort	2.332*	18.331	12.116	41.042	22.880	64.928	25.017
WmLondres	0.214*	5.394	5.078	30.658	11.247	53.833	15.853
WmMadrid	0.544*	28.484	12.129	46.094	27.270	65.484	28.484
WmMilan	0.519*	35.445	35.406	50.623	34.852	60.538	35.406
WmNuevayork	0.681*	4.909	5.755	36.166	18.312	64.112	24.221
WmTokio	1.207*	10.732	10.701	19.139	9.307	41.390	10.701
Deceases	0.513*	5.187	5.288	5.913	4.161	7.295	5.288
SpaMovSpec	7.824*	168.443	56.282	53.019	51.034	53.070	52.883
Exchange	0.011*	47.169	35.914	33.658	39.600	27.009	36.075
Gasoline	0.000*	7.353	8.995	9.394	6.365	19.412	7.329
MortCanc	0.152*	8.130	6.145	31.959	1.844	46.068	2.630
MortMade	2.582*	13.471	35.704	56.770	51.644	52.227	12.953
Books	1.849*	18.596	14.479	18.871	16.656	20.948	11.838
FreeHouPrize	1.547*	9.491	12.482	31.042	6.549	50.029	11.493
Prisoners	0.178*	1.786	1.906	14.123	6.422	35.766	9.371
TurIn	0.561*	6.781	5.482	12.795	4.614	32.355	6.671
TurOut	0.232*	35.128	8.219	11.965	10.860	37.078	10.784
TUrban	1.707*	8.341	6.054	6.431	4.729	17.642	8.694
HouseFin	3.028*	21.422	17.257	20.053	18.059	43.246	22.495

Table 7 Friedman's test ranking

MAPE		MASE		MdAPE	
L-Co-R	1.55	L-Co-R	2.63	L-Co-R	1.90
Theta	3.30	Theta	2.85	ARIMA	2.98
ARIMA	3.32	RW	3.60	Theta	3.10
RW	4.28	ETS	3.62	RW	4.15
ETS	4.40	Croston	4.60	ETS	4.23
Croston	5.00	ARIMA	4.65	Croston	5.30
Mean	6.15	Mean	6.05	Mean	6.35

Control algorithms are located in first row

- Horizon 1: significant differences exist between L-Co-R and the rest of the method for MAPE. With respect to MASE, there exist significant differences between the L-Co-R algorithm and Mean, ARIMA, and Croston, although it is not appropriate to assure that with methods ETS, RW, and Theta. Regarding MdAPE, L-Co-R has significant differences with all methods except ARIMA, as can be seen Table 11.

Table 8 Friedman's test ranking

MAPE		MASE		MdAPE	
L-Co-R	1.00	L-Co-R	1.70	L-Co-R	1.00
ARIMA	3.33	ARIMA	3.13	Theta	3.25
Theta	3.45	Theta	3.30	ARIMA	3.33
RW	4.35	RW	4.10	RW	4.10
ETS	4.68	ETS	4.63	ETS	4.48
Croston	5.05	Croston	5.00	Croston	5.45
Mean	6.15	Mean	6.15	Mean	6.40

Control algorithms are located in first row

Table 9 Friedman's test ranking

MAPE		MASE		MdAPE	
L-Co-R	1.00	L-Co-R	1.26	L-Co-R	1.05
ARIMA	3.40	ARIMA	3.24	Theta	2.53
Theta	3.42	Theta	3.42	ARIMA	2.55
RW	4.37	RW	4.26	RW	4.11
ETS	4.61	ETS	4.61	ETS	4.39
Croston	5.11	Croston	5.05	Croston	5.10
Mean	6.11	Mean	6.16	Mean	6.26

Control algorithms are located in first row

Table 10 Friedman's test ranking

MAPE		MASE		MdAPE	
L-Co-R	1.00	L-Co-R	1.18	L-Co-R	1.18
ARIMA	3.26	ARIMA	3.02	ARIMA	2.91
Theta	3.59	Theta	3.65	Theta	3.59
RW	4.44	RW	4.41	RW	4.29
ETS	4.76	ETS	4.74	ETS	4.74
Croston	4.88	Croston	4.94	Croston	5.24
Mean	6.05	Mean	6.05	Mean	6.06

Control algorithms are located in first row

- Horizon 6: L-Co-R has significant differences with all methods used, for every measure considered, as Table 12 shows.
- Horizon 12: there are significant differences among the control algorithm, L-Co-R, and the rest of the methods in all cases, as can be observed in Table 13.
- Horizon 24: as with horizons 6 and 12, there are also significant differences between L-Co-R and other methods, as Table 14 shows.

Table 11 Adjusted *p* values of Holm's procedure between the control algorithm (L-Co-R) and the other methods for MAPE, MASE, and MdAPE with respect to horizon 1

MAPE		MASE		MdAPE	
Mean	1.654E-11	Mean	5.340E-07	Mean	7.311E-11
Croston	4.412E-07	ARIMA	3.034E-03	Croston	6.454E-07
ETS	3.020E-05	Croston	3.839E-03	ETS	6.654E-04
RW	6.635E-05	ETS	1.432E-01	RW	9.890E-04
ARIMA	9.367E-03	RW	1.535E-01	Theta	7.898E-02
Theta	1.041E-02	Theta	7.419E-01	ARIMA	1.156E-01

Values lower than $alpha = 0.05$ indicate significant differences between L-Co-R and the corresponding algorithm

Table 12 Adjusted *p* values of Holm's procedure between the control algorithm (L-Co-R) and the other methods for MAPE, MASE, and MdAPE with respect to horizon 6

MAPE		MASE		MdAPE	
Mean	4.742E-14	Mean	7.311E-11	Mean	2.684E-15
Croston	3.055E-09	Croston	1.361E-06	Croston	7.311E-11
ETS	7.463E-08	ETS	1.854E-05	ETS	3.640E-07
RW	9.395E-07	RW	4.427E-04	RW	5.681E-06
Theta	3.352E-04	Theta	1.917E-02	ARIMA	6.654E-04
ARIMA	6.654E-04	ARIMA	3.698E-02	Theta	9.889E-04

Values lower than $alpha = 0.05$ indicate significant differences between L-Co-R and the corresponding algorithm

Table 13 Adjusted *p* values of Holm's procedure between the control algorithm (L-Co-R) and the other methods for MAPE, MASE, and MdAPE with respect to horizon 12

MAPE		MASE		MdAPE	
Mean	3.238E-13	Mean	2.874E-12	Mean	1.051E-13
Croston	4.704E-09	Croston	6.417E-08	Croston	7.372E-09
ETS	2.690E-07	ETS	1.856E-06	ETS	1.856E-06
RW	1.540E-06	RW	1.866E-05	RW	1.328E-05
Theta	5.517E-04	Theta	2.078E-03	ARIMA	3.611E-04
ARIMA	6.337E-04	ARIMA	4.862E-03	Theta	4.165E-04

Values lower than $alpha = 0.05$ indicate significant differences between L-Co-R and the corresponding algorithm

In conclusion, it is possible to confirm that the L-Co-R method is able to achieve a better forecast in majority of cases even when the horizon grows, comparing with the other 6 methods utilized and concerning to 3 different quality measures.

Table 14 Adjusted p values of Holm's procedure between the control algorithm (L-Co-R) and the other methods for MAPE, MASE, and MdAPE with respect to horizon 24

MAPE		MASE		MdAPE	
Mean	8.646E-12	Mean	4.421E-11	Mean	4.421E-11
Croston	1.609E-07	Croston	3.357E-07	Croston	4.306E-08
ETS	3.757E-07	ETS	1.563E-06	ETS	1.563E-06
RW	3.414E-06	RW	1.263E-05	RW	2.581E-05
Theta	4.775E-04	Theta	8.551E-04	Theta	1.134E-03
ARIMA	2.240E-03	ARIMA	1.240E-02	ARIMA	1.918E-02

Values lower than $alpha = 0.05$ indicate significant differences between L-Co-R and the corresponding algorithm

Acknowledgments This work has been supported by the regional projects TIC-3928 and -TIC-03903 (Feder Funds), the Spanish project TIN 2012-33856 (Feder Founds), TIN 2011-28627-C04-02 (Feder Funds).

References

1. Araújo, R.: A quantum-inspired evolutionary hybrid intelligent apporach fo stock market prediction. Int. J. Intell. Comput. Cybern. **3**(10), 24–54 (2010)
2. Bowerman, B., O'Connell, R., Koehler, A.: Forecasting: Methods and Applications. Thomson Brooks/Cole, Belmont, CA (2004)
3. Box, G., Jenkins, G.: Time series analysis: forecasting and control. Holden Day, San Francisco (1976)
4. Brockwell, P., Hyndman, R.: On continuous-time threshold autoregression. Int. J. Forecast. **8**(2), 157–173 (1992)
5. Broomhead, D., Lowe, D.: Multivariable functional interpolation and adaptive networks. Complex Syst. **2**, 321–355 (1988)
6. Carse, B., Fogarty, T.: Fast evolutionary learning of minimal radial basis function neural networks using a genetic algorithm. In: Proceedings of Evolutionary Computing. LNCS, vol. 1143, pp. 1–22 Springer, Heidelberg (1996)
7. Castillo, P., Arenas, M., Merelo, J., and Romero, G.: Cooperative co-evolution of multilayer perceptrons. In: Mira, J., lvarez, J.R. (eds.) Computational Methods in Neural Modeling, LNCS, vol. 2686, pp. 358–365. Springer, Heidelberg (2003)
8. Chan, K., Tong, H.: On estimating thresholds in autoregressive models. J. Time Ser. Anal. **7**(3), 179–190 (1986)
9. Clements, M., Franses, P., Swanson, N.: Forecasting economic and financial time-series with non-linear models. Int. J. Forecast. **20**(2), 169–183 (2004)
10. Du, H., Zhang, N.: Time series prediction using evolving radial basis function networks with new encoding scheme. Neurocomputing **71**(7–9), 1388–1400 (2008)
11. Eshelman, L.: The chc adptive search algorithm: how to have safe search when engaging in nontraditional genetic recombination. In: Proceedings of 1st Workshop on Foundations of Genetic Algorithms, pp. 265–283 (1991)
12. García-Pedrajas, N., Hervas-Martínez, C., Ortiz-Boyer, D.: Cooperative coevolution of artificial neural network ensembles for pattern classification. IEEE Trans. Evol. Comput. **9**(3), 271–302 (2005)

13. Harpham, C., Dawson, C.: The effect of different basis functions on a radial basis function network for time series prediction: a comparative study. Neurocomputing **69**(16–18), 2161–2170 (2006)
14. Holm, S.: A simple sequentially rejective multiple test procedure. Scand. J. Stat. **6**(2), 65–70 (1979)
15. Hyndman, R., Koehler, A.: Another look at measures of forecast accuracy. Int. J. Forecast. **22**(4), 679–688 (2006)
16. Hyndman, R.J., Khandakar, Y.: Automatic time series forecasting: the forecast package for r. J. Stat. Softw. **27**(3), 1–22 (2008)
17. Jain, A., Kumar, A.: Hybrid neural network models for hydrologic time series forecasting. Appl. Soft Comput. **7**(2), 585–592 (2007)
18. Li, M., Tian, J., Chen, F.: Improving multiclass pattern recognition with a co-evolutionary rbfnn. Pattern Recogn. Lett. **29**(4), 392–406 (2008)
19. Lukoseviciute, K., Ragulskis, M.: Evolutionary algorithms for the selection of time lags for time series forecasting by fuzzy inference systems. Neurocomputing **73**(10–12), 2077–2088 (2010)
20. Ma, X., Wu, H.: Power system short-term load forecasting based on cooperative co-evolutionary immune network model. In: Proceedings of 2nd International Conference on Education Technology and Computer, pp. 582–585 (2010)
21. Makridakis, S., Andersen, A., Carbone, R., Fildes, R., Hibon, M., Lewandowski, R., Newton, J., Parzen, E., Winkler, R.: The accuracy of extrapolation (time series) methods: Results of a forecasting competition. J. Forecast. **1**(2), 111–153 (1982)
22. Makridakis, S., Hibon, M.: The m3-competition: results, conclusions and implications. Int. J. Forecast. **16**(4), 451–476 (2000)
23. Maus, A., Sprott, J.C.: Neural network method for determining embedding dimension of a time series. Commun. Nonlinear Sci. Numer. Simul. **16**(8), 3294–3302 (2011)
24. Parras-Gutierrez, E., Garcia-Arenas, M., Rivas, V., del Jesus, M.: Coevolution of lags and rbfns for time series forecasting: L-co-r algorithm. Soft Comput. **16**(6), 919–942 (2012)
25. Pea, D.: Análisis de Series Temporales. Alianza Editorial (2005)
26. Potter, M., De Jong, K.: A cooperative coevolutionary approach to function optimization. In: Proceedings of Parallel Problem Solving from Nature, LNCS, vol. 866, pp. 249–257. Springer, Heidelberg (1994)
27. Rivas, V., Merelo, J., Castillo, P., Arenas, M., Castellano, J.: Evolving rbf neural networks for time-series forecasting with evrbf. Inf. Sci. **165**(3–4), 207–220 (2004)
28. Sheskin, D.: Handbook of parametric and nonparametric statistical procedures. Chapman & Hall/CRC, Boca Raton (2004)
29. Snyder, R.: Recursive estimation of dynamic linear models. J. Roy. Stat. Soc. Ser. B (Methodological) **47**(2), 272–276 (1985)
30. Takens, F.: Dynamical systems and turbulence, Lecture Notes In Mathematics, vol. 898, Chapter Detecting Strange Attractor in Turbulence, pp. 366–381. Springer, New York, NY (1980)
31. Tong, H.: On a threshold model. Pattern Recogn. signal process. NATO ASI Ser. E: Appl. Sc. **29**, 575–586 (1978)
32. Tong, H.: Threshold models in non-linear time series analysis. Springer, Berlin (1983)
33. Whitehead, B., Choate, T.: Cooperative-competitive genetic evolution of radial basis function centers and widths for time series prediction. IEEE Trans. Neural Netw. **7**(4), 869–880 (1996)
34. Winters, P.: Forecasting sales by exponentially weighted moving averages. Manage.Sci. **6**(3), 324–342 (1960)