Chapter 2
Using Meta-Learning in Automatic Demand Forecast with a Large Number of Products

Luis Gutiérrez and Marcel Goic

Abstract Demand analysis is one of the cornerstones of any supply chain management system, and most of the critical operational decisions in the supply chain rely on accurate demand predictions. Although a large body of academic literature proposes various forecasting methods, there are still important challenges when using them in practice. The common problem is that firms need to decide about thousands of products, and the demand patterns could be very different between them. In this setting, frequently, there is no single forecasting method that works well for all products. While some autoregressive models might work well in some cases, the demand for other products might require an ad-hoc identification of trend and seasonality components. In this chapter, we present a methodology based on meta-learning that automatically analyzes several features of the demand to identify the most suitable method to forecast the demand for each product. We apply the methodology to a large retailer in Latin America and show how the methodology can be successfully applied to thousands of products. Our analysis indicates that this approach significantly improves the firm's previous practices, leading to important efficiency gains in the supply chain.

Keywords Forecasting · Meta-learning · Time series · Retailing

2.1 Introduction

The retail industry faces a dynamic and competitive landscape that has been confronted with the irruption of digital channels, the emergence of new formats,

L. Gutiérrez · M. Goic (✉)
Department of Industrial Engineering, University of Chile, Santiago de Chile, Chile
e-mail: mgoic@uchile.cl

L. Gutiérrez
e-mail: luisgutierrez@uchile.cl

M. Goic
Instituto de Sistemas Complejos de Ingeniería, Santiago, Chile

J. L. García Alcaraz et al. (eds.), *Supply Chain Management Strategies and Methodologies*,
Lecture Notes in Logistics, https://doi.org/10.1007/978-3-031-32032-3_2

and the increasing use of technology in the value chain. Among the long-term trends consolidated in recent years is the automation of various processes, ranging from inventory management to self-checkout terminals. In this research, we propose a methodology to automate demand forecast at the product-store level, which is an important input for several key processes such as assortment planning or inventory management. For instance, to automate the replenishment of stores from the distribution centers, we need to project how much product will be sold shortly in each store. Accurate forecasting has important consequences for operation performance. If the forecast underestimates the demand, the products will be out of stock, harming sales. If the forecast overestimates the demand, the inventory cost would be unnecessarily high, and it might even force the implementation of aggressive price reductions to reduce stocks.

The academic literature provides numerous methodologies to forecast demand in the retail industry (Ma et al. 2016; Huber and Stuckenschmidt 2020). However, practical implementations of automatic forecasting systems imply important methodological challenges. First, most retailers consider a large assortment of thousands of SKUs in several dozen stores, which require the completion of several thousands of forecasting tasks. Although computational power is not an important barrier to estimating a large number of statistical models, there is a more fundamental difficulty in automating demand forecasting. The underlying time series of sales of different products can be radically different, and there is no universal model to provide the best solution for all cases. While a simple autoregressive model can provide satisfactory solutions in some cases, other cases might require a more comprehensive identification of seasonal components. A common practice to deal with this problem is to either commit to a forecasting model that works well on average or assign human analysts to inspect the series and decide case by case. In this research, we propose a methodology to automatically select the best estimation method for each series, facilitating the automation of critical processes without sacrificing forecasting accuracy.

The need to use different forecasting models comes from the existence of distinct components in different demand series. To illustrate the point, in Fig. 2.1, we display the time series of sales of four different products in our dataset. For product A, we observe very pronounced spikes in demand. As this product belongs to the toy category, those spikes are associated with seasonal events such as Christmas or Children's Day, which are strongly associated with larger purchases in the toy category. For product B, demand is higher in the second part of every year, but that is mainly associated with year seasonality and not with a single event. This pattern is relatively common to items in the clothing category, where the demand tends to be very cyclical. In this set, we also have products with no evident seasonal patterns, such as products C and D. Product C presents large variations in sales. However, those occur at different times of the year, possibly associated with promotions or other unobservable factors. On the other hand, product D presents less variation over time, with almost no acute spikes in the observational period.

Overall, we observe series with very different components requiring different modeling approaches. To automate the forecast, we need to estimate every case

Fig. 2.1 Illustration of several time series with different seasonality and trend components

adequately. However, some models provide better results in some cases, and others perform better in others. Our solution is based on a technique called meta-learning, in which a machine learning model decides the best model to use in each case based on observable characteristics of the series, such as the trend and seasonality strength, as well as the size of the autoregressive components. To calibrate this model, we need to produce many forecasts using different models to identify which performs best under different conditions.

In addition to proposing a methodology to automatically select the best forecasting model for each series, using historical data, we evaluate the impact of utilizing this approach on the accuracy of the forecast, and we demonstrate that it could lead to better results. Furthermore, we conducted a business evaluation using a controlled experiment to compare product sales and inventory levels for products. We used the methodology to decide product replenishment against a control where orders were decided using standard business practices. Here we found that the model can indeed improve operational efficiency in practice. In this project, we developed a predictive model to generate an accurate automatic forecast for various products, thus reducing logistics and inventory management costs in the supermarket industry.

The rest of the article is organized as follows. In Sect. 2, we review the relevant literature. Section 3 introduces the methodology we use to build forecasts for many products. Then, in Sect. 4, we describe the empirical setting and provide descriptive statistics of the thousands of products we consider in the empirical evaluation. In Sect. 5, we present the result, and we conclude in Sect. 6 with the main takeaways of our research and a discussion with some avenues for future research.

2.2 Literature Review

This research is associated with three streams of research. First, from a substantive perspective, we relate to a vast literature exploring efficient demand estimation in the retail industry. Second, from a methodological perspective, our research is connected to recent advances in meta-learning. Lastly, from an operational perspective, we aim to produce forecasts with minimal human intervention and therefore, our research also relates to the literature on retail automation. Next, we discuss these three streams sequentially.

Regarding demand estimation, previous literature has recognized that the forecasting approach depends on the nature of the decisions they support. For instance, Fildes et al. (2022) pose that strategic, tactical, and operational decisions require different methods and data aggregation levels. In this work, we provide product and store-level forecasting to support operational decisions such as order sizes and inventory volumes. Since the introduction of retail scanner data, various methods have been proposed to forecast sales. While a common practice in the industry is using regressions (e.g., Macé and Neslin 2004) or autoregressive time series models (e.g., Srinivasan et al. 2008), recent methodological advances have motivated a large number of investigations using more sophisticated forecasting models. For instance, Ali et al. (2009) compare a variety of autoregressive, stepwise, and support vector regression models to forecast demand in the presence of promotion and found that, with more detailed input data, machine learning models can significantly improve the forecasts. More recently, Spiliotis et al. (2020) compare statistical and machine learning methods to forecast daily demand and conclude that the latter reduces the bias and leads to more accurate predictions. Unlike these systematic evaluations that evaluate the aggregated performance of different forecasting models, our research aims to identify the best model for each case. In addition, while most of these studies consider a few dozen scenarios, our model is devoted to providing adequate demand forecasting for thousands of product-store combinations.

The desire to have estimation methods that can be generalized to multiple prediction instances has a long tradition in the forecasting literature. More than 30 years ago, Mahmoud et al. (1988) already posed that no one sales forecasting method is appropriate for every situation (p. 54). While the problem was identified a long time ago, it was not until the last decade that the literature has provided more systematic approaches to address it. Early approaches to finding general forecasting models within a given domain rely on aggregation methods (for example, Horváth and Wieringa 2008). However, we believe these approaches are better suited for cases with a relatively short number of temporal observations for each unit, which is less of a concern in our empirical application. Another approach to aim for generalizability is using forecasting ensembles, where multiple models and data sources of different types are combined to produce a unified forecast (Wu and Levinson 2021). Our empirical analysis considers ensembles as potential candidates to generate the best predictions. However, we consider the possibility that one model by itself could be the most suitable for specific instances.

The methodology we used to forecast the demand at the product-store level is based on meta-learning. The basic idea behind meta-learning is using a classifying method to select the most suitable model for a given time series (for a similar methodology, see Prudêncio and Ludermir 2004). Unlike ensemble learning, which combines multiple forecasts, we aim to select the best model for each case in meta-learning. With the proliferation of a wide gamut of time-series models, the need for some guidelines to decide on the best modeling approach has become more pressing. Early guidelines mostly relied on visual examination of the series (Pegels 1969) or qualitative rules (Collopy and Armstrong 1992). More recently, meta-learning methods have taken advantage of the important advances in machine learning to use a classification model to decide the most promising approach as a function of a large number of features characterizing a given time series (Talagala et al. 2018). Using a wide range of univariate time series from different domains, Wang et al. (2009) identify six clusters of series that might require different forecasting techniques. Similarly, Lemke and Gabrys (2010) identify an extensive set of features describing the time series and another set of features to characterize the forecasting methods. More recently, Ma and Fildes (2021) applied meta-learning methods in retail and demonstrated that they could significantly improve forecasting efficacy. Although they evaluate meta-learning using a publicly available dataset, we effectively use this approach to support decision-making in the retail industry. In terms of the methodology, we find that the addition of a final step, in which we discard those models with worse performance, could play a critical role in facilitating the classifier to select the best model for each forecasting task.

To conclude this review, our research is also related to previous work on retail automation. Considering the massive nature of retail operations and the high competition in the retail markets, there is constant pressure to systematize and automate processes (Begley et al. 2019). The number of applications that automatize key retail decisions is vast. These include the evaluation of promotional effectiveness with a minimum of analyst intervention (Abraham and Lodish 1987), the dynamic adjustment of store item-level prices (Zhou et al. 2009), and the delivery of automatic responses triggered by consumer actions (Goic et al. 2021) to name a few. The main goal of this research is to provide an automatic demand forecast at the product-store level. Although we expect that automation can lead to better forecasting in the long term, we aim to provide predictions that are, at least, as good as the current business practices that require manual examination of thousands of series.

2.3 Methodology

As illustrated in Fig. 2.2, the proposed methodology consists of four main steps. First, we produce forecasts for many cases using various models and compute error metrics for each model and case (1). Second, we generate several features to characterize each case (2). Third, we use those features to train a meta-learning model that indicates

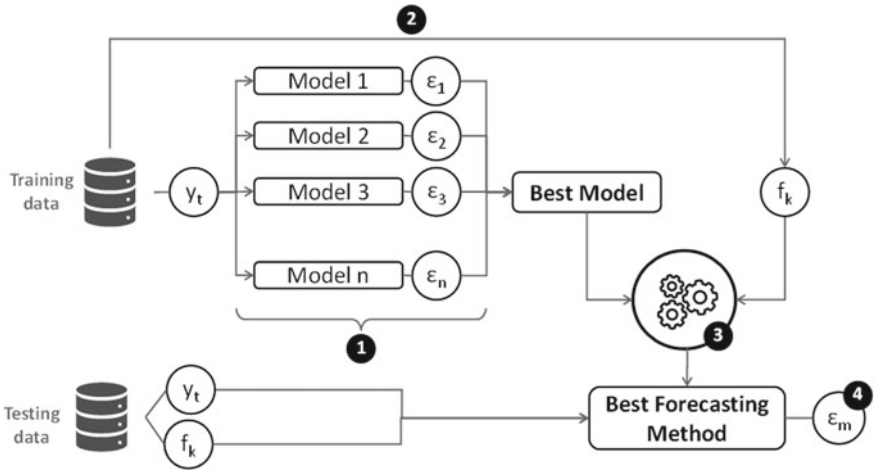

Fig. 2.2 Schematic representation of the proposed methodology

which model leads to smaller errors for given values of features (3). We conclude by applying the results of the meta-learning and evaluating its performance (4).

In the following subsections, we discuss each of these components in more detail: Step (1) is described next in the sub-section "Forecasting with alternative models". Step (2) is described later in the sub-section "Extraction of time series features". Step (3) is presented in the sub-section "Model selection through meta-learning". The evaluation of the whole methodology using the best method is presented in the section "Results".

2.3.1 Forecasting with Alternative Models

We start the methodology by estimating a variety of forecasting models for a large number of products. The objective of this task is twofold. First, it allows us to verify that no single model generates the most accurate prediction for most cases, which provides empirical justification for including a meta-learning process to assign product demand patterns to models. Second, the results of these models work as an input for the calibration of the meta-learning algorithm. In fact, the assessment of the forecasting errors gives us the basis for the construction of classification labels that will be used in the training of the meta-learning step.

The models that we consider in the evaluation are:

- **Moving Average (MA)**: This is the model used by the firm before the implementation of the meta-learning, and it generates the forecast for the next period as a weighted mean of the observed sales in the last two periods (Johnston et al. 1999).

- **Autoregressive Integrated Moving Average (ARIMA)**: On a given period, the values of the time series depend on their lagged values and lagged errors. The series is further differentiated to estimate the model to allow nonstationary processes (Newbold 1983).
- **Holt-Winters (HW)**: This model expands the simple exponential smoothing approach by allowing trends in the forecasting. Thus, the method comprises three smoothing equations for the level, the trend, and the seasonal components (Chatfield 1978).
- **Exponential smoothing state space model with Box-Cox transformation, ARMA errors, Trend, and Seasonal components (TBATS)**: This model uses a combination of exponential smoothing and Box-Cox transformations to accommodate multiple seasonal components automatically. Each seasonal component is modeled by a trigonometric representation based on a Fourier series (De Livera et al. 2011).
- **Time-dertificial neural networks (TDANN)**: This model uses a flexible neural network architecture to model the time series. In this structure, we use lagged values as inputs to the network (Clouse et al. 1997).
- **Seasonal-Trend decomposition using LOESS (STL)**: This model allows the time series's decomposition into three components: seasonality, trend, and residuals. To combine these components, this model uses a robust local regression approach to outliers (Cleveland et al. 1990).
- **Ensemble (EN)**: In this approach, the forecast corresponds to the combination of multiple models. While the literature suggests alternative approaches to combining models, in our case, we simply consider a simple average that often outperforms more complex combination schemes (Bates and Granger 1969).

2.3.2 Model Selection Through Meta-Learning

To select the best model for each time series, we use meta-learning. To perform meta-learning, we need to generate a dataset with all the available time series. For each series, we need (i) a label indicating which model had the most accurate prediction for this series and (ii) several features to characterize them a priori. With these components, the problem translates into a standard classification model. The labels with the best model are obtained from the extensive forecasting with alternative models we explained in the previous subsection. The process of extracting time-series features is explained in depth in the following subsection.

We split the database into training and testing subsets using standard supervised learning approaches. The model is calibrated using the training data and then evaluated in the testing data. In our case, we use a random sample of 80% of the product-store series for training and the remaining 20% for testing. Although there are many alternative methods to perform the classification task, following previous work on meta-learning, we use a random forest model (Talagala et al. 2018). In our case, the random forest is produced, averaging 1,000 trees. We tried alternative specifications

with a larger number of trees without observing a meaningful improvement in the classifier's performance.

The labels indicating which candidate is the best model are based on the Mean Absolute Error (MAE). Since the label is used to guide which model performs better for each time series shape, to feed the random forest classifier, we only consider the case in which there is a clear winner among the competing model. Of the 5,000 time series analyzed, there are 1,103 series where there is no meaningful difference in the prediction errors of at least two models, which we discarded from the analysis. Thus, the classification is trained with 3,897 series. It is possible that other methods could perform better without removing those cases from the training set, but this filter proved to lead to better forecasting results for our application.

Another variation in the classifier proved to enhance the meta-learning significantly. Instead of calibrating the classifier to select the best model among all possible methods, we calibrate it to choose between the two models with the best overall performance. Restricting the classification to only those models with the smallest forecasting errors reduces the potential gain of the automation of model selection. In fact, as we will see in the result section, every model provides the best forecast for at least a few cases. Therefore, removing models will lead to a worse possible solution for those series. Notice, however, that the gain in the forecasting capabilities only materializes if the classifier effectively identifies the best model for each series. However, with more labels, the classification task becomes more difficult. Thus, the key tradeoff is between reducing the potential forecasting gains and augmenting the classification errors. As we will see in the result section, in our empirical application, the reduction in the classification error more than compensates for the selection of suboptimal methods, and meta-learning with the best models leads to better results overall.

2.3.3 Extraction of Time Series Features

To calibrate a meta-learning step, we need to connect the performance of all forecasting methods to a series of observable features of the forecasting task. In this project, these observable features correspond to characteristics of the shape of the underlying time series. For instance, we consider the strength of the seasonal and trend components. The basic idea is that some methods might be more suitable to capture those components than others and that the meta-learning step can identify those patterns by observing the performance of several methods in thousands of cases.

We closely follow previous literature on time-series meta-learning to define the list of time-series features to use in the empirical analysis. For each demand series of product-store combination, we compute 15 features such as trend, seasonal strength, and autocorrelation coefficients, as well as metrics of the internal variability such as entropy, spikiness, and maximum level shifts (Talagala et al. 2018; Ma and Fildes 2021). To illustrate how different time series differ depending on the values of these

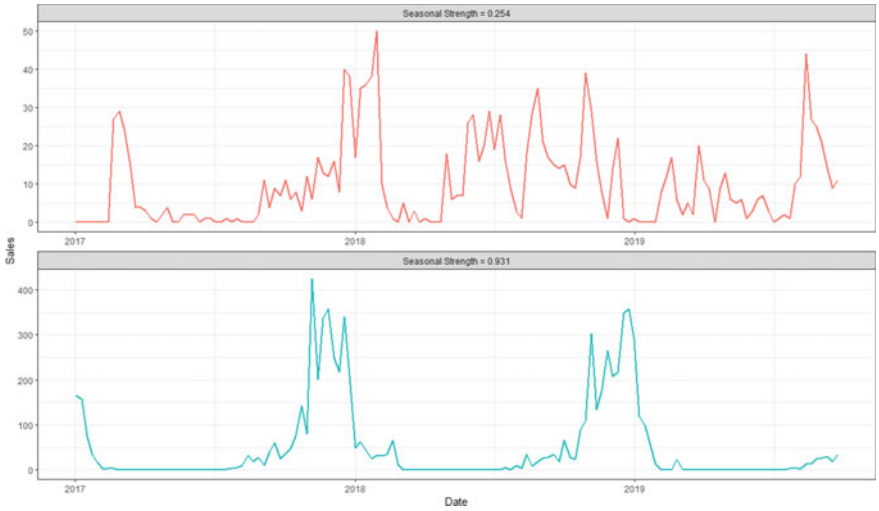

Fig. 2.3 Visual representation of two series of sales with different Seasonal Strengths

features, in Fig. 2.3, we display two series of demand with fairly different values for Seasonal Strength (by construction, the Seasonal Strength takes values in the range [0, 1]). In the bottom panel, we display a series with high Seasonal Strength. In this case, sales during the summer times (November–March, in the southern hemisphere) are much higher than in the rest of the year. In the top panel, we display a series with low Seasonal Strength, and, in this case, it is much more difficult to anticipate what would be the weeks with higher sales. In terms of the forecast, the need for a model that properly controls for seasonality appears to be more critical in the second series.

2.3.4 Execution and Evaluation of Meta-Models

To complete the methodology, we apply the classifier for many products, then evaluate to what extent the resulting predictions improve concerning standard forecasting tools. Considering that we apply the proposed methodology to many products, we need an aggregated performance metric. In our case, we use a weighted Mean Absolute Percentage Error (wMAPE) in which we give larger weight to products with larger sales levels. Our choice is justified because of its scale independence and consistency with the business objective of having more accurate predictions for those products with a larger impact on revenues (Narayanan et al. 2019).

Table 2.1 Descriptive statistics of the demand series for different product-stores combinations

		Weekly sales [units]		Price [CL$]	
Product family	N° Products	Mean	Max	Mean	Max
15	297	24.9	903	5,186	172,914
27	4,703	40.5	4,355	3,221	170,540

2.4 Empirical Setting

From a practical point of view, we are interested in automatizing demand forecasting to use those estimates to feed different operational processes. The focal decision in this research is the daily number of units to distribute from the central warehouses to all stores scattered throughout the territory. On the one hand, considering the limited storage space in the store, demand overestimation could lead to high operational costs. On the other hand, demand underestimation could lead to lost sales due to an out-of-stock. While we formally analyze the inventory reorder process, the forecast could also be used to support other decisions, such as assortment or promotional planning.

We consider 5,000 demand series of different product-store combinations in the empirical evaluation. The time series correspond to 143 weeks of sales from January 2017 to September 2019 for the clothing and toys categories. These series span 200 families of products and 130 stores in Chile. It is worth noting that not all product families are sold in all stores. Due to the constant product introduction, these two product categories are precisely among those the company has faced more difficulties in generating forecasting at the product-store level. The constant variation in the product offering motivates us to forecast at the product family and not at the SKU level. In Table 2.1, we display descriptive statistics of the demand for both product categories.

Statistics from Table 2.2 indicate that most of the series we consider in this numerical analysis correspond to clothing, which tends to have larger sales than the toys category, which also tends to have larger prices. For our analysis, the key insight from these statistics is that the demand series might be fairly different between products, providing further qualitative support to the need for a meta-learning classifier that guides the best model to forecast each series.

2.5 Results

According to the methodology presented in Sect. 3, several components are worth reporting. We first describe the results of the forecasting of all independent standard models. Then we describe the implementation of the time-series feature extractions. These two components are the primary inputs for the meta-learning stage that we

Table 2.2 Forecasting Error across models for the first week

Model	MAE	Sd	wMape (%)
HW	13.7	20.5	33.2
TBATS	15.9	29.3	38.6
NNAR	17.9	25.0	43.5
STL	18.4	27.3	44.5
ARIMA	19.1	35.4	46.4
MA	19.2	25.5	46.5
EN	13.6	21.2	33.0
Mean	16.8	26	40.8

present next. We conclude this section using the forecasting models to evaluate the business impact.

2.5.1 Forecasting Through Standard Models

We first estimate each of the seven forecasting models for each 5,000-time series to complete 35,000 forecasting tasks. The majority of these models require the calibration of hyper-parameters. For TBATS, we need to determine if Box-Cox transformation is required or for the ARIMA models, and we need to decide the number of lags to use. We tune all these hyperparameters using cross-validation.

In this exercise, the forecasts correspond to the daily sales of the last four weeks of the time series. This forecasting window is chosen to match the typical target for inventory reorders. Figure 2.4 illustrates the forecasts of all individual methods for a selected time series. Although the series largely differ in features (trend, seasonality, spikiness, etc.), this example represents a common pattern we find in most series: the predictions are not radically different between models. While this indicates that any model could provide a reasonable approximation, it also suggests that it might be difficult to classify the best model for a given series. Beyond the illustration of a given series, Table 2.2 reports the forecasting errors for the first week of forecasting for all models across the 5,000 series. In this table, we include the MAE we use to compare predictions between models for a given series and the wMAPE we use later to evaluate the performance across series. Consistent with the previous example, these results indicate that all proposed models are competitive, with relatively small differences in the aggregated performance metrics between the best and worst models.

While this indicates that any model could provide a reasonable approximation, it also suggests that it might be difficult to classify the best model for a given series. Beyond the Illustration of a given series, Table 2.2 reports the forecasting errors for the first week of forecasting for all models across the 5,000 series. In this Table 2.2, we include the MAE we use to compare predictions between models for a given series and the wMAPE that we use later to evaluate the performance across series.

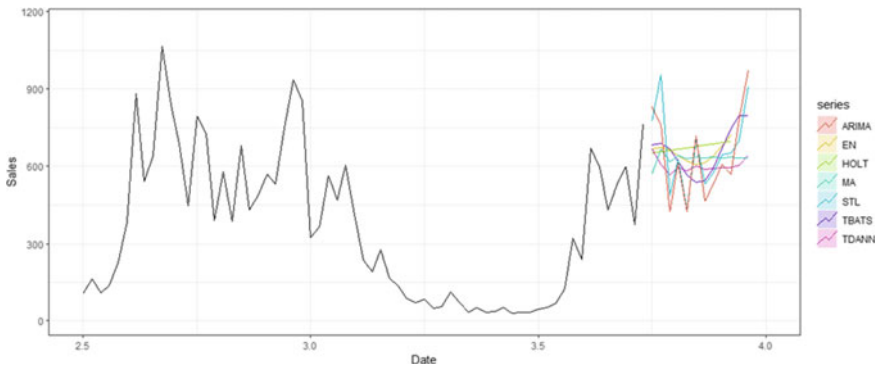

Fig. 2.4 Illustration of alternative forecasting results for a selected series

Consistent with the previous example, these results indicate that all proposed models are competitive, with relatively small differences in the aggregated performance metrics between the best and worst models.

To complement previous results, in Table 2.3, we display the forecasting errors for all four weeks we used in these numerical exercises. As expected, the further the forecasting window is in the future, the lower the accuracy of the prediction. However, the notion that the differences between models are small remains.

Recall that our methodology uses the forecasting results from individual models to calibrate a classification model that determines the best model to predict each series. In this regard, the forecasting of individual models is the primary source to build the labels of the classification model. We use the smallest forecasting error for each case to produce these labels. The frequencies of these labels are displayed in Fig. 2.5, where we further decompose them by week. For instance, the ARIMA model has the smallest forecasting errors in 17.9% of the series in week 1. Similarly, the ensemble produces the best results in 13.7% of the series for the same week.

Considering that we had previously found that the forecast errors were not dramatically different between models, it may not be surprising that we now find that no

Table 2.3 Forecasting errors by week

Model	S1	S2	S3	S4	Mean
HW	13.7	15.7	16.4	19.2	16.3
TBATS	15.9	16.2	16.8	18.2	16.8
NNAR	17.9	18.9	21.7	19.9	19.6
STL	18.4	18.8	21.0	20.8	19.8
ARIMA	19.1	17.5	18.8	19.8	18.8
MA	19.2	18.5	20.4	19.9	19.5
EN	13.6	14.4	15.2	17.0	15.1
Weekly mean	16.8	17.1	18.6	19.3	18.0

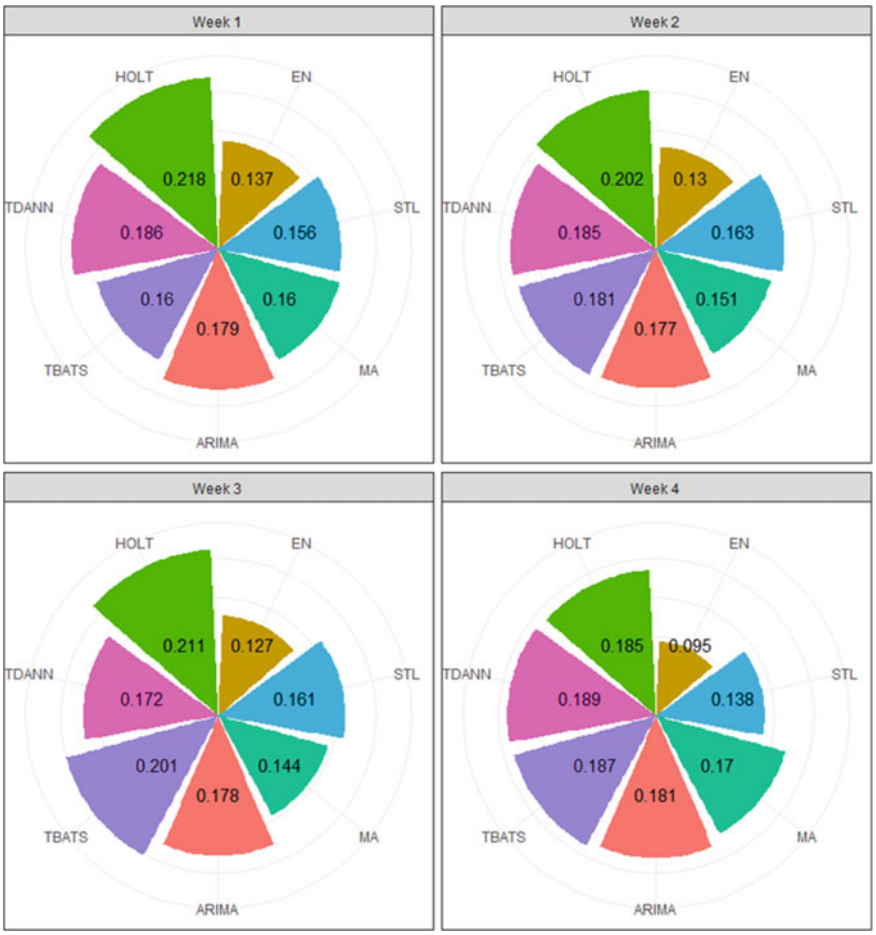

Fig. 2.5 Fraction at which each model provides the minimum forecasting error

model is the best alternative for most cases. It is possible, however, that a particular model could be consistently better, but only by a small margin. The results in Fig. 2.5 indicate that this is not the case and that some models work well for some series, and others predict better in other cases. This is precisely the pattern that justifies the need for a classifier to guide the decision of which model should be used for each specific prediction task.

The comparison across models reveals that the ensemble is the preferred model in the least number of cases. This is somewhat surprising considering that overall is the method with the smallest mean error. To conciliate these two empirical findings, it is worth emphasizing that the ensemble derives from averaging multiple models. Thus, while this approach generates consistently good solutions, it is often the case that there is one specific model that works better for that particular case. While taking

averages warrants the production of good models, at the same time, it is influenced by relatively bad models, making it difficult to produce the best solution.

2.5.2 Generation of Features

As the methodology section explains, we compute features closely following what previous literature has used to characterize time series. This extraction considers trends, seasonality, and autoregressive factors, among others. In Table 2.4, we display the list of the time-series features we use for meta-learning, along with their corresponding descriptive statistics. For a complete study of feature extraction, see Wang et al. (2006).

According to the descriptive statistics presented in Table 2.4, except for the spike, the features extracted from the different time series present significant dispersion. Consequently, the observed time series differ in their shapes, providing enough variation to learn about their incidence in the performance of each model.

Table 2.4 List of time-series features for meta-learning with the corresponding descriptive statistics for the case of study

Variable	Description	Min	Mean	Max	Sd
Trend	Strength of trend	0,000	0,131	0,815	0,111
Spike	Spikiness	0,000	0,000	0,001	0,000
Linearity	Linearity	−5,934	0,371	9,025	1,914
Curvature	Curvature	−5,087	−0,381	4,696	1,420
Seasonal	Seasonal strength	0,228	0,600	0,970	0,156
Entropy	Shannon entropy	0,598	0,890	1,000	0,065
Xacf1	First ACF of the series	0,017	0,572	0,938	0,150
Xacf10	SS of the first ACF of the series	0,006	0,924	5,646	0,753
Diff1acf1	First AF of the series differences	−0,651	−0,271	0,348	0,118
Diff1acf10	SS of the first 10 ACF of the first differences	0,039	0,179	0,949	0,080
Diff2acf1	First ACF of the first differences	−0,804	−0,561	−0,031	0,082
Diff2acf10	SS of the first 10 ACF of the second differences	0,159	0,435	1,774	0,135
Eacf1	First ACF of the remainder series	−0,387	0,370	0,842	0,172
Eacf10	Sum of squares of first 10 ACF of remainder series	0,005	0,352	2,150	0,257
Seasacf1	Autocorrelation coefficient at the first seasonal lag	−0,292	0,191	0,589	0,155

SS = Sum of the squares

2.5.3 *Meta-Learning*

Considering this is one of the most critical steps in the methodology, we describe two variants to learn from the best modeling approach to conduct the forecast for each series. Although both versions use a Random Forest to classify, we consider two different sets of models in which the Random Forest must classify. First, we feed the meta-learner with all forecasting models, and then we restrict the classification to the two models with the best overall performance. A perfect classifier would benefit from selecting from a larger set of models. However, more candidates make the classification task more complex, and therefore, which approach would lead to better results is an open empirical question.

Before presenting the results of using a meta-learner to select the best model, in Fig. 2.6, we display the mean value for all time-series features depending on the model with the best performance.

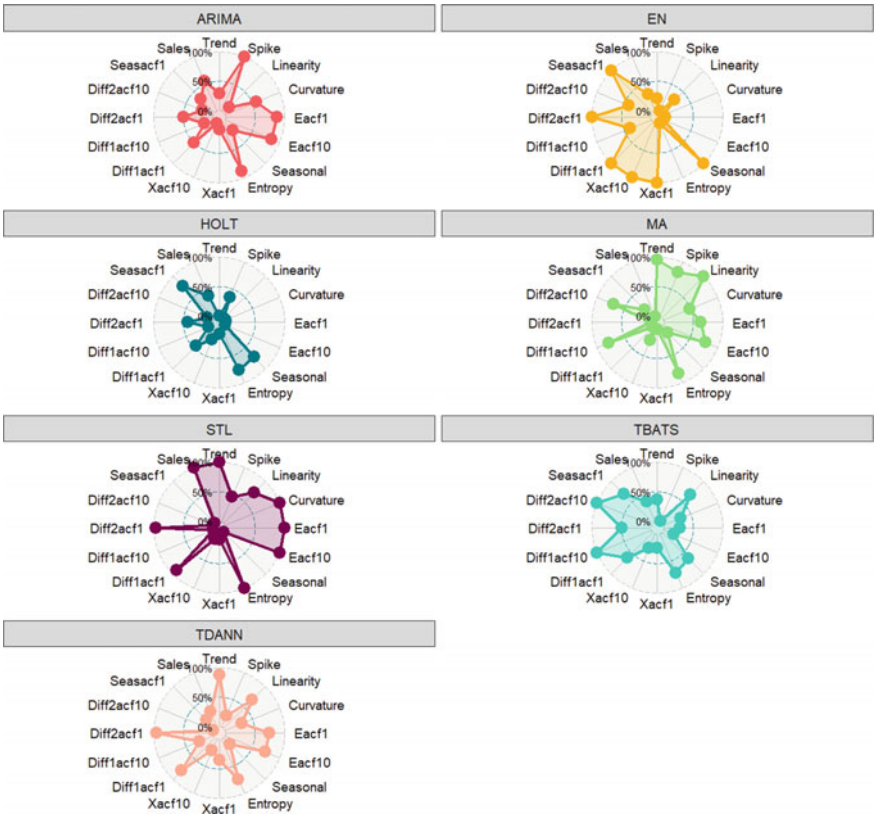

Fig. 2.6 Mean attribute value depending on which is the preferred model

According to these results, we corroborate that some models tend to perform better for specific profiles of attributes. For instance, when STFL is preferred, the underlying time series tend to have large values for curvature, eafc1, and eafc10. Similarly, a moving average is preferred for series with large values for trend, linearity, and spike. These results prove that meta-learning can effectively identify the underlying patterns connecting time-series features and model performance.

2.5.3.1 Classification with all Models

In this first exercise, the meta-learning step must decide the best model among seven competing alternatives. Table 2.5 reports the error of the Meta-Forecast against all other contenders and the fraction at which each model ended up being the best forecast (Win Rate).

Results from Table 2.5 indicate that the meta-forecast, along with the Holt-Winters model has the highest win rate among all. This provides preliminary evidence that using a model classifier can positively impact the system's overall performance. Notice, however, that in terms of the forecasting error, the meta-forecast does not provide the best results and simpler approaches, such as the ensemble or Holt-Winters, perform better on average. This indicates that while meta-forecast is

Table 2.5 Performance of meta-Forecast against individual models (first exercise)

Model	MAE	WMAPE (%)	Win rate (%)
HOLT	12.9	31.8	19.3
TBATS	14.8	36.5	12.8
STLF	17.0	41.9	14.0
NNAR	17.2	42.4	16.5
MM	18.3	45.1	14.2
ARIMA	18.5	45.6	14.1
ENSAMBLE	12.7	31.3	9.2
Meta-forecast	14.9	36.7	19.3

Table 2.6 Performance of meta-forecast against individual models (second exercise)

Model	MAE	WMAPE (%)	Win rate (%)
HOLT	12.9	31.8	19.3
TBATS	14.8	36.5	12.8
STLF	17.0	41.9	14.0
NNAR	17.2	42.4	16.5
MM	18.3	45.1	14.2
ARIMA	18.5	45.6	14.1
ENSEMBLE	12.7	31.3	9.2
Meta-forecast	11.0	27.1	24.9

frequently the best solution, the classifier could make serious classification mistakes and some series were probably forecasted with models with large errors. These results motivate an alternative and more conservative approach in which the classifier only selects among those models that perform well on average, as we explore next.

2.5.3.2 Classification with the Best Two Models on Average

In this second exercise, the classifier only considers two labels associated with the Holt-Winters and the ensemble model that performed better on average. Table 2.6 reports the errors of this new meta-forecast against all other contenders and the fraction at which each model leads the smallest forecasting error (Win Rate).

Compared to the previous case, this new meta-forecaster leads to much better results and overperforms all other models in all relevant metrics. The meta-forecast model not only provides a significant reduction in average error metrics with a wMAPE of 27.1%, which is 4.2% points better than the closest competitor (Ensemble) and more than 18% points better than a simple ARIMA model. These numbers lead the meta-forecast to provide the very best solution in 24.9% of the cases, which is almost 10% more than the closest competitor.

Overall, these results indicate that meta-learning can significantly boost accuracy to make better predictions regarding detailed retail demand sales. However, this gain is not automatic, and it might be necessary to learn the best configuration for the classifier to achieve the best performance.

2.5.4 Business Evaluation

In previous sections, we have shown that the use of meta-learning helps to automate the forecasting process, allowing an algorithm to decide the most suitable model to estimate each combination of products and stores. Furthermore, the resulting forecasts could even lead to more accurate predictions. In this section, we empirically test whether these improvements can be effectively applied in a real setting and evaluate their impact on relevant business metrics.

To measure the impact of the forecasting automation, we evaluate their impact on the process of product replenishment that requires estimating the future demand at the product-store level. Our evaluation is based on a controlled experiment in the clothing department, where a selected group of products and stores operated their replenishment process using the automatic forecasting methodology proposed in this chapter, and a comparable group of products continued their replenishment processes using standard business practices. While in the treatment, we forecast the demand using the automatic meta-learner; in control, the forecast was performed by analysts who calibrate simple autoregressive models, and they can make a judgment call to overwrite the forecast if they consider it necessary. The treatment and control groups

Table 2.7 The daily mean of sales inventory between treatment and control conditions

	Treatment	Control
Sales	277.1	250.5
Inventory	18,644.4	19,096.6

were selected to have similar demand levels pre-treatment, and the experiment lasted two weeks.

Indeed, the automation of the forecasting process brings several benefits that can only be observed in the mid-term. These include more consistent decision-making, the fastest processing, and cost savings associated with the process. For this evaluation, we will focus on the impact that can be measurable in the short term. More precisely, we look at the inventory levels and total sales. We expect that if the forecasting is successful, it should lead to lower inventory levels and more sales. Although we do not expect the forecasting to increase the demand, a more precise forecast should be associated with a smaller number of out-of-stocks and positively affect sales. Table 2.7 reports the daily mean for sales and inventory for this experiment.

The treatment and control groups were selected to be balanced. Therefore, the treatment's larger sales and smaller inventory provide preliminary evidence that the forecast can positively affect both metrics. However, a formal analysis requires detailed control for sales levels and temporal variations. To do so, we exploit the panel data structure of the experimental setting and estimate the following two regression models:

$$\text{sales}_{\text{ist}} = \alpha_i^1 + \beta_s^1 + \gamma_t^1 + \delta^1 \cdot \text{Treat}_{\text{ist}} + \varepsilon_{\text{ist}}^2 \tag{2.1}$$

$$\text{inventory}_{\text{ist}} = \alpha_i^2 + \beta_s^2 + \gamma_t^2 + \delta^2 \cdot \text{Treat}_{\text{ist}} + \varepsilon_{\text{ist}}^2 \tag{2.2}$$

The key variable in this regression is $\text{Treat}_{\text{ist}}$ that takes the value 1 if the product i in store s, in day t was replenished using the automatic forecasting methodology. The dummy variables $(\alpha_i^k, \beta_s^k, \gamma_t^k)$ control for product, store, and day-fixed effects $(k \in \{1, 2\})$. According to our previous discussion, we expect that $\delta^1 > 0$ meaning that the automatic forecasting model increased the sales volume on average, and $\delta^2 < 0$, meaning that the automatic forecasting model decreased the inventory levels. The results of the regression models are displayed in Table 2.8. In the table, we include two versions of the Eq. (2.1) and (2.2) that differ in whether we control for stores or not. In all cases, we reported clustered standard errors by product and day. In the analysis, we observe the sales of all products for all days in the experiment ($N =$ 9,705), but there is an imperfect inventory collection. Therefore we only observe a fraction of them ($N = 5{,}470$).

Results from Table 2.8 confirm our hypothesis about the direction of the impact of a successful implementation of automatic forecasting. In fact, we find evidence of a positive effect on sales and a negative effect on inventory levels.

Table 2.8 Regression results for the evaluation of the implementation of automatic forecasting using meta-learning

Dependent Var	Sales		Inventory	
Model	(1a)	(1b)	(2a)	(2b)
Treat	0.573* (0.249)	0.531* (0.235)	−4.87* (2.21)	−5.47* (2.36)
Fixed effect				
Product	Yes	Yes	Yes	Yes
Day	Yes	Yes	Yes	Yes
Store	No	Yes	No	Yes
Observations	9,705	9,705	5,470	5,470

2.6 Discussion and Future Research

Modern retailing faces important challenges. The constant increase in product variety and the growing pressure to increase supply chain processes' efficiency have pushed for demand forecasting automation. Recent advances in data analytics offer a wide range of models that can be applied to improve forecasting. However, the suitability of the models depends on the case, and there is no universal best model. With retailers having to plan inventories of thousands of products in hundreds of stores, manually choosing the best forecasting model is costly and can often be inaccurate.

In our research, we present a methodology that takes advantage of recent advances in meta-learning to select the best model for each forecasting task automatically. In this chapter, we describe the methodology and then numerically demonstrate that meta-learning can significantly improve forecasting accuracy. Furthermore, we apply our approach in a controlled experiment and show that replenishment can benefit by reducing inventory levels and increasing sales. From a methodological point of view, it is important to notice that there is a tradeoff between the use of multiple forecasting models and the difficulty in classifying models in the meta-learning phase. In our case, we found that restricting the set of eligible models to only those that perform well on average leads to better overall performance.

To the best of our knowledge, this is one of the first studies showing that meta-learning can provide value in the retail industry. However, we identify several limitations and avenues for future research. First, we concentrate the analysis on only two product categories (clothing and toys) in a single retail chain. Despite expecting that the main findings generalize to other scenarios, more research is needed to understand the boundaries of the application of this technology. Second, in the empirical analysis, we focused on a limited number of forecasting models. Although our list is representative of the most common forecasting approaches, the list can be enhanced with other models, such as gradient boost (Chen and Guestrin 2016) or Prophet (Taylor and Letham 2018). Third, our application only considers Random Forest as a classification technique.

Further analysis could consider the exploration of alternative classifiers such as Naïve Bayes classifiers (Rish 2001) or Support Vector Machines (Pisner and Schnyer 2020). A final idea for future research is to use meta-learning insights to create customized ensembles. Although we considered a statistic ensemble in our work, creating different ensembles depending on the time series features might lead to further improvements in the forecast.

This research illustrates how recent data analytics and automation advances can impact a regional retailer. While the technology is mature enough to impact today, we expect that this type of initiative will continue playing an important role in improving the operational efficiency in the industry and will become part of the standard way of operating shortly.

References

Abraham MM, Lodish LM (1987) Promoter: an automated promotion evaluation system. Mark Sci 6(2):101–123. https://doi.org/10.1287/mksc.6.2.101

Ali ÖG, Sayın S, Van Woensel T, Fransoo J (2009) SKU demand forecasting in the presence of promotions. Expert Syst Appl 36(10):12340–12348. https://doi.org/10.1016/j.eswa.2009.04.052

Bates JM, Granger CW (1969) The combination of forecasts. J Oper Res Soc 20(4):451–468. https://doi.org/10.1057/jors.1969.103

Begley S, Hancock B, Kilroy T, Kohli S (2019) Automation in retail: an executive overview for getting ready. McKinsey & Company Retail Insights

Chatfield C (1978) The holt-winters forecasting procedure. J Roy Stat Soc: Ser C (appl Stat) 27(3):264–279. https://doi.org/10.2307/2347162

Chen T, Guestrin C (2016) Xgboost: a scalable tree boosting system. In: Proceedings of the 22nd ACM sigkdd international conference on knowledge discovery and data mining, pp 785–794. https://doi.org/10.1145/2939672.2939785

Cleveland RB, Cleveland WS, McRae JE, Terpenning I (1990) STL: a seasonal-trend decomposition. J Off Stat 6:3–73

Clouse DS, Giles CL, Horne BG, Cottrell GW (1997) Time-delay neural networks: representation and induction of finite-state machines. IEEE Trans Neural Netw 8(5):1065–1070. https://doi.org/10.1109/72.623208

Collopy F, Armstrong JS (1992) Rule-based forecasting: development and validation of an expert systems approach to combining time series extrapolations. Manag Sci 38(10):1394–1414. https://doi.org/10.1287/mnsc.38.10.1394

De Livera AM, Hyndman RJ, Snyder RD (2011) Forecasting time series with complex seasonal patterns using exponential smoothing. J Am Stat Assoc 106:1513–1527. https://doi.org/10.1198/jasa.2011.tm09771

Fildes R, Ma S, Kolassa S (2022) Retail forecasting: research and practice. Int J Forecast 38(4):1283–1318. https://doi.org/10.1016/j.ijforecast.2019.06.004

Goic M, Levenier C, Montoya R (2021) Drivers of customer satisfaction in the grocery retail industry: a longitudinal analysis across store formats. J Retail Consum Serv 60:102505. https://doi.org/10.1016/j.jretconser.2021.102505

Horváth C, Wieringa JE (2008) Pooling data for the analysis of dynamic marketing systems. Stat Neerl 62(2):208–229. https://doi.org/10.1111/j.1467-9574.2007.00382.x

Huber J, Stuckenschmidt H (2020) Daily retail demand forecasting using machine learning with emphasis on calendric special days. Int J Forecast 36(4):1420–1438. https://doi.org/10.1016/j.ijforecast.2020.02.005

Johnston FR, Boyland JE, Meadows M, Shale E (1999) Some properties of a simple moving average when applied to forecasting a time series. J Oper Res Soc 50(12):1267–1271. https://doi.org/10.1057/palgrave.jors.2600823

Lemke C, Gabrys B (2010) Meta-learning for time series forecasting and forecast combination. Neurocomputing 73(10–12):2006–2016. https://doi.org/10.1016/j.neucom.2009.09.020

Ma S, Fildes R, Huang T (2016) Demand forecasting with high dimensional data: the case of SKU retail sales forecasting with intra- and inter-category promotional information. Eur J Oper Res 249:245–257. https://doi.org/10.1016/j.ejor.2015.08.029

Ma S, Fildes R (2021) Retail sales forecasting with meta-learning. Eur J Oper Res 288(1):111–128. https://doi.org/10.1016/j.ejor.2015.08.029

Macé S, Neslin SA (2004) The determinants of pre-and postpromotion dips in sales of frequently purchased goods. J Mark Res 41(3):339–350. https://doi.org/10.1509/jmkr.41.3.339.359

Mahmoud E, Rice G, Malhotra N (1988) Emerging issues in sales forecasting and decision support systems. J Acad Mark Sci 16(3):47–61. https://doi.org/10.1177/009207038801600308

Newbold P (1983) ARIMA model building and the time series analysis approach to forecasting. J Forecast 2(1):23–35. https://doi.org/10.1002/for.3980020104

Pegels CC (1969) Exponential forecasting: some new variations. Manag Sci 15(5):311–315. https://doi.org/10.1287/mnsc.15.5.311

Pisner DA, Schnyer DM (2020) Support vector machine. In: Machine learning: methods and applications to brain disorders. Academic Press, pp 101–121. https://doi.org/10.1016/B978-0-12-815739-8.00006-7

Prudêncio RB, Ludermir TB (2004) Meta-learning approaches to selecting time series models. Neurocomputing 61:121–137. https://doi.org/10.1016/j.neucom.2004.03.008

Narayanan A, Sahin F, Robinson EP (2019) Demand and order-fulfillment planning: the impact of point-of-sale data, retailer orders and distribution center orders on forecast accuracy. J Oper Manag 65(5):468–486. https://doi.org/10.1002/joom.1026

Rish I (2001) An empirical study of the naive Bayes classifier. In: IJCAI 2001 workshop on empirical methods in artificial intelligence, pp 41–46

Spiliotis E, Makridakis S, Semenoglou AA, Assimakopoulos V (2020) Comparison of statistical and machine learning methods for daily SKU demand forecasting. Oper Res Int J 22:2037–3061. https://doi.org/10.1007/s12351-020-00605-2

Srinivasan S, Pauwels K, Nijs V (2008) Demand-based pricing versus past-price dependence: a cost–benefit analysis. J Mark 72(2):15–27. https://doi.org/10.1509/jmkg.72.2.15

Talagala TS, Hyndman RJ, Athanasopoulos G (2018) Meta-learning how to forecast time series. Monash Econom Bus Stat Work Pap 6(18):1–29

Taylor SJ, Letham B (2018) Forecasting at scale. Am Stat 72(1):37–45. https://doi.org/10.1080/00031305.2017.1380080

Wang X, Smith K, Hyndman R (2006) Characteristic-based clustering for time series data. Data Min Knowl Disc 13(3):335–364. https://doi.org/10.1007/s10618-005-0039-x

Wang X, Smith-Miles K, Hyndman R (2009) Rule induction for forecasting method selection: meta-learning the characteristics of univariate time series. Neurocomputing 72(10–12):2581–2594. https://doi.org/10.1016/j.neucom.2008.10.017

Wu H, Levinson D (2021) The ensemble approach to forecasting: a review and synthesis. Transp Res Part C: Emerg Technol 132:103357. https://doi.org/10.1016/j.trc.2021.103357

Zhou W, Tu YJ, Piramuthu S (2009) RFID-enabled item-level retail pricing. Decis Support Syst 48(1):169–179. https://doi.org/10.1016/j.dss.2009.07.008