Using Meta-Learning in Automatic Demand Forecasts with a Large Number of Products

Preprint · July 2022

CITATIONS
0

READS
50

2 authors, including:

Marcel Goic
University of Chile
30 PUBLICATIONS **280** CITATIONS

SEE PROFILE

Using Meta-Learning in Automatic Demand Forecasts with a Large Number of Products

Luis Gutiérrez (luisgutierrez@uchile.cl)
Marcel Goic (mgoic@uchile.cl)

July 2022

Abstract

Demand analysis is one of the cornerstones of any supply chain management system, and most of the key operational decisions in the supply chain rely on accurate demand predictions. Although there is a large body of academic literature proposing a variety of forecasting methods, there are still important challenges when using them in practice. A common problem is that firms need to decide about thousands of products and the patterns of demand could be very different between them. In this setting, oftentimes 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 previous practices of the firm leading to important efficiency gains in the supply chain.

Keywords: Forecasting, Meta-Learning, Time Series, Retailing.

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, and the increasing use of technology in the value chain (Goic and Olivares, 2019). Among the long-term trends that have consolidated in recent years is the automation of a variety of 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 is going to be sold in the near future in each store. Accurate forecasting has important consequences for operation performance. If the forecast underestimates the demand, the products will be out of stock, having a negative impact on 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, Fildes, Huang, 2016; Huber, 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 in some cases a simple autoregressive model can provide satisfactory solutions, 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 key processes without sacrificing forecasting accuracy.

The need of using different forecasting models comes from the existence of distinct components in different demand series. To illustrate the point, in Figure 1, we display the time series of sales of four different products. For product A, we observe very pronounced spikes in the 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 clearly higher in the second part of every year, but that is mostly associated with year seasonality and not to a single event. This pattern is fairly common to items in the clothing category where the demand tends to be very cyclical. In this set, we also have products with no clear seasonal patterns such as products C and D. On one hand, product C presents large

variations in sales, but 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.

Figure 1: Illustration of several time series with different seasonality and trend components.

Overall, we observe series with very different components requiring different modeling approaches. To automate the forecast, we need to provide an adequate estimation for every case. However, some models provide better results in some cases and other models perform better in other cases. 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 a large number of forecasts using different models to identify which one 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 in which we compare sales and inventory levels for products in which we use the methodology to decide product replenishment, against a control where orders were decided using the standard business practices. Here we found that the model can indeed improve

operational efficiency in practice. To summarize, in this project we developed a predictive model to generate an accurate automatic forecast for a wide variety of products, thus reducing logistics and inventory management costs in the supermarket industry.

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

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, Ma, and Kolassa, (2019) pose that strategic, tactical, and operational decisions require different methods and data aggregation levels. In this work, we focus on providing forecasting at the product and store levels to support operational decisions such as order sizes and inventory volumes. Since the introduction of retail scanner data, a wide variety of 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, Pauwels, and Nijs, 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 particular 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-stores combinations.

The desire of having estimation methods that can be generalized to multiple prediction instances has a long tradition in the forecasting literature. More than 30 years ago,

Mahmoud, Rice, and Malhotra (1988) already posed that *no one sales forecasting method is appropriate for every situation* (p 54). While the problem has been identified a long time ago, it has not been 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 (see, for example, Horvath 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). In our empirical analysis, we consider ensembles as potential candidates to generate the best predictions, but we consider the possibility that one model by itself could be the most adequate for certain 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 adequate model for a given time series (Prudencio and Ludermir, 2004). Unlike ensemble learning, which combines multiple forecasts, in meta-learning, we aim to select the best model for each case. 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, Hyndman, and Athanasopoulos, 2018). Using a wide range of univariate time series from different domains, Wang, Smith-Miles, and Hyndman (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) apply meta-learning methods in a retail setting and demonstrate they can significantly improve forecasting efficacy. Although they evaluate meta-learning using a publicly available dataset, in our work 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, Hancock, Kilroy, and Kohli, 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, and Piramuthu, 2009), and the delivery of automatic responses triggered by consumer actions (Goic, Rojas and Saavedra, 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.

3 Methodology

As illustrated in Figure 2, the proposed methodology consists of four main steps. First, we produce forecasts for a large number of cases using a variety of models and compute error metric for each model and case (1). Second, we generate several features to characterize each case (2), and third, we use those features to train a meta-learning model that indicates 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).

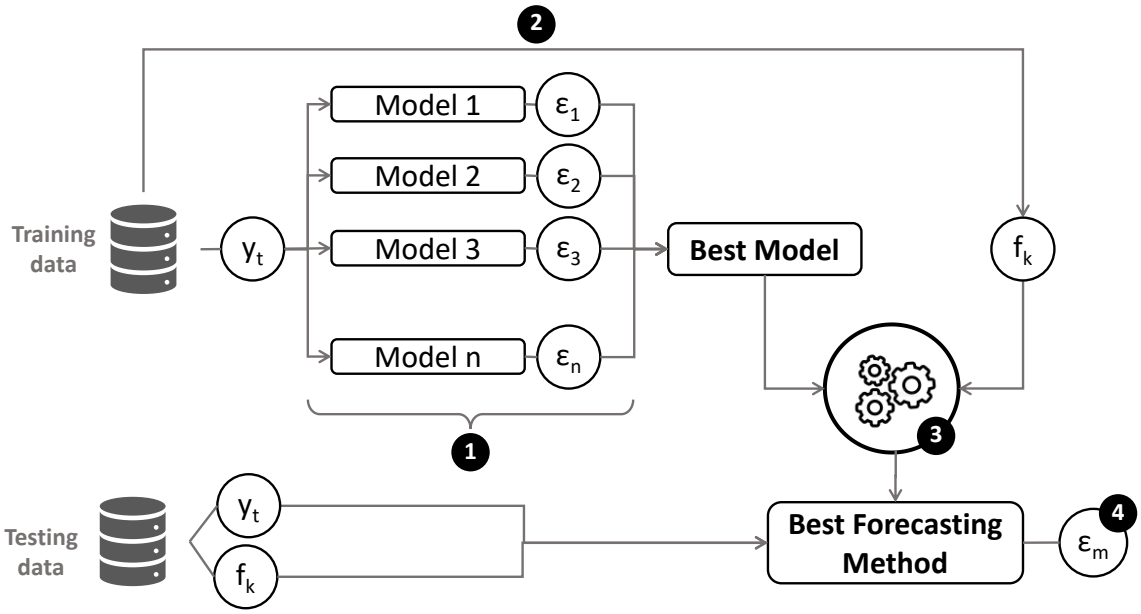

Figure 2: Schematic representation of the proposed methodology.

In the next subsections, we discuss each of these components in more detail.

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 there is no single model that generates the most accurate prediction for a majority of cases, which provides empirical justification for the inclusion of 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. 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, Boyland, Meadows, and Shale, 1999).
- **Autoregressive Integrated Moving Average (ARIMA):** The values of the time series on a given period depend on their lagged values and lagged errors. To estimate the model, the series are further differentiated to allow for 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 automatically accommodate multiple seasonal components. Each of these seasonal components is modeled by a trigonometric representation based on a Fourier series (De Livera, Hyndman, Snyder, 2011).
- **Time-delay artificial 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, Giles, Horne, and Cottrell 1997).
- **Seasonal-Trend decomposition using LOESS (STL):** This model allows for the decomposition of the time series into three components including seasonality, trend, and residuals. To combine these components, this model uses a local regression approach that is robust to outliers (Cleveland, Cleveland, McRae, Terpenning, 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).

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 and, 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 gotten from the extensive forecasting with alternative models that we explained in the previous subsection. The process to extract time-series features is explained in depth in the next subsection.

Using, standard supervised learning approaches, we split the database into training and testing subsets. 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, Hyndman, and Athanasopoulos, 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 performance of the classifier.

The labels indicating which candidate is the best model are based on the Mean Absolute Error (MAE). Since the label are used to guide which model performs better for each shape of the time series, 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.

There is another variation in the classifier that proved to significantly enhance meta-learning. This is, 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 performance overall. 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 any of the models will necessarily 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 here 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 the meta-learning with the best models leads to better results overall.

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.

To define the list of time-series features to use in the empirical analysis, we closely follow previous literature on time-series meta-learning and 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, Hyndman, Athanasopoulos 2018; Ma and Fildes, 2021). To illustrate how different time series differ depending on the values of these features, in Figure 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.

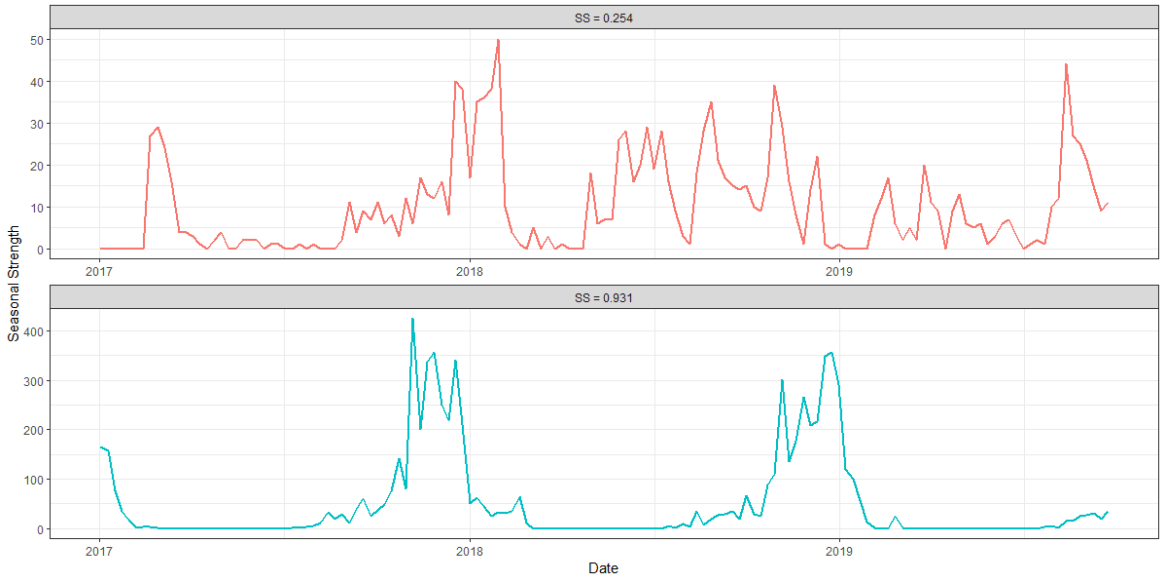

Figure 3: Visual representation of two series of sales with different Seasonal Strength.

3.4 Execution and Evaluation of Meta-models

To complete the methodology, we apply the classifier for a large number of products and then we evaluate to which extent the resulting predictions improve with respect to standard forecasting tools. Considering that we apply the proposed methodology to a large number of products, we need an aggregated metric of performance. 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 not only justified because of its scale independence, but also because it is consistent with the business objective of having more accurate predictions for those products with a larger impact on revenues (Narayanan, Sahin and Robinson, 2019).

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 through the territory. On one hand, and considering the limited storage space in the store, demand overestimation could lead to large operational costs. On the other hand, demand underestimation could lead to lost sales due to out of stocks. While we formally analyze the inventory reorder process, the forecast could also be used to support other decisions such as assortment or promotional planning.

In the empirical evaluation, we consider 5,000 demand series of different product-store combinations. 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 1, we display descriptive statistics of the demand for both product categories.

		Weekly Sales [units]		Price [CLP]	
Product Category	No Products	Mean	Max	Mean	Max
Toys	297	24.9	903	5,186	172,914
Clothing	4,703	40.5	4,355	3,221	170,540

Table 1: Descriptive statistics of the demand series for different product-stores combinations. Prices are reported in the Chilean currency (CLP).

Statistics from Table 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 the purpose of 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.

5 Results

According to the methodology presented in Section 3, there are several components that 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 basic inputs for the meta-learning stage that

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

5.1 Forecasting through standard models

We first estimate each of the seven forecasting models for each of the 5,000 time series to complete a total of 35,000 forecasting tasks. The majority of these models require the calibration of hyper-parameters. For instance, for TBATS we need to determine if Box-Cox transformation is required, or for the ARIMA models, 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 4 weeks of the time series. This forecasting window is chosen to match the typical target for inventory reorders. In Figure 4, we illustrate the forecasts of all individual methods for a selected time series. Although the series largely differ in terms of their features (trend, seasonality, spikiness, etc.), this example represents a common pattern we find in most of the 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 what is the best model for a given series. Beyond the illustration of a given series, Table 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 that we use to compare predictions between models for a given series, and the wMAPE that 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.

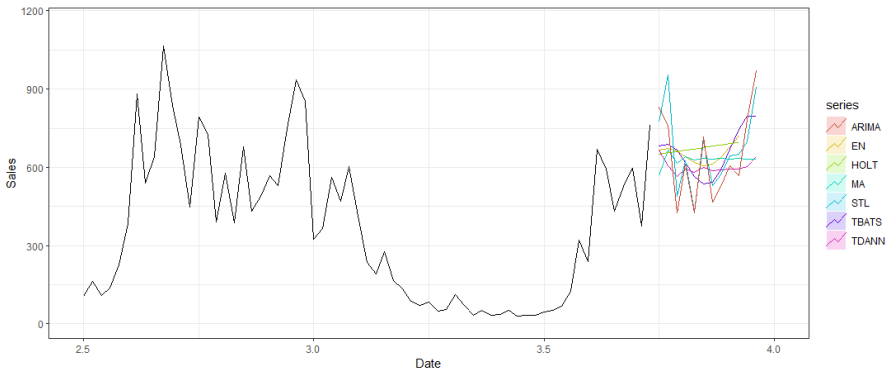

Figure 4: Illustration of alternative forecasting results for a selected series.

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%

Table 2: Forecasting Error across models for the first week.

To complement previous results, in Table 3 we display the forecasting errors for all four weeks we use in these numerical exercises. As expected, the further in the future the forecasting window is, the lower the accuracy of the prediction. However, the notion that the differences between models are small, remains.

Model	w1	w2	w3	w4	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

Table 3: Forecasting errors by week.

Recall that in our methodology, the forecasting results from individual models are used to calibrate a classification model that determines what is 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. To produce these labels, we use the smallest forecasting error for each case. The frequencies of these labels are displayed in Figure 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 model is the best alternative for the majority of the cases. It is possible, however, that a particular model could be consistently better, but only by a small margin. The results in Figure 5 indicate that this is indeed not the case and that some models work well for some series and other models 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.

Figure 5: Fraction at which each model provides the minimum forecasting error.

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 which make it difficult to produce the very best solution.

5.2 Generation of Features

As explained in the methodology section, 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 4, we display the whole list of the time-series features we use for meta-learning along with their corresponding

descriptive statistics. For a more complete study of feature extraction, see Wang, Smith, and Hyndman (2006).

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 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

Table 4: List of time-series features for meta-learning with the corresponding descriptive statistics for the case study.

According to the descriptive statistics presented in Table 4, except for *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 in each model.

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 perform the classification, 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 certainly benefit from selecting from a larger set of models. However, more candidates make the classification task more difficult, 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 Figure 6 we display the mean value for all time-series features depending on which was the model with the best performance. According to these results, we corroborate that some models tend to perform better for certain 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. Overall, these results provide preliminary evidence that meta-learning can be effective in identifying the underlying patterns connecting time-series features and model performance.

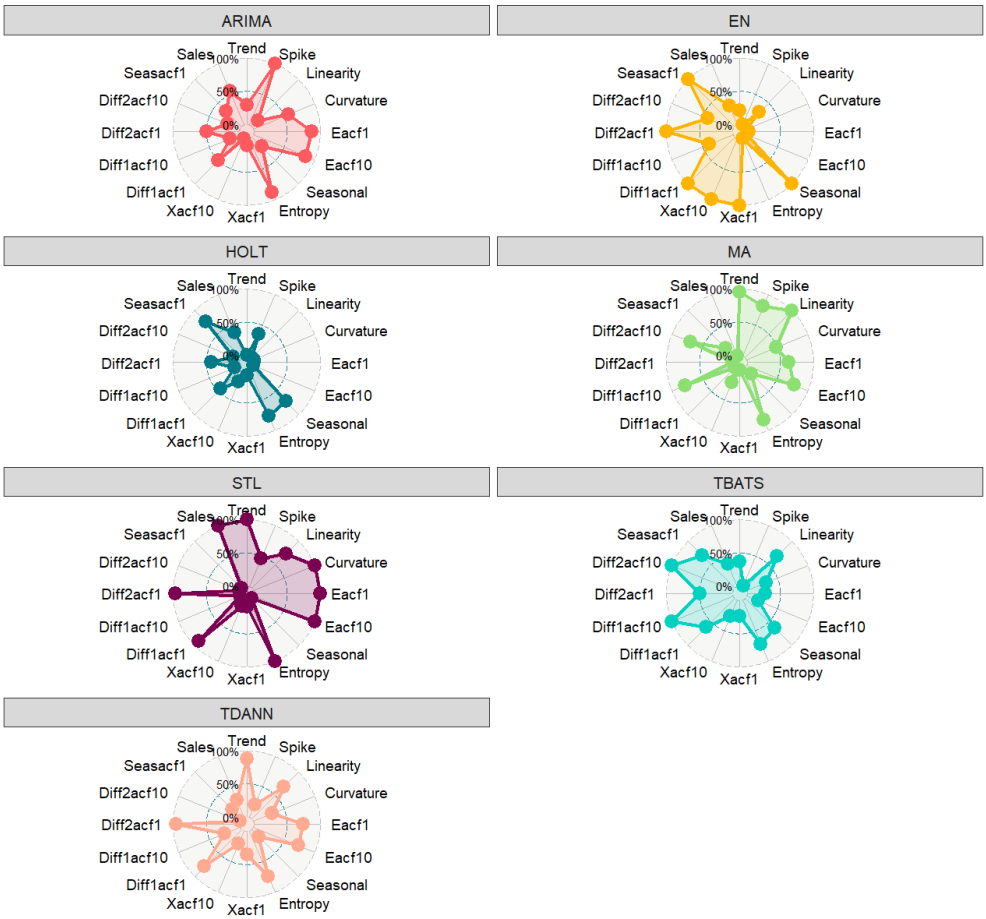

Figure 6: mean attribute value depending on which is the preferred model.

Classification with all models

In this first exercise, the meta-learning must decide the best model among seven competing alternatives. Table 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).

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 5: Performance of meta-Forecast against individual models.

Results from Table 6 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 have a positive impact on the overall performance of the system. 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 oftentimes the best solution, the classifier could make important 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.

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 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).

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%**

Table 6: Performance of Meta-Forecast against individual models.

Compared to the previous case, this new meta-forecaster leads to much better results and it overperforms all other models in all relevant metrics. In fact, the meta-forecast model

not only provides a significant reduction in average error metrics with a wMAPE of 27.1%, which is 4.2 percentage points better than the closest competitor (Ensemble) and more than 18 percentage 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 add a meaningful boost of accuracy to make better predictions in the case of 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.

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 adequate model to produce the estimation for each combination of product and stores. Furthermore, the resulting forecasts could even lead to more accurate predictions. In this section, we empirically test if these improvements can be effectively applied in a real setting and we evaluate their impact on relevant business metrics.

To measure the impact of the forecasting automation, we evaluate their impact in the process of product replenishment that requires estimation of 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 continue their replenishment processes using the standard business practices. While in the treatment, we forecast the demand using the automatic meta-learner, in the 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 selection of treatment and control groups was made to have similar demand levels pre-treatment and the experiment lasted two weeks.

Certainly, the automation of the forecasting process brings several benefits that can only be observed in the mid-term. Those include more consistent decision-making, fastest processing, and cost saving 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 by itself can increase the demand, a more precise forecast should be associated with a smaller number of out-of-stocks and therefore have a positive effect on sales levels. Table 7 reports the daily mean for sales and inventory for this experiment.

	Treatment	Control
Sales	277.1	250.5
Inventory	18644.4	19096.6

Table 7: Daily mean of sales inventory between treatment and control conditions.

The treatment and control groups were selected to be balanced and therefore, the larger sales and smaller inventory in the treatment provide preliminary evidence that the forecast can have a positive effect on 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:

$$sales_{ist} = \alpha_i^1 + \beta_s^1 + \gamma_t^1 + \delta^1 \cdot \mathrm{Treat}_{ist} + \varepsilon_{ist}^2 \quad (1)$$

$$inventory_{ist} = \alpha_i^2 + \beta_s^2 + \gamma_t^2 + \delta^2 \cdot \mathrm{Treat}_{ist} + \varepsilon_{ist}^2 \quad (2)$$

The key variable in this regression is $\cdot\mathrm{Treat}_{ist}$ 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. Results of the regression models are displayed in Table 8. In the table, we include two versions of the equation (1) and (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 recollection of inventory and therefore we only observe a fraction of them (N=5,470).

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

Table 8: Regression results for the evaluation of the implementation of automatic forecasting using meta-learning.

Results from Table 9, 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.

6 Discussion and Future Research

Modern retailing faces important challenges. The constant increase in product variety and the growing pressure to increase the efficiency of supply chain processes have pushed for automation in demand forecasting. 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 automatically select the best model for each forecasting task. In this chapter, we describe the methodology and then we numerically demonstrate that meta-learning can have a meaningful impact on improving forecasting accuracy. Furthermore, we apply our approach in a controlled experiment and show that the replenishment process can benefit from it by reducing inventory levels and increasing sales. From a methodological point of view, it is important to notice that there is a trade-off between the use of multiple forecasting models and the difficulty in classifying models in the meta-learning phase. In particular, 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 we consider 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, in our application, we only consider 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 the insights of meta-learning to create customized ensembles. Although in our work we did consider a statistic ensemble, the creation of different ensembles depending on the features of the time series might lead to further improvements in the forecast.

To conclude, in this research we have illustrated how recent advances in data analytics and automation can have a real impact on a regional retailer. While the technology is mature enough to have an impact today, we expect that this type of initiatives will continue playing an important role in improving the operational efficiency in the industry and they will become part of the standard way of operating in the near future.

7 References

Abraham, M. M., & Lodish, L. M. (1987). Promoter: An automated promotion evaluation system. *Marketing Science*, 6(2), 101-123.

Ali, Ö. G., Sayın, S., Van Woensel, T., & Fransoo, J. (2009). SKU demand forecasting in the presence of promotions. *Expert Systems with Applications*, 36(10), 12340-12348.

Aburto, L., & Weber, R. (2007). Improved supply chain management based on hybrid demand forecasts. *Applied Soft Computing, 7(1), 136-144.*

Bates, J. M., & Granger, C. W. (1969). The combination of forecasts. *Journal of the operational research society*, 20(4), 451-468.

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. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 27(3), 264-279.

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).

Cleveland, R. B., Cleveland, W. S., McRae, J. E., & Terpenning, I. (1990). STL: A seasonal-trend decomposition. *Journal of Official Statistics*, 6(1), 3-73.

Clouse, D. S., Giles, C. L., Horne, B. G., & Cottrell, G. W. (1997). Time-delay neural networks: Representation and induction of finite-state machines. *IEEE Transactions on Neural Networks*, 8(5), 1065-1070.

Collopy, F., & Armstrong, J. S. (1992). Rule-based forecasting: Development and validation of an expert systems approach to combining time series extrapolations. *Management science*, 38(10), 1394-1414.

De Livera, A. M., Hyndman, R. J., & Snyder, R. D. (2011). Forecasting time series with complex seasonal patterns using exponential smoothing. *Journal of the American statistical association*, 106(496), 1513-1527.

Fildes, R., Ma, S., & Kolassa, S. (2019). Retail forecasting: Research and practice. *International Journal of Forecasting* (in press).

Goic, M., & Olivares, M. (2019). Omnichannel analytics. In *Operations in an Omnichannel World* (pp. 115-150). Springer, Cham.

Goic, M., Levenier, C., & Montoya, R. (2021). Drivers of customer satisfaction in the grocery retail industry: A longitudinal analysis across store formats. *Journal of Retailing and Consumer Services*, 60, 102505.

Horváth, C., & Wieringa, J. E. (2008). Pooling data for the analysis of dynamic marketing systems. *Statistica Neerlandica*, 62(2), 208-229.

Huber, J., & Stuckenschmidt, H. (2020). Daily retail demand forecasting using machine learning with emphasis on calendric special days. *International Journal of Forecasting*, 36(4), 1420-1438.

Johnston, F. R., Boyland, J. E., Meadows, M., & Shale, E. (1999). Some properties of a simple moving average when applied to forecasting a time series. *Journal of the Operational Research Society*, 50(12), 1267-1271.

Kang, Y., Hyndman, R. J., & Li, F. (2020). GRATIS: GeneRAting TIme Series with diverse and controllable characteristics. *Statistical Analysis and Data Mining: The ASA Data Science Journal, 13(4), 354-376.*

Lemke, C., & Gabrys, B. (2010). Meta-learning for time series forecasting and forecast combination. *Neurocomputing, 73(10-12), 2006-2016.*

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. *European Journal of Operational Research*, 249(1), 245-257.

Ma, S., & Fildes, R. *(2021).* Retail sales forecasting with meta-learning. *European Journal of Operational Research, 288(1), 111-128.*

Macé, S., & Neslin, S. A. (2004). The determinants of pre-and postpromotion dips in sales of frequently purchased goods. *Journal of Marketing Research*, 41(3), 339-350.

Mahmoud, E., Rice, G., & Malhotra, N. (1988). Emerging issues in sales forecasting and decision support systems. *Journal of the Academy of Marketing Science*, 16(3), 47-61.

Newbold, P. (1983). ARIMA model building and the time series analysis approach to forecasting. *Journal of forecasting*, 2(1), 23-35.

Pegels, C. C. (1969). Exponential forecasting: Some new variations. *Management Science*, 311-315.

Pisner, D. A., & Schnyer, D. M. (2020). Support vector machine. In *Machine learning* (pp. 101-121). Academic Press.

Prudêncio, R. B., & Ludermir, T. B. (2004). Meta-learning approaches to selecting time series models. *Neurocomputing*, 61, 121-137.

Narayanan, A., Sahin, F., & Robinson, E. P. (2019). Demand and order-fulfillment planning: The impact of point-of-sale data, retailer orders and distribution center orders on forecast accuracy. *Journal of Operations Management*, 65(5), 468-486.

Rish, I. (2001). An empirical study of the naive Bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence* (Vol. 3, No. 22, pp. 41-46).

Spiliotis, E., Makridakis, S., Semenoglou, A. A., & Assimakopoulos, V. (2020). Comparison of statistical and machine learning methods for daily SKU demand forecasting. *Operational Research*, 1-25

Srinivasan, S., Pauwels, K., & Nijs, V. (2008). Demand-based pricing versus past-price dependence: a cost–benefit analysis. *Journal of Marketing*, 72(2), 15-27.

Talagala, T. S., Hyndman, R. J., & Athanasopoulos, G. *(2018).* Meta-learning how to forecast time series*. Monash Econometrics and Business Statistics Working Papers, 6(18), 16.*

Taylor, S. J., & Letham, B. (2018). Forecasting at scale. *The American Statistician*, 72(1), 37-45.

Wang, X., Smith, K., & Hyndman, R. (2006). Characteristic-based clustering for time series data. *Data mining and knowledge Discovery*, 13(3), 335-364.

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.

Wu, H., & Levinson, D. (2021). The ensemble approach to forecasting: A review and synthesis. *Transportation Research Part C: Emerging Technologies*, 132, 103357.

Zhou, W., Tu, Y. J., & Piramuthu, S. (2009). RFID-enabled item-level retail pricing. *Decision Support Systems*, 48(1), 169-179.