Cross-Item Learning for Volatile Demand Forecasting: An Intervention with Predictive Analytics

Howard Hao-Chun Chuang
College of Commerce, National Chengchi University
Taipei 11605, Taiwan
chuang@nccu.edu.tw

Yen-Chun Chou
College of Commerce, National Chengchi University
Taipei 11605, Taiwan
yenchun@nccu.edu.tw

Rogelio Oliva*
Mays Business School, Texas A&M University
College Station, TX 77843
roliva@tamu.edu
*Corresponding Author

Please cite as:

Chuang, HHC, YC Chou, and R Oliva. 2021. Cross-item learning for volatile demand forecasting: An intervention with predictive analytics. *Journal of Operations Management* (forthcoming).

Cross-Item Learning for Volatile Demand Forecasting: An Intervention with Predictive Analytics

Abstract

Despite its importance to OM, demand forecasting has been perceived as a "problem-solving" exercise; most empirical work in the field has focused on explanatory models but neglected prediction problems that are part of empirical science. The present study, involving one of the leading electronics distributors in the world, aims to improve prediction accuracy under high demand volatility for procurement managers to make better inventory decisions. In response to requests for an integrated forecasting methodology, we undertook an iterative process based on three guiding principles — *data pooling*, *theory-informed feature engineering*, and *ensemble-based machine learning*. The resulting framework managed to improve forecast accuracy significantly and is applicable to a broad range of situations. We present reflections and insights derived abductively through engagement with managers in this problem situation. This "problem-driven" process corresponds to intervention as a research strategy that can foster theoretical and methodological innovations in OM. Our contribution goes beyond the development of the prediction framework as it elucidates ways OM researchers could leverage theoretical foundations to inform feature derivation and model construction. We posit that this work points to a way forward to the combination of OM principles with the emerging innovations in data science and artificial intelligence.

Key words: demand forecasting; lead time; feature engineering; machine learning; data analytics; intervention-based research.

1 Introduction

Forecasting demand over lead time, that is, total demand between the current period and anticipated period of order delivery, is a perennial problem in operations management (OM) given the importance of demand estimation to inventory control and related planning and sourcing decisions (Fildes et al. 2008, Syntetos et al. 2016). Classical inventory models often assume decision-makers possess knowledge about the distribution or model structures of stochastic demand, and parameters of the presumed distribution to be estimable given demand observations.

Such distributional and functional assumptions are, however, often violated in real world operations, a problem exacerbated in the electronics and semiconductor supply chains characterized by short product life cycles and non-stationary demand. The present study addresses a challenging demand forecasting problem.

Our research partner is one of the largest electronics distributors in the world, henceforth referred as *Alpha*. *Alpha* procures from global suppliers (e.g., Intel, Texas Instruments) semiconductor components that are warehoused and eventually distributed in response to production demands of electronics manufacturing services (EMS) companies mostly based in East Asia (e.g., Foxconn, Quanta Computer). Despite massive sales revenue (US$18 billion in 2018), *Alpha's* profit margins are tight, compromised by high inventory holding and obsolescence costs primarily attributable to seemingly unpredictable demand; inventory and accounts receivable (i.e., recently sold inventory) represent, respectively, 33% and 49% of total assets. How to more accurately predict demand is a salient problem for *Alpha*, whose financial performance is contingent on stocking decisions that are based on demand forecasts.

The demand forecasting problem *Alpha* faces is common to many firms in high-tech manufacturing sectors (Fu and Chien 2019). *Alpha*, like these other firms, conducts a periodic review inventory management system. Procurement managers review every week, for each component, in-warehouse and in-transit stocks and determine order quantities based on a forecast for demand over supply lead times. Although more accurate demand predictions would afford procurement managers better anchoring points for improving inventory decisions, generating reliable forecast for demand over lead time proves unexpectedly complex. Demand observations in manufacturing supply chains are typically intermittent, lumpy, or erratic (Syntetos et al. 2005), that is, items tend to exhibit hard to predict consecutive zero realizations and spikes for non-zero realizations (Syntetos et al. 2016). Some firms even consider production demand with isolated spikes and frequent zeros to be non-forecastable (Bachman et al. 2016).

High demand uncertainty being the norm in this industry sector, demand forecasting risk increases over long lead times (de Treville et al. 2014), resulting in high safety stocks that translate into financial strain. Supply side inventory control could be improved if the downstream EMS firms were able to share reliable advance demand information in the presence of highly variable demand, that is, a rolling forecast for future production demand (Terwiesch et al. 2005). The rolling forecast received by *Alpha*, however, is noisy and deemed unreliable by procurement executives.

In the aggregate, these issues – long supply lead time, unknown demand structures, and noisy and unreliable rolling forecast signals – make the demand forecasting problem practically as well as theoretically challenging.

In response to requests for an integrated forecasting methodology for volatile demand patterns, we undertook with *Alpha* managers an iterative process of developing and refining an approach for tackling this recurring forecasting problem. Our test case required us to forecast the demand of 426 items required by multiple plants of a major EMS client. Our initial attempt revealed that individual time series methods grounded in historical demand could not outperform a simple baseline (moving average) in the presence of intermittent and erratic demand patterns. Further conversations with management revealed the need for a robust method that could leverage information across items. This motivated us to reframe the prediction problem and pool demand data temporally and across items into a single model, as opposed to fitting individual-item models, to learn common functional relationships for the prediction tasks. The aggregation of items into a single model, i.e., *cross-item learning*, is methodologically robust (Ban et al. 2019, Wu et al. 2021) and becoming increasingly popular (Bojer and Meldgaard 2021). Our first iteration tested the single model approach using a machine learning algorithm to exploit data features from relevant demand forecasting theories. Results, albeit encouraging, barely matched the performance of the baseline model. Our second iteration incorporated rolling forecast data features based on operational understanding of supply chain physics. Further improvements were achieved in the third iteration with enhancements to the machine learning methods to address forecast robustness. Satisfied with results in the test case, we tested the proposed methods on a broader data set from an EMS client with more heterogeneous items and longer lead time. The multi-week test corroborated the value and applicability of cross-item learning. The test of multiple EMS clients also helped validate the parsimony and appropriateness of the feature set made available to the algorithms. We further evaluated as a potential avenue for improvement the relative importance of demand and forecasting signals and explored all demand patterns for which our methods failed to yield accurate forecasts. This concluding reflection identified recurring challenges currently being used by management to improve customer communication channels.

The present study shows that it is possible to improve prediction performance through active engagement with managers in the problem situation (problem owners) when established demand forecasting methods fail to perform. This "problem-driven" process corresponds to intervention

(i.e., confrontation of a situation with a theoretical framework via a methodology) as a strategy for fostering research innovations in OM (Oliva 2019). The iterative process went beyond solving a problem instance and yielded insights based on three guiding principles — *data pooling for cross-item learning*, *theory-informed feature engineering*, and *ensemble-based machine learning*. By enhancing predictive analytics protocols with OM principles, our intervention managed to improve forecast accuracy and, without deterioration of sales or service levels, reduce inventory holdings for the focal EMS by 5%, a reduction of inventory value of approximately USD$3 million. With the success application to the focal EMS and the verified effectiveness for a second EMS (see §4), *Alpha* executives introduced the new forecasting methodology to other major EMS clients that together constitute more than $10 billion annual sales (56% of total sales).

Our resulting solution is closely related to efforts by the forecasting community that has begun to leverage machine learning (e.g., Barker 2020, Gilliland 2020, Montero-Manso et al. 2020). Although our modeling effort is not the first to apply machine learning to time series demand forecasts (e.g., Carbonneau et al. 2008, Gutierrez et al. 2008, Hill et al. 1996), our integrated forecasting approach (as opposed to fitting individual time series models) is distinct in that we introduce several techniques (e.g., cross-item learning, model stacking) that have proven to be useful on separated occasions and relatively new to the OM community. Our framework allows for adoption of as many predictor variables as desirable and does not require knowledge of the underlying structure of demand processes. We show that cross-item learning improves performance by itself and its efficacy can be substantially enhanced by a wider set of features. More importantly, our modeling tactics guided by operational and statistical principles can be transferred to other OM settings. The core contribution of our intervention to empirical OM lies in articulating and validating the value that OM principles offer in informing those computational learning techniques. Specifically, our paper elucidates how OM researchers can leverage theoretical understanding of forecast settings (e.g., long lead time with access to a rolling forecast signals) to develop sensible, effective data pooling and feature engineering, indispensable steps often overshadowed by the allure of learning algorithms. While data and feature engineering are crucial to the success of any business predictive analytics initiative empowered by machine learning, it is our OM contextual and theoretical knowledge that enable us to generate useful features for prediction.

The remainder of the paper describes chronologically our intervention and development effort

and closes with analysis of the theoretical and methodological implications of our results. In §2, we describe the problem situation and examine past demand observations as well as rolling forecast signals from an EMS client. An unsuccessful improvement attempt based on established models and methods is reported in §3.1. In the remainder of §3, we show iterations through which the approach substantially improved prediction accuracy. In §4, we corroborate the effectiveness of the proposed method by testing data sets from a different EMS client, perform diagnostics to identify possible improvements, and report on the deployment strategy and initial results. In §5, we reflect on the intervention process, articulate theoretical and methodological insights derived from it, and assess the generalizability and transferability of those insights. We conclude by summarizing our contribution and articulating directions for future studies.

2 Problem Situation

Our intervention is aimed at helping *Alpha* predict downstream EMS production demand for semiconductor components over a stable lead time L (ten weeks) from vendors.[1] We analyze 52 weekly demand observations over a full calendar year for 426 semiconductor items requested by a major EMS client with eleven production plants located in southeast China. Each item is unique to the requesting plant and all items are specific to the focal EMS and not usable for other of *Alpha's* EMS clients (i.e., no pooling of components across clients). Note that the 426 items represent *all* the items supplied by *Alpha* to the focal EMS, i.e., our selection of items was not a sample. *Alpha* managers turned down our suggestion to focus the analysis based on an ABC item classification arguing that regardless of cost, all items were equally relevant and operationally related for production planning. Failure of delivery of inexpensive items could jeopardize overall shipments by the EMS. Nevertheless, even though these were supplies for manufacturing, *Alpha* could not infer production demand dependencies as most EMS clients pull items for their production schedule from multiple distributors and without revealing the full bill of materials of their products[2].

[1] The production lead time of semiconductor components lies in general between ten and twelve weeks. Lead time, however, varies item by item and occasional shortages may extend the lead time for certain items. *Alpha* takes the stance that forecasting over normal identical lead times suffices procurement decision support purposes.

[2] Among the 426 items, only 44 (0.05%) pairs exhibited absolute pairwise correlation greater than or equal to 0.8 for the 52-weeks demand observations. For each of the highly correlated pairs, we estimated the correlation coefficients of the first and second half of the year sample. The two vectors of 44 coefficients had a correlation of only -0.12,

Figure 1 presents an example of demand observations for five representative items over 52 weeks. Demand realizations tend to be sparse (i.e., a series of zeros before a large order) and non-stationary. We were informed that other EMS clients exhibit similar demand patterns.

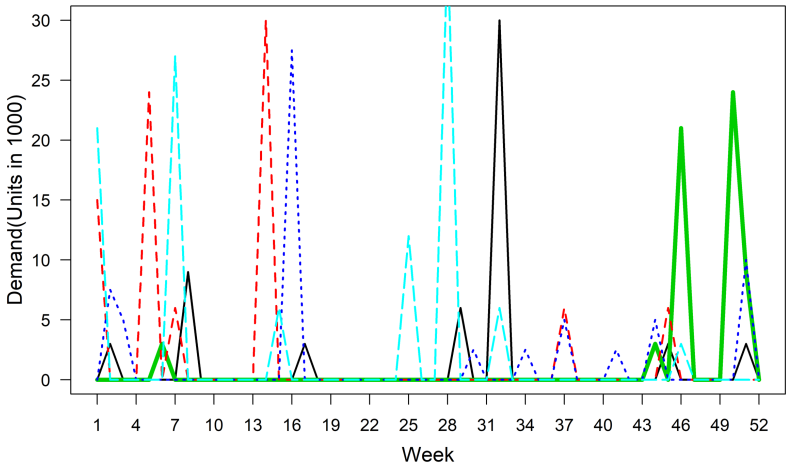

Figure 1: Example demand patterns

We categorize the 426 weekly demand series into smooth, intermittent, erratic, and lumpy based on metrics proposed by Syntetos et al. (2005), (i) average interval between non-zero demand realizations (p), and (ii) the square of the coefficient of variation of a demand series (CV^2). Syntetos et al. (2005) also suggest cut-off values (p=1.32 and CV^2=0.49) widely used by subsequent studies for demand classification (e.g., Boylan et al. 2008, do Rego and de Mesquita 2015). A large p indicates low demand frequency (i.e., more zeros), a large CV^2 high demand volatility. Figure 2 illustrates the categorization of demand observations. Of 426 items in our sample, only 60 (14.1%) belong to the *smooth* category; the remaining 366 fall into the intermittent (211 items, 49.5%), erratic (37 items, 8.7%), and lumpy (118 items, 27.7%) categories, which exhibit highly skewed demand distributions inherently difficult to predict (der Auweraer et al. 2019).

confirming non-stable dependencies in the observed demands.

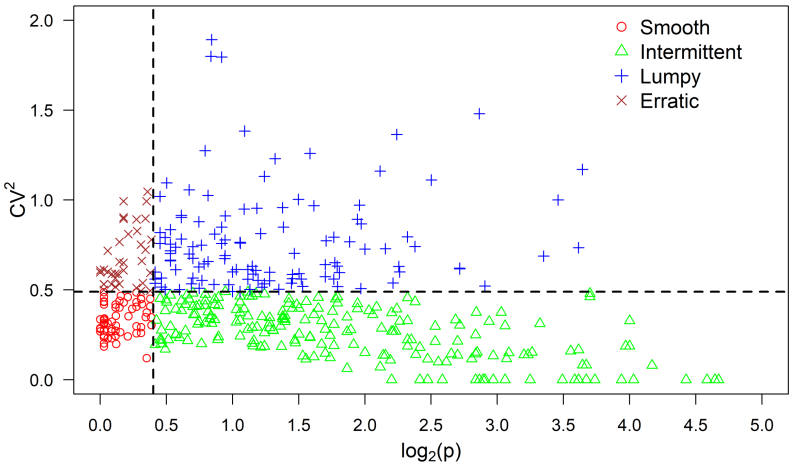

Figure 2: Demand interval and volatility of 426 items

In addition to past demand realizations observed by the distributor, the downstream EMS client provides rolling forecast signals as leading indicators of future demand. Each week, in addition to placing the final order for the week, the EMS client provides a forecast of expected demand for the ten upcoming weeks – the lead time L for *Alpha*'s inventory planning. Using the Martingale Model of Forecast Evolution (MMFE) method (Heath and Jackson 1994), we calculate forecast evolution vectors (i.e., differences between forecast signals released at week t and week t-1) for the 426 items in our sample. For each item, we compute L vectors of forecast evolution, one for every week of the lead time L, and fit a multivariate normal distribution that, per the MMFE, should serve as the basis for simulating demand over the future lead time, conditioning on the most recently available rolling forecast information.

Unfortunately, the computed vectors violate key MMFE assumptions. First, according to the Henze-Zirkler and Royston tests (Korkmaz et al. 2014), none of the 426 items' forecast evolutions exhibit multivariate normality (at α=0.01 level). More important, the assumption that downstream forecasts improve over time (i.e., a forecast made one week ago should be closer to actual demand than a forecast made L weeks ago) also fails to hold. Table 1 reports the average mean absolute error (MAE) for all items for the forecast made one, five, and ten weeks before. As can be seen in the table, $|Error_i(1)|$ has a larger median, interquartile range, and maximum; clearly the most recent forecast is not closer to demand than forecasts released five or ten weeks before. This quick analysis confirms a statement by one of the procurement executives when we first requested the rolling forecast data: "[T]he rolling forecast signals are way too noisy and have no value to us. We would rather ignore those troublesome data in forecasting processes."

Table 1 Mean absolute value of rolling forecast errors across all items

	Min	25%tile	Median	Mean	75%tile	Max
$\lvert Error_i(1)\rvert$	48.7	2,208.1	9,292.5	27,482.7	26,518.3	575,163.7
$\lvert Error_i(5)\rvert$	80.3	2,210.0	7,547.8	20,664.7	21,435.3	415,839.7
$\lvert Error_i(10)\rvert$	96.2	2,023.5	6,913.5	19,696.9	19,871.0	433,883.3

The foregoing issues and responses pushed us to focus on developing forecast models based on past demand observations. Surprisingly, *Alpha* employed a simple, perhaps naïve approach to develop demand forecasts over lead time on a weekly basis: it computed an eight-week moving average (8wMA) of past weekly observations and multiplies by L weeks as the point estimate (Trapero et al. 2019a) for future demand over lead time. Internal analysis showed the 8wMA to perform reasonably well for tested items. Executives asked us to improve upon this simplistic baseline and develop an integrated approach using the full year (52 weeks) data for model fitting. They further designated all the items' demand over the first ten weeks in the following year (i.e., weeks 53-62) as test data to assess the out-of-sample prediction performance. Although demand patterns exhibit different levels of volatility and frequency, managers hoped that some common patterns might be leveraged to generate demand forecasts.

3 Iterative Intervention

This section describes major phases of our intervention process. We first summarize our attempt to leverage various single-item methods to improve on the moving average forecasting method used by *Alpha*. Despite their relative refinements — e.g., consideration for sporadic demand and non-stationary components — none of the proposed methods was able to consistently improve the forecast accuracy of the baseline model used by *Alpha:* a simple moving average. The failure of these 'more sophisticated' methods to reduce demand forecast error triggered the 'reframing exercise' (Chandrasekaran et al. 2020) that resulted in the revised intervention and the consequential improvements. The remaining subsections show how we tackled failure modes and incrementally improved our solution based on theoretical frameworks and operational understanding.

3.1 Initial Intervention

The distribution of demand patterns depicted in Figure 2 suggests a variety of time series prediction problems. The operations and supply chain management literatures offer no unified theory or model for such demand forecasting tasks (Syntetos et al. 2009). A fundamental reason for developing alternative forecasting methods (or, as computer scientists call them, learning

algorithms) is the *no-free-lunch* theorem of machine learning, that is, no single method outperforms all other methods in all tasks (Wolpert 1996). Thus, our first attempt to address the prediction problem and improve on the algorithm used by *Alpha* was to employ an array of time series modeling techniques that matched the demand characteristics of the 426 items in our test case. Details of all tested approaches and algorithm implementations are provided in Appendix A.

We addressed non-smooth demand patterns using two fundamental methods for sporadic demand forecasting, Croston's (1972) exponential smoothing (Cr_Exp) and moving average (CR_MA). We also include the Syntetos-Boylan-Approximation (SBA) (Syntetos and Boylan 2001), which reduces bias in the Croston method by introducing a correction factor into the prediction equation demand. Lastly, we included the Teunter-Syntetos-Babai method (TSB) (Teunter et al. 2011), which multiplies the forecast of non-zero demand by the predicted probability of non-zero demand.

We also tested three methods capable of handling non-stationary demand processes: Autoregressive Integrated Moving Average (ARIMA), Error Trend Seasonality (ETS) (Hyndman and Athanasopoulos 2018), and the Trigonometric model – an advanced ETS with Box-Cox transformation, ARMA errors, and Trend and Seasonal components (De Livera et al. 2011) – that employs the above transformations to model complex patterns in time series (TBATS). These three methods are parametric and impose some functional structures on the underlying stochastic demand processes.

Finally, we employed two non-parametric univariate methods – feed forward neural nets (NN) with one hidden layer (Hyndman and Athanasopoulos 2018) and multilayer perception (MLP) (Ord et al. 2017) – that rely on artificial neural networks and perform well with non-smooth time-series forecasting (e.g., Gutierrez et al. 2008, Kourentzes 2013, Lolli et al. 2017, Mukhopadhyay et al. 2012). Collectively, our initial attempt to tackle the prediction problem employed nine models, all more sophisticated than the baseline 8wMA used by *Alpha*.

We fitted the nine time series models for each of the 426 item using the 52 weeks' observations (t=1, …, 52). Since procurement decisions on safety stock are entirely contingent on aggregate demand over lead time L (Bruzda 2020, Cobb et al. 2015, Trapero et al. 2019b), *Alpha's* goal is to predict total demand over lead time (i.e., L=10 weeks) and ensure that orders placed at the current week would be able to cover the supply deficit after the lead time.

As mentioned earlier, *Alpha* gave us demand observations over the first L=10 weeks of the

following year as test samples for assessing the predictive power of each model. We evaluated model performance by computing the SLE (squared log error) of total demand over the lead time, that is, $SLE_i = (\log(\sum_{t=53}^{62} F_{it} + 1) - \log(\sum_{t=53}^{62} D_{it} + 1))^2$ $(i = 1, \ldots, 426)$. The forecast error metric SLE was chosen per *Alpha*'s request to make the error metric scale independent (i.e., comparable across demand patterns with different means) and avoid the zero-denominator issue in percentage metrics.[3] The SLE has the additional benefit of penalizing under-estimates (i.e., forecast $F <$ actual D) more than over-estimates (Kannan et al. 2020), a highly desirable property in our setting given that insufficient supply to EMS clients incurs an extremely high cost of goodwill loss for *Alpha*. Figure 3 reports box and whisker plots of the SLE distribution for all ten forecasting methods across the 426 items. Because of the right skewness of the SLE distributions (i.e., the mean was much larger than the median error) management chose the median and interquartile range as key measures of performance.

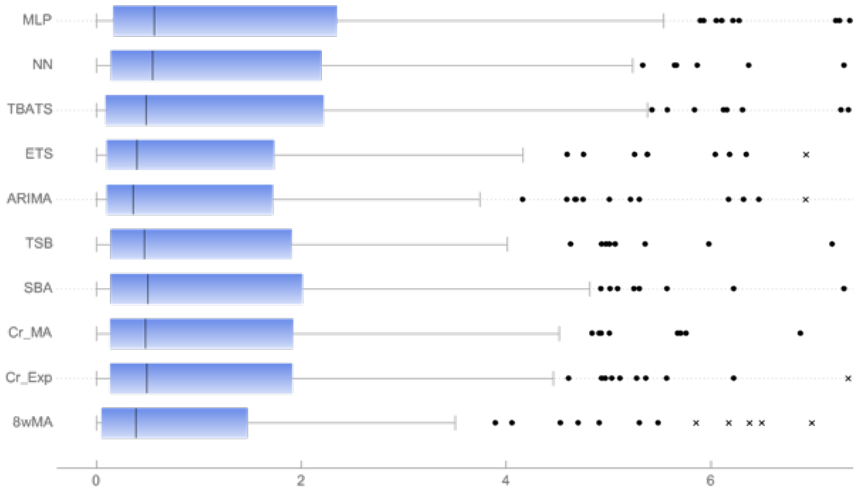

Figure 3: Squared Log Error (SLE) distribution of forecasting methods across the sample of 426 items

Note: The line within the box is the median, the box covers the Inter Quartile Range (IQR), and whiskers cover points within 1.5 times the IQR beyond the end of the box. Far outliers beyond three times the IQR are represented by "x." The graph range does not cover all far outliers.

Surprisingly, none of the proposed techniques improved on the SLE performance of the baseline 8wMA model; the 8wMA yielded close to the lowest median SLE and tightest IQR.[4] We

[3] Results were consistent between assessing prediction performance with mean absolute error (MAE) and SLE. *Alpha* opted for scale-free SLE comparable across items and without the biases of MAE for non-smooth demand (Morlidge 2015).

[4] The ARIMA model had a mean SLE 6.6% smaller than the 8wMA, but its IQR was 13.7% larger.

further analyzed the number of items for which each method is the best choice (see Appendix A). The high dispersion of best-performing methods confirmed management's concern that it would be difficult to *a priori* identify the best model for each item and that doing so would result in too many different models to maintain for thousands of items. Although the results reported are based on a single round test, the results show why predicting non-smooth demand over lead time has long been considered a difficult problem in supply chain forecasting (Lolli et al. 2017, Mukhopadhyay et al. 2012). That several items had SLE>50 for all forecasting methods suggested that historical data did not hold useful information for predicting future demand (Bachman et al. 2016). Although evidence indicated that they could not be predicted with backward-looking methods, management insisted these items be retained, as they were a significant part of the forecasting challenge.

In meetings following this initial exploration, managers re-emphasized the goal to predict entire demand over lead time and to reduce the total forecast error. This led us to reassess the necessity of characterizing week-to-week changes in limited demand data and to recalibrate the forecast target. They also asked whether it was possible to develop an integrated modeling framework that did not require fitting many individual time series models. Managers' request for "a" method for all items helped us realize that we had fallen into the trap of "time series modeling," our first attempt having been to fit n individual models to weekly demand series. In the wake of these initial discussions with managers, we re-examined known theories and methods and, through multiple iterations, developed the framework described below.

3.2 Problem Reframing – Data Aggregation

Our conversations with management after the initial failures revealed the possibility of reformulating the demand forecast target for stocking decisions and interpreting the data as an n by T panel. In reframing the problem, we also adopted two data aggregation principles for temporal variance reduction and integrated model estimation.

Our first aggregation is driven by obviating the need to dwell on the week-to-week variation that is hard to capture and not aligned with procurement decisions. Since the goal of procurement managers is to predict *total* demand over lead time (L=10 weeks), the target forecast should be in line with the operational requirement. Week-to-week differences in forecast accuracy within the lead time are not considered as crucial for stocking decisions (Bruzda 2020, Cobb et al. 2015, Trapero et al. 2019b). The cumulative demand and errors over lead time remains the focal issue

for up-to-date forecasting research (Kourentzes et al. 2020, Prak and Teunter 2019, Trapero et al. 2019a). Accordingly, we aggregated demand observations by L weeks and redefine the forecasting goal to generate demand estimates for the full lead time. Referred to in the literature as *temporal aggregation* (Rostami-Tabar et al. 2013), this strategy has shown that transforming non-smooth data series into lower frequencies stabilizes variance and improves forecast performance (Petropoulos and Kourentzes 2015, Kourentzes and Petropoulos 2016).

Specifically, demand observations (w_t) in weeks t=1,..,T are aggregated into total demand over lead time in a block as the prediction target based on $y_j = \sum_{k=(j-1)*L+1}^{j*L} w_k, \forall j = 1, \dots, T/L$. For a single item with demand series of T weeks, *temporal aggregation* results in T/L non-overlapping blocks. Like the classical demand pooling that effectively reduces demand variances in supply chains, aggregating weekly demand into a block of sum of demand over L weeks reduces the impact of zero demand weeks and averages out the spikes in raw weekly demand.

Note that it is customary to use non-overlapping temporal aggregation as a robust filter for high frequency components (Athanasopoulos et al. 2017). Although, as an alternative to non-overlapping aggregation, one can make every consecutive block pair differ by one week, doing so creates strong dependencies across blocks that are not suitable for statistical modeling approaches assuming independent observations (Hastie et al. 2016). Non-overlapping aggregation also results in an extra parameter – the width between two consecutive blocks – upon which to decide. We chose to begin by assessing the efficacy of non-overlapping aggregation, as we could easily relax the restriction later.[5]

Our second aggregation is motivated by the possible dependency among the demand patterns of individual items. Although Alpha has no detailed knowledge about family structures of item sets used by EMS clients, or specific bills of materials, executives are adamant that they are required to supply all requested items that are part of manufacturing inventories and operationally-related (Orlicky 1975, Hopp and Spearman 2011). This suggests that realized demands across items and plants, in spite of individual differences, are likely to share some common patterns motivating a cross-item learning initiative (Ban et al. 2019, Bojer and Meldgaard 2021). Hence, in addition to the item-wise temporal aggregation with a frequency of L weeks based on operational

[5] Later tests with overlapping blocks led to poor prediction performance and the idea was abandoned. Managers also did not like the added requirement to optimize for the overlapping parameter.

lead time, we performed *cross-item aggregation* by pooling temporarily aggregated observations of individual items into an n items by T/L blocks panel data set. With this method, it is technically possible for cross-item or cross-store prediction models to capture both common effects and individual heterogeneities. The resulting data structure enables learning algorithms (e.g., regression, tree, neural nets) to leverage multi-item information for cross-learning of demand patterns (Loureiro et al. 2018, Ren et al. 2015, Ren and Choi 2016, Ren et al. 2017). Cross-individual pooling has been found to be useful for prediction problems in retailing (Ban et al. 2019, Chuang et al. 2016), and has the added benefit of compensating for information loss in temporal aggregation, which reduces the number of available observations for model building at the item level (Petropoulos and Korentzes 2015).

The aggregated blocks of n items by T/L blocks panel data set (a total of $n*T/L$ observations) serves as the target (y) for predictive modeling of total demand over lead time (L weeks). In presenting this *data pooling* to *Alpha* managers, we put forward the argument that cross-learning effects are more salient when stochastic demands share the common error distribution after adjusting for their means (Ban et al. 2019). Managers concurred that these effects are likely to be present in our case because item demands all originate from the same EMS client. The cross-learning approach has also been shown to outperform traditional univariate methods in finance (Wu et al. 2021), automotive (Gonçalves et al. 2021), and retail (Spiliotis et al. 2021) forecasting studies, in addition to well accommodating missing values and limited observations (Hartmann et al. 2015). For multiple longitudinal series prediction tasks, cross-learning approaches are increasingly popular and proven effective in many forecasting competitions (Bojer and Meldgaard 2021).

The effectiveness of this cross-item modeling approach, nonetheless, relies on the input factors provided to the forecasting algorithm. The goal is to present the data in a format that enables the algorithm to detect patterns relevant to the prediction problem. In this instance, we used our understanding inventory and logistics models and forecasting methods to develop features hypothesized to contain information relevant to the prediction problem.

3.3 First Iteration – Fixed Effects and Past Demand Features

For business prediction tasks, *feature engineering* is perhaps more crucial to prediction accuracy than the machine learning algorithms themselves (Kuhn and Johnson 2019). Features, numerical representations of specific aspects of data, sit between data and models in predictive analytics

(Zheng and Casari 2018). The critical task here is to create features (i.e., predictor variables) that serve as inputs to subsequent *machine learning* (i.e., model fitting). The creation of input features is critical but difficult without a solid understanding of the problem domain. Wu et al. (2021) show, in a finance application, that the success of cross-learning relies on feeding algorithms with a set of idiosyncratic features rooted in theoretical concepts. Analogously, for our prediction problem, we develop three sets of features based on theoretical principles of econometric modeling and demand forecasting.

The first set of features aims to cover *base heterogeneity*. In the theory of panel data modeling and cross-learning for prediction, heterogeneities come from cross-sectional units and time periods (Wooldridge 2010). Hence, to control for individual and time fixed effects, we create binary variables using one-hot encoding for all item IDs, production plant IDs, and block period index (i.e., 1, 2,…, T/L). The block period index absorbs seasonal/trend effects that are rudimentary elements of time-series forecasting theory (Hyndman and Athanasopoulos 2018).

The second set of features from historical demand capture the *recent demand level* based on the theoretical perspective of arguably all forecasting methods in the literature, that is, past demand data are correlated with future demand and are legitimate inputs for generating demand predictions (Hyndman and Athanasopoulos 2018, Utley and Gaylord May 2010). We compute three statistics as input features from demand in the previous block (i.e., a lag of one)[6], specifically, the sum of demand of the previous block (x_1) and sum of demand in the first (x_2) and second (x_3) halves of the previous block. To capture within-block variation not reflected in total demand, we consider the feature expansion (Zheng and Casari 2018) of sub demand level (x_2 and x_3) in addition to total level (x_1). The three features subsume and expand the best-performing baseline moving average.

From statistical modeling and forecasting theories, time-variant *level* is not sufficient for non-smooth demand prediction (Thomopoulos 2015). Hence, our third set of features attempt to capture *recent demand volatility*. We focus on sporadicity and variability that are crucial for intermittent, erratic, or lumpy demand modeling (Syntetos et al. 2005). For sporadicity, we consider the level of and density of zeros in demand observations. We capture how sporadic the demand pattern is by computing the number of zero-demand weeks in the block (x_4), the number of zero-demand

[6] Note that we look into a lag of only one block to reduce information losses, that is, after the *data pooling* phase, each item has only T/L=5 blocks/observations in the panel data set. Using a lag of higher order would reduce the number of available blocks for training and validation.

weeks in the first (x_5) and second (x_6) halves of the block, and the number of consecutive zero-demand weeks of the block (x_7). As a measure of demand variability in the previous block (x_8) we use the median absolute deviation (Rousseeuw and Croux 1993) of weekly demand (median of $|x_i - \text{median}(x)|$ for $x_1, x_2, \ldots, x_n$), as opposed to the standard deviation because the former statistic is resilient to outliers. Table 2 provides a succinct summary of the features derived in this stage.

Including the binary variables for item, plant, and periods, the data set for our prediction problem has more than 400 features (i.e., regressors). The number of predictors being nontrivial relative to the sample size, we used *machine learning* to tackle the high-dimensional input data. We employ statistical machine learning algorithms to approximate the functional relationship between the target y and an array of predictors $\boldsymbol{x}$. We use random forests (RF) (Breiman 2001), an ensemble learning algorithm where the ensemble refers to a set of weak learners (e.g., simple classification and regression trees) combined to solve a particular problem (Mendes-Moreira et al. 2012). RF creates many bootstrapped samples with randomly selected features and fits a regression tree to each sample with incomplete features. Averaging predictions across the regression trees yields the final forecast. RF is well known for its prediction accuracy, and popular for being much easier to train than deep learning approaches.

To predict the 426 items' total lead time demand over L=10 weeks as described in §2, we first created training and validation sets. Recall that we have a panel data structure with sample size 1704 = 426 items * 4 blocks (T/L=50/10=5, minus 1 because of the lagged block for x_1-x_8). We use the first three blocks for training (n=1278, 426 items, each with three L-week blocks) and the last block (n=426 items, each with one L-week block) for validation. The training set allows algorithms to fit functions from data, that is, to take features as inputs in order to generate outputs in the form of total demand over the next L weeks by minimizing some error/loss metrics, such as mean squared error (MSE) and mean squared log error (MSLE). The validation set is for assessing the trained models' out-of-sample prediction performance before applying the models to the entirely unknown out-of-sample test set (weeks 53-62). This validation procedure enables analysts to fine-tune hyper-parameters of learning algorithms and avoids over-fitting. Training errors significantly lower than validation and test errors suggest models with poor generalizability.

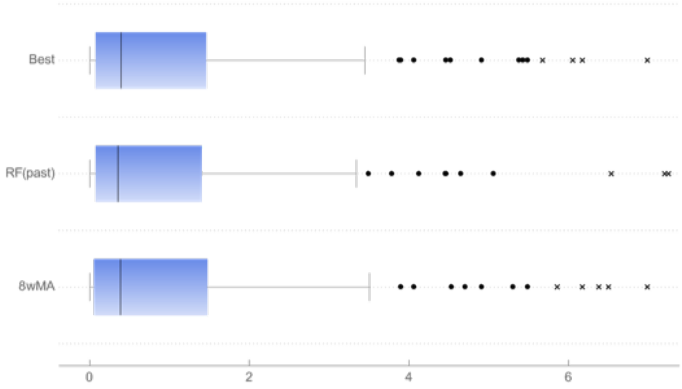

Figure 4: Squared Log Error (SLE) distribution of forecasting methods
Note: The distribution for 8wMA is the same as reported in Figure 3.

We trained the RF using the open-source machine learning platform H2O in *R* (Cook 2016). We use the validation set to set the number of trees, maximum depth, number of randomly selected feature in a tree, and minimum data points in a leaf. The setup of hyper-parameters of predictive machines is reported in Appendix C. Figure 4 shows the distribution of SLEs for the 426 items (RF(past)). For reference, the figure includes the SLE distribution for the 8wMA (management current practice) as well as the distribution of SLEs selecting for each item the *Best* of the ten methods described in §3.1 during the validation period and using that method to predict for the test set.

Relative to current management practice (8wMA), we find RF(past) to reduce the median SLE by 7.5% (from 0.385 to 0.356) and IQR by 6.6% (from 1.425 to 1.331) as well as outperform the item-wise selection of the *Best* method, having a median 9.8% lower and an IQR 3.8% narrower. *Alpha* managers found particularly appealing that, with the data structure for cross-item learning, only one integrated model, as opposed to hundreds of individual time series models, is needed to generate *n* predictions. We informed managers, however, that up to this point our features encompass just base heterogeneity and demand statistics that reflect recent demand level and volatility. Even though test results validated the value of *cross-item learning* and showed that one model could elevate forecast accuracy, we expected that more predictors could further improve forecasting accuracy. That the small reduction in SLE seems to imply that those features are insufficient for lead time demand forecasting motivated us to proceed to the next phase.

3.4 Second Iteration – Rolling Forecast Features

Like many forecast models, our predictive machine in the previous phase relies on input features predicated on the fundamental assumption in time series modeling that future demand is dependent

on past demand and fixed effects. Leading demand signals, such as advance order data, are nevertheless intended to contain information useful for prediction in supply chain operations (Terwiesch et al. 2005, Utley and Gaylord May 2010). Despite our unsuccessful attempt to fit MMFE assumptions and managers' skeptical views of rolling forecast prior to our initial intervention, we determined to derive two sets of features to leverage *forward-looking information* embedded in noisy forecast signals.

For any week t, downstream forecast signals of different lengths and versions are available for the next L weeks. Theoretically, the latest version carries the most up-to-date information (Cakanyildirim and Roundy 2002, Heath and Jackson 1994), from which we derive six features (x_9-x_{14}) on level and sporadicity of *leading order signals*. Note that the calculation of those features follows a similar logic used in developing the two sets of past demand features. In addition to those signals for future demand at week t+1 to t+L, historical forecast signals for past demand in recent periods may carry extra information for future demand, because the discrepancies between historical signals and demand realizations could be related to future demand propensity (Cakanyildirim and Roundy 2002, Albey et al. 2015). Hence, we derive another set of features (x_{15}-x_{17}) on *retrospective order signals*. Details on computing rolling forecast features are reported in Appendix B. Table 2 briefly summarizes the operationalization and rationale of all features derived from our iterative investigations into literature and discussions with management.

Table 2 List of features

Feature		Description*	Rationale
Base heterogeneity	Item	Binary indicators	To capture individual item heterogeneity.
	Plant	Binary indicators	To capture individual plant heterogeneity.
	Periods	Binary indicators	To capture seasonal/time effects.
Recent demand level	x_1	sum of demand of preceding block	x_1-x_3: Developed to capture inertia in realized demand levels and within-block changes; the three features retain and expand the efficacy of the base MA that is a rudimentary model in demand forecasting theory (Hyndman and Athanasopoulos 2018).
	x_2	sum of demand in the first half of preceding block	
	x_3	sum of demand in the second half of preceding block	
Recent demand volatility	x_4	number of zero-demand weeks in preceding block	x_4-x_6: Developed to capture the number of "zeros" as sporadicity measures in past demand; important indicators for non-smooth demand modeling (Syntetos et al. 2005, Teunter et al. 2011).
	x_5	number of zero-demand weeks in the first half of preceding block	
	x_6	number of zero-demand weeks in the second half of preceding block	
	x_7	number of consecutive zero-demand weeks in preceding block	x_7: Developed to capture the timing of non-zeros; important indicators for non-smooth demand modeling (Syntetos et al. 2005, Li and Lim 2018).
	x_8	median absolute deviation of demand in preceding block	x_8: Developed to capture the variability in past demand; important indicator for generic demand modeling (Rousseeuw and Croux 1993, Syntetos et al. 2005, Thomopoulos 2015).
Leading order signals	x_9	sum of diagonal forecast of succeeding block	x_9-x_{11} : Developed to capture the latest information about demand levels and changes over future lead time (Heath and Jackson 1994, Terwiesch et al. 2005, Albey et al. 2015).
	x_{10}	sum of diagonal in the first half of succeeding block	
	x_{11}	sum of diagonal in the second half of succeeding block	
	x_{12}	number of zeros in the diagonal rolling forecast of succeeding block	x_{12}-x_{14}: Developed to capture the latest information about demand sparsity over future lead time (Heath and Jackson 1994, Terwiesch et al. 2005, Albey et al. 2015).
	x_{13}	number of zeros in the diagonal of the first half of succeeding block	
	x_{14}	number of zeros in the diagonal of the second half of succeeding block	
Retrospective order signals	x_{15}	sum of Fcst(-1) of preceding block	x_{15}-x_{17}: Developed to capture latent production demand (measured by last production forecast) and to complement to realized demand (x_1-x_3) (Cakanyildirim and Roundy 2002, Albey et al. 2015).
	x_{16}	sum of Fcst(-1) in the first half of preceding block	
	x_{17}	sum of Fcst(-1) in the second half of preceding block	

* Preceding block refers to one temporal aggregation window of last *L* weeks. Succeeding block refers to one temporal aggregation window of next *L* weeks.

Using the rolling forecast features (x_9-x_{17}) identified in Table 2, we execute *machine learning*

using RF. Relative to the RF(past) benchmark established in the previous section, we find the forecast features improve the median SLE by 24% (from 0.356 to 0.272) and the SLE IQR by 14% (from 1.331 to 1.148) (see RF(fcst) in Figure 5). Furthermore, when combining all the developed features, the RF algorithm reduces, relative to RF(past), the median SLE for the 426 items by 38% (from 0.356 to 0.218) and the SLE IQR by 32% (from 1.331 to 0.902) (see RF(all) in Figure 5). These performance gains attest to the value of more comprehensive *feature engineering*.

3.5 Third Iteration – Machine Learning Improvements

We then adopted a more sophisticated ensemble learning algorithm utilizing gradient boosting machines (GBM) (Friedman 2001) capable of extrapolating predictions beyond the observed data range. RF, by averaging predictions of trees, is unable to extrapolate observations (Hastie et al. 2016, Zhang et al. 2017). Further, GBM follows a sequential learning protocol, i.e., a *boosting* process, in which each tree tries to reduce prediction errors from previous trees (Kuhn and Johnson 2013), whereas in RF each regression tree learns from data independently. The gradient of the loss function is a mathematically general representation of residuals. GBM excels at reducing prediction bias, but is prone to over-fitting (Natekin and Knoll 2013).

We used eXtreme gradient boosting (XGB) (Chen and Guestrin 2016), which augments GBM by enhancing regularization terms to prevent over-fitting, considers both gradients and hessians of loss functions in creating splits, and improves computation efficiency. XGB outperforms other deep/shallow learning models in many prediction tasks on non-perpetual and structured data (Sjardin et al. 2016). Although XGB does not improve the aggregate performance of RF(all) (see XGB in Figure 5), averaging the predictions of RF(all) and XGB reduced the RF(past) median SLE by 45% (from 0.356 to 0.197) and the IQR by 39% (from 1.331 to 0.817). Whereas RF(all) alone realizes a substantial improvement over the baseline, XGB, with its extrapolative power, adds value to the demand prediction problem, the averaging method improving the median SLE over its two inputs by 10% and 13% and IQR performance by 10% and 9%. The method is easy to use and grounded on “model stacking”, a practically prevalent and theoretically effective approach in machine learning for predictive analytics (Cook 2016).

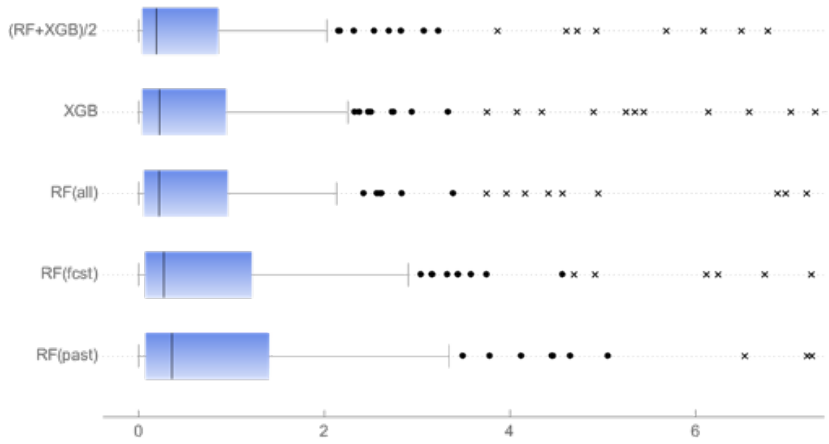

Figure 5: Squared Log Error distribution of ML methods across 426 items in sample

Note: The distribution for RF(past) is the same as reported in Figure 4

To sum up, the initial failure of individual time series models to predict erratic demand patterns lead us to a reframing of the forecasting problem considering the possibility of data aggregation (across time and across items). Through an iterative process, we proposed and tested specific techniques for data analysis to improve forecast accuracy. Rather than having to fit 426 individual time series models for all items recounted in §3.1, we end up with one predictive machine grounded in two ensemble learning algorithms. The combination of ensemble regression trees enables us to capture overall patterns across item-plant pairs while accommodating local heterogeneities. More significantly, the predictive machine accommodates as many features as desirable and does not presume the underlying structure of demand processes.

4 Assessment of Proposed Method

Satisfied with improvements in forecast accuracy achieved with the foregoing averaging method (i.e., (RF+XGB)/2), management requested that we test it with demand from other EMS clients. We report here findings of these additional tests as robustness checks and diagnoses aimed at identifying other possible improvements. Finally, we report on the deployment strategy and initial results from the adoption of the predictive modeling framework.

4.1 Tests with Other Data Set

Alpha selected one of its most complex customers in terms of item heterogeneity and lead time as a test for assessing the appropriateness of the proposed model over multiple prediction rounds. The test, involving ~1000 items from 15 production plants, differs from the foregoing case in several respects. First, in addition to offering more information for model training, the increased number of items represents higher item heterogeneity. Second, items for this EMS client had a longer lead

time from the supply side, twelve weeks, suggesting higher demand uncertainty over a longer planning horizon.[7] Third, we were asked to report prediction performance across ten runs on a rolling basis, that is, for total demand from weeks 76-87, 77-88, …, 85-96.

Numbers of items across prediction rounds varied as the EMS client's production schedule changed, the weekly prediction challenge ranging from 822 to 967 items.[8] Accordingly, the weighted average improvement across the ten out-of-sample weeks (weighted by the number of observed items in each week) relative to 8wMA was a 62.7% reduction in median SLE with an improvement range of 50%-69%. The reduction in IQR of the SLE distribution was consistently above 97% over the ten runs, with a weighted average of 97.5%. Figure 6 shows the SLE distribution for the base and proposed (ML) methods of rolling predictions for forecast windows with the best (weeks 79-90), median (weeks 82-93), and least (weeks 85-96) improvement in median and IQR. The proposed method dominates the base method even for the case with the least improvement. Indeed, the variance in improvement performance seems to stem more from the adequacy of the base method in any given window than from differences in the performance of the ML method. Moreover, the variance in the number of cases included each week indicates a slight but significant ($p < 0.001$) improvement in the median of the SLE distribution as more items are included in the model.

Despite the significant prediction error reduction shown in Figure 6, a few outlying items with large SLE are present in all tested periods (as is the case with the 426 items in Figure 5). Management attributed these outlying errors to sudden changes in EMS clients' product lines or production schedules. Such demand patterns are essentially unpredictable based on past demand records and forward forecast signals (Bachman et al. 2016).

---

[7] This test was also performed assuming the same lead time for all components (see footnote 1). Technically, it is possible to extend our method to items with different lead times by introducing a binary array of input features to represent the different length of lead time or normalizing the total demand by lead time length.

[8] The models were trained on an Intel i9 2.6 GHz desktop with 64 GB DRAM. A couple of minutes were required to train the RF and XGB models in *R*, given hyper-parameters. Computing time increases in the number of random searches for hyper-parameters, and varies with the number of trees and sampling fraction of observations/columns for each tree. That this could take several hours is not a concern to *Alpha* because of available multi-core and cloud computing.

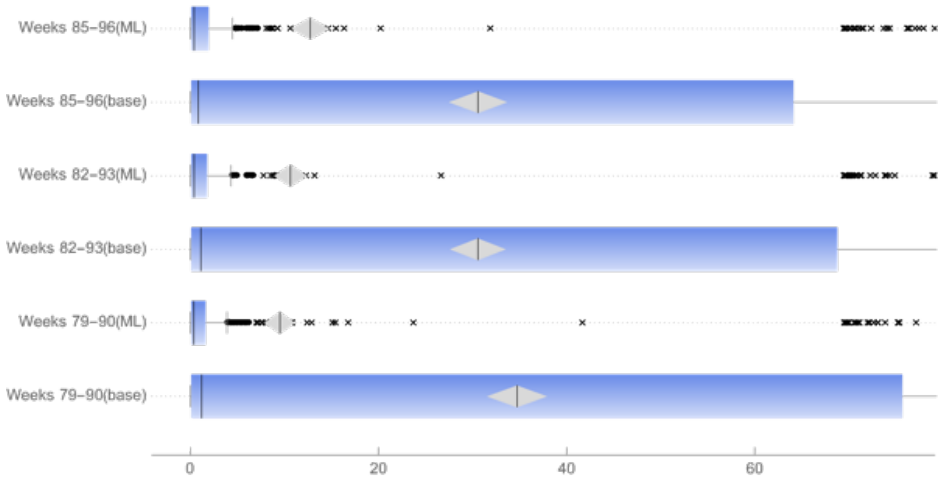

Figure 6: Squared log error distribution of base and ML method

Note: The gray triangle denotes the mean SLE and its 95% confidence interval.

Alpha deemed the test results to demonstrate the method to be robust and adaptive, effective not only in situations in which historical data could be leveraged to make sensible predictions (e.g., the aforementioned EMS client with 426 items), but also in situations in which current forecasting practices were yielding poor results (the latter EMS client with ~1000 items). The performance of the cross-item analytics, moreover, is consistent across clients with varying item heterogeneity and lead time intervals. In particular, the proposed method's ability to accrue cross-item learning from balanced and unbalanced panel data structures easily accommodates short life cycle items, as limited demand observations make individual model fitting difficult. In the following section we evaluate, using our original data set, the relative importance of the features used by the ML model and examine items for which the method still provides a poor forecast.

4.2 Relative Importance of Features

Both RF and XGB construct ensembles of regression trees, in which each tree embeds multiple binary splits based on different features. Although post-fitting interpretation is more difficult with an ensemble of trees than with a single tree with full transparency, recent developments in explainable machine learning enable us to identify important predictors in the post-training stage (Kuhn and Johnson 2013).

Figure 7 shows relative feature importance in the RF(all) and XGB models described in §3.5. For RF(all), importance is determined by whether a predictor is selected to split during tree construction processes and how much the squared error of all trees drops due to a predictor. Specifically, for each tree, a feature's attributed reduction in error is the difference in variance of

response (y) within the node and response variance of its children nodes (Hastie et al. 2016). Relative influence is obtained by dividing each feature's variance reduction to total variance reduction. For the XGB model, we calculate, instead of variance reduction, the average gain obtained by a feature. The gain metric is a function of gradient and hessian statistics as well as two regularization terms in the loss function (Chen and Guestrin 2016). Relative influence is obtained by dividing each feature's gain to total gains. Computational details on the importance of each feature in ensembles of randomized trees can be seen in Louppe et al. (2013) and Kazemitabar et al. (2017).

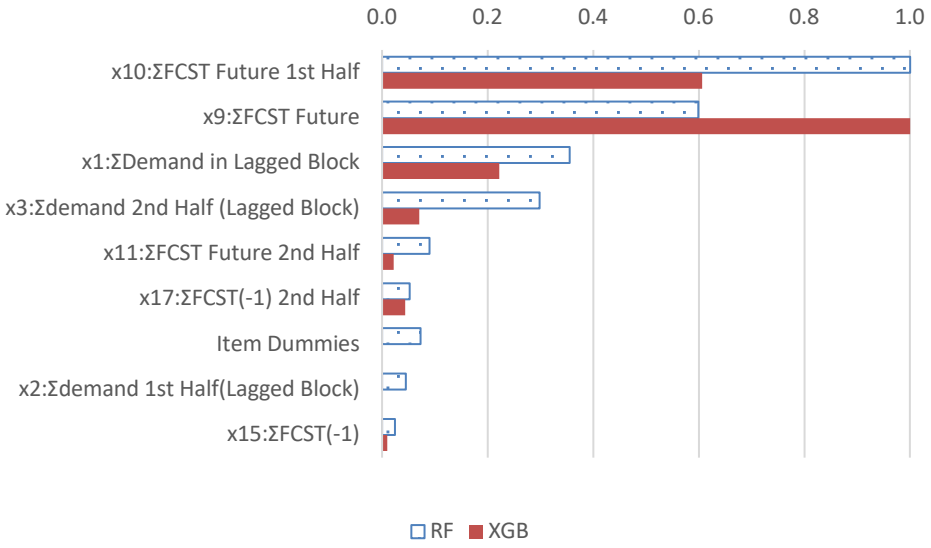

Figure 7: Relative feature importance in RF and XGB

A number of observations can be made from Figure 7. First, the features derived from the rolling forecast matrix – x_9, x_{10}, and x_{11} – are the most salient in reducing forecast errors. Our understanding of the MMFE idea and theoretical value of advance demand information helps us reassess the value of those noisy signals. Although the item-by-item rolling forecast signals neither seem helpful nor improve over time, the two models seem to be able to learn from cross-item signals and extract useful information for demand predictions. Second, consistent with time series modeling theory, past demand observations (i.e., the sum of lead time in the previous block) – x_1, x_2, and x_3 – do seem to have an effect on reducing prediction errors, although learning algorithms clearly derive more value from forward-looking (albeit noisy) signals than from historical demand. Third, features from recent forecast signals FCST(-1) for already realized demand, that is, x_{17} and x_{15}, and item fixed effects have much smaller effects. Features related to the number of zeros in rolling forecasts and past demand are not detected as useful in machine learning processes despite

the large number of lumpy and intermittent demand patterns.

Our findings regarding relative feature importance corroborate the efficacy of *feature engineering* for *machine learning* to generate high-quality predictions. A key lesson is that features guided by OM theory (e.g., forecast evolutions) are key to learn from noisy data and improve prediction performance. Although noisy signals may not be informative on an item-by-item basis, pooling information enables us to detect existing patterns across items. Observed a procurement executive at *Alpha*:

> It is a big surprise to see that features based on rolling forecast signals carry such importance. We definitely will elicit more advance demand information and try to integrate related features into machine learning.

Having assessed the importance of features, we investigate items with abnormally high SLE. Examining 56 items with SLE greater than seven revealed 53 of the 56 items to have zero demand in the test period (weeks 53-62). The remaining three items had just one week with non-zero demand. These outliers were abrupt changes in the demand signal, i.e., most of the items had been suddenly discontinued or replaced by alternatives. Management suggested that the zeros were likely caused by changes in the EMS client or end-of-projects that could not be predicted by patterns learned from demand- or rolling forecast-based features. The lack of significance of features describing the number of zeros in past demand and rolling forecasts (see Figure 7), despite the large number of lumpy and intermittent demand patterns, lends credence to this interpretation. Developing codifiable features on such changes on a regular and timely basis for algorithms to learn and adapt was, however, not feasible. *Alpha* explicitly recognized that forecasting methods in general are not immediately responsive to abrupt discontinuities. Instead of further pushing for improvement of our modeling approach, management is looking for other channels of information (beyond the rolling forecast) that better capture the signal of when an item will be suddenly discontinued or replaced. Furthermore, *Alpha* has asked engineers and procurement managers to pay extra attention to items with high validation error, i.e., items with less predictable demand given available features. Specifically, procurement managers now must report any information about "operational changes/anomalies" in the EMS client's request for the items and engineers are required to assess if items exhibit over-fitting (inconsistencies between in-sample and out-of-sample prediction performance) that could signal a possible need to tune the model.

We also pointed out in discussions with *Alpha* that demand uncertainty could be reduced if

the long supply lead time (10-12 weeks) could be shortened and shared the results of a simple test showing a lower forecast error under reduced lead time. *Alpha* responded that although conceptually appealing, key suppliers' long lead time is irreducible due to the physics of semiconductor manufacturing processes. Moreover, even with annual revenue of more than $18 billion, *Alpha*, as a distributor and buffer in the supply chain, has limited power to influence vendors and clients. *Alpha* managers, nevertheless, took the ability to work with shorter lead times as further evidence of the framework's flexibility and robustness.

4.3 Deployment

Based on these results, management decided to roll out the methods to different EMS clients and production plants. For actual deployment of predictive analytics, management opted for the (RF+XGB)/2 model described in §3.5 with all the features listed in Table 2. While the focal EMS of the study represented approximately 3% of *Alpha's* sales, the framework has been rolled out to predict demand for all major EMS clients with large transaction volume and hundreds of items. Combined, these large EMS clients represent 56% of *Alpha's* sales. The predictive machine for each EMS (with its corresponding plants and products) is re-calibrated on a monthly basis as new data becomes available. Thus, addition and termination of items in *Alpha's* operation is reflected in the continual data engineering and machine learning processes.

The central objective is to use forecasts as better anchoring points for improving stock control. Feedback from *Alpha* has been positive and procurement managers no longer rely on 8wMA to make inventory decisions. Improvement performance has not yet been assessed for all clients, but associates responsible for implementation of the framework report that for one major EMS client forecast error has been reduced by 30% or more for 50% of items and by at least 10% for up to 80% of items. *Alpha*'s management acknowledges that more accurate lead time demand forecasting has enabled the firm to reduce inventory holdings and scrapping of obsolete material without compromising the service level to EMS clients. For the focal EMS of this study, average inventory holdings have dropped by 5%, a reduction of inventory value of approximately $3M. As it is still managers who make ordering decisions after considering the additional information they now possess, we cannot attribute cost improvement solely to better forecasts. But given managers' tendency to anchor to a point (see Harvey 2001 for extensive discussion of the research in this area), providing better forecasts of demand over lead time arguably improves the anchoring point and should thus relate, to some extent, to better procurement decision-making.

5 Discussion

The iterative process described above — where frameworks to analyze data are tested in their ability to create accurate and robust forecasts, and insights are drawn from these tests to further improve the frameworks to analyze the data — improved lead time demand forecast accuracy at our research site. The iterative process also affords an opportunity to reflect on the theory and methods used to improve the problem situation in an example of intervention-based research (IBR) (Chandrasekaran et al. 2020), in which insights emerge from mismatches between expected and actual outcomes (Oliva 2019).

The intervention strategy described in §3 can be mapped into multiple iterations of what Oliva (2019) calls Mode 1 of intervention-based research, where theories (T) and methods (M) are confronted with a problem situation (S) with the goal of improving S (see Figure 1 in Oliva 2019). In our setting, the purpose of the iterative process is to develop more accurate, robust forecasts. Although better forecasts should in turn lead to better re-stocking decisions (Syntetos et al. 2009), because *Alpha* explicitly limited the scope of our intervention to the forecasting process, the test of the usefulness of our theories and methods is their ability to generate accurate forecasts. The results of the modeling effort, however, are still empirically validated. Failure to improve forecasts or unexpected outcomes (surprises) generate reflections on the appropriateness of the theories and methods and redefinition of what might be considered a useful theory (see Figure 8).

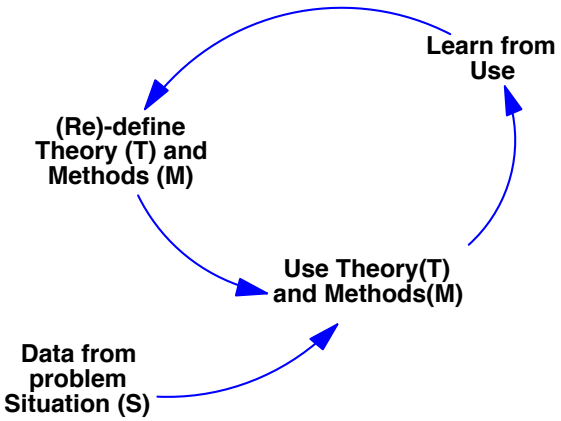

Figure 8: Iterative learning from interventions

We posit that the fine-tuned set of complementary guidelines yielded by the above-described iterative process constitutes a generalizable research contribution. Recognizing that the distinction between T and M is often not clear and insights largely overlap (see §2.2 in Oliva 2019), by contrasting our findings with current theoretical and methodological guidelines we can push the

lessons beyond the specifics of the *Alpha* site. Their importance notwithstanding, context-driven *data pooling* and theory-informed *feature engineering* tend to be obscured by the shadow of *machine learning* in the wave of big data and artificial intelligence (AI). As opposed to many machine learning applications that treat the model as a black box to be fed all sorts of related information (Rudin and Radin 2019), we found that it was our understanding of OM that allowed us to reach better problem formulation, feature derivation, and model construction. We posit this work hints to what the future of OM might be in the context of emerging AI. While there is no way that humans can compete with the pattern identification capabilities of algorithms, we show that pre-processing the data in accordance with theory played an important part in achieving the improvements we obtained. Below, we elucidate and contrast with existing theoretical and methodological guidelines these important elements.

5.1 Data Pooling

Data pooling is built upon *temporal aggregation*, which has become increasingly popular in business forecasting (Boylan and Babai 2016, Kourentzes and Athanasopoulos 2020, Syntetos et al. 2016) as a way to derive low frequency time series from high frequency observations. Although aggregation across time has been validated as a useful approach for tackling non-smooth demand patterns widely seen in various industry sectors (Nikolopoulos et al. 2011), related studies almost exclusively identify patterns from temporally aggregated univariate series via distribution or function fitting. Ours differs from prior work by pooling the low-frequency time series into a panel data structure. Temporal aggregation focuses on within-item data pooling, and aggregating individuals afterwards aims to enable potential cross-item learning.

Cross-item learning from panel data aims to combine cross-sectional items for fitting models that reveal overall demand patterns, a sharp contrast with fitting individual time series models that capture exclusively item-specific patterns. While many demand forecast models in the literature are trained in a series-wise fashion (Spiliotis et al. 2021), the renowned M4 competition with 100,000 real-world time series has shown that cross-learning models effectively enhance forecasting performance (Makridakis et al. 2020). The aggregate approach has proven successful for product demand forecasting in various consumer (Ren et al. 2017, Ban et al. 2019, Spiliotis et al. 2021) and industrial markets (Gonçalves et al. 2021). Hu et al. (2019) show that, even by just clustering items based on similarities, fitting common curves without input features improves demand forecast accuracy. Our results in the electronics distribution sector justify the utility of

cross-individual learning for demand forecasting in industrial markets. Notwithstanding differences in market types, panel data structures introduce more information on common effects into demand modeling processes. Theoretically, by exploring between- as well as within-item variation, fitted models can improve prediction performance by obtaining parameter estimates with high efficiency (using more samples) while accounting for individual heterogeneity (Greene 2011). In addition to leveraging information across individuals, cross-item learning addresses that individual time series modeling is unable to capture common patterns and vulnerable to missing values and limited observations (Hartmann et al. 2015). Furthermore, cross-item learning affords the flexibility to include additional features, whether time variant or not, at the macro (e.g., plant and quarter fixed effects) or micro level (e.g., item-specific and block-varying attributes).

The idea of combining temporal aggregation (that alleviates within volatility) and cross-item learning (that serves between variation) thus constitutes a theoretical proposition that can be employed for non-smooth series of hundreds or thousands of items common in manufacturing and retailing settings. In line with previous successful OM applications in apparel retailing (Chuang et al. 2016, Ban et al. 2019) and recent success of employing a cross-individual machine learning approach to financial forecasting (Wu et al. 2021), we posit cross-item learning is theoretically well-grounded when considering manufacturing inventory operational requirements. With sensible input features, cross-learning excels in extracting common patterns in demand targets and the strategy is worth exploring in subsequent studies of operational prediction problems.

5.2 Feature Engineering

Feature engineering creates predictor variables and serves as the backbone of predictive modeling efforts (Kuhn and Johnson, 2019). The success of image recognition by convolutional neural networks has led to a claim that deep learning frees analysts from feature creation (Zheng and Casari 2018). This claim may hold for image or voice recognition tasks, but prediction performance for most management and social problems still relies on relevant predictors identified by analysts, and feature engineering remains indispensable to the success of business prediction tasks (Martinez et al. 2020). A common strategy of feature engineering in statistical learning is to create as many non-linear transformations of raw data variables and all possible two-way interaction terms (Kuhn and Johnson 2019) in regression models, such as Lasso and elastic net (James et al. 2013), that can handle high-dimensional inputs. An alternative strategy in machine learning is to feed into models all raw variables, such as support vector machines and Gaussian

process models (Theodoridis 2015), that count on kernel functions to reach high-dimensional feature expansion (Hastie et al. 2016).

These two feature engineering paradigms in statistical/machine learning exploit computational power and reduce human involvement. These approaches, however, can drop theoretically relevant factors due to collinearities or become combinatorically cumbersome when the dimension of raw input variables is high, resulting in high noise-to-signal ratios that undermine prediction performance. Instead of brute-force methods to transform data signals, our intervention adopts a theory-informed approach to feature engineering. As described earlier, we construct five sets of features following theoretical perspectives of forecasting, OM, and econometrics. For instance, motivated by individual forecasting techniques for non-smooth demand, in addition to rudimentary features for demand levels, we devise an array of features related to zeros for demand sparsity and timing. Although revealed by our post-modeling analysis to be less important than others, the sparsity and timing features are guided by theoretical ideas and could prove useful in other demand forecasting settings. Note that we cannot generalize which features will be more useful *a priori*, since features that appear weak in some sites may become strong in other settings. Hence, our advice to researchers and managers is to include features grounded on theoretical principles and allow learning algorithms identify the useful features.

Predicated on the rudimentary theoretical principle that future demand is dependent on past realizations, our initial machine learning with past demand and base heterogeneity features achieved marginal improvements in forecast accuracy. Guided by OM principles, we salvaged data originally discarded by management and further extracted features from the rolling forecast matrix to create leading indicators. Unlike MMFE techniques that attempt to capture all observed stochastic variation (i.e., forecast evolutions) via distributional tools, we generate a set of features using the latest information about future and recent demand. Post-modeling analysis of relative feature importance provides empirical support for this process, as both RF and XGB identify the leading demand indicators as the most important features.

Although we tackle rolling forecast via a data-driven alternative to MMFE techniques (Heath and Jackson 1994) with somewhat vulnerable assumptions, the data-driven features are motivated by, and benefit from, OM theories of the value of advance demand signals with regular updates (Utley and Gaylord May 2010). A key lesson from our feature engineering journey is that, although brute-force transformation and raw computational power are easily adopted and do not require

much "brain effort," it remains crucial that derivation of features be guided by theoretical and contextual principles.

5.3 Machine Learning

Machine learning was the least effort-consuming aspect of our intervention. The major learning algorithms are not new, but recent improvements in computational tools have made learning from high-dimensional data much more accessible. There are, however, insights to be derived from our experience. First, note that our recommendation to *Alpha* was based on a combination of ensemble learning techniques – RF and XGB – instead of other algorithms for high-dimensional inputs, such as Lasso regression and artificial neural networks (James et al. 2013). Our strategy is based on the theoretical virtues of tree methods being robust to outliers and missing data and insensitive to variable scales that often create numerical problems for other algorithms. These properties of ensemble machine learning techniques are highly desirable for *cross-item learning* that absorbs multiple demand series with broad scale differences. Second, RF was initially chosen because of its conceptual simplicity, practical applicability (e.g., sales forecast in Ferreira et al. 2016), and theoretical efficacy in error variance reduction. However, RF's inability to extrapolate led us to complement it with XGB, which possesses extrapolative power and error bias reduction capabilities (Hastie et al. 2016).

Because RF and XGB both aggregate predictions from many tree models, point forecasts are, by construction, products of hybrid models. Combining point predictions from different models or experts into a hybrid forecast has a long tradition in the forecasting literature (Armstrong 2001) and time series modeling (Zhang 2003). A hybrid forecast, that is, a weighted linear combination of predictions, is commonly generated by identifying weights using the validation error of each prediction. One could also identify weights using meta-learners, for example, a least squares model with non-negative weights in which each prediction is a regressor for the validation target (Breiman 2001). We opted for a simple average of predictions, machine learning theorists having shown generalization (out-of-sample prediction) errors to be lower for averaged predictions than for single model prediction (see Mendes-Moreira et al. 2012 for a discussion). A simple average of predictions is easy to use, theoretically effective, and lowers the risk of over-fitting validation sets in attempting to find "optimal" weights.

Note that the extent of our testing does not preclude the possible existence of better-performing alternatives. It is, however, essentially impossible to exhaustively test every

multivariate and aggregate forecasting approach (e.g., dynamic panel data models, vector autoregressions, multiple aggregation prediction algorithm with regressors, recurrent neural networks) and their potential combinations. Our exploration of alternatives was limited by the constraints of an intervention in the real world – cost and timelines of the solutions – and a satisficing solution was deemed appropriate. A comprehensive assessment of multivariate models, with and without aggregation, and their potential combination based on their performance attributes would be a substantial contribution to the focal demand forecasting problem and related literature.

5.4 Generalizability

Although *machine learning* is an important part of our intervention, *data aggregation* strategy and *feature engineering* based on OM principles constitute our generalizable contributions to the theory and practice of predictive analytics. In fact, the theory-informed features derived from our intervention have broad application beyond ensemble learning algorithms. Researchers could, for example, fit panel data regression or structural simulation models based on those input features for the sake of more transparency, or use more obscure black boxes, such as deep neural nets, while retaining the benefit of features crafted from theoretical knowledge. For instance, RF models have been shown to effectively improve the performance of job dispatching and outperform widely used dispatching rules for difficult flexible job shop scheduling problems (Jun et al. 2019). Machine learning performance being a function of the input features engineered by researchers possessing theoretical knowledge of scheduling, we posit that it is contextual knowledge and theoretical understanding that gives OM researchers an edge over computer scientists and statisticians in creating input features and better applying machine learning in operational problems.

Our intervention has led to the development of a predictive modeling framework for a demand forecasting problem that is not *Alpha*-specific, but general to many firms in manufacturing supply chains (Fu and Chien 2019). Methodological innovations, like many scientific developments, are driven by empirical irregularities, that is, unsuccessful applications of oft-used theories and methods. The processes of field validation and deriving theoretical implications imparts to the proposed framework a more robust grounding and higher degree of generalizability, major criteria for assessing theoretical contributions of empirical work (Oliva 2019). It should be noted, however, that intervention-based insights, like the insights from case studies, are intended to generalize to theory as opposed to other populations. An intervention is not a 'sample' in the common statistical

sense, but rather is a rich base from which to develop generalizable theoretical insights. Yin (2003, p. 10) calls this generalizing process 'analytic generalization' as opposed to the 'statistical generalization' that is required from sampling studies and Meredith (1998, p. 450) calls for 'theoretic generalizability' from field research, as opposed to the 'assumptive generalizability' that is required from rationalistic research.

Indeed, our proposed guidelines will be useful in other settings only to the extent that those other settings share the attributes of the problem that we explore here: supply for manufacturing processes in innovative markets (short product life) and where the supplier does not have access to the bill-of-materials and/or the master production schedule. First, the demand patterns seen by *Alpha* – rapidly changing, non-smooth, and violating MMFE assumptions – are common and expectable for upstream members of a supply chain, wherein the supply lead time tends to be longer than downstream channels. Second, despite decades of work to achieve information integration in supply chains (Cai et al. 2010, Wei et al. 2020), and the demonstrated effects that they have on performance (e.g., Devaraj et al. 2007, Yuen and Thai 2016, Gu et al. 2017), this lack of integration is still common in industry (see Bughin et al. 2017, IBM Institute for Business 2019) due to various barriers such as mistrust, cost, and information risks (e.g., Harland et al. 2007, Vafaei-Zadeh et al. 2020). Thus, although our case illustration cannot be claimed to be generalizable to a specific population, the guiding principles have a broad range of application.

One may nevertheless still question whether this type of predictive effort constitutes a theoretical contribution. How does predictive analytics — long considered operations research or engineering work — fit into OM empirical research? Can we claim a theoretical contribution when we have neither shown statistical significance of regressors nor explicitly attempted to reject a null hypothesis? We ground our response in Fisher's (2007) argument that empirical research in OM naturally embodies both descriptive and prescriptive elements. Prediction problems that require prescriptive interventions are under-appreciated in OM (Terwiesch 2019). Predictive modeling, in parallel with explanation through hypothesis testing, has been part of empirical science in other fields (Shmueli 2010, Shmueli and Koppius 2011). The explanation-prediction dichotomy and its different evaluation standards have undergone critical examinations in the overall science community (see Schumeli 2010 for an in-depth discussion). In an intriguing essay, Hofman, et al. (2017) point out the pitfalls of the social science research paradigm that relies on unbiasedness and significance of parameter estimates to justify and validate generic human explanations formed as

theories. Hofman et al. (2017) articulate why explanatory and predictive research — similarly driven by empirical phenomenon and validated by empirical data — complement each other and enhance theory development in social sciences. In line with this perspective, we posit that our intervention employing predictive analytics, despite its problem-solving origin, can lead to generalizable contributions to theory development in OM and supply chain demand forecasting.

6 Concluding Remarks

Adopting intervention as a research strategy (Oliva 2019) aimed at developing theoretical and methodological contributions based on practice, we report the outcome of an intervention undertaken to improve demand forecasting. After an unsuccessful problem-solving attempt in the field, we develop a methodological framework that substantially reduces forecast errors at our research site and yields generalizable insights. We show OM frameworks and perspectives to be critical inputs to data pooling for cross-item learning and theory-driven feature engineering, and that, absent appropriate prediction units and informative features, machine learning cannot work effectively. We further show it to be possible for one aggregate model to outperform many individual time series models. Cross-individual learning empowered by ensemble machine learning is potentially a new paradigm for demand forecasting (Bojer and Meldgaard 2021). By introducing transparency to such powerful methods (often considered unexplainable black boxes), feature importance metrics (James et al. 2013) inform feature development through further learning. Subsequent studies are encouraged to examine, refine, and improve the guidelines generated by our proposed framework. Empirical OM research is not solely about offering generic explanations; solving prediction problems that are often among managers' major concerns affords rich opportunities for theoretical and methodological development. Our predictive modeling journey presents the OM community with a salient case for exploiting data analytics and machine learning to develop new theories, methods, and principles for addressing operational problems (Misic and Perakis 2020). Engaging practitioners in interventions is one way for OM researchers to obtain useful ideas for innovations and ensure relevance by disseminating those ideas to a broader audience.

Finally, our experience allows us to also reflect on the broader role of pursuing an IBR strategy (Oliva 2019, Chandrasekaran et al., 2020) in the context of new technological developments. Despite the inherent risks of an intervention might not yield adventitious data that requires new theorizing — see discussion of 'unexpected outcomes' in §2.1 of Oliva (2019) — it

should be noted that a) new technology, by virtue of being new, involves unforeseen execution challenges, and b) the adoption of new technologies will normally require an adaptation of the technology to make it workable within a specific organizational backdrop. Both characteristics surround the adoption with uncertainty and, thus, make it a fertile field for insights and further theoretical developments. Our intervention leverages this opportunity by assisting in the adoption of data analytics and machine learning in an operational context. We were able to develop a new framework informed by the specifics of the context — in this case a forecasting, supply chain, and operations management context — from the mismatches between expectations or requirements and what we achieved in each iteration. It is these mismatches, the unexpected outcomes, and the extent that we took them seriously to act on them, that drove the insights for improvement. Our resulting forecasting approach is a very different proposal than the approach a team of data or computer scientists would have developed. This is because we made sense of the mismatches from the forecasting, supply chain, and operations management perspectives. Interestingly, our proposal does not inform the technology *per se* — a typical outcome of a design science perspective — but rather it provides guidelines on how improving the inputs makes the technology more successful in this context. That is, we use the theory to better inform how to leverage the new technology. We believe that there are ample opportunities for this type of theoretical development using IBR strategy and push our theories to be more *practical* and *relevant*.

Acknowledgments

The authors thank *Alpha* and its managers for their willingness to engage in the intervention and generous contributions of time, insights, and access to relevant data. The first two authors also wish to thank the E.SUN Commercial Bank for supporting their research.

References

Albey, E., Norouzi, A., Kempf, K. G., & Uzsoy, R. (2015) Demand modeling with forecast evolution: An application to production planning. *IEEE Transactions on Semiconductor Manufacturing*, 28(3): 374-384.

Armstrong, J. S. (2001) Combining forecasts. In *Principles of Forecasting*, Armstrong, J. S. (ed.). Kluwer Academic Publishers, New York. pp: 417-439.

Athanasopoulos, G., Hyndman, R. J., Kourentzes, N., & Petropoulos, F. (2017) Forecasting with temporal hierarchies. *European Journal of Operational Research*, 262(1): 60-74.

Bachman, T. C., Williams, P. J., Cheman, K. M., Curtis, J., & Carroll, R. (2016) PNG: Effective inventory control for items with highly variable demand. *Interfaces*, 46(1): 18-32.

Ban, G., Gallien, J., & Mersereau, A. J. (2019) Dynamic procurement of new products with covariate information: The residual tree method. *Manufacturing & Service Operations Management*, 44(2):165-186.

Barker, J. (2020) Machine learning in M4s. *International Journal of Forecasting*, 36(1): 150-155.

Bojer, C. S., & Meldgaard, J. P. (2021) Kaggle forecasting competitions: An overlooked learning opportunity. *International Journal of Forecasting*, in press.

Boylan, J. E., & Babai, M. Z. (2016) On the performance of overlapping and non-overlapping temporal demand aggregation approaches. *International Journal of Production Economics*, 181(A): 136-144.

Boylan, J. E., Syntetos, A. A., & Karakostas, G. C. (2008) Classification for forecasting and stock control: A case study. *Journal of the Operational Research Society*, 59(4): 473-481.

Breiman, L. (2001) Random Forests. *Machine Learning*, 45(1): 5-32.

Bruzda, J. (2020) Demand forecasting under fill rate constraints: The case of re-order points. *International Journal of Forecasting*, 36(4): 1342-1361.

Bughin, J., LaBerge, L., & Mellbye, A. (2017) The case for digital reinvention. *McKinsey Quarterly*, https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/the-case-for-digital-reinvention#

Cai, S., Jun, M., & Yang, Z. (2010) Implementing supply chain information integration in China: The role of institutional forces and trust. *Journal of Operations Management*, 28(3): 257-268.

Carbonneau, R., Laframboise, K., & Vahidov, R. (2008) Application of machine learning techniques for supply chain demand forecasting. *European Journal of Operational Research*, 184(3): 1140-1154.

Cakanyildirim, M., & Roundy, R. O. (2002) SeDFAM: semiconductor demand forecast accuracy model. *IIE Transactions*, 34(5): 449-465.

Chandrasekaran, A., de Treville, S., & Browning, T. (2020) Editorial: Intervention-based research (IBR) – What, where, and how to use it in operations management. *Journal of Operations Management*, 66(4): 370-378.

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*, San Francisco, CA, pp 785-794.

Chuang, H. H., Oliva, R., & Perdikai, O. (2016) Traffic-based labor planning in retail stores. *Production and Operations Management*, 25(1): 96-113.

Cobb, B. R., Johnson, A. W., Rumi, R., & Salmeron, A. (2015) Accurate lead time demand modeling and optimal inventory policies in continuous review systems. *International Journal of Production*

Economics, 163: 124-136.

Cook, D. (2016) *Practical Machine Learning with H2O*. O'Reilly Media Inc., Sebastopol, CA.

Croston, J. D. (1972) Forecasting and stock control for intermittent demands. *Operational Research Quarterly*, 23(3): 289-303.

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.

de Treville, S., Schurhoff, N., Trigeorgis, L., & Avanzi, B. (2014) Optimal sourcing and lead-time reduction under evolutionary demand risk. *Production and Operations Management*, 23(12): 2103-2117.

Devaraj, S., Krajewski, L., & Wei, J. C. (2007). Impact of eBusiness technologies on operational performance: The role of production information integration in the supply chain. *Journal of Operations Management*, 25(6): 1199-1216.

der Auweraer, S. V., Boute, R. N., & Syntetos, A. A. (2019) Forecasting spare part demand with installed base information: A review. *International Journal of Forecasting*, 35(1): 181-196.

do Rego, J. R., & de Mesquita, M. A. (2015) Demand forecasting and inventory control: A simulation study on automotive spare parts. *International Journal of Production Economics*, 161: 1-16.

Ferreira, K. J., Lee, B. H. A., & Simchi-Levi, D. (2016) Analytics for an online retailer: Demand forecasting and price optimization. *Manufacturing & Service Operations Management*, 18(1): 69-88.

Fildes, R., Nikolopoulos, K., Cone, S. F., & Syntetos, A. A. (2008) Forecasting and operational research: A review. *Journal of the Operational Research Society*, 59(9): 1150-1172.

Fisher, M. (2007) Strengthening the empirical base of operations management. *Manufacturing & Service and Operations Management*, 9(4): 368-382.

Friedman, J. H. (2001) Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5): 1189-1232.

Fu, W., & Chien, C. (2019) UNISON data-driven intermittent demand forecast framework to empower supply chain resilience and an empirical study in electronics distribution. *Computers & Industrial Engineering*, 135: 940-949.

Gilliland, M. (2020) The value added by machine learning approaches in forecasting. *International Journal of Forecasting*, 36(1): 161-166.

Gonçalves, J. N., Cortez, P., Carvalho, M. S., & Frazao, N. M. (2021) A multivariate approach for multi-step demand forecasting in assembly industries: Empirical evidence from an automotive supply chain. *Decision Support Systems*, in press.

Greene, W. H. (2011) *Econometric Analysis*, seventh edition, Pearson, New York.

Gu, Q., Jitpaipoon, T., & Yang, J. (2017) The impact of information integration on financial performance:

A knowledge-based view. *International Journal of Production Economics*, 191: 221-232.

Gutierrez, R. S., Solis, A. O., & Mukhopadhyay, S. (2008) Lumpy demand forecasting using neural networks. *International Journal of Production Economics*, 111(2): 409-420.

Harland, C., Caldwell, N., Powell, P., & Zheng, J. (2007) Barriers to supply chain information integration: SMEs adrift of eLands. *Journal of Operations Management*, 25(6): 1234–1254.

Harvey, N. (2001) Improving judgement in forecasting. In *Principles of Forecasting*, Armstrong, J. S. (ed.). Kluwer Academic Publishers: New York. pp: 59-80.

Hartmann, C., Hahmann, M., Lehner, W., & Rosenthal, F. (2015) Exploiting big data in time series forecasting: A cross-sectional approach. *IEEE International Conference on Data Science and Advanced Analytics*, 1-10.

Hastie, T., Tibshirani, R., & Friedman, J. (2016) *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd edition. Springer, New York.

Heath, D.C., & Jackson, P.L. (1994) Modeling the evolution of demand forecasts with application to safety stock analysis in production/distribution systems. *IIE Transactions*, 26(3): 17-30.

Hill, T., O'Connor, M., & Remus, W. (1996) Neural network models for time series forecasts. *Management Science*, 42(7): 1082-1092.

Hofman, J. M., Sharma, A., & Watts, D. J. (2017) Prediction and explanation in social systems. *Science*, 355: 486-488.

Hopp, W. J. & Spearman, M. L. (2011) *Factory Physics*, 3rd edition. Waveland Press, Inc.: Long Grove, IL.

Hu, K., Acimovic, J., Erize, F., Thomas, D. J., & Van Mieghem, J. A. (2019). Forecasting new product life cycle curves: Practical approach and empirical analysis. *Manufacturing & Service Operations Management*, 21(1): 66-85.

Huang, Z., Wang, Z., & Zhang, R. (2019) Cascade2vec: Learning dynamic cascade representation by recurrent graph neural networks. *IEEE Access*, 7: 144800-144812.

Hyndman, R. J., & Athanasopoulos, G. (2018) *Forecasting: Principles and Practice*. 2nd edition. OTexts, Melbourne, Australia.

Hyndman, R. J., & Khandakar, Y. (2008) Automatic time series forecasting: The forecast package for R. *Journal of Statistical Software*, 27(3): 1-22.

Hyndman, R. J., Koehler, A. B., Ord, J. K., & Snyder, R. D. (2008) *Forecasting with Exponential Smoothing: The State Space Approach*. Springer-Verlag, Berlin.

Hyndman, R., Athanasopoulos, G., Bergmeir, C., Caceres, G., Chhay, L., O'Hara-Wild, M., Petropoulos, F., Razbash, S., Wang, E., & Yasmeen, F. (2020) *forecast: Forecasting functions for time series and linear models*. R package version 8.12, http://pkg.robjhyndman.com/forecast.

IBM Institute of Business (2019) *Build Your Trust Advantage: Leadership in the Era of Data and AI*

Everywhere. IBM Corporation, Armonk, NY.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013) *An Introduction to Statistical Learning with Applications in R*. Springer, New York.

Jun, S., Lee, S., & Chun, H. (2019) Learning dispatching rules using random forest in flexible job shop scheduling problems. *International Journal of Production Research*, 57(10): 3290-3310.

Kannan, B. A., Kodi, G., Padilla, O., Gray, D., & Smith, B. C. (2020) Forecasting spare parts sporadic demand using traditional methods and machine learning - a comparative study. *SMU Data Science Review*, 3(2).

Kazemitabar, J., Amini, A., Bloniarz, A., & Talwalkar, A. S. (2017) Variable importance using decision trees. *Proceedings of Advances in Neural Information Processing Systems*: 425-434.

Korkmaz, S., Goksuluk, D., Zararsiz, G. (2014) MVN: An R package for assessing multivariate normality. *The R Journal* 6(2): 151-162.

Kourentzes, N. (2013) Intermittent demand forecasting with neural networks. *International Journal of Production Economics*, 143(1): 198-206.

Kourentzes, N., & Athanasopoulos, G. (2016) Elucidate structure in intermittent demand series. *European Journal of Operational Research*, 288(1): 141-152.

Kourentzes, N., & Petropoulos, F. (2016) Forecasting with multivariate temporal aggregation: The case of promotional modeling. *International Journal of Production Economics*, 181(Part A): 145-153.

Kourentzes, N., Trapero, J. R., & Barrow, D. K. (2020) Optimising forecasting models for inventory planning. *International Journal of Production Economics*, 225: 107597.

Kuhn, M., & Johnson, K. (2013) *Applied Predictive Modeling*. Springer, New York.

Kuhn, M., & Johnson, K. (2019) *Feature Engineering and Selection*: *A Practical Approach for Predictive Models*. CRC Press, Boca Raton, FL.

Li, C., & Lim, A. A. (2018) A greedy aggregation-decomposition method for intermittent demand forecasting in fashion retailing. *European Journal of Operational Research*, 269(3): 860-869.

Lolli, F., Gamberini, R., Regattieri, A., Balugani, E., Gatos, T., & Gucci, S. (2017) Single-hidden layer neural networks for forecasting intermittent demand. *International Journal of Production Economics*, 183(A): 116-128.

Louppe, G., Wehenkel, L., Sutera, A., & Geurts, P. (2013) Understanding variable importances in forests of randomized trees. *Proceedings of Advances in Neural Information Processing Systems*: 431-439

Loureiro, A. L. D., Migueis, V. L., & da Silva, L. F. M. (2018) Exploring the use of deep neural networks for sales forecasting in fashion retail. *Decision Support Systems* 114: 81-93.

Makridakis, S. (1993) Accuracy measures: theoretical and practical concerns. *International Journal of Forecasting*, 9: 527-529.

Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2020) The M4 competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, 36(1): 54-74.

Martinez, A., Claudia, S., Pereverzyev Jr., S., Pirker, C., & Haltmeier, M. (2020) A machine learning framework for customer purchase prediction in the non-contractual setting. *European Journal of Operational Research*, 281(3), 588-596.

Mendes-Moreira, J., Soares, C., Jorge, A. M., & De Sousa, J. F. (2012) Ensemble approaches for regression: A survey. *ACM Computing Surveys*, 45(1): Article No. 10.

Meredith, J. (1998) Building operations management theory through case and field research. *Journal of Operations Management* 16: 441-454.

Misic, V. V. & Perakis, G. (2020) Data analytics in operations management: A review. *Manufacturing & Service Operations Management*, 22(1): 158-169.

Morlidge, S. (2015) Measuring the quality of intermittent-demand forecasts: It's worse than we've thought. *Foresight: International Journal of Applied Forecasting*, 37: 37-42.

Montero-Manso, P., Athanasopoulos, G., Hyndman, R. J., & Talagala, T. S. (2020) FFORMA: Feature-based forecast model averaging. *International Journal of Forecasting*, 36(1): 86-92.

Mukhopadhyay, S., Solis, A. O., & Gutierrez, R. S. (2012) The accuracy of non-traditional versus traditional methods of forecasting lumpy demand. *Journal of Forecasting*, 31(8): 721-735.

Natekin, A. & Knoll, A. (2013) Gradient boosting machines, a tutorial. *Frontiers in Neurorobotics*, 7: Article No. 21.

National Electronic Distributors Association (2003) *Quantifying The Value of Authorized Distribution*. Thomas and Joan Read Center for Distribution Research and Education. Texas A&M University. College Station, TX.
https://ecia.memberclicks.net/assets/docs/ValueStudy/TAMU%20Executive_Summary_Updated.pdf

Nikolopoulos, K., Syntetos, A. A., Boylan, J. E., Petropoulos, F., & Assimakopoulos, V. (2011) An aggregate-disaggregate intermittent demand approach (ADIDA) to forecasting: An empirical proposition and analysis. *Journal of the Operational Research Society*, 62(3): 544-554.

Oliva, R. (2019) Intervention as a research strategy. *Journal of Operations Management*, 65(7): 710-724.

Ord, J. K., Fildes, R., & Kourentzes, N. (2017) *Principles of Business Forecasting*, 2nd edition. Wessex Press Publishing Co.

Orlicky, J. (1975) *Material Requirements Planning: The New Way of Life in Production and Inventory Management*. McGraw-Hill: New York.

Petropoulos, F., & Kourentzes, N. (2015) Forecast combinations for intermittent demand. *Journal of the Operational Research Society*, 66(6): 914-924.

Prak, D., & Teunter, R. (2019) A general method for addressing forecasting uncertainty in inventory models.

International Journal of Forecasting, 35(1): 224-238.

Ren, S., Chan, H., & Ram, P. (2017) A comparative study on fashion demand forecasting models with multiple sources of uncertainty. *Annals of Operations Research*, 257: 335-355.

Ren, S., & Choi, T. (2016) Selection and industrial applications of panel data base demand forecasting models. *Industrial Management & Data Systems*, 116(6): 1131-1159.

Ren, S., Choi, T., & Liu, N. (2015) A comparative study on fashion demand forecasting models with multiple sources of uncertainty. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(3): 411-421.

Rostami-Tabar, B. Babai, M. Z., Syntetos, A., & Ducq, Y. (2013) Demand forecasting by temporal aggregation. *Naval Research Logistics*, 60(6): 479-498.

Rousseeuw, P. J., & Croux, C. (1993) Alternatives to the median absolute deviation. *Journal of the American Statistical Association*, 88(424): 1273-1283.

Rudin, C., & Radin, J. (2019) Why are we using black box models in AI when we don't need to? A lesson from an explainable AI competition. *Harvard Dara Science Review* 1(2). https://doi.org/10.1162/99608f92.5a8a3a3d

Shmueli, G. (2010) To explain or to predict? *Statistical Science*, 25(3):289-310.

Shmueli, G., & Koppius, O. R. (2011) Predictive analytics in information systems research. *MIS Quarterly*, 35(3):553-572.

Sjardin, B., Massaron, L., & Boschetti, A. (2016) *Large Scale of Machine Learning with Python*. Packt Publishing, Birmingham, UK.

Spiliotis, E., Makridakis, S., Semenoglou, A., & Assimakopoulos, V. (2021) Comparison of statistical and machine learning methods for daily SKU demand forecasting. *Operational Research*, in press.

Syntetos, A. A., & Boylan, J. E. (2001) On the bias of intermittent demand patterns. *International Journal of Production Economics*, 71(1-3): 457-466.

Syntetos, A. A., Boylan, J. E., & Croston, J. D. (2005) On the categorization of demand patterns. *Journal of the Operational Research Society*, 56(5): 495-503.

Syntetos, A. A., Boylan, J. E., & Disney, S. M. (2009) Forecasting for inventory planning: a 50-year review. *Journal of the Operational Research Society*, 60(sup1): s149-s160.

Syntetos, A. A., Babai, Z., Boylan, J. E., Kolassa, S., & Nikolopoulos, K. (2016) Supply chain forecasting: Theory, practice, their gap and the future. *European Journal of Operational Research*, 252(1): 1-26.

Terwiesch, C. (2019) Empirical research in operations management: From field studies to analyzing digital exhaust. *Manufacturing and Service Operations Management*, 21(4): 713-722.

Terwiesch, C., Ren, Z. J., Ho, T. H., & Cohen, M. A. (2005) An empirical analysis of forecast sharing in the semiconductor equipment supply chain. *Management Science*, 51(2): 208-220.

Teunter, R., Syntetos, A. A., & Babai, M. Z. (2011) Intermittent demand: Linking forecasting to inventory obsolescence. *European Journal of Operational Research*, 214(3): 606-615.

Theodoridis, S. (2015) *Machine Learning: A Bayesian and Optimization Perspective.* Academic Press, London.

Thomopoulos, N. T. (2015) *Demand Forecasting for Inventory Control*. Springer, New York.

Trapero, J. R., Cardos, M., & Kourentzes, N. (2019a) Quantile forecast optimal combination to enhance safety stock estimation. *International Journal of Forecasting*, 35(1): 239-250.

Trapero, J. R., Cardos, M., & Kourentzes, N. (2019b) Empirical safety stock estimation based on kernel and GARCH models. *Omega*, 84: 199-211.

Utley, J. S., & Gaylord May, J. (2010) The use of advance order data in demand forecasting. *Operations Management Research*, 3(1-2): 33-42.

Vafaei-Zadehm A., Ramayah, T., Hanifah, H., Kurnia, S., & Mahmud, I. (2020) Supply chain information integration and its impact on the operational performance of manufacturing firms in Malaysia. *Information & Management*, 57(8): 103386.

Wei, S., Ke, W., Liu, H., & Wei, K. K. (2020) Supply chain information integration and firm performance: Are explorative and exploitative IT capabilities complementary or substitutive? *Decision Sciences*, 51(3): 464-499.

Wolpert, D. H. (1992) Stacked generalization. *Neural Networks*, 5(3): 241-259.

Wolpert, D. H. (1996) The lack of a priori distinctions between learning algorithms. *Neural Computation*, 8(7): 1341-1390.

Wooldridge, J. M. (2010) *Econometric Analysis of Cross Section and Panel Data*, 2nd edition. The MIT Press, Cambridge, MA.

Wu, W., Chen, J., Yang, Z., & Tindall, M. L. (2021) A cross-sectional machine learning approach for hedge fund return prediction and selection. *Management Science*, in press.

Yin, R. K. (2003) *Case Study Research: Design and Methods*, 3rth ed. Sage: Thousand Oaks.

Yuen, K. F., & Thai, V. V. (2016) The relationship between supply chain integration and operational performances: A study of priorities and synergies. *Transportation Journal*, 55(1): 31-50.

Zhang, G. P. (2003) Time series forecasting using a hybrid ARIMA and neural network model. *Neural Computing*, 50: 159-175.

Zhang, H., Nettleton, D., & Zhu, Z. Regression-enhanced random forests. *JSM Proceedings, Section on Statistical Learning and Data Science, American Statistical Association*: 636-647.

Zheng, A., & Casari, A. (2018) *Feature Engineering for Machine Learning*. O'Reilly Media Inc., Sebastopol, CA.

Appendix A: Tested time series models

The Appendix provides details of the time series models tested in our initial intervention reported in §3.1. We fit demand observations for each of the 426 items to all models and obtain item-specific parameters for out-of-sample prediction. We begin with Croston's (1972) exponential smoothing and moving average. Given demand observations, the two methods independently analyze periods with and without zero values. Let q_i denote the ith non-zero quantity and a_i the number of periods between q_{i-1} and q_i. Croston's exponential smoothing (Cr_Exp) employs two equations:

$$\hat{q}_i = (1-\alpha)\hat{q}_{i-1} + \alpha q_{i-1}$$
$$\hat{a}_i = (1-\alpha)\hat{a}_{i-1} + \alpha a_{i-1}$$

where α is the smoothing parameter between 0 and 1. The two estimates are updated only if demand occurs in period t. The final output of the model is:

$$\hat{y}_t = \frac{\hat{q}_i}{\hat{a}_i}.$$

Croston's moving average (Cr_MA) differs only by replacing the two exponential smoothing equations with moving average equations. The Croston method having been shown to be biased, the Syntetos-Boylan-Approximation (SBA) (Syntetos et al. 2005) corrects the bias by adjusting the prediction equation thus:

$$\hat{y}_t = \left(1 - \frac{\alpha}{2}\right)\frac{\hat{q}_i}{\hat{a}_i}.$$

The Teunter-Syntetos-Babai method (TSB) uses two smoothing equations for the non-zero demand probability p_t and non-zero demand size z_t. The method updates the demand probability in every period and is unbiased. Let d_t be a binary variable denoting the occurrence of any non-zero demand in period t (d_t=1 if y_t>0, otherwise d_t=0). The TSB method has two updating schemes:

$$\hat{p}_t = \begin{cases} if\ d_t = 0,\ \hat{p}_{t-1} + \beta(0 - \hat{p}_{t-1}), \hat{z}_t = \hat{z}_{t-1},\ \hat{y}_t = \hat{p}_t\hat{z}_t \\ if\ d_t = 1,\ \ \hat{p}_{t-1} + \beta(1 - \hat{p}_{t-1}), \hat{z}_t = \hat{z}_{t-1} + \alpha(z_t - \hat{z}_{t-1}),\ \hat{y}_t = \hat{p}_t\hat{z}_t \end{cases}.$$

For each item, the parameters of the foregoing methods are estimated with the aid of the *tsintermittent* package in *R*. Optimal smoothing parameters α and β are found by minimizing the mean absolute rate error function (Kourentzes 2013) within the sample period.

ARIMA and ETS exponential smoothing are standard, common times series models. Both methods are capable of handling non-stationary demand processes, and the latter generalizes the Holt-Winters exponential smoothing based on a state-space modeling framework (Hyndman et al. 2008). Let p denote the number of autoregressive terms, d be the number of differences needed for stationarity, and q be the number of lagged forecast errors; the ARIMA(p, d, q) model (Hyndman

and Athanasopoulos 2018) can be written as:

$$\left(1-\phi_1 B-\cdots-\phi_p B^p\right)(1-B)^d y_t = u\left(1-\phi_1-\cdots-\phi_p\right)+(1+\theta_1 B+\cdots+\theta_q B^q)\varepsilon_t$$

where B is the backshift operator (i.e., $B^p y_t = y_{t-p}$) and u the mean of $(1-B)^d y_t$. The ETS exponential smoothing model has eighteen possible combinations from error (additive or multiplicative), trend (none, additive, or damped), and seasonal (none, additive, or multiplicative) components. Each combination has its own set of prediction equations (see §7 in Hyndman and Athanasopoulos 2018 for detailed expressions).

The TBATS model (De Livera et al. 2011), which mixes ideas from ETS and ARIMA, has more complicated forms. The method first applies Box-Cox transformation to observed y_t thus:

$$y_t^{\omega} = \begin{cases} \dfrac{y_t^{\omega}-1}{\omega} \; if \; \omega \neq 0 \\ \log(y_t) \; if \; \omega = 0 \end{cases}.$$

The transformed y_t^{ω} is modeled as a set of equations:

$$y_t^{\omega} = l_{t-1} + \phi b_{t-1} + \sum_{i=1}^{M} s_{t-m_i}^i + d_t$$

$$l_t = l_{t-1} + \phi b_{t-1} + \alpha d_t$$

$$b_t = (1-\phi)\phi b + \phi b_{t-1} + \beta d_t$$

$$d_t = \sum_{i=1}^{p} \phi_i d_{t-i} + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j} + \varepsilon_t$$

$$s_t^i = \sum_{j=1}^{k_i} s_{j,t}^i$$

where l_t denotes the local level, b_t is the short-term and b the long-term trend, and d_t is an ARMA(p, q) error process, ε_t white noise, and s_t^i a collection of Fourier-like/trigonometric terms for seasonal components. For each item, the parameters of the ARIMA, ETS, and TBATS models are estimated using the *forecast* package in *R* (Hyndman and Athanasopoulos 2018). The functions automatically return each item's specification and parameters by minimizing the Akaike information criterion among all specifications perceived by algorithms to be appropriate for training data. Hyndmand and Khandakar (2008) and Hyndman et al. (2020) describe in detail how the *forecast* package performs automatic model selection and estimation for the ARIMA, ETS, and TBATS models.

The last two methods, NN and MLP, are feedforward neural networks with three major components – input, hidden layers with non-linear activation functions, and output layers. As the input layer uses lagged demand (y_{t-1}, y_{t-2},…, y_{t-p}) as input variables, NN and MLP are essentially

complex non-linear auto-regression models (Hyndman and Athanasopoulos 2018). NN is a neural autoregression with one hidden layer. MLP constructs multiple hidden layers to approximate unknown functional forms. The output layer denotes target demand to be forecasted. Let h_j denote the j_{th} neuron in the hidden layer; NN with one hidden layer with k neurons can be expressed as:

$$h_j = g(b_j + \sum_{i=1}^{p} w_{ij} y_{t-i}), \forall\, j = 1, \ldots, k$$

$$y_t = g(b_o + \sum_{m=1}^{k} w_m h_m)$$

where $g(\,)$ is a non-linear activation function. For each item, we use the sigmoid function for $g(\,)$ and apply the *nnetar*/*neuralnet* functions in *R* to train NN/MLP, respectively. The parameters are estimated by gradient descent for mean squared error minimization with 20 repetitions of randomly initialized weights. Numbers of input lags (tested from four to twelve) and hidden neurons (twenty random draws between a quarter and two-thirds of input nodes) are determined by minimizing the Akaike information criterion.

Analyzing the number of items for which each method is the best choice revealed the 8wMA to be the most successful model, providing the best fit for 24% of the 426 items. Although it was expected that no method would perform well for all items (as suggested by the very nature of the diversity of patterns described in Figure 2), it was disappointing to find the 8wMA to continue to be the best method, and the top four methods to account for only 68% of the best methods identified.

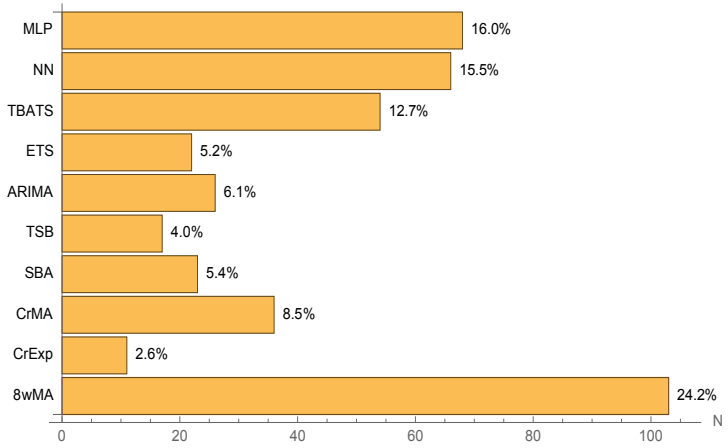

Figure A1. Number of items for which each method is the best performing

Appendix B: Operationalization of rolling forecast features

Figure B1 illustrates that at week t downstream forecast signals of different lengths and versions are available for the next L weeks. The j inside Fcst(-j) denotes the forecast released j weeks prior to the target week. For time t+1, one has already received L signals, whereas for time t+L one has only one signal at time t. Technically, the diagonal elements in the upper triangle forecast matrix – {Fcst(-1) for t+1}, {Fcst(-2) for t+2},…, {Fcst(-L) for t+L} – in Figure B1 carry the latest information about total production demand over future lead time at time t. Hence, we derive six features: sum of diagonal forecast (x_9), sum of diagonal in the first (x_{10}) and second (x_{11}) halves of the next L weeks, number of zeros in the diagonal (x_{12}), and number of zeros in the diagonal of the first (x_{13}) and second (x_{14}) halves of the next L weeks. Although off-diagonal elements in the upper triangle matrix in Figure B1 carry information about forecast revisions that could be additional features, we focus on the diagonal elements to capture the most up-to-date information, the forecast revisions being fairly noisy.

Week	Demand	Fcst(-1)	Fcst(-2)	Fcst(-3)	Fcst(-4)	....	Fcst(-L)
t+1	?	15884	22333	19210	31610	....	17294
t+2	?	X	13616	17472	18472	....	18870
t+3	?	X	X	30646	23517	....	7399
...	?	X	X	X	21751	....	34110
...	?	X	X	X	X	....	19152
...	?	X	X	X	X	....	1216
...	?	X	X	X	X	....	19152
t+L	?	X	X	X	X	X	15020

Figure B1 Rolling forecast for one item

In addition to *leading order signals* for future demand at time t+1 to t+L, we consider *retrospective order signals*, i.e., Fcst(-1) for demand realizations in the previous block. As illustrated in Figure B2, the Fcst(-1) vector in weeks t to t-L+1 stands for last signals of EMS client demand in retrospect. Computing the sum of Fcst(-1) for the L demand points (x_{15}) as a feature helps to capture the association between retrospective forecast signals and future demand. Following the feature expansion idea in x_2 and x_3, we compute the sum of Fcst(-1) in the first (x_{16}) (i.e., Fcst(-1) for $D(t)$,…, $D(t\text{-}L/2\text{+}1)$) and second (x_{17}) (Fcst(-1) for $D(t\text{-}L/2\text{+}1)$,…, $D(t\text{-}L\text{+}1)$) halves of the past L weeks. In short, we leverage our operational understanding of the rolling forecast matrix to create nine features, six (x_9-x_{14}) of which are leading indicators of future demand. The other three (x_{15}-x_{17}) are indicators of forecasts of recent demand.

Block	Week	Demand	Fcst(-1)	Fcst(-2)	Fcst(-3)	Fcst(-4)
b	t-L+1	3000	6828	14970	19898	21751
…	…	…	16524	16005	26187	23090
b	t-2	9000	33819	12887	16204	22321
b	t-1	15000	13512	1843	8598	5522
b	t	3000	11092	24057	29681	22652
b+1	t+1	?	1216	16650	19152	17294
b+1	t+2	?	X	12658	15020	18870
…	…	?	X	X	x	x
b+1	t+L	?	X	X	X	x

Figure B2 Past FCST(-1) for demand realizations for one item

Appendix C: Hyper-parameter tuning for ensemble learning

Both RF and GBM require hyper-parameter tuning to perform appropriately. Since an exhaustive search of parameter space is not feasible, we follow a common practice and conduct a random search over a grid of hyper-parameter values (Cook 2016). Table C1 shows the grid settings for the RF and XGB search. For each algorithm, we randomly select 100 combinations of hyper-parameters to train the model and choose the best-performing set based on its out-of-sample prediction performance on the validation set. For RF(all), we set the number of trees to 500, maximum depth to 30, fraction of random samples to 0.6, and minimum data points in a leaf to five. For XGB, we set the number of trees to 500, learning rate to 0.036, fraction of random samples to 0.6, maximum allowed depth of each tree to five, and minimum data points in a leaf to four.

Table C1

Model	Hyper-parameters	Values
RF	number of trees	[100, 1000] with an increment of 100
	maximum tree depth	[3, 10] with an increment of 1
	fraction of random samples in a tree	[0.5, 1] with an increment of 1
	minimum data points in a leaf	[1, 10] with an increment of 1
XGB	number of trees	[100, 1000] with an increment of 100
	fraction of random samples in a tree	[0.5, 1] with an increment of 1
	maximum tree depth	[3, 10] with an increment of 1
	minimum data points in a leaf	[1, 10] with an increment of 1
	learning rate	[0.01, 0.05] with an increment of 0.005