Decision Support Systems 139 (2020) 113401

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

Forecasting demand profiles of new products

R.M. van Steenbergen*, M.R.K. Mes

Department of Industrial Engineering and Business Information Systems, Faculty of Behavioural, Management and Social sciences, University of Twente, P.O. Box 217, 7500 AE Enschede, the Netherlands

ARTICLE INFO

Keywords:
New product forecasting
Pre-launch forecasting
Random Forest
Quantile regression Forest
Inventory management

ABSTRACT

Nowadays, many companies face shorter product life cycles, increasing the need to properly forecast demand for newly introduced products. These forecasts allow them to support operational decisions, such as procurement and inventory control. However, forecasting the demand of new products is challenging compared to existing products, since historical sales data is not available as an indicator of future sales. Moreover, little attention has been paid in literature to quantitative methods for new product forecasting, especially with respect to quantifying the uncertainty in demand. In this paper, we present a novel demand forecasting method denoted by DemandForest, which combines K-means, Random Forest, and Quantile Regression Forest. This machine learning-based approach combines the historical sales data of previously introduced products and product characteristics of existing and new products to make prelaunch forecasts and support inventory management decisions for new products. DemandForest clusters and predicts demand patterns, and predicts the quantiles of the total demand during an introduction period. We validate and illustrate our approach for forecasting and inventory management using real-world data sets of several companies. Compared to several benchmark methods, DemandForest provides the most accurate predictions, resulting in potential inventory savings of around 15% depending on lead times and service levels.

1. Introduction

New product forecasting is challenging compared to forecasting demand of existing products since historical data is not available as an indicator of future demand. However, because many industries are facing shorter product life cycles [1–3], new product forecasting gains importance. Besides the challenges regarding the lack of historical data, there is limited analysis time and there exists a general uncertainty related to consumer acceptance and competitive reactions [1,3–6]. Despite their complexity, these initial forecasts are essential for the operations of a company, as they guide important decisions like capacity planning, procurement and inventory control [3,5,7]. Because these decisions are guided by forecasting, a proper sales forecasting approach is key to prevent complications during or right after the product launch. Poor forecasts can result in stock-outs or overstock situations, which have a direct impact on the company's profitability and may also decrease customer satisfaction and market share [1,2,8].

Since companies require an increasing number of new product forecasts, the need for analytical approaches to new product forecasting rises. Goodwin et al. [6] argue that quantitative models should be at the core of the forecasting process of new products. However, Kahn [10] states that the use of analytical methods for new product forecasting is still limited among companies. According to Kahn [9], expert opinions, surveys and the average sales of similar products are the most widespread techniques for predicting the demand of new products. Kahn [10] points out that these techniques should focus on creating meaningful estimations to anticipate the risks of new product introductions. Meaningful estimations should not only focus on point forecasts but should provide insight into the level of uncertainty surrounding new product introductions by means of prediction intervals. Nevertheless, researchers paid little attention to analyzing the uncertainty among new products [6].

To create new product forecasts and quantify the uncertainty, we develop a novel new product forecasting method called *DemandForest* to support supply chain planners. DemandForest is a hybrid method combining K-means clustering, Random Forest [11] and Quantile Regression Forest [12]. Forecasts are generated by utilizing the historical demand of existing products and the product characteristics of both new and existing products. The approach is widely applicable to companies in different industries due to the general applicability of the Random Forest algorithm. Furthermore, the Quantile Regression Forest (QRF) algorithm quantifies the uncertainty of the demand and can be used to construct prediction intervals and support inventory management decisions.

* Corresponding author.
E-mail address: r.m.vansteenbergen@utwente.nl (R.M. van Steenbergen).

https://doi.org/10.1016/j.dss.2020.113401
Received 28 January 2020; Received in revised form 28 July 2020; Accepted 5 September 2020
Available online 09 September 2020

This research contributes to the field of new product forecasting in several ways. First, to the best of our knowledge, this research is the first application of QRFs in new product forecasting, while we also evaluate the impact on inventory management. Second, we propose an extension of QRF by fitting theoretical distributions to the quantiles. Third, we propose a synthetic data set that can be used for future comparisons between various new product forecasting methodologies. Finally, we show that the feature importance and list of comparable products can be extracted from Random Forest algorithms to provide valuable insights for supply chain planners.

The remainder of this paper is structured as follows. In the next section, we discuss work related to new product forecasting based on product characteristics. After describing the problem in Section 3, we propose our method DemandForest in Section 4. Section 5 presents the data and experimental setup. The results obtained from experiments are reported in Section 6. Finally, we provide the conclusions of our research in Section 7.

2. Literature review

Quantitative methods for new product forecasting are mainly based on analogous forecasting to overcome the lack of demand history [13]. This means that these models use data of similar products to generate forecasts for new products. In this section, we focus on these quantitative analogous forecasting methods. The critical assumption with analogous forecasting is that similarity between products translates into similar demand patterns. However, there are no assurances that the historical demand of analogous products corresponds to the future demand of new products [10] and this should be handled with care. Methods based on diffusion models, such as the Bass model proposed by [14], are out of scope of this section. Diffusion models are mainly used to model emerging technologies or new-to-the-world products [9], such as the adoption of electric cars in Europe. Applying these models for predicting the demand of new products on SKU-level might lead to significant errors.

An early forecasting model is proposed by Neelamegham and Chintagunta [15]. This model was designed to predict the number of viewers of movies in their introduction week, using movie attributes such as genre and presence of movie stars. The model showed proper predictions on movie-country level. A Bayesian approach was applied to provide a measure of uncertainty of the forecasts. Thomassey and Fiordaliso [16] proposed a forecasting method for mid-term forecasting in the apparel industry. First, a K-means algorithm clusters the demand patterns of existing products into distinctive profiles. Afterward, they trained a decision tree with the C4.5 algorithm based on price, starting time of sales, and life span of items. With this classification tree, Thomassey and Fiordaliso [16] could predict the demand profile of future items. The predicted profile, which is given by the mean pattern of the underlying existing products, was used as a forecast for future items. Thomassey and Fiordaliso [16] showed that the decision tree classification performed better than models such as the Naive Bayesian classifier and the nearest neighbor classifier. One year later, Thomassey and Happiette [17] applied the same procedure with neural clustering and classification techniques. Demand profiles were derived by using a Self-Organising Map and K-means clustering. Predicting the demand profile was performed by a Probabilistic Neural Network. However, this model did not perform better than Naive Bayes classification. An extreme learning machine (ELM) was applied by Sun et al. [18] to forecast the sales of fashion items, which outperformed other neural network-based methods. The relationship between sales and product characteristics, such as color, size, and price, were investigated.

Szozda [19] proposed a forecasting method that compared the initial sales of a relatively new product to the initial sales of existing products. Hence, this method did not generate a pre-launch forecast but used the initial sales as input. The time series were adjusted in order to find similar demand patterns with a different scale. The demand shape of the existing product with the highest similarity was used to adjust the sales forecast of the new product. Basallo-Triana et al. [2] applied a fuzzy Gustafson-Kessel algorithm for clustering the time series of analogous products. Instead of generating a pre-launch forecast, Basallo-Triana et al. [2] assigned new products to a cluster based on the initial sales. For each time period, new products are assigned to a cluster.

Fallah Tehrani and Ahrens [20] forecast sales combining classification and regression models. A probabilistic approach identified the class of products in terms of sale. Thereafter, a kernel machine approach was used to predict the number of sales. The combined approach showed robust and promising results. Baardman et al. [3] proposed a scalable algorithm that iteratively determined distinctive groups of similar existing products. New products were probabilistically assigned to these groups based on their product features using multinomial logistic regression. The pre-launch forecasts were generated using multiple linear regression, support vector machines and neural networks. The application of multiple algorithms was also investigated by Loureiro et al. [8] for the prediction of the total sales of new fashion items. The predictions were based on product characteristics and the expectation level of the sales. With k-fold cross-validation, multiple algorithms, such as deep neural networks, Random Forest and support vector regression were analyzed. The lowest forecast errors were achieved by a deep neural network and a Random Forest algorithm. Although deep neural networks showed good potential, Loureiro et al. [8] suggest that a Random Forest algorithm is more suitable in practice, because it provides satisfactory predictive performance and the training process is less complex.

To conclude, several analytical methods have been proposed for new product forecasting, but most literature about new product forecasting has not considered the uncertainty in the demand for new products and its impact on inventory management practices. In this paper, we present a pre-launch forecasting approach that estimates the uncertainty of demand with Quantile Regression Forests and we evaluate the performance of the approach with multiple inventory management cases. Furthermore, we combine several concepts from related literature, such as predicting clustered demand patterns [16] and applying Random Forest as a suitable algorithm in practice [8].

3. Problem description

We study the problem of multi-period forecasting of the demand for new products. We use a fixed forecast horizon (i.e., the introduction period) of T periods (e.g., weeks), and generate a forecast at $t = 0$ for each period $t = 0, \ldots, T$. This forecast will not be updated when new data becomes available during the forecast horizon. Since we consider a pre-launch forecast, no historical demand data is available on the new product $y \in \mathscr{Y}$. However, we assume that historical demand data on comparable products $x \in \mathscr{X}$ is available. We use $Z = \mathscr{X} \cup \mathscr{Y}$. Each product $z \in \mathscr{Z}$ can be characterized by a set of features $f \in \mathscr{F}$, and the feature values f_z of both the existing and new products are known at $t = 0$. From now on, we express t relative to the start of the introduction period, for both, existing and new products.

The forecast should consist of two elements: (i) the total demand during the T periods and (ii) a profile representing the division of total demand over the T periods. The total demand d_x^h of an existing product x is given by the sum of demands during the T periods in the forecasting horizon by: $d_x^h = \Sigma_{t=0}{}^T d_{x,t}$. The normalized demand profile p_x of product x is given by the vector $p_x = (p_{x,t})_{\forall t}$, with $p_{x,t} = d_{x,t}/d_x^h$. The forecast for new products $y \in \mathscr{Y}$ should now be made by finding analogous products $x \in \mathscr{X}$ based on the feature values $f_z \ \forall z \in \mathscr{Z}$; the forecast should utilize historical demand data $d_x{}^h$, p_x, and $\widehat{d}_{x,t}$ of these analogous products x to forecast the demand of new products y given by d_y^h, p_y, and $d_{y,t}$. We denote the forecasted values of these demand components by $\widehat{d}_y^{\,h}$, $\widehat{p}_y$, and $\widehat{d}_{y,t}$ respectively. Besides these point forecasts, we also want to establish prediction intervals $[L_y, U_y]$ for $\widehat{d}_y^{\,h}$,

Table 1
Brief description of the companies and new products (similar characteristics to article Y are highlighted in bold).

Product	Supplier	Category	Price ()	Sales channel	Article type	Total demand
Y	**Sup1**	**Cat1**	**8**	**Chan1**	**Stat1**	**181**
X1	Sup2	**Cat1**	**9**	Chan2	Stat2	**186**
X2	**Sup1**	Cat2	17.12	**Chan1**	**Stat1**	300

which we transform to intervals $[L_{y,t}, U_{y,t}]$ for $\widehat{d}_{y,t}$, $\forall t$ (for the bounds, we typically use the 5th and 95th percentiles).

To illustrate the differences and similarities between characteristics, demand patterns, and total demand, we extracted three sample products from one of the case studies considered in this paper, with $X1$ and $X2$ representing existing products and Y the new product, see Table 1. In this example, the characteristics and the demand data of both the existing products and the new product are known. We see that product Y and $X1$ have the same category, price, and total demand. Product Y and $X2$ are also comparable, regarding the supplier, sales channel and article type, but the demand is almost twice as high for article $X2$. The normalized demand patterns throughout the introduction period (18 weeks in this example) are visualized in Fig. 1. We observe that the demand of product Y and $X2$ mainly occurs in the last weeks of the introduction period. On the other hand, product $X1$ shows a peak in one of the first weeks, showing that most of the demand occurs much earlier in the introduction period. Hence, if we want to make a demand forecast for the new product Y, we should ideally combine the demand data of the two comparable existing products $X1$ and $X2$, where we use the total demand of $X1$ and the pattern of $X2$.

To evaluate the classification performance in our methods, we use the classification accuracy and Cohen's kappa score (Cohen, 1960). The classification accuracy is the ratio of the number of correct predictions to the total number of predictions. The kappa is generally thought to be more robust compared to the classification accuracy, because it corrects the accuracy for chance (see Appendix A for the computation of kappa).

We measure the forecasting performance with the Root Mean Square Error (RMSE) for each individual period t for each product y:

$$RMSE_y(t) = \sqrt{\frac{\sum_{t}^{T} (\widehat{d}_{y,t} - d_{y,t})^2}{T}} \tag{1}$$

We do not use the more common Mean Average Percentage Error (MAPE), since the MAPE should not be used for evaluating forecasts with values of zero or close to zero (Hyndman et al., 2006) and the products in our data sets regularly have demands of zero in specific time periods.

Besides evaluating the predictions with the RMSE, we evaluate the performance of the prediction intervals. Several metrics have been proposed in the literature [21–23]. The metrics we will use are the Prediction Interval Coverage Probability (PICP) [23] and the Prediction Interval Normalized Average Width (PINAW) [24]:

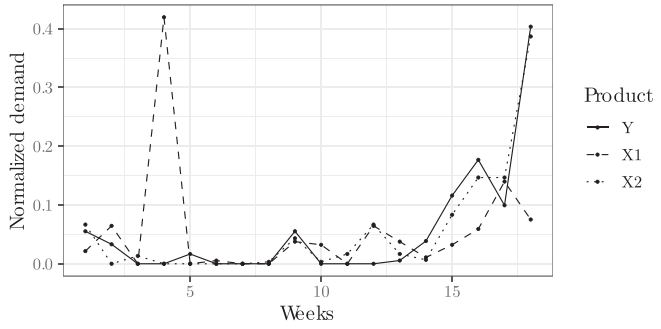


Fig. 1. Example of three demand patterns.

$$PICP_y = \frac{1}{t} \sum_{t=1}^{T} c_{y,t} \tag{2}$$

$$PINAW_y = \frac{1}{T} \sum_{t=1}^{T} \frac{(U_{y,t} - L_{y,t})}{R_t} \tag{3}$$

where $c_{y,t} = 1$ if $d_{y,t} \in [L_{y,t}, U_{y,t}]$, otherwise $c_{y,t} = 0$, and $R_t = max_{i \in y}(d_{i,t}) - min_{i \in y}(d_{i,t})$. The PICP should be equal to the target probability of the interval, thus with a 90% prediction interval the PICP should also be 90%. The lower the PINAW, the smaller and more valuable the intervals are. When we measure the RMSE, PICP, and PINAW for the complete horizon h, $\widehat{d}_{y,t}$ will be replaced by $\widehat{d}_y^{\,h}$.

In addition to evaluating the quality of the forecasts, we evaluate the benefit of these forecasts for inventory management decisions. Since the forecasts will be used for managing stock levels, the resulting performance of the inventory management decisions might be even more important than the forecast accuracy. The challenge is to provide a certain service level against minimal costs. We use the Cycle Service Level (CSL), which is the ratio of cycles that did not resulted in a stock-out relative to the total number of cycles. The CSL of product y is formulated as:

$$CSL_y = 1 - \frac{S_y}{N_y} \tag{4}$$

where S_y is the number of stock-out occasions during the introduction period of product y and N_y is the total number of cycles. A stock-out occasion occurs when the inventory on hand becomes negative. A new inventory cycle starts when an order is received.

Besides the CSL as performance measure for inventory management, we evaluate four different inventory costs: ordering costs, holding costs, excess holding costs, and lost sales costs. The ordering costs are the costs for each order placed during the introduction period. The holding costs are the costs of keeping products in inventory during the introduction period, defined as a percentage of the purchase costs of a product. Excess holding costs are considered as a penalty for keeping too many products in inventory, which is defined as the expected holding costs after the introduction period. With a high number of products in inventory and a low demand of the new product, the excess holding costs increase. The last type of cost is the lost sales costs, which is a penalty for having too few products in inventory, such that some orders cannot be satisfied.

4. Solution methodology

To generate multi-period pre-launch forecasts for new products, we propose a new method denoted by DemandForest. With DemandForest, we combine K-means, Random Forest, and Quantile Regression Forest algorithms to generate a forecast for the demand of a new product. With these machine learning algorithms, we can apply DemandForest to a wide range of companies and train the algorithms with data of a specific company to generate predictions for their new products. This method is inspired by the work of Thomassey and Fiordaliso [16] and Loureiro et al. [8]. We combine the concept of profile predictions of Thomassey and Fiordaliso [16] with Random Forests as applied in Loureiro et al. [8]. Furthermore, we enhance the methodology by predicting quantiles with Quantile Regression Forests and extend this with the use of theoretical distribution functions in order to improve quantiles when a limited number of existing products is available. First, we briefly discuss K-means, Random Forest, and Quantile Regression Forest in Sections 4.1, 4.2, and 4.3. Thereafter, we describe the steps of DemandForest in Section 4.4 and describe the extension to improve quantiles in Section 4.5.

4.1. K-means

For clustering the demand patterns, we use the K-means algorithm. The K-means algorithm is the most widely used clustering method in practice and is effective and efficient in most cases [25,26]. It is also applied to cluster demand patterns [16,27–29]. K-means partitions the observations into K clusters, such that each observation is assigned to the cluster that has the nearest mean. K-means aims to minimize the within cluster sum of squares. Since K-means is a greedy algorithm, it converges to local minima based on the initial partition. To overcome this limitation and increase the chance of finding the global minimum, the algorithm can be run with multiple initial partitions. The partition with the smallest sum of the squared errors can be chosen as final partition [26].

The most important parameter choice of the K-means algorithm is the number of clusters K [26]. To determine the number of clusters, it is recommended to use the majority vote of multiple cluster validity indices, since one index does not demonstrates clear advantages over other indices [30]. This research applies the Davies-Bouldin Index Davies-Bouldin Index [31], the Silhouette Coefficient [32], and the Calinski-Harabasz Index [33]. These are among the top indices recommended by [30] and also used by for example [34–36]. The indices are all based on the comparison of the tightness within a cluster and the separation between clusters. The Davies-Bouldin Index measures the ratio between the average distance from the samples in a cluster to its cluster center and the distance between the cluster centers. The Silhouette Coefficient compares the distance between each sample in the same cluster to the samples in the neighboring cluster. The Calinski-Harabasz Index determines the tightness and separation with the variability of the observations within each cluster and the variability between points in different clusters. The recommended number of clusters is obtained with higher scores of the Silhouette Coefficient and Calinski-Harabasz Index and lower scores of the Davies-Bouldin Index.

4.2. Random forest

Random Forest (RF) is an ensemble of decision trees for both classification and regression, introduced by Breiman [11]. Random Forest grows a set of parallel decision trees. The final prediction of an RF is the majority vote of each tree (with classification) or the mean prediction of the individual trees (for regression). RF is widely used due to its efficiency, and robustness to outliers and noise [11,37]. It overcomes overfitting and obtains substantially better results due to bootstrapping and the random selection of features. There are only two main parameters, which are relatively simple to tune and provide proper results over a wide range of values. Additionally, Random Forests can be applied to various types of data sets, because it can handle continuous and categorical data, as well as handling missing values [38]. RF comes with two components that provide insight into the model: feature importance and proximity. The importance is a measure of how much individual features contribute to the overall performance. The proximity is a measure of comparability between samples. The proximity between two samples is the percentage of trees in which they end up in the same leaf node. Among others, the proximity can be used for outlier detection, clustering, and missing data imputation [39].

4.3. Quantile regression forest

Meinshausen (2006) introduced Quantile Regression Forest (QRF), which gives a non-linear and non-parametric way of estimating conditional quantiles. QRF grows a set of trees in the same way as Random Forest. However, instead of keeping only the mean of the observations, QRF keeps all observations for each leaf node in each tree. Using all observations, QRF approximates the full conditional distribution and can provide a prediction interval for new observations. Quantile Regression Forest has recently been successfully applied in quantifying uncertainties in other fields, such as photovoltaic electricity production [40], weather forecasting [41], soil mapping [42], and wind power forecasting [43].

4.4. DemandForest

DemandForest begins with the data of the existing products x of a company (i.e., all previously introduced products). The data of these products consists of the product features f_x, the normalized demand patterns over the forecast horizon p_x, and the total demand during the complete forecasting horizon $d_x{}^h$. Thereafter, the K-means algorithm is used to cluster the demand patterns p_x into distinct profiles. The number of profiles (i.e., clusters) is determined by the majority vote of the Davies-Bouldin Index, the Silhouette Coefficient, and the Calinski-Harabasz Index. After clustering the demand patterns into profiles, a Random Forest is trained based on the set of features $\mathcal{F}$ to classify the profile. Besides the classification of the profile based on the features, a Quantile Regression Forest algorithm is trained on the features to predict the total demand $d_x{}^h$. The advantage of a QRF algorithm compared to a RF algorithm, is that not only the total demand, but also the full conditional distribution can be estimated. The full conditional distribution provides insight into the potential uncertainty of the total demand of a new product. By training both the RF and the QRF algorithm, the DemandForest model is capable of predicting the profiles, total demand, and distributions for new products.

In the next phase, the data of the new products y are utilized. The only data about the new products y that is known beforehand are the product features f_y. The product features are the input to the trained RF and QRF algorithms. The RF algorithm, which is trained to predict the profile, determines the predicted profile $\widehat{p}_y$ for the new products y. The QRF algorithm, which is trained to predict the total demand, determines the total demand $\widehat{d}_y^{\,h}$ and the conditional distributions given the new product features f_y. From the distribution, we can derive a lower bound L_y and an upper bound U_y for the total demand of each product. The lower and upper bound can be for example the 5^{th} and 95^{th} percentiles of the distribution. In the last step, the predicted profile with a ratio for each time period t and the total demand are multiplied to obtain a forecast $\widehat{d}_{y,t}$ for the T periods. To obtain the prediction intervals $[L_{y,t}, U_{y,t}]$ for the T periods, the predicted profile and the lower and upper bounds can be multiplied. Since it is not possible to order or sell half a product, the forecast in each time period t is rounded towards the closest integer. The method is illustrated schematically in Fig. 2. Besides predicting the demand and the lower and upper bounds, safety stock levels can be determined using the conditional distribution. The quantiles of the conditional distribution translate to the probability of not having a stock-out, which is equal to the target Cycle Service Level. In this way, we do not have to estimate the variability of the demand to determine stock levels. Instead, we can directly use the quantile predictions that correspond to certain target Cycle Service Levels. Hence, DemandForest is not only a forecasting method, but also functions as a system to support inventory management decisions.

4.5. Extension of DemandForest

Due to a limited number of comparable products, it might be possible that the predicted empirical distribution of a new product only consists of a limited number of values. For example, the upper quantiles in the empirical distribution in Fig. 3 are limited to 38, whereas the Log-Normal and Gamma distribution provide a smoother and presumably more accurate distribution.

To overcome this limitation of the empirical distribution of the QRF algorithm, we extend this algorithm by fitting a theoretical distribution over the empirical distribution. Because the demand data in all data sets are right-skewed and non-negative, we will fit Log-Normal and Gamma distributions to the distributions generated by the QRF algorithm. The

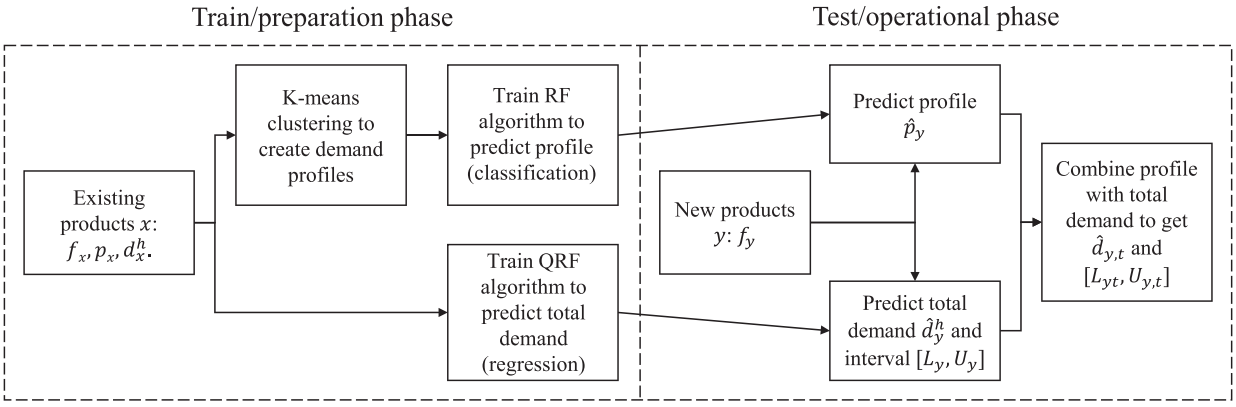


Fig. 2. Schematic overview of DemandForest.

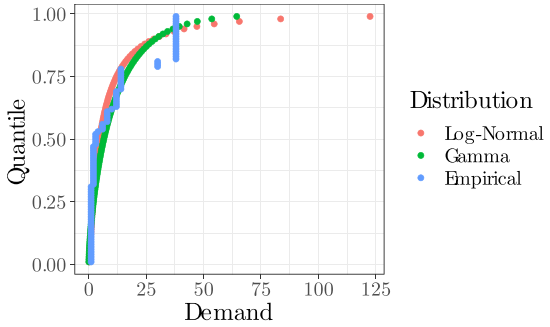


Fig. 3. Example of an empirical distribution and fitted theoretical distributions.

Log-Normal distribution is often used to model the demand in an inventory model, see, e.g., [44,45]. The Gamma distribution is also often used for the distribution of demand within inventory control literature [46–48]. To fit distributions, we use quantiles 0.01 to 0.99 with a step size of 0.01 and fit the Log-Normal and Gamma distribution to this data of each prediction.

5. Experimental setup

We evaluate the forecasting performance of DemandForest with multiple real-world data sets and a synthetic data set. We describe the data collected from industry partners in Section 5.1. In Section 5.2, we describe the synthetic data set. In Section 5.3, we define two benchmark methods that we use for comparison. Finally, the experimental setup and inventory cases are described in Section 5.4.

5.1. Data from industry partners

We collected data sets from five companies, which we anonymize using the letters A to E. The product characteristics and historical sales data of previously introduced products were retrieved from the databases of the inventory management systems of these companies. For this research, the obtained historical sales data is recorded weekly.

Company A is the only company that keeps track of the date at which products were introduced. For the other companies, we needed to estimate the introduction date. We assume that the week of introduction is the week in which the first sale occurs. This results in a bias in the data, since products may have been introduced earlier and not sold directly. Spare parts are an example of products that are not likely to be sold directly after their introduction. The inability to retrieve the actual introduction date of the products is a limitation we need to take into account in the remainder of this paper. Once we determined the introduction date, the sales data of the first 18 weeks (approximately four months) of the products were selected. From these products, only products were included that were kept in inventory and of which the first 18 weeks of historical sales were available. Products with stock-outs during these 18 weeks were included in the selection. However, products that were in total more than 14 days out of stock were excluded because the historical sales data of these products did not reflect the actual demand properly.

A brief description of each company and the data is given in Table 2, where the characteristics have been defined by the companies themselves. We intended to select data sets from companies operating in different industries and markets that sell different types of products. With this diversity of data sets, we can analyze whether the proposed methods are applicable for a wide range of companies. Company A has put effort in their categories and product groups. These product characteristics are numerical instead of categorical. Therefore, comparable categories and groups have values close to each other. For example, categories with electrical devices all have a value between 300 and 399, whereas categories with tableware have a value between 800 and 899. Therefore, we handle these categories and groups as numerical values. Additionally, the collection type, space facing, and circle type are also numerical. Therefore, the only categorical characteristic of Company A is the supplier. For the other companies, all product characteristics, except the prices and margins, are categorical. For example, the data set of Company E contains 48 suppliers, eight categories and 44 different brands.

5.2. Synthetic data set

Besides the real-world data sets, we also evaluate the proposed methods with a synthetic data set [49]. This data set can also serve as a benchmark instance for future research. Since we generate the synthetic data set from a given model, there certainly exist relationships between the product characteristics and the demand. In the real-world data sets, this may not be so clear. Therefore, validating the proposed methods with both a synthetic data set and real-world data sets is interesting. When the proposed methods can achieve similar results for the synthetic data set and the company data sets, it underlines the applicability of DemandForest in practice.

The products from our synthetic data set are characterized by color, category, brand, and price. The names of the colors, brands, and categories are chosen arbitrarily. The demand during T time periods is defined by a profile and the total demand. The profiles are, arbitrarily, one with a 10% exponential increase per time period, one with a 10% exponential decrease per time period, and one stable profile. The total demand is generated randomly with the Gamma distribution using $\alpha = 2$ and $\beta = 150$. The color and price relate to the demand, whereas the category and brand relate to a profile. Specific colors, categories, and brands relate to a specific demand segment or profile based on categorical distributions. Categories and brands relate in 80% of the cases to a specific profile and otherwise randomly to one of the other

Table 2
Brief description of the companies and new products.

Comp.	Industry type	Market	Type of products	Product characteristics	#prod.
A	Retail	B2C	Household items	Supplier, category, sales price, margin, collection type, product group, space facing, circle type	16,229
B	*E*-commerce	B2B & B2C	Lighting	Supplier, category, purchase price, sales channels, article type	3197
C		B2C	Sanitary ware	Supplier, category, subcategory, subsubcategory, sales price, margin, product type, brand, brand collection	592
D	Wholesale	B2B	Agricultural machinery (spare) parts	Supplier, category, purchase price, brand	1172
E	Wholesale	B2B	Garden tools and forestry machines	Supplier, category, purchase price, brand	660

profiles. We divide the demand into five equal segments, to relate the color to the demand. Colors relate in 80% of the cases to a specific demand segment and otherwise randomly to one of the other segments. The price is a numeric characteristic that is inversely proportional to the demand: $price = 2000/demand$. We apply noise to this inverse proportional relationship by adding a coefficient of variation of 0.5 to the price.

5.3. Benchmark methods

To compare the proposed methods with the current situation, we ideally use the actual historical manual forecasts of planners. However, this data is not available. Therefore, we need to define other types of benchmarks.

The first benchmark we define is Zero Rule, abbreviated to *ZeroR*, which simply predicts the average output of the training data for the test data [50]. In our case, it uses the average demand of each period t of the training data as a forecast for each new product. For prediction intervals or safety stock calculations, we extend ZeroR by using the quantiles of each period t from the training data. ZeroR is a simple benchmark, but it does not distinguish between product characteristics and does not accurately represents the current situation of forecasting at the selected companies. Therefore, we also want to use a method that imitates the decisions of a human planner. Planners usually discuss the forecasts during Sales and Operations Planning (S&OP) meetings with managers and they often base their initial forecasts on a similar existing product. According to the involved companies, a planner is usually able to manually identify a similar existing product.

To imitate the manual identification of similar products, we can use the proximity measure that can be extracted from a trained Quantile Regression Forest algorithm. The proximity between two products is the percentage of trees in which they end up in the same leaf node. Since a QRF algorithm partitions the products based on the characteristics for which the demand is similar, i.e., products with both similar characteristics and similar demand are likely to end up in the same leaf nodes. Hence, we can identify existing products with a high similarity, i.e., high proximity, compared to new products. We use the proximities of the Quantile Regression Forest to create our second benchmark, which we denote by One Proximity, abbreviated to *OneP*. OneP uses the proximity meausure to determine for each new product the one most similar existing product. The total demand of the most similar existing product is then used as prediction for the total demand of the new product. By using the total demand of the most similar existing product, OneP resembles the domain knowledge or experience of a planner. Together with the total demand, we use the average profile of all products as profile for OneP. We multiply the total demand of the most similar existing product with the profile to obtain the forecast for each new product.

By using the most similar product from the proximity, OneP does not come with quantiles. To use OneP for inventory management, we assume a Normal distribution for the uncertainty of demand. A Normal distribution is commonly applied in practice. Also at our industry partners, typically a Normally distributed monthly demand of new products is assumed, with a coefficient of variation of 0.45. Because the total demand is in our case approximately four months, we scale the coefficient of variation: $0.45 \times \sqrt{4} = 0.9$. By using the total demand from the most similar existing product and the uncertainty represented by the Normal distribution, we can set target service levels with OneP for the inventory management cases.

5.4. Experimental design

To evaluate the performance of our method DemandForest, we carry out experiments in several phases. The introduction period in the data sets consist of $T = 18$ weeks (approximately four months). First, we partition each data set into a training set (i.e., existing products) and a

test set (i.e., new products). The training set consists of a random sample of 75% of the data, while the test set consists of the remaining 25%. For the clustering with K-means, we run the algorithm 25 times, as suggested by Jain [26], to increase the chance of finding the global minimum. Considering the number of trees in the Random Forest and Quantile Regression Forest algorithms, usually 500 trees are sufficient to stabilize the performance (Oshiro et al., 2012). Nevertheless, to ensure that the number of trees is sufficiently large for all experiments, we set the number of trees in the Random Forest and Quantile Regression Forest to 2000. Furthermore, based on the out-of-bag predictions of the profile and total demand, we explore the values for mtry for each data set. We evaluate the value for mtry from one to the number of features $|F|$. We choose the mtry with the highest out-of-bag accuracy (for the profile prediction) and the lowest out-of-bag Root Mean Square Error (RMSE) (for demand prediction). Then, we evaluate the individual steps of DemandForest, the combined forecast, and the performance in the inventory management cases.

Since new products are not included in the K-means clustering, we do not know the actual profiles and cannot determine the accuracy and kappa. Nevertheless, we can determine the profiles to which K-means would have assigned the new products. We determine the Euclidean distance between the demand patterns of new products and the profiles of K-means. The profile with the lowest Euclidean distance is assigned to the new products. Thereafter, we determine the accuracy and kappa score.

After evaluating the profile prediction with the accuracy and kappa score, we determine the total demand and 90% prediction intervals of the total demand. We evaluate the forecast error of the total demand with the RMSE, and the prediction intervals with the PICP, and the PINAW. Thereafter, we combine the predicted profiles with predictions of the total demand. These combined forecasts are evaluated with the RMSE considering each time period t.

After evaluating the forecasts, we apply DemandForest in three inventory management cases. We do not evaluate the products with their specific lead times. Instead, we consider three cases for the lead times. The first two cases include inventory replenishments with lead times of one and six weeks. For the third case, we only order once at the beginning of the introduction period.

The replenishment policy we use for the inventory cases is a (R, s, S) policy, which stands for a policy with a review time R, reorder level s and order-up-to-level S. For all cases, we use a review time R of one week (i.e., one period). The reorder level s is the similar to the expected demand during review time and lead time plus the safety stock. Generally, the safety stock calculation is based on the Normal distribution and determined by the target Cycle Service Level (CSL) and the standard deviation during review time and lead time. With a target CSL of 95%, the reorder level s is given by the 95^{th} percentiles of the Normal distribution with a mean and standard deviation of the demand. However, in the case of DemandForest, we do not use the Normal distribution, nor do we determine the standard deviation of the demand. Instead, we directly determine the reorder level s with the quantile prediction of DemandForest that corresponds to the target CSL. With a target CSL of 95%, we set the reorder level s to the 0.95 quantile of the predicted demand during the review and lead time. This value is equal to the 95^{th} percentile of the expected demand. In our analysis, the order-up-to-levels are set equal to the reorder levels, to obtain a good overview of the achieved CSLs compared to the quantiles. We do not determine Economic Order Quantities (EOQs) since the general EOQ calculations assume a known and stable demand, which is not the case for new products. For all cases, we evaluate the quantiles (i.e., target CSLs) and actual CSLs for the quantiles between 0.5 and 0.99 with a step size of 0.01. It is expected that the actual CSLs are equal to the used quantiles.

Besides the consistency of the service levels, we evaluate four different costs: ordering costs, holding costs, excess holding costs, and lost sales costs. To compare the results of the different data sets, we assume similar costs. We assume ordering costs of 25 euros for each individual order. For the holding costs, we assume 25% of the purchase price of a product as holding costs of keeping one product in inventory for one year. Besides regular holding costs, we take excess holding costs into account. Excess holding costs are the expected holding costs after the introduction period. When too many products are put in inventory in the introduction period, it may take a while to sell the excess products. Therefore, we penalize too many products into inventory, especially when the actual demand is low, by considering excess holding costs. To determine these excess holding costs, we assume a stable demand after the introduction period. We determined for each product the ratio between the average demand during the introduction period and the average demand in the year after the introduction period, if available in the databases of the companies. For example, when the average demand in the introduction period is 50 per period and the average demand after the introduction is 75 per period, the ratio is 1.5. When there is no demand data available, because the historical demand data is limited, we multiply the actual demand with the average ratio of the other products in the data set. For the synthetic data set, we make an assumption for the ratios: for the increasing profile, we use a factor of three, for the stable a ratio of one, and for the decreasing profile a ratio of 1/3. With these ratios, we determine the time that products will be in stock and the corresponding excess holding costs.

The last type of costs we consider are lost sales costs. We assume that when a product is out of stock, consumers go to a competitor and the sale is lost, which in principle might also affect future sales. To take this risk into account, we assume lost sales costs of two times the margin. For the data sets where this margin is unavailable, we assume a margin equal to the purchasing price.

The methods and experiments described in this paper are implemented in the R environment (R Core Team, 2014) using the packages Ranger [7] and fitdistrplus [51]. The Ranger package contains Random Forest [11] and Quantile Regression Forest [12], and fitdistrplus is used for the maximum likelihood estimations of the Gamma distribution.

6. Experimental results

To test the performance and validity of the proposed methods, we present the experimental results of DemandForest and the benchmark methods ZeroR and OneP in this section. We test the methods with the synthetic data set as well as with the data sets from the five companies. We cluster the demand profiles in Section 6.1. In Section 6.2, we present the results of predicting the profiles by the Random Forest algorithm and we discuss the predictions of the total demand of the Quantile Regression Forest in Section 6.3. In Section 6.4, we provide the forecast error for all methods. The performance of the quantile predictions is analyzed in Section 6.5 using the inventory cases. Finally, we show in Section 6.6 the feature importance and top five comparable products, which serve as additional insights for supply chain planners. Since we analyzed multiple methods, data sets, and inventory management cases, we only present the most interesting observations in the main text. For a complete overview of results, we refer to Appendix B.

6.1. Clustering

Unanimously recommended by the Davies-Bouldin Index, Silhouette Coefficient, and Calinski-Harabasz Index, the optimal number of clusters is two for all company data sets. For the synthetic data set, the optimal number of clusters according to the three cluster validity indices is three, as also defined in Section 5.2. The profiles of Company A, clustered by K-means, are displayed in Fig. 4. In the figure, we observe one concave increasing profile and one rather linear profile. This means that some products are mainly sold in the first weeks after the introduction, while other products are more stable during the weeks. Similar shapes of the demand profiles are obtained among the other

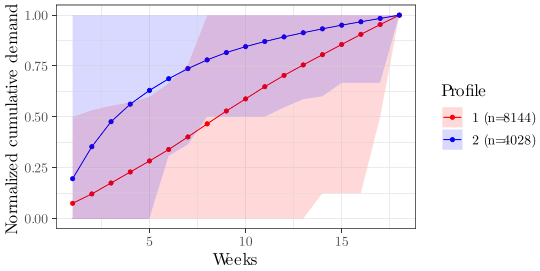


Fig. 4. Clustered profiles of Company A.

companies. The cluster with the peak in the first week can be partly the result of our assumption that the introduction period starts with the first sale.

6.2. Predicting profiles

With the Random Forest algorithm, we train and predict the profile for the products in the test set. We determine the actual classes for the test set by assigning the profile that has the minimum Euclidean distance to the demand pattern of each product. The accuracy and kappa determined afterward for each data set are presented in Table 3. We observe that the accuracy for each data set, except for data set C, is around 0.80. Nevertheless, Cohen's kappa shows less promising results. The kappa is relative good for the synthetic data set, moderate for Company A and E, and poor for Company B, C, and D.

These results show that it can be difficult to predict a clustered demand pattern with a Random Forest based on product characteristics. Nevertheless, we will apply these predicted profiles to the forecasts to analyze the performance of the complete forecast. First, we discuss the other component of DemandForest, the prediction of the total demand with a Quantile Regression Forest.

6.3. Predicting total demand

We predict the total demand and the 90% prediction interval for DemandForest, the extensions with the Gamma and Log-Normal distribution, and the benchmark methods OneP and ZeroR. For these predictions, we determine the RMSE, the PICP, and the PINAW, as discussed in Section 5.4. The results of all methods are displayed in Table 4. In the table, we see that DemandForest achieves a lower RMSE than the OneP and ZeroR method in almost all cases. Only for the data set of Company D, the OneP method is more accurate. Regarding the extended methods, the DemandForest + Log-Normal results in less accurate predictions than the regular DemandForest, whereas DemandForest + Gamma improves the accuracy for all data sets except for Company C.

The PICP shows the realized prediction interval, which should be close to the target prediction interval (in this case 90%), whereas the PINAW indicates the interval width, which should be small. For the synthetic data set, DemandForest + Gamma shows the best results, with a PICP of around 90% and one of the smallest PINAWs. The PINAW of DemandForest is smaller, but the PICP is less than 90%. DemandForest + Gamma is also the best for all company data sets. For all these data sets, the PICP is around 90% with a low PINAW. For Company C, DemandForest + Log-Normal also performs good, with a comparable PICP and PINAW. The OneP method shows small values for PINAW for Company C, D, and E, but the PICP for these data sets is only around 80%.

To further analyze the prediction intervals, we compute the 50%, 80%, 90%, and 98% prediction intervals of the total demand for all methods for the synthetic data set and plot the actual prediction intervals in Fig. 5. We analyze the consistency between the prediction intervals based on the quantile predictions (i.e., the target intervals) and the actual prediction intervals. The closer the actual intervals are to the target, the more reliable the intervals of a certain method. The DemandForest methods and the ZeroR method obtain prediction intervals close to the target prediction intervals, whereas the OneP method show deviations on the bottom and the top of the interval.

Table 3
Predictive performance of Random Forest for the profiles at each data set.

	Synthetic	A	B	C	D	E
Accuracy	0.824	0.802	0.746	0.568	0.826	0.770
Kappa	0.736	0.534	0.200	0.197	0.007	0.489

6.4. Forecasting performance of DemandForest

By combining the predicted profile with the total demand and 5^{th} and 95^{th} percentiles, we obtain a forecast with a 90% prediction interval on a weekly level for 18 weeks. We use the 90% prediction interval for the analysis of the forecasting performance of each method and data set. Table 5 shows the forecast errors RMSE, PICP, and PINAW.

The forecasts of the DemandForest methods overall provide the lowest forecast error. The results of Company C are the only exception with ZeroR as the lowest RMSE. Furthermore, at Company D, ZeroR provides a lower forecast error than DemandForest and DemandForest + Gamma. This corresponds with the lower kappa of Company C and D and relative higher prediction errors of the total demand, although it is not completely similar. The extensions of DemandForest provide a comparable (Synthetic, A, and B) or lower (C, D, and E) RMSE than the standard DemandForest.

Similar to the total demand prediction, the PICP is ideally 90%, whereas the PINAW should be small. For the PICP, the ZeroR method generally performs the best, with values around 90%. For the Synthetic data and the data sets of company A and B, the PICPs of the DemandForest methods are too low. This decrease of the PICP compared to the results of the total demand in Table 4 is likely to be caused by the profile, which does not seem to translate the prediction intervals well over the weekly time periods. The higher PICPs for the data sets of company C, D, and E are presumably caused by more homogeneous demand patterns. Although the predictability of the profiles was not high, the combination of the profile and the total demand resulted in proper intervals during the weekly time periods. The PINAWs are rather similar between the methods, only the PINAWs of the OneP and ZeroR method of the synthetic data set are much larger than the DemandForest methods.

We compute the 50%, 80%, 90%, and 98% prediction intervals of the weekly demand for all methods for the synthetic data set and plot the actual prediction intervals in Fig. 6. The consistency between the target intervals of 50%, 80%, 90%, and 98% and the actual prediction intervals indicates the reliability of the intervals of a certain method. The DemandForest methods provide intervals narrower than they should be, whereas the OneP method provides intervals that are too wide. Despite the lower forecasting performance, the prediction intervals of the ZeroR method are the most accurate. This is probably because of using the weekly percentiles of the training data, instead of applying profiles. For a further investigation on the quality of other percentiles, we study the impact of the 50th to the 99th percentile in the inventory management decisions in the next subsection.

6.5. Performance inventory management cases

Since the primary purpose of the forecasts is to support the decisions on inventory levels, the benefit in inventory management cases may be even more important than the forecast accuracy itself. The results of inventory management decisions demonstrate the actual quality of the

Table 4
Predictive performance of Quantile Regression Forest for the total demand and 90% prediction interval at each data set.

		Synthetic	A	B	C	D	E
RMSE	DemandForest	120.6	7.58	399.5	17.7	15.1	20.0
	DF + Gamma	119.7	7.50	386.0	17.8	14.8	19.7
	DF + Log-Normal	120.0	7.81	400.9	19.0	15.6	18.9
	OneP	161.8	8.14	529.0	23.2	14.2	27.3
	ZeroR	212.6	8.77	485.6	18.6	17.4	27.4
PICP	DemandForest	87.6%	93.3%	84.9%	94.6%	93.9%	91.5%
	DF + Gamma	90.6%	94.9%	87.6%	93.9%	92.2%	91.5%
	DF + Log-Normal	90.6%	94.0%	86.7%	93.2%	91.5%	90.3%
	OneP	96.8%	91.2%	82.5%	78.4%	81.2%	81.2%
	ZeroR	93.2%	90.0%	91.0%	95.3%	95.2%	95.2%
PINAW	DemandForest	23.4%	4.5%	12.6%	22.8%	15.5%	27.4%
	DF + Gamma	23.6%	4.0%	11.9%	18.6%	13.3%	25.0%
	DF + Log-Normal	25.0%	4.2%	15.9%	18.4%	14.6%	34.0%
	OneP	76.3%	5.1%	15.5%	18.2%	11.0%	21.6%
	ZeroR	55.9%	5.2%	19.3%	29.1%	14.4%	32.8%

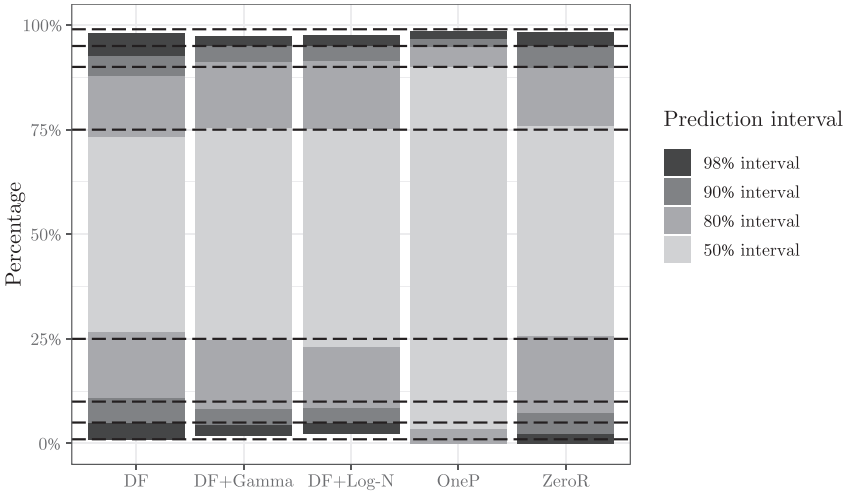


Fig. 5. Prediction intervals per method for the total demand in the synthetic data set.

forecasts. In this subsection, we discuss the expected outcomes of using the forecasts and intervals as input for inventory management decisions, in other words, using DemandForest for prescriptive analytics. For all methods, we analyze the consistency between the quantiles, which are the target service levels, and the actual Cycle Service Levels (CSLs) and we compute the resulting inventory costs. The consistency between the quantiles and the actual CSLs is important because it shows if the target that is set as input is achieved on various levels by the different methods. When the actual CSLs deviate largely from the targets, the method is not reliable, and companies cannot use it to achieve the desired service levels. Besides checking the consistency, we evaluate the inventory costs to analyze whether DemandForest can provide its forecasts and service levels against competitive costs.

First, we discuss the consistency between the quantiles and CSLs. To check the consistency, we analyze the results for the quantiles between 0.50 and 0.99 with a step size of 0.01. For the cases with a one-time

Table 5
Predictive performance of DemandForest and the 90% prediction interval at each data set.

		Synthetic	A	B	C	D	E
RMSE	DemandForest	10.8	0.681	32.5	2.22	3.59	3.61
	DF + Gamma	10.8	0.676	32.0	2.15	3.48	3.56
	DF + Log-Normal	10.7	0.683	32.6	1.83	3.03	3.31
	OneP	13.2	0.705	40.2	2.57	4.11	5.36
	ZeroR	15.2	0.784	36.4	1.78	3.15	4.24
PICP	DemandForest	70.5%	62.2%	66.2%	88.0%	97.5%	93.8%
	DF + Gamma	74.2%	56.7%	71.3%	87.5%	97.5%	93.6%
	DF + Log-Normal	74.9%	59.7%	67.6%	87.4%	97.4%	93.9%
	OneP	95.4%	49.9%	85.1%	85.4%	97.1%	92.1%
	ZeroR	89.9%	86.2%	85.7%	88.9%	97.4%	94.2%
PINAW	DemandForest	16.1%	4.1%	11.3%	26.3%	2.7%	9.5%
	DF + Gamma	16.3%	3.5%	19.3%	21.2%	2.4%	9.1%
	DF + Log-Normal	17.3%	3.8%	14.2%	20.8%	3.0%	13.2%
	OneP	52.6%	3.6%	13.8%	19.0%	2.2%	9.5%
	ZeroR	38.5%	5.1%	17.9%	33.2%	1.4%	8.8%

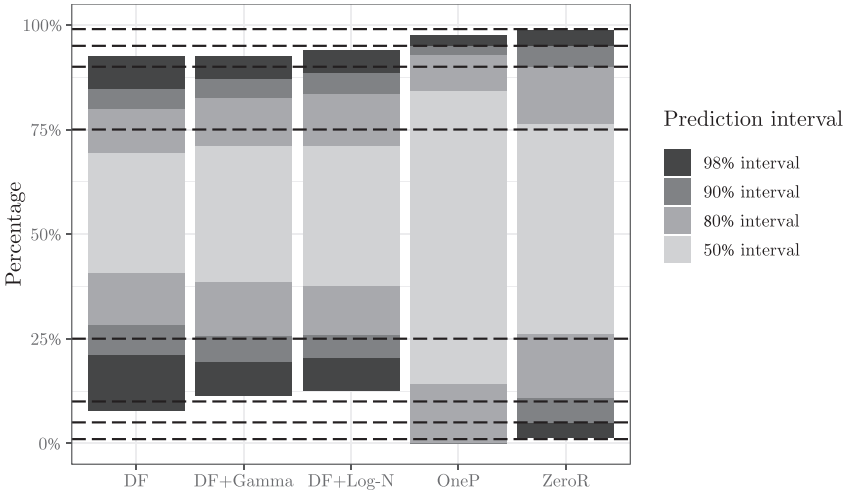


Fig. 6. Prediction intervals per method for the weekly demand in the synthetic data set.

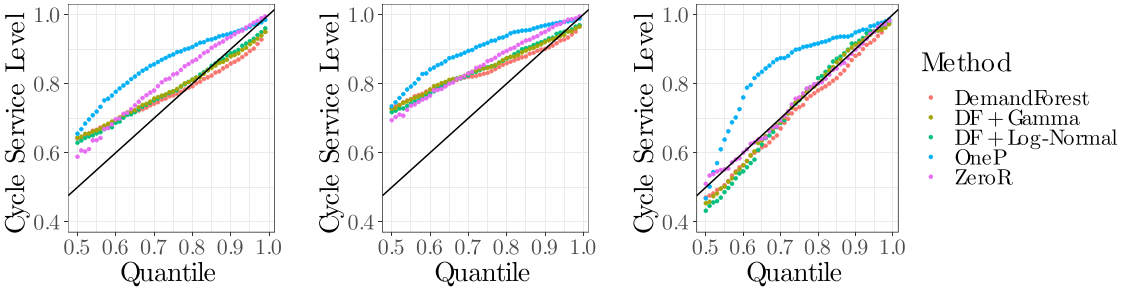
order, it is likely that DemandForest shows proper consistency since we use the QRF algorithm only. However, with the replenishment cases, we multiply the quantiles with the profiles. Hence, the actual CSLs might differ from the given quantile. A method is more reliable when the deviations between the quantiles and the CSLs are small because then a CSL similar to the target service level is achieved.

In Fig. 7, we show the quantiles and service levels for the synthetic data set for each case and method. All methods, except OneP, show reliable CSLs for the one-time order case (Fig. 7c). The CSLs for the OneP method are only comparable to the quantiles around 0.5 and close to 1. For the replenishment cases, we observe different results. Although the prediction intervals of the DemandForest methods were narrower than the target prediciton intervals, all methods, including the DemandForest methods, show actual CSLs higher than the quantiles. Again, the OneP method provides overall the most upward deviation and the DemandForest methods are the most reliable. Considering the lead time of one week, the ZeroR method is a lot higher for almost all quantiles. The DemandForest methods provide results that are closest to the actual CSLs between quantile 0.7 and 0.9, with no distinctive differences between the alternatives. Nevertheless, for the quantiles below 0.8, all the methods result in higher service levels than targeted, whereas the methods achieve lower actual CSLs above the 0.8 quantile. For the results of the data sets of the companies, we refer to Appendix B. Similar to the results of the synthetic data set, the DemandForest methods are the most consistent and the OneP method the least consistent.

Next, we consider the inventory costs for specific values for the Cycle Service Levels, namely 75%, 90%, and 95%, for each case. We consider the actual CSLs and not the quantiles, since similar quantiles lead to different CSLs, inventory levels, and incomparable costs between the different methods. To analyze the costs at the specified CSLs, all methods were analyzed on the quantiles 0.50 to 0.99 with a step size of 0.01. The costs of the actual CSLs closest to respectively 75%, 90%, and 95% are presented. Table 6 shows the inventory costs of the synthetic data set for the three different CSLs and for all methods. We observe that the DemandForest methods obtain lower results for all cases and service levels. Especially the regular DemandForest achieved low costs. ZeroR results in the highest costs, especially for the one-time order case. For the inventory costs for the data sets of the companies, we refer to Appendix B. For the data sets of the companies, the DemandForest methods achieved overall the most robust and competitive

Table 6
Inventory costs (in k euros) for each method on the synthetic data set (lowest costs are highlighted in bold).

Case	CSL	DF	DF + Gamma	DF + Log-Normal	OneP	ZeroR
Replenish, LT = 1	75%	**335**	341	339	440	458
	90%	**234**	239	238	279	285
	95%	209	**208**	210	225	251
Replenish, LT = 6	75%	**307**	318	319	445	513
	90%	**187**	192	194	263	391
	95%	**166**	169	172	222	457
One-time order	75%	182	**180**	180	239	657
	90%	**149**	149	152	229	1211
	95%	164	**161**	171	330	1673



(a) Replenish, LT=1 (b) Replenish, LT=6 (c) One-time order

Fig. 7. Consistency between quantiles and Cycle Service Levels of the synthetic data set.

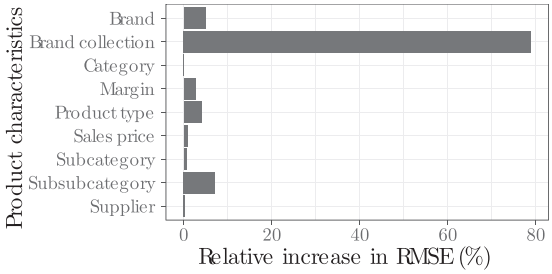


Fig. 8. Permutation feature importance of the total demand of Company C.

Table 7
A new product and the five most comparable products.

Article code	Color	Category	Brand	Price	Total demand
1501	Black	Tablets	Animity	51.4	71
231	Black	Tablets	Supranu	56.8	84
336	Black	Tablets	Supranu	46.46	42
756	Black	Tablets	Kayosis	59.39	28
491	Black	Computers	Verer	53.91	42
111	Black	Tablets	Hyperive	34.26	67

costs. Nevertheless, in some cases, the OneP and ZeroR methods obtained the lowest costs.

6.6. Feature importance and comparable products

The feature importance and a top five comparable products determined from the Random Forest algorithms can provide additional information to supply chain planners. First, we describe the concept of the feature importance. Thereafter, we provide an example of a top five comparable products.

The feature importance is a measure of how much individual features (e.g., product characteristics) contribute to the overall performance of a prediction. There are several methods for determining the feature importance, such as the Gini importance and the permutation importance. The Gini importance can be directly derived from Random Forest algorithms, but the main disadvantage is the bias towards continuous and high cardinality features, which are commonly present in our data sets. Therefore, we apply the permutation feature importance. The permutation feature importance can be determined with the out-of-bag data of Random Forest algorithms. It does not have the bias, but can overestimate correlating features [52]. The permutation feature importance can be determined by sequentially permuting each feature and calculating the decrease of the performance (e.g., the RMSE). With a large decrease, the feature contributes significantly to the overall performance and is regarded as important. With a low decrease in the performance, a feature is considered less important. Hence, the feature importance can show which product characteristics have the most influence on the predictions.

Fig. 8 shows feature importance for the data set of Company C as an example. The brand collection is by far the most important feature for both the profile and the total demand. This brand collection indicates the specific product lines of a brand. Permuting the category hardly changes the RMSE, which means that the category has hardly any predictive value for the demand. The brand and brand collection as well as the category, subcategory, and subsubcategory are hierarchical characteristics, which are handled as separate features in the Random Forest algorithms. The features with the lowest hierarchy (i.e., the brand collection and subsubcategory) show the highest importance. This indicates that more specific features can improve predictions.

The feature importance provides an overview of important characteristics in the complete data set but does not show product-specific insights. To provide more specific insights for a specific forecast, we can extract the top five comparable products from the proximity matrix of a Random Forest. As an example, we show the top five comparable products for one of the products of the synthetic data set in Table 7. In this table, we see that especially the first comparable product is similar in terms of characteristics and total demand. Providing supply chain planners with a top five for each individual forecast enhances the interpretability of DemandForest, which can support the adoption of using DemandForest. Furthermore, it can provide valuable inputs for S&OP meetings with colleagues and managers.

7. Conclusions

We developed a novel approach called DemandForest that provides pre-launch forecasts for the demand of new products during their introduction period. Our approach, relying on Random Forest algorithms, utilizes product characteristics of new and existing products to predict a profile and the total demand of these new products during the introduction period. Our approach also provides prediction intervals and quantiles, which can be used to support decisions in inventory management. The forecasts are based on the historical demand of existing comparable products to overcome the challenge of new product forecasting: the lack of historical data. In this way, DemandForest is an automated data-driven approach that is able to provide estimations reducing the need of human judgment, extensive analysis or years of experience. This method is especially valuable for companies with a large number of new product introductions each year. By using the forecasts, prediction intervals, quantiles and the most important product characteristics resulting from DemandForest, companies can support, enhance and also automate decision making in inventory management of new products.

We assessed the quality and performance of the proposed methods on a synthetic data set as well as various real-life data sets provided by five companies, from retail, e-commerce, wholesale and both B2B and B2C markets. With this variety of data sets, we showed that DemandForest is a generalizable computational approach that provides meaningful estimations (i.e., including uncertainty of demand) for both forecasts and inventory management decisions for new products. Considering the different data sets and cases, the DemandForest methods provide a higher forecasting quality, more reliable prediction intervals and service levels, and comparable or lower inventory costs than the benchmark methods. The extensions of the Log-Normal and Gamma distributions provided slightly better results than the regular empirical distributions of the QRF algorithm of DemandForest.

Besides the development of DemandForest, we contribute to new product forecasting techniques by exploring the performance of Quantile Regression Forest to quantify the uncertainty of demand. Furthermore, we showed that the empirical quantiles could be improved by fitting theoretical distributions. As additional insight for supply chain planners, we showed the possibility of extracting the most comparable products from the proximity of a Random Forest. In addition, we provided a synthetic data set that can be used for future validations of new product forecasting and inventory management methodologies. Future work can focus on more reliable methods to derive the profiles, including additional input features (e.g., seasonality, weather, or internet traffic), and combining it with other algorithms such as artificial neural networks, gradient boosting or support vector machines.

Appendix A. Computation of the kappa score

The kappa score is given by:

$$\kappa = \frac{p_o - p_e}{1 - p_e}, \tag{A.1}$$

where p_o gives the proportion of units in which agreed and p_e gives the proportion of units for which agreement is expected by chance. These values are given by

$$p_o = \frac{TP + TN}{TP + TN + FP + FN},$$
$$p_e = \frac{((TN + FP)(TN + FN) + (TP + FP)(TP + FN))^2}{(TP + TN + FP + FN)},$$

where T, F, P, N denote True, False, Positive and Negative respectively.

Appendix B. Extensive numerical results

The results for the inventory management cases of the five data sets of the companies is given in the figures and tables below. Figs. B.9, B.10, B.11, B.12, and B.13 show the consistency between the quantiles and the CLSs, whereas Tables B.8, B.9, B.10, B.11, and B.12 describe the inventory costs of each method at CSLs of 75%, 90%, and 95%. The inventory costs are missing when the highest quantile of the method did not provide a value close to a certain CSL.

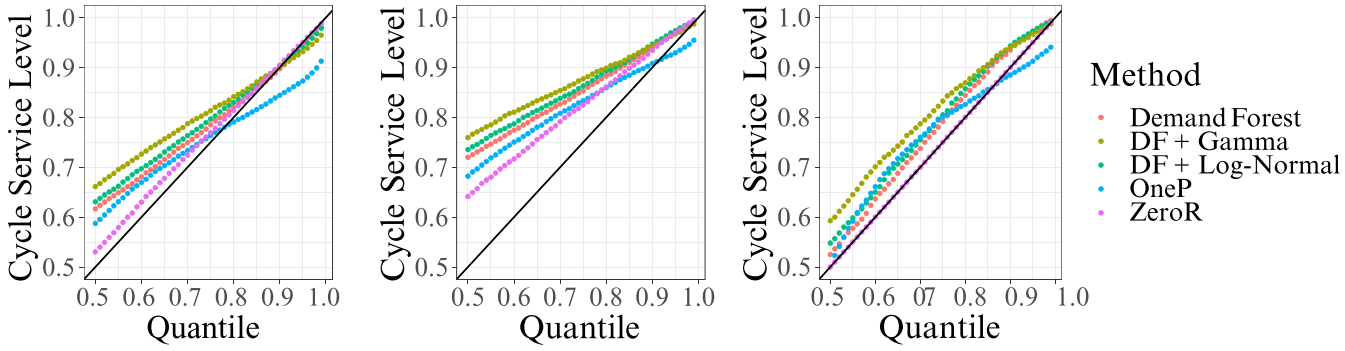


Fig. B.9. Consistency between quantiles and Cycle Service Levels of Company A.

Table B.8
Inventory costs (in k euros) for each method on the data set of company A (lowest costs are highlighted in bold).

Case	CSL	DF	DF + Gamma	DF + Log-Normal	OneP	ZeroR
Replenish, LT = 1	75%	1017	1011	1016	**971**	984
	90%	895	897	898	807	**793**
	95%	828	842	842	–	**652**
Replenish, LT = 6	75%	495	496	495	**474**	508
	90%	445	447	448	**406**	469
	95%	454	458	460	**367**	592
One-time order	75%	**52**	54	56	86	277
	90%	102	**99**	105	208	885
	95%	197	**181**	192	388	1957

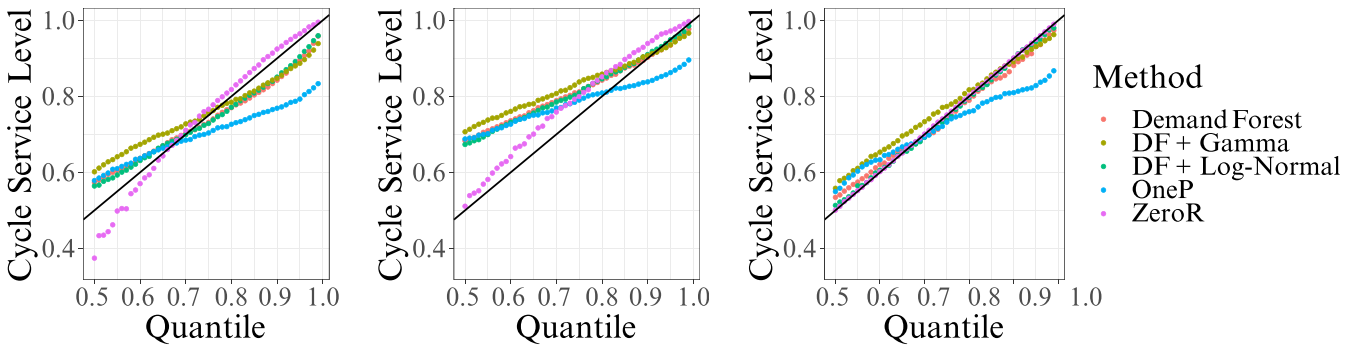


Fig. B.10. Consistency between quantiles and Cycle Service Levels of Company B.

Table B.9
Inventory costs (in k euros) for each method on the data set of company B (lowest costs are highlighted in bold).

Case	CSL	DF	DF + Gamma	DF + Log-N	OneP	ZeroR
Replenish, LT = 1	75%	942	954	937	**835**	1446
	90%	684	649	**612**	–	1267
	95%	691	**602**	621	–	1804
Replenish, LT = 6	75%	1049	1074	1066	**951**	1638
	90%	1039	**863**	875	873	4888
	95%	1377	1246	**1236**	–	12,564
One-time order	75%	943	906	**896**	1054	4139
	90%	1596	1219	**1200**	–	23,659
	95%	3266	**2480**	2616	–	57,791

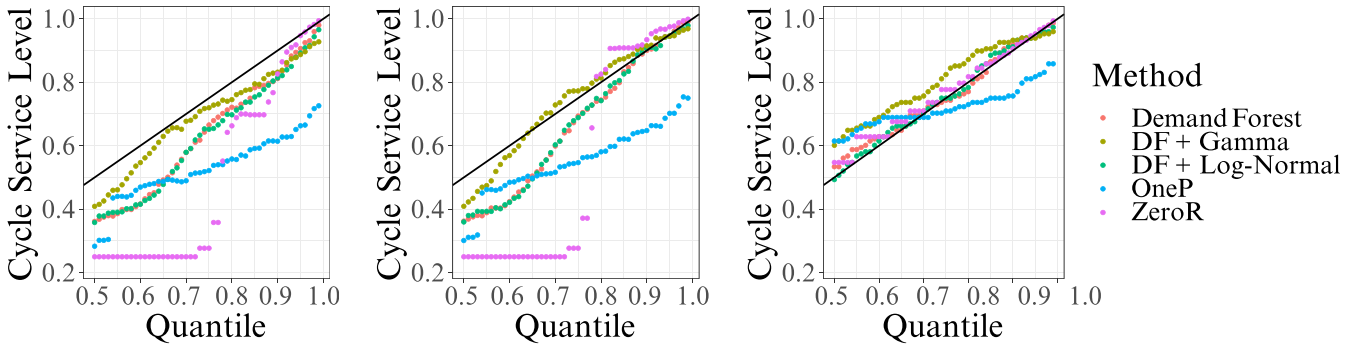


Fig. B.11. Consistency between quantiles and Cycle Service Levels of Company C.

Table B.10
Inventory costs (in k euros) for each method on the data set of company C (lowest costs are highlighted in bold).

Case	CSL	DF	DF + Gamma	DF + Log-N	OneP	ZeroR
Replenish, LT = 1	75%	113	116	112	–	**91**
	90%	306	262	200	–	**77**
	95%	1579	514	430	–	**97**
Replenish, LT = 6	75%	**108**	113	109	119	131
	90%	174	150	125	–	**96**
	95%	477	306	227	–	**165**
One-time order	75%	**99**	103	100	113	115
	90%	156	**134**	136	–	246
	95%	796	453	**355**	–	937

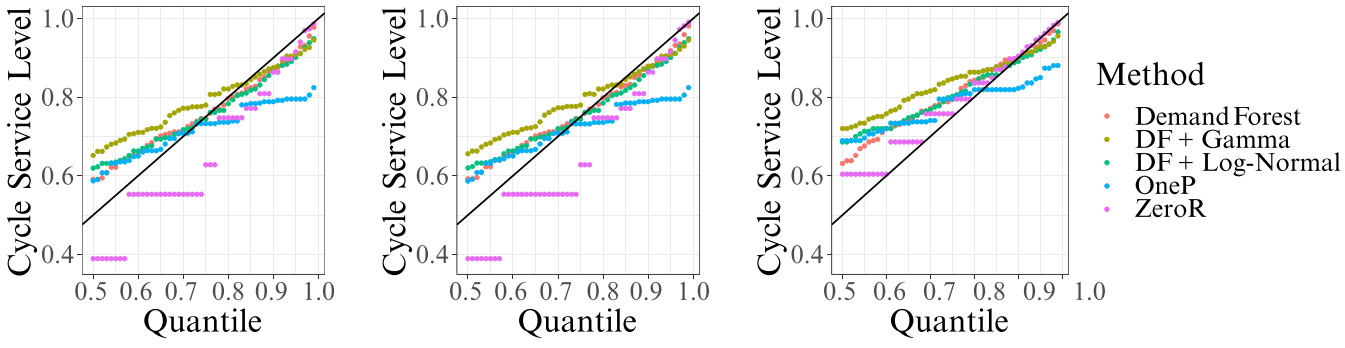


Fig. B.12. Consistency between quantiles and Cycle Service Levels of Company D.

Table B.11
Inventory costs (in k euros) for each method on the data set of company D (lowest costs are highlighted in bold).

Case	CSL	DF	DF + Gamma	DF + Log-N	OneP	ZeroR
Replenish, LT = 1	75%	40	40	**38**	65	44
	90%	149	**73**	97	–	129
	95%	411	**209**	657	–	313
Replenish, LT = 6	75%	41	40	**38**	69	44
	90%	156	**75**	102	–	129
	95%	437	**221**	711	–	353
One-time order	75%	33	34	**33**	58	38
	90%	70	58	**53**	59	122
	95%	**203**	227	310	–	697

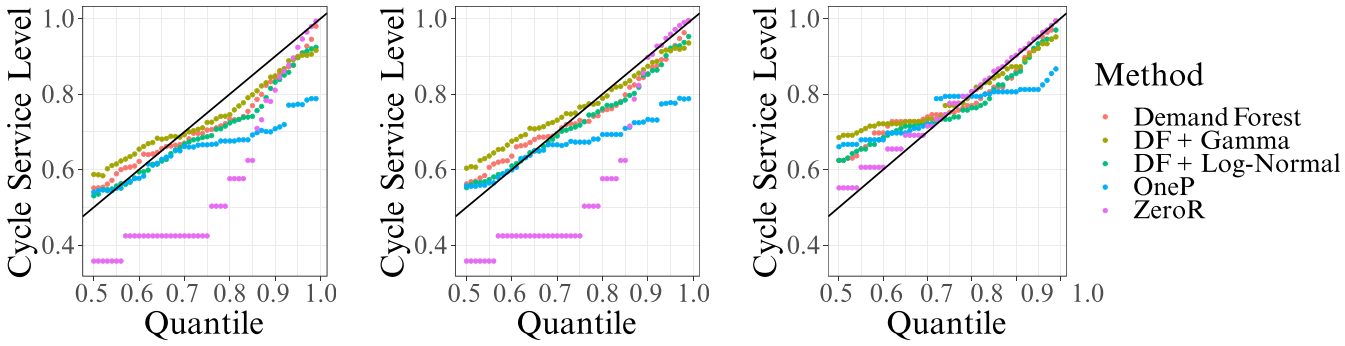


Fig. B.13. Consistency between quantiles and Cycle Service Levels of Company E.

Table B.12
Inventory costs (in k euros) for each method on the data set of company E (lowest costs are highlighted in bold).

Case	CSL	DF	DF + Gamma	DF + Log-N	OneP	ZeroR
Replenish, LT = 1	75%	146	138	**129**	245	207
	90%	458	291	336	–	**149**
	95%	1011	–	–	–	**161**
Replenish, LT = 6	75%	177	**150**	152	271	216
	90%	486	371	505	–	**199**
	95%	1218	1494	7045	–	**279**
One-time order	75%	204	204	**189**	231	215
	90%	1258	874	1026	–	**349**
	95%	1932	3583	4133	–	**1221**

References

[1] H. Lee, S.G. Kim, H.-w. Park, P. Kang, Pre-launch new product demand forecasting using the Bass model: a statistical and machine learning-based approach, Technol. Forecast. Soc. Chang. 86 (2014) 49–64, https://doi.org/10.1016/j.techfore.2013.08.020.

[2] M.J. Basallo-Triana, J.A. Rodríguez-Sarasty, H.D. Benitez-Restrepo, Analogue-based demand forecasting of short life-cycle products: a regression approach and a comprehensive assessment, Int. J. Prod. Res. 55 (2017) 2336–2350, https://doi.org/10.1080/00207543.2016.1241443.

[3] L. Baardman, I. Levin, G. Perakis, D. Singhvi, Leveraging Comparables for new product sales forecasting, Prod. Oper. Manag. 27 (2018) 2340–2343, https://doi.org/10.1111/poms.12963.

[4] G. Assmus, New product forecasting, J. Forecast. 3 (1984) 121–138, https://doi.org/10.1002/for.3980030202.

[5] D. Voulgaridou, K. Kirytopoulos, V. Leopoulos, An analytic network process approach for sales forecasting, Oper. Res. 9 (2009) 35–53, https://doi.org/10.1007/s12351-008-0026-2.

[6] P. Goodwin, K. Dyussekeneva, S. Meeran, The use of analogies in forecasting the annual sales of new electronics products, IMA J. Manag. Math. 24 (2013) 407–422, https://doi.org/10.1093/imaman/dpr025.

[7] M.J. Wright, P. Stern, Forecasting new product trial with analogous series, J. Bus. Res. 68 (2015) 1732–1738, https://doi.org/10.1016/j.jbusres.2015.03.032.

[8] A. Loureiro, V. Miguéis, L.F. da Silva, Exploring the use of deep neural networks for sales forecasting in fashion retail, Decis. Support. Syst. 114 (2018) 81–93, https://doi.org/10.1016/j.dss.2018.08.010.

[9] K. Kahn, An exploratory investigation of new product forecasting practices, J. Prod. Innov. Manag. 19 (2002) 133–143, https://doi.org/10.1016/S0737-6782(01)00133-3.

[10] K.B. Kahn, Solving the problems of new product forecasting, Business Horizons 57 (2014) 607–615, https://doi.org/10.1016/j.bushor.2014.05.003.

[11] L. Breiman, Random forests, Mach. Learn. 45 (2001) 5–32, https://doi.org/10.1023/A:1010933404324.

[12] N. Meinshausen, Quantile Regression Forests, J. Mach. Learn. Res. 7 (2006) 983–999.

[13] K.C. Green, J.S. Armstrong, Structured analogies for forecasting, Int. J. Forecast. 23 (2007) 365–376, https://doi.org/10.1016/j.ijforecast.2007.05.005.

[14] F.M. Bass, Comments on "a new product growth for model consumer durables the Bass model", Manag. Sci. 50 (2004) 1833–1840, https://doi.org/10.1287/mnsc.1040.0300.

[15] R. Neelamegham, P. Chintagunta, A Bayesian model to forecast new product performance in domestic and international markets, Mark. Sci. 18 (1999) 115–136, https://doi.org/10.1287/mksc.18.2.115.

[16] S. Thomassey, A. Fiordaliso, A hybrid sales forecasting system based on clustering and decision trees, Decis. Support. Syst. 42 (2006) 408–421, https://doi.org/10.1016/j.dss.2005.01.008.

[17] S. Thomassey, M. Happiette, A neural clustering and classification system for sales forecasting of new apparel items, Appl. Soft Comput. 7 (2007) 1177–1187, https://doi.org/10.1016/j.asoc.2006.01.005.

[18] Z.-L. Sun, T.-M. Choi, K.-F. Au, Y. Yu, Sales forecasting using extreme learning machine with applications in fashion retailing, Decis. Support. Syst. 46 (2008) 411–419, https://doi.org/10.1016/j.dss.2008.07.009.

[19] N. Szozda, Analogous forecasting of products with a short life cycle, Decision

Making Manufact. Serv. 4 (2010) 71–85, https://doi.org/10.7494/dmms.2010.4.2.71.
[20] A. Fallah Tehrani, D. Ahrens, Enhanced predictive models for purchasing in the fashion field by using kernel machine regression equipped with ordinal logistic regression, J. Retail. Consum. Serv. 32 (2016) 131–138, https://doi.org/10.1016/j.jretconser.2016.05.008.
[21] P. Kabaila, The relevance property for prediction intervals, J. Time Ser. Anal. 20 (1999) 655–662, https://doi.org/10.1111/1467-9892.00163.
[22] T. Gneiting, A.E. Raftery, Strictly proper scoring rules, prediction, and estimation, J. Am. Stat. Assoc. 102 (2007) 359–378, https://doi.org/10.1198/016214506000001437.
[23] A. Khosravi, S. Nahavandi, D. Creighton, Construction of optimal prediction intervals for load forecasting problems, IEEE Trans. Power Syst. 25 (2010) 1496–1503, https://doi.org/10.1109/TPWRS.2010.2042309.
[24] A. Khosravi, S. Nahavandi, D. Creighton, A neural network-GARCH-based method for construction of prediction intervals, Electr. Power Syst. Res. 96 (2013) 185–193, https://doi.org/10.1016/j.epsr.2012.11.007.
[25] X. Wu, V. Kumar, J. Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G.J. McLachlan, A. Ng, B. Liu, P.S. Yu, Z.-H. Zhou, M. Steinbach, D.J. Hand, D. Steinberg, Top 10 algorithms in data mining, Knowl. Inf. Syst. 14 (2008) 1–37, https://doi.org/10.1007/s10115-007-0114-2.
[26] A.K. Jain, Data clustering: 50 years beyond K-means, Pattern Recogn. Lett. 31 (2010) 651–666, https://doi.org/10.1016/j.patrec.2009.09.011.
[27] M. Espinoza, C. Joye, R. Belmans, B. DeMoor, Short-term load forecasting, profile identification, and customer segmentation: a methodology based on periodic time series, IEEE Trans. Power Syst. 20 (2005) 1622–1630, https://doi.org/10.1109/TPWRS.2005.852123.
[28] C.-J. Lu, L.-J. Kao, A clustering-based sales forecasting scheme by using extreme learning machine and ensembling linkage methods with applications to computer server, Eng. Appl. Artif. Intell. 55 (2016) 231–238, https://doi.org/10.1016/j.engappai.2016.06.015.
[29] J. Huber, A. Gossmann, H. Stuckenschmidt, Cluster-based hierarchical demand forecasting for perishable goods, Expert Syst. Appl. 76 (2017) 140–151, https://doi.org/10.1016/j.eswa.2017.01.022.
[30] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J.M. Pérez, I. Perona, An extensive comparative study of cluster validity indices, Pattern Recogn. 46 (2013) 243–256, https://doi.org/10.1016/j.patcog.2012.07.021.
[31] D.L. Davies, D.W. Bouldin, A cluster separation measure, IEEE Trans. Pattern Anal. Mach. Intell. PAMI-1 (1979) 224–227, https://doi.org/10.1109/TPAMI.1979.4766909.
[32] P.J. Rousseeuw, Silhouettes: a graphical aid to the interpretation and validation of cluster analysis, J. Comput. Appl. Math. 20 (1987) 53–65, https://doi.org/10.1016/0377-0427(87)90125-7.
[33] T. Calinski, J. Harabasz, A dendrite method for cluster analysis, Commun. Statist. Theor. Methods 3 (1974) 1–27, https://doi.org/10.1080/03610927408827101.
[34] S. Asteriadis, K. Karpouzis, N. Shaker, G.N. Yannakakis, Towards detecting clusters of players using visual and gameplay behavioral cues, Procedia Comput. Sci. 15 (2012) 140–147, https://doi.org/10.1016/j.procs.2012.10.065.
[35] X. Fu, X. Chen, Y.-T. Shi, I. Bose, S. Cai, User segmentation for retention management in online social games, Decis. Support. Syst. 101 (2017) 51–68, https://doi.org/10.1016/j.dss.2017.05.015.
[36] S.-T. Wang, Integrating KPSO and C5.0 to analyze the omnichannel solutions for optimizing telecommunication retail, Decis. Support. Syst. 109 (2018) 39–49, https://doi.org/10.1016/j.dss.2017.12.009.
[37] K. Fawagreh, M.M. Gaber, E. Elyan, Random forests: from early developments to recent advancements, Syst. Sci. Control Eng. 2 (2014) 602–609, https://doi.org/10.1080/21642583.2014.956265.
[38] J. Ali, R. Khan, N. Ahmad, I. Maqsood, Random forests and decision trees, IJCSI Int. J. Comput. Sci. Issu. 9 (2012) 272–278.
[39] L. Breiman, Manual - setting up, using, and understanding random forests v4.0, (2003) https://www.stat.berkeley.edu/ breiman/Using_random_forests_v4.0. pdf, 2003 (accessed 5 September 2019).
[40] M. Zamo, O. Mestre, P. Arbogast, O. Pannekoucke, A benchmark of statistical regression methods for short-term forecasting of photovoltaic electricity production. Part II: probabilistic forecast of daily production, Sol. Energy 105 (2014) 804–816, https://doi.org/10.1016/j.solener.2014.03.026.
[41] M. Taillardat, O. Mestre, M. Zamo, P. Naveau, Calibrated ensemble forecasts using quantile regression forests and ensemble model output statistics, Mon. Weather Rev. 144 (2016) 2375–2393, https://doi.org/10.1175/MWR-D-15-0260.1.
[42] K. Vaysse, P. Lagacherie, Using quantile regression forest to estimate uncertainty of digital soil mapping products, Geoderma 291 (2017) 55–64, https://doi.org/10.1016/j.geoderma.2016.12.017.
[43] A. Lahouar, J. Ben Hadj Slama, Hour-ahead wind power forecast based on random forests, Renew. Energy 109 (2017) 529–541, https://doi.org/10.1016/j.renene.2017.03.064.
[44] B.R. Cobb, R. Rumí, A. Salmerón, Inventory management with log-normal demand per unit time, Comput. Oper. Res. 40 (2013) 1842–1851, https://doi.org/10.1016/j.cor.2013.01.017.
[45] A. Gholami, A. Mirzazadeh, An inventory model with controllable lead time and ordering cost, log-normal-distributed demand, and gamma-distributed available capacity, Cogent Business & Manage. 5 (2018) 1469182, https://doi.org/10.1080/23311975.2018.1469182.
[46] K. Namit, J. Chen, Solutions to the < Q,r > inventory model for gamma leada€ time demand, Int. J. Phys. Distrib. Logist. Manag. 29 (1999) 138–154, https://doi.org/10.1108/09600039910264713.
[47] K. Ramaekers, G.K. Janssens, On the choice of a demand distribution for inventory management models, Euro. J. Indust. Eng. 2 (2008) 479, https://doi.org/10.1504/EJIE.2008.018441.
[48] G. Nenes, S. Panagiotidou, G. Tagaras, Inventory management of multiple items with irregular demand: a case study, Eur. J. Oper. Res. 205 (2010) 313–324, https://doi.org/10.1016/j.ejor.2009.12.022.
[49] R.M. van Steenbergen, M.R.K. Mes, A synthetic data set with new product demand and characteristics, Research data 1 (2020), 10.17632/g3v9xcxjgc.1.
[50] M.F. Amasyali, O.K. Ersoy, A Study of Meta Learning for Regression, Technical Report 386, ECE Technical Reports (2009).
[51] M.L. Delignette-Muller, C. Dutang, fitdistrplus: An R package for fitting distributions, J. Stat. Softw. 64 (2015) 1–34, https://doi.org/10.18637/jss.v064.i04.
[52] C. Strobl, A.-L. Boulesteix, T. Kneib, T. Augustin, A. Zeileis, Conditional variable importance for random forests, BMC Bioinformatics 9 (2008) 307, https://doi.org/10.1186/1471-2105-9-307.

Robert M. van Steenbergen is a PhD candidate within the department of Industrial Engineering and Business Information Systems (IEBIS) at the University of Twente, The Netherlands. He received an MSc in Industrial Engineering and Management in 2019. His research interests include supply chain management, machine learning, intelligent logistics, simulation optimization, and humanitarian logistics.

Martijn R.K. Mes is an Associate Professor within the department of Industrial Engineering and Business Information Systems (IEBIS) at the University of Twente, The Netherlands. He holds an MSc in Applied Mathematics (2002) and a PhD in Industrial Engineering and Management at the University of Twente (2008). His research interests include freight transportation, multi-agent systems, vehicle routing problems, approximate dynamic programming, stochastic optimization, optimal learning, machine learning, discrete event simulation, and simulation optimization.