Islam and Amin *J Big Data* (2020) 7:65
https://doi.org/10.1186/s40537-020-00345-2

Journal of Big Data

RESEARCH Open Access

Prediction of probable backorder scenarios in the supply chain using Distributed Random Forest and Gradient Boosting Machine learning techniques

Samiul Islam and Saman Hassanzadeh Amin*

*Correspondence:
saman.amin@ryerson.ca
Department of Mechanical and Industrial Engineering, Ryerson University, Toronto, ON, Canada

Abstract

Prediction using machine learning algorithms is not well adapted in many parts of the business decision processes due to the lack of clarity and flexibility. The erroneous data as inputs in the prediction process may produce inaccurate predictions. We aim to use machine learning models in the area of the business decision process by predicting products' backorder while providing flexibility to the decision authority, better clarity of the process, and maintaining higher accuracy. A ranged method is used for specifying different levels of predicting features to cope with the diverse characteristics of real-time data which may happen by machine or human errors. The range is tunable that gives flexibility to the decision managers. The tree-based machine learning is chosen for better explainability of the model. The backorders of products are predicted in this study using Distributed Random Forest (DRF) and Gradient Boosting Machine (GBM). We have observed that the performances of the machine learning models have been improved by 20% using this ranged approach when the dataset is highly biased with random error. We have utilized a five-level metric to indicate the inventory level, sales level, forecasted sales level, and a four-level metric for the lead time. A decision tree from one of the constructed models is analyzed to understand the effects of the ranged approach. As a part of this analysis, we list major probable backorder scenarios to facilitate business decisions. We show how this model can be used to predict the probable backorder products before actual sales take place. The mentioned methods in this research can be utilized in other supply chain cases to forecast backorders.

Keywords: Inventory management, Product backorder, Machine learning, Gradient boosted machine, Supply chain management, Big data

Introduction

When a customer orders a product, which is not available in the store or temporary out of stock, and the customer decides to wait until the product is available and promised to be shipped, then this scenario is called backorder of that specific product [1, 2]. If backorders are not handled promptly, they will have high impacts on the respective company's revenue, share market price, customers' trust, and may end up losing the customer

or sale order. On the other hand, the prompt actions to satisfy backorders put enormous pressure on different stages of the supply chain which may exhaust the supply chain processes or may appear with extra labor and/or production costs, and associated shipment expenses [3, 4]. Moreover, the uncertainty in customers' demands causes difficulty in forecasting the demand which makes the traditional supply chain management systems less effective in many ways such as inaccurate demand forecasting or misclassifying of back-ordered products [5, 6]. Nowadays, some companies predict the backorders of products by applying machine learning prediction processes to overcome the associated tangible and intangible costs of backorders [7].

Machine learning models may misclassify many records if the dataset contains misleading or missing information. This issue is a challenge to analyze the dataset of this study. There are very high negative and positive values in several predicting features of this dataset. Our dataset contains the number of negative records related to the inventory. The negative inventory level suggests that the current stock level of the product is less than zero. The causes and effects of negative inventory are well understood in the supply chain industry [8]. An inventory level dependent ordering model discusses the relationship between inventory level and demand which reflects how negative inventory level can affect the demand [9]. Recent studies suggest that the correlation factor among the product variety, sales, and inventory level is biasing the inventory level [10, 11].

Backorder aging prediction can be feasible for the market with non-volatile demand where the lead time, price per unit, quantity of placed order, and product stock level are the main drivers [12]. However, a sudden change in the demand may raise other risk flags associated with the supply chain and may lead to a loss [13, 14]. To cope with the challenges of stochastic demand, a few researchers developed multi-objective inventory models [15]. It has been proven mathematically that the hybrid backorder (i.e., fixed and time-weighted backorder) inventory model is more efficient than the fixed backorder inventory model in the market with volatile demand. To subside the stochastic demand problem, forecasting partial backorders based on the periodic count on the current stock level seems profitable, but this process may exhaust the local inventory system [8].

In this work, a flexible inventory solution is provided by listing easily understandable probable backorder scenarios. "Literature review" section of this paper focuses on the literature review. Then, "Dataset and exploratory analysis" section is devoted to the dataset and exploratory analysis. The methodology is described in "Methodology" section. Then, the experiments and the results are discussed in "Experiment" and "Results on test data" sections, respectively. Besides, conclusions are provided in "Conclusions" section.

We have performed some hypothesis tests considering backorder scenarios. The outcomes of the hypothesis's tests are helpful to choose the appropriate machine learning model for prediction. Distributed Random Forest (DRF) and Gradient Boosting Machine (GBM) techniques [16, 17] are chosen on the H2O platform. To resolve the imbalanced class problem, a synthetic minority oversampling technique (SMOTE) [18] on the target class is selected. We have divided our predicting features in different ranges, and we have passed it to GBM and DRF models for prediction. Besides, the actual data is fed to those models. It can be observed that the two models show different characteristics in the test run. This research considers backorder forecasting using DRF and GBM and reports easily understandable probable backorder decision scenarios.

Literature review

Machine Learning (ML) techniques enable us to forecast accurately multiple aspects related to supply chain management such as demand, sale, revenue, production, and backorder. ML approaches have been used to predict manufacturers' garbled demands where some researchers applied a representative set of ML-based and traditional forecasting methods to the data to compare the precision of those used methods [19]. Those researchers found that the average performances of the ML method did not outperform the traditional methods, but when a Support Vector Machine (SVM) was trained on several demand-series, it produced the most precise predictions [20]. The same researchers extended their research works using Support Vector Machines (SVM) and Neural Networks (NN) [21]. They have found that the techniques of applying machine learning models provided noticeable improvements over the traditional models [22].

An analysis of the supply chain's demand prediction was carried out by applying the Support Vector Regression (SVR) method in the paper of Guanghui [23]. The outcome of that investigation indicated that the prediction performance of SVR is superior to Radial Basis Function (RBF) [24], as SVR produced smaller results of the relative mean square error along with higher forecast precision of the supply chain. However, several factors were not taken into account in that research (e.g., imbalance class problem, application of machine learning techniques like neural network and ensemble methods due to the limitations of the computational resources).

To minimize the supply chain and inventory control costs, a risk-based dynamic backorder replenishment planning framework was proposed by Shin et al. [25] applying the Bayesian Belief Network. A similar framework was prescribed by Acar and Gardner [26], using optimization and simulation techniques. Rodger [12] presented a risk triggering model using fuzzy feasibility Bayesian probabilistic evaluation of backorder.

To deal with the imbalanced class problem efficiently, ML classifiers were examined in [27] to identify a suitable forecasting model. To carry out this task, they applied different measures along with the ensemble learning. The results of that investigation showed that the ensemble learning method provided feasible performance when precision-recall curves were considered, and also minimized the computational costs. They also suggested applying different ML algorithms such as SVM and NN for the verification of potential performance improvements. Prak and Teunter [28] investigated the prediction uncertainty in an inventory model, and they proposed a framework to estimate the demand to obtain more accurate inventory decisions.

The competition among different ML techniques produces a higher rate of accuracy of forecasts which improvises the necessitous decisions to increase revenue. Dancho [29] predicted the product backorders using a stack-ensemble machine learning approach. The author also discussed the cost–benefit of early prediction of the backorder. However, the demonstration of a probable backorder situation has not been discussed in that paper. The research in [30] and [31] have proposed an order policy-based inventory system model to observe the performance using ARIMA models, Theta method, and multiple temporal aggregation techniques. Performances of ML models with and without Google trends were measured to identify the trend of oil consumption in the paper of Yu et al. [32]. Comparisons among different error measures such as Mean Square Error (MSE) and Root Mean Square Error (RMSE) are shown in the research of Hyndman and

Koehler [33] to indicate the models' performance. Kim and Kim [34] introduced a new metric of measuring the performance known as Mean Arctangent Absolute Percentage Error (MAAPE), and they compared with the other ML error calculations. Martínez et al. [35] evaluated the performance of ML models such as Lasso, Extreme learning machine, and Gradient tree boosting to forecast future purchase trends. The efficiency and impact of different types of forecasting methods were measured for promotional products in business in the research of De Baets and Harvey [36].

Some related papers to our research have been classified in Table 1. Based on the literature review, there are some research gaps. To our knowledge, just a few researchers [10, 11] have explored the negative values in the supply chain data. Besides, very few papers have considered flexible inventory control and cost minimization techniques separately, but none of them provided the probable backorder scenarios in inventory management.

As our research problem falls under the decision-making problem, we have investigated the decision tree-based predictive modeling approach in this research. Decision trees are supervised learning techniques that can be used to solve both classification and regression problems. Each branch of a decision tree is highly traceable, and a decision tree model can be easily interpreted without having expert knowledge in the predictive modeling domain [37]. These characteristics of the decision tree make it popular among the organizational decision-makers to solve different decision-making problems. However, when the input data size is very large, predictions using decision trees suffer from both execution time and performances. If the training data size is big, the tree construction during the training phase of a decision tree based predictive model increases the computational complexity in terms of memory consumption and execution time [38]. Moreover, the constructed trees with big input data suffer from the same input label distribution among different classes which lowers the predictive performance of a decision tree model [39]. To overcome these problems, a few researchers adopted the tie braking method [40] for a decision tree. The tie-breaking method increases the performance of model. However, the memory overflow problem remains in the big data environment [41]. Random sampling [42], tree pruning on input space [43], and few-shot samplings [44] are some techniques proposed by a few researchers to undersize the input space.

Table 1 Review of the related papers

Authors	Prediction domain	ML models	Performance metrics	Flexible inventory control with ranged data	Probable decision scenarios
Carbonneau et al., (2008)	Manufacturers' garbled demands	SVM, NN	✓		
Guanghui (2012)	Supply chain's demand	SVR, RBF	✓		
Shin et al., (2012)	Backorder replenishment planning			✓	✓
de Santis et al., (2017)	Material backorder in supply chain	LOGIST, CART, Ensemble	✓		
Prak and Teunter (2019)	Prediction uncertainty			✓	✓
Proposed work	Product backorder in supply chain	DRF, GBM	✓	✓	✓

These proposed techniques decrease the computational complexity, but also they reduce the model's performance. The different data types and data structures in the input space also affect the performance of a tree-based model [37]. In this study, we have proposed the technique of clustering the input space to reduce the data dimensionality and the computational cost. This ranged based clustering technique also improves the model predictive performances where there are many ties. Besides, we have used a correlational factor among clusters to have better performance. The main research contributions of this work are as follow.

Proposing a ranged based clustering method to reduce the dimensionality of input space and to decrease the computational cost.
Implementing the correlational factor among ranged clusters in the input space to increase the performance of the model.
Developing a tunable ranged based model for flexible inventory control by incorporating both negative and positive data types.
Demonstrating probable backorders scenarios using a decision-based approach where the number of ties for classification is minimized.

To our knowledge, the range-based clustering to reduce data dimensionality, combined with the association establishment among clusters for better prediction accuracy is new. The ranged method that is used in this research can be easily tunable based on the types of businesses. We have chosen a tree-based method in our study because the tree-based algorithm is appraised as one of the easiest and strongest supervised machine learning techniques which is widely used by many researchers e.g., [45, 46]. It is well established that predictive models incorporating tree-based techniques provide high accuracy along with the interpretive ease. Tree-based algorithms are very effective at mapping nonlinear associations which are the shortcomings of other available linear methods [47]. Decision trees, random forest, and gradient boosting are some examples of tree-based methods that are used in the data science research domain frequently. Some researchers e.g., [45, 46, 48] have preferred the GBM model for prediction purposes as it provides very good accuracy when it is tuned correctly. Considering these advantages, GBM and DRF are utilized in this paper.

Dataset and exploratory analysis

The dataset of this research has been published in Kaggle. It is divided into the training and testing datasets. Each dataset contains 23 attributes with 1,687,862 and 242,077 observations for the training and testing sets, respectively. Both datasets contain a mix of features with floating-point, integer, and string values. For this study, we intend to use the most common data attributes that can be readily available for any business. Hence, we have chosen inventory, lead time, sales, and forecasted sale as our predicting variables and 'went on backorder' as our response variable. Our target variable is labeled with two classes. Hence, this scenario falls under the binary classification problem. The inventory feature indicates an available stock of products, although it contains high numbers of negative records. The negative inventory may arise due to the machine or human error. It may also occur when a shipment is recorded as complete before it arrives. The

'lead time' feature indicates the elapsed time between the placement of products' orders and delivery of those products to the customers. The lead time in our dataset ranges from 0 to 52 weeks. The sales features are divided into four parts as one-month sale, three months sale, six months sale, and nine months sale. The forecasted sale is divided into three columns showing the forecast of three months, six months, and nine months.

Figure 1 shows the distribution of some samples in the dataset. It is observable in this figure that the data points of different features have many outliers with different ranges. In both training and testing datasets, a large number of missing values across the predicting variables are observed. Moreover, our response variable is highly imbalanced with 0.669% data from 'Yes' class, and 99.33% data from 'No' class. Figure 1 depicts how the data samples are distributed among two classes, where 0 indicates 'No' class or non-backorder items and 1 indicates 'Yes' class or backorder items.

Hypothesis testing

The central limit theorem states that the distribution of the sample is normal if the sample size is large or greater than 30. For large size samples, the distribution of the data can be ignored, and parametric tests can be applied. In this study, the Wilcoxon rank-sum test with continuity correction is used for the hypothesis test.

We would like to examine the relationship between the products that went on backorder with some features of our training dataset. First, we want to see whether the current level of products' stock affects the decision of the backorder or not. We have assumed a null hypothesis that if the stock level of a product reaches zero, it results in backorder. The significance Alpha level of 0.05 is selected, which means that there is a 5% chance of rejecting the null hypothesis when the hypothesis is true. It has been observed that the p-value for this null hypothesis is far below the significance level. Hence, we cannot accept this hypothesis.

In the next stage, it is assumed that the most sold items per month went on backorder. To consider this assumption, all the sales columns are added, and the average is calculated. It is observed that the p-value for this null hypothesis is far below the significance level. Therefore, the alternative hypothesis is true.

It is examined whether the lead time factor or the high forecasted demands cause backorders. It is observed that the p-value is less than the significance level in both

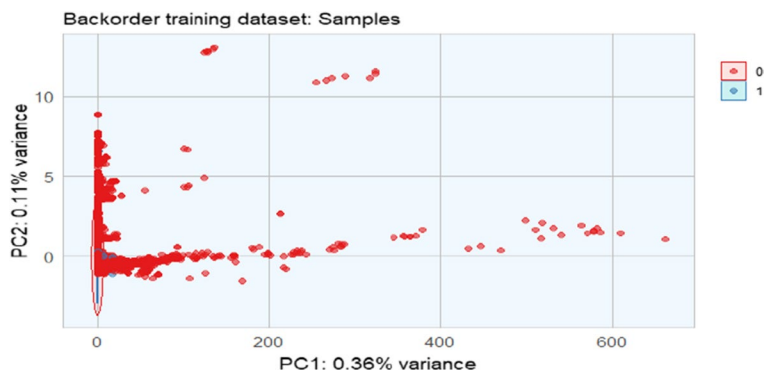

Fig. 1 Distribution of samples among features

cases. Thus, the null hypothesis cannot be accepted. Table 2 includes a summary of the hypothesis testing results and decisions.

Methodology

In this study, we build our proposed model mainly focusing on sales and forecasted sales. As the product stock level and the lead time of products are commonly known attributes, we include these two factors with the sales and forecasted sales data.

Prediction outline

The proposed model can be described using a five-level decision tree to predict back-order products. In the first level, the current inventory level is considered. Lead time, past sales, forecasted sales, and prediction decisions are in levels two, three, four, and five, respectively. Figure 2 shows the different levels of the prediction method.

Converting into different range levels isolates the nodes and they may not have the same association with other features as the actual numerical values will change to the categorical levels. For instance, when the inventory level is converted into different range groups, it can be considered as an isolated node as the actual value in this feature is not changed. To minimize this issue, the associated factors of inventory level and other features are captured. After converting the inventory level into ranges, the range levels are multiplied with the association factors. A similar idea is used in path analysis. If A and B are two independent nodes and both are associated with a third node, C, then the association between A and B will be the multiplication of the correlation coefficient of AC and BC.

In this study, the Spearman correlation method is utilized to capture the association. A Spearman correlation coefficient measures the strength of association between two variables [49]. In the real world, most of the data can be classified as nonparametric. Spearman's method can handle this issue efficiently. If some non-linear relationships among two variables exist, Pearson correlation would produce zero which can be interpreted as no linear relationship among two variables. Spearman's method can figure out both linear and non-linear relationships among two variables. Moreover, this technique can be used for both continuous and discrete variables.

Table 2 Hypothesis testing summary

Question	Hypothesis	Result analysis	Decision
a) Are the products out of stock resulting back-orders?	H_0: $n = 0$; where n = Quantity of product in the stock Ha: $n \geq 1$	p-value < 2.2e−16, for $\alpha = 0.05$	Alternative hypothesis is true
b) Were the most sold items per month producing back-ordered?	H_0:$Backorder_{Yes} = n$, n = Average number of products sold per month	p-value < 2.2e−16, for $\alpha = 0.05$	The alternative hypothesis is true
c) Are the lead time factors producing backorders?	H_0: $Backorder_{Yes} = factors_{lead-time}$.	p-value < 2.2e−16, for $\alpha = 0.05$	The alternative hypothesis is true
d) Are the forecasted demands of products resulting in backorders?	H_0: $Backorder_{Yes} = Forecast_{high}$	p-value < 2.2e−16, for $\alpha = 0.05$	The alternative hypothesis is true

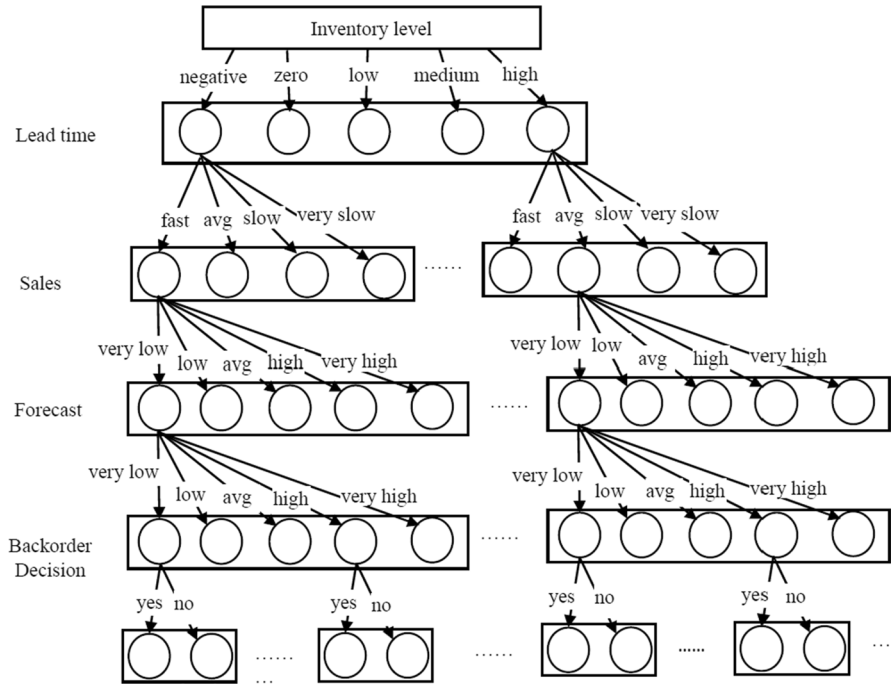

Fig. 2 Decision tree structure

Inventory level

In the current inventory stock, there are negative, zero, and positive values. To determine the stock level, the safety stock is calculated. The safety stock is calculated by assuming the most common service level in the retail industry which is 90% [50, 51]. The average demand is denoted by D_{avg}. Besides, the desired service level is shown by Z, and the standard deviation of lead time is expressed as σ_{LT}. The desired service level of 90% is considered in this experiment. The corresponding Z value is fetched from the normal distribution chart which gives the value of 1.28. The safety stock is calculated for each item using Eq. (1).

$$SafetyStock = Z * \sigma_{LT} * D_{avg} \tag{1}$$

The safety stock is subtracted from the current stock value and is distributed among five different ranges, namely, negative, zero, low, medium, and high levels. These five levels denote the status of the current inventory level.

Negative stock values refer to the situation where the inventory count of the products turns to less than zero. It may happen due to several reasons such as accidental duplication of sales.

The zero-stock level happens if no physical unit of products is in the current stock. The low, medium, and high stock levels can be defined based on the prescribed ranges of supply chain managers. This range may vary based on the business types.

Generally, there is a correlation between lead time and inventory. Low cost or fast selling products may have a very fast lead time. However, large and expensive items usually have a long lead time. Similarly, actual sales and forecasted demand have a

relational factor with the current stock level. These correlation factors are used to intensify the five levels of inventory which can be shown using Eq. (2).

$$Inventorylevel = Inventorylevel_{status} * \rho_{IL} * \rho_{IS} * \rho_{IF} \quad (2)$$

$Inventorylevel_{status}$ denotes the numerical representation of five levels. Negative, zero, low, moderate, and high levels are shown by 1, 2, 3, 4, and 5, respectively.

ρ_{IL} denotes the Spearman correlation factor between inventory and lead time.

ρ_{IS} denotes the Spearman correlation factor between inventory and sales.

ρ_{IF} denotes the Spearman correlation factor between inventory and forecasted sale.

Lead time

In this research, the lead time is grouped into four different categories including fast, moderate, slow, and very slow. Then, it is converted from the week into the number of days, and it is grouped in a range as 0 to 10 days as fast lead time, 11 to 40 days as moderate lead time, 49 to 120 days as slow lead time, and 121 to 364 days as a very slow one. The ranges may vary based on the type of business. Generally, faster sale items may have faster lead time and vice versa. We measure the linear relation among lead time, inventory, average forecast, and average sales. Then, we multiply these factors by the four different ranges of lead time as shown in Eq. (3).

$$Leadtime\ level = Leadtime\ level_{status} * \rho_{IL} * \rho_{LS} * \rho_{LF} \quad (3)$$

$Leadtime\ level_{status}$ denotes the numerical representation of the four levels (the fast level as 1, medium level as 2, slow level as 3, and a very slow level as 4).

ρ_{IL} denotes the Spearman correlation factor between inventory and lead time.

ρ_{LS} denotes the Spearman correlation factor between lead-time and sales.

ρ_{LF} denotes the Spearman correlation factor between lead-time and forecasted sales.

Sales

The sales quantities are grouped into five different ranges which are very-low, low, moderate, high, and very high. The number of ranges can be tuned based on business requirements and policies. Generally, the number of items sold has a direct impact on the inventory level, lead time, and forecasted sale. These impact factors are used to strengthen the sales range as shown in Eq. (4).

$$Sales\ level = Sales\ level_{status} * \rho_{Is} * \rho_{LS} * \rho_{SF} \quad (4)$$

$Sales\ level_{status}$ denotes the numerical representation of five levels, i.e., very-low level as 1, low level as 2, moderate level as 3, high level as 4, and very high level as 5.

ρ_{IS} denotes the Spearman correlation factor between inventory and sales.

ρ_{LS} denotes the Spearman correlation factor between lead-time and sales.

ρ_{SF} denotes the Spearman correlation factor between sales and forecasted sale.

Forecasted sale

Companies can forecast based on their sales data. In this study, the forecast is divided into five different ranges such as very-low, low, moderate, high, and very high. The range

numbers can be changed based on the business type and requirements. The average percentage error is calculated to report the forecasting error using Eq. (5).

$$\text{The average percentage error for forecast, } \varepsilon = \frac{\sum |\text{V}_{\text{SF}}|}{\sum \text{Actual Sales}} * 100 \tag{5}$$

V_{SF} is the variance between the actual sales and forecasted sales. The range of forecast is divided by the forecasting error to eliminate the effect of deviation as shown in Eq. (6).

$$\textit{Forecast level} = \textit{Forecast level}_{status}/\varepsilon \tag{6}$$

$\textit{Forecast level}_{status}$ denotes the numerical representation of the five levels, i.e., very-low or 1, low or 2, moderate or 3, high or 4, and very high level or 5.

Prediction equations and notations

In this subsection, the notations and two equations are discussed.

Inv_i: Five levels of inventory status (Negative, Zero, Low, Moderate, High).
LT_j: Four levels of lead time status (Fast, Moderate, Slow, Very slow).
S_k: Five levels of sales status (Very low, Low, Moderate, High, Very High).
F_l: Five levels of forecast status (Very low, Low, Moderate, High, Very High).
ρ_{IL}: Spearman correlation factor between inventory and lead time.
ρ_{IS}: Spearman correlation factor between inventory and sales.
ρ_{IF}: Spearman correlation factor between inventory and forecasted sale.
ρ_{LS}: Spearman correlation factor between lead-time and sales.
ρ_{LF}: Spearman correlation factor between lead-time and forecasted sale.
ρ_{SF}: Spearman correlation factor between sales and forecasted sale.
ε: Average percentage forecast error.
D_{yes_x}: Number of yes decision nodes.
D_{no_x}: Number of no decision nodes.
x: Set of n number of items.
Equations (7) and (8) calculate the final backorder decisions.

$$D_{yes} = \sum_{x=1}^{n} \left[\left[\left[\left\{ \sum_{i=1}^{5} Inv_i * (\rho_{IL} + \rho_{IS} + \rho_{IF}) \right\} * \left\{ \sum_{j=1}^{4} LT_j * (\rho_{IL} + \rho_{LS} + \rho_{LF}) \right\} \right] * \sum_{k=1}^{5} S_k * (\rho_{IS} + \rho_{LS} + \rho_{SF}) \right] * \sum_{l=1}^{5} F_l/\varepsilon \right] \tag{7}$$

$$D_{no} = \sum_{x=1}^{n} \left[\left[\left[\left\{ \sum_{i=1}^{5} Inv_i * (\rho_{IL} + \rho_{IS} + \rho_{IF}) \right\} * \left\{ \sum_{j=1}^{4} LT_j * (\rho_{IL} + \rho_{LS} + \rho_{LF}) \right\} \right] * \sum_{k=1}^{5} S_k * (\rho_{IS} + \rho_{LS} + \rho_{SF}) \right] * \sum_{l=1}^{5} F_l/\varepsilon \right] \tag{8}$$

Because the number of decision nodes grows exponentially based on the number of items, we use a tree-based machine learning model to carry out the solution. One advantage of this approach is that we can get the interpretable backorder decision scenarios from the tree-based approach regarding the dataset. There are some tree-based approaches, and among them, the widely used algorithms are Decision Trees, Random Forest, Bagging, Boosting, Xgboost, and Gradient Boosted Machine (GBM). The inability to handle continuous numerical variables and overfitting are the disadvantages of decision trees. Random Forest (RF) algorithm can cope with large datasets including multiple dimensions. It can perform both classification and regression. RF is widely used for unsupervised clustering in the real world. It works as a black box when large proportions of data are missing or the data dimensionality is unknown. In this study, RF is chosen as the baseline model. GBM is a type of boosting algorithm which applies gradient descent technique to minimize the error rate which has made this model popular among many researchers in recent years [45–47]. The GBM is selected in this study as the second model for the backorder prediction solution.

Experiment

Experimental environment setup

We have initialized the H_2O cluster to run our targeted algorithms. Our datasets are in the data frame format, and H_2O requires the data in the H_2O frame format. Therefore, the datasets are converted into the H_2O frame.

Model construction

The H_2O models are constructed for both synthetic minority oversampling techniques and random oversampling techniques. The feature variables related to inventory, lead time, forecast, and sales along with the response variable for backorder are selected from the actual dataset. These features are converted into different class ranges and are multiplied by previously captured correlational factors from the actual dataset.

Two GBM models are constructed using the actual data and the converted data. Two DRF models are also constructed by a similar process. To speed up the models' training, a separate validation frame is used, which is 35% of the training dataset. The number of trees is tuned from 50 to 1000 to have an early stopping or fast runtime. The learning rate is set to 0.1 for both models. The lower learning rate provides a fine output, although it may slow down the training process if it is tuned below the range of 0.01. The maximum depth is set to 10 to avoid overfitting of the model and to get interpretable outcomes. The higher depth may provide higher accuracy, but the computational time will increase. The sample rate is set to 0.9 which means that 90% of the training data rows will be considered for each tree. The column sample rate is set to 1 to consider 100% of the columns per split of the trees.

The first GBM model with actual data has specified the three months sales as the most important factor for identifying back order (see Fig. 3). Besides, it has shown 9 months forecasts, and three months forecasts, as the second and third important factors, respectively.

Using the proposed grouping methods, the GBM model has identified the inventory level as the most important feature, nine months sales, and one-month sales, as the second and

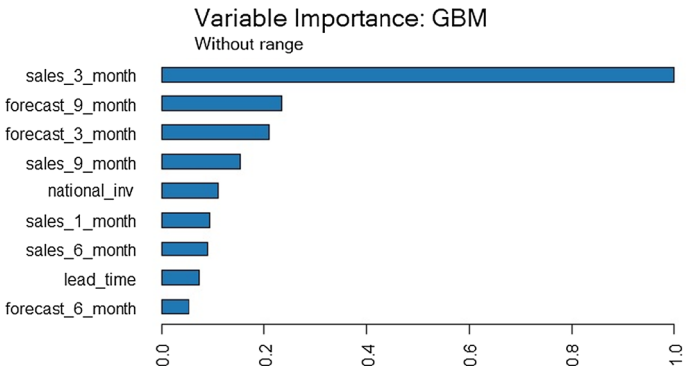

Fig. 3 GBM model's important features with actual data

third important elements for predicting backorder as shown in Fig. 4. A backorder scenario where the inventory is negative/low/very low/ and the sale is moderate/high matches the second GBM model.

The first DRF model with actual data has considered three months forecasts as the main predicting factor, nine months forecasts, and national inventory as the second and third important factors for predicting backorder as shown in Fig. 5. On the other hand, the second DRF model with the ranged data has considered national inventory, nine-month sales,

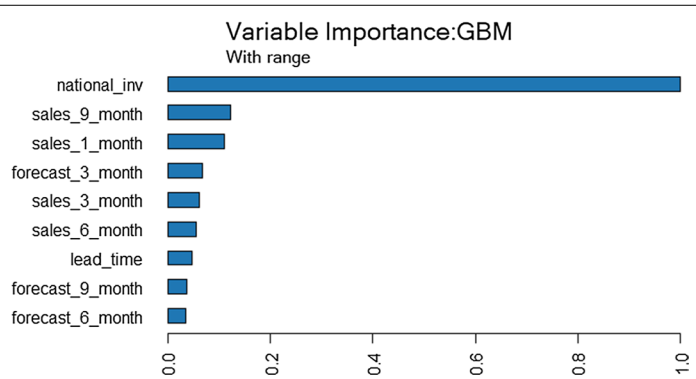

Fig. 4 GBM model's important features with range converted data

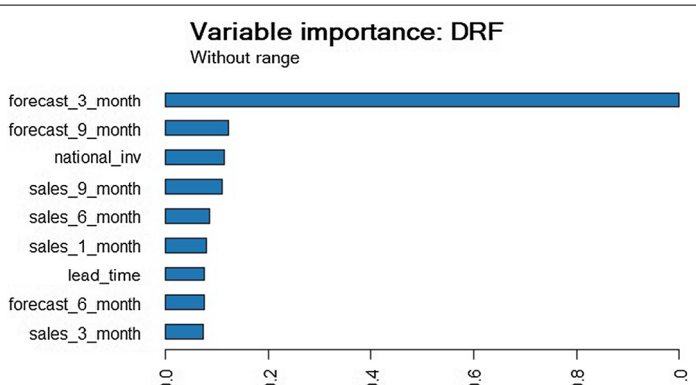

Fig. 5 DRF model's important features with actual data

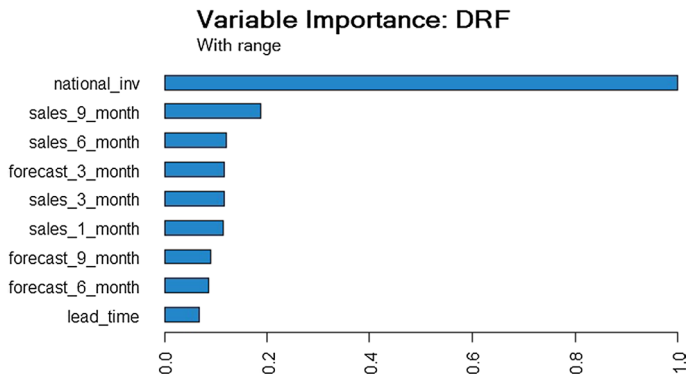

Fig. 6 DRF model's important features with range converted data

Table 3 Models' performances during the training phase

Performance metric	GBM (with actual data)	GBM (with ranged data)	DRF (with actual data)	DRF (with ranged data)
LogLoss	0.016	0.019	0.016	0.029
AUC	0.994	0.979	0.987	0.985
Mean per class error	0.029	0.089	0.030	0.060

Table 4 Models' performances during the testing phase

Performance metric	GBM (with actual data)	GBM (with ranged data)	DRF (with actual data)	DRF (with ranged data)
LogLoss	0.098	0.029	0.036	0.042
AUC	0.795	0.946	0.787	0.959
Mean per class error	0.423	0.07	0.430	0.103

and six-month sales as the top three important factors for predicting backorders as shown in Fig. 6.

Model evaluation

Table 3 shows that the GBM and DRF models with actual data and ranged training data have no significant difference in terms of the Area Under the Curve (AUC) value. The mean per class error is high for the RF model with the ranged data.

Making predictions

The constructed models are used on the testing dataset, and the performances are observed by visualizing Receiver Operating Characteristics (ROC) Curve along with AUC.

Results on test data

Model performance

Table 4 shows that how GBM and DRF models perform when they have been exposed to the actual and ranged testing dataset. The mean classification error per class of the GBM model with actual data and the DRF model with actual data is high. Moreover, the AUC is low for both models with actual data which indicates the overfitting of those two models in the training phase. One of the main reasons for the overfitting of a model is the high diversity of the data in the test phase. It can be observed that the AUC and mean class errors are slightly decreased compared to the training phase. This slight variation is tolerable in the test phase.

Confusion matrix

The first thing we would like to focus on is the model's Confusion matrix as it is one of the easiest ways to get a glimpse of the correctness of the model. Most of the performance measures depend on the different term values of the Confusion matrix. The confusion matrix consists of four terms, namely true positive (TP), true negative (TN), false positive (FP), and false negative (FN). Different performance measures are elaborated for the prediction run of one model to show how each term can be interpreted. To do so, the confusion matrix of the GBM model trained with ranged data is fetched.

The distributions of TP, FP, TN, FN are shown as a Confusion matrix in Table 5. According to Table 5, the classifier has perfectly classified 11,915 products as a backorder. They were marked as went on backorder on the dataset. Therefore, the TP instances are 11,915. This predictive model also classified 426,522 products correctly which did not go as the backorder. So, we get 426,522 TN instances. The classifier marked 3265 products as they did not go on backorder which did go on backorder in the dataset. Besides, 1518 products were wrongly classified as went on backorder. Hence, we get 3265 FN instances and 1518 FP instances. A high number of TN instances are observed because most of the products in the dataset did not go on backorder.

To evaluate the model's strength, several performance measures are investigated. First, we have looked at how many backorder and non-backorder decisions have been classified by the model correctly. The classification accuracy of the model in our case is 0.9892 which reflects that it predicted approximately 98 products out of every 100 products correctly, whether those go on backorder or not. The next measure we have looked at is what proportions of the products that are predicted as going to the backorder, actually became backorder. We have done this by dividing the actual true positive instances with the cumulative true positive and false positive instances. The proposed

Table 5 Confusion matrix for GBM with ranged data

Models' prediction	Actual values in the dataset	
	Backorder	Non-backorder
Backorder	TP 11,915	FP 1518
Non-backorder	FN 3265	TN 426,522

Table 6 Models' characteristics during the testing phase

Performance metric	GBM trained with ranged data	GBM trained with actual data	DRF trained with ranged data	DRF trained with actual data
Classification accuracy	0.9892	0.7919	0.9835	0.8436
Precision	0.8869	0.6896	0.8231	0.7213
Recall/sensitivity	0.7849	0.5876	0.8488	0.6893
Specificity	0.9964	0.7991	0.9986	0.8407
F1 score	0.7845	0.6345	0.8357	0.7049
Misclassification error	0.0107	0.2080	0.0164	0.1563

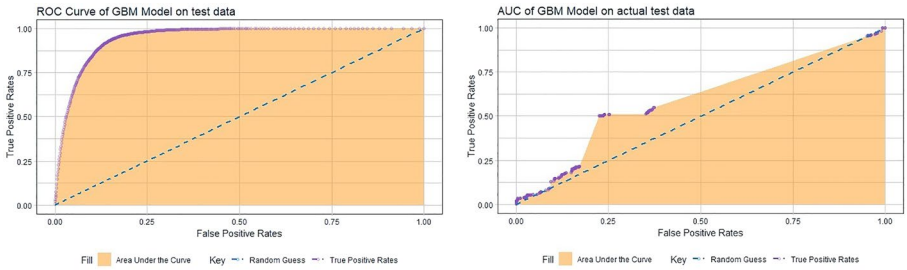

Fig. 7 Area under the curve of GBM models which are trained differently

model predicts 88% of the products correctly that went on backorder. In contrast, the specificity of the model is also calculated to figure out the percentage of correctly classified non-back ordered products by the model, which is 99%. The probable reason for the difference between correctly classified true positive instances (88%) and true negative instances (99%) is because of the imbalanced class distribution. The error rate of the proposed model is also calculated as 0.0107 which shows the model's prediction strength.

Comparison table for four models

For simplicity, we did not elaborate on each term of sections 6.2.1 to 6.2.6 for all models. The results for all measures are reported in Table 6. From Table 6, it is observed that the overall performances of the models with the ranged data are higher than the others. The GBM and DRF have performed almost similar for the ranged data. However, DRF performed better than GBM for the actual data. Both GBM and DRF with ranged data have misclassified 1 product out of 100 products. On the other hand, the models trained with the actual data have misclassified 15–0 products out of 100 products.

ROC-AUC curve

In this part, we would like to show the performances of our model by visualizing Receiver Operating Characteristics (ROC) Curve along with AUC. The ROC curve tells us how our models have performed throughout the prediction phase for all possible threshold values whereas the AUC represents the performance summary in a single value. The higher AUC value can lead to a more accurate predictive model. The ROC curve is plotted considering True Positive Rates (TPR) in the y-axis, and False Positive

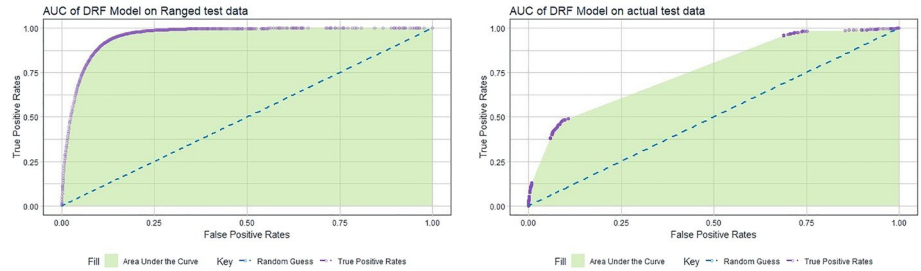

Fig. 8 Area under the curve of DRF models which are trained differently

Rates (FPR) in the x-axis on a scale from 0 to 1. The TPR and FPR are calculated for each threshold point of the classification process. The threshold points are the probability values that have been used to determine the class.

Figures 7 and 8 depict the performances of the models. The diagonal dotted lines represent the random guessing states for each threshold. The colored portion of Figs. 7 and 8 denote the area under the curve. Figure 8 shows the AUC of the DRF models. Both GBM and DRF trained with the ranged dataset have produced similar characteristics. However, the GBM and DRF models that have been trained with the actual data have acted differently in the test phase. The DRF has produced more accurate results than the GBM in this case.

Tree investigation for probable backorder cases

We investigate the tree of our models. The investigation begins with the fetching of the tree that has produced the highest AUC. As mentioned in "Model construction" section, we have selected the max depth of 10. Figure 9 shows the AUC achieved at different depths. This figure illustrates that the selected model reached AUC of almost 94% at Depth 9. The trees are sorted based on their AUC values in the descending order, and the first tree is pulled out. To understand how the tree is grown, the sample representation for each level is presented.

Figure 10 shows that the negative inventory level has produced the probability of positive 0.19 which means that the products with negative inventory may lead to the

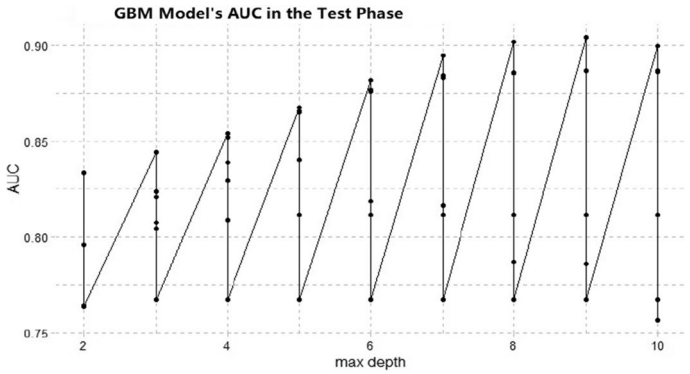

Fig. 9 AUC of GBM trained with ranged data at different depth

backorder in 19 cases among 100 cases. In addition, 17 products out of 100 which have zero or no stock may go to the backorder. It is also noticed that the high and the medium inventory levels produce a 30% chance together that the items may not become a backorder. The sign in the prediction denotes the binary class (yes or no, 0 or 1). In our case, the positive (+) sign in prediction denotes the chance to become backorder, and the negative (−) sign denotes the probability of non-backorder.

Figure 11 shows that the slow and very slow lead time produces a 25% chance of backorder scenario. There is a 15% chance that the product is not a backorder product if the lead time is in the moderate range.

Figure 12 depicts that the moderate and high sales levels produce 9% chances of backorder whereas the low and very low sales denote that there is a 2% chance for non-backorder.

Figure 13 shows that if the forecast level is very high, there is a 20% chance for the product to be backorder. These backorder chances are almost 8% for the moderate forecast level, and 5% for the high forecast level.

Figure 14 depicts that if the inventory level is negative or zero, there are 16 to 20% chances of backorder for different lead time levels. For the low inventory level, if the lead time level is moderate, the product may become backorder in 14% of cases.

Figure 15 shows that if the inventory level is negative or zero, there are 16 to 19% chances of backorder for different sales levels. For the low inventory level, if the sales level is either high, low, or moderate, the product may be considered backorder in 16% of cases.

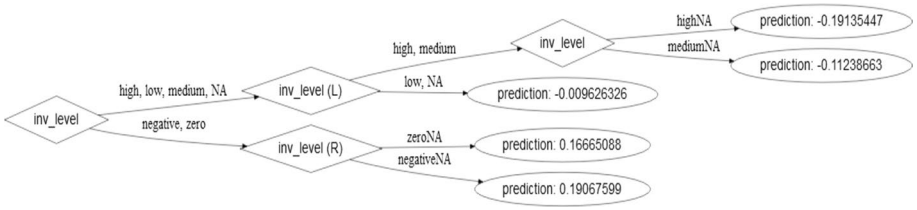

Fig. 10 GBM model's inventory level distribution for backorder class

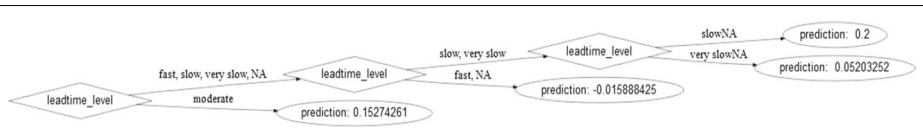

Fig. 11 GBM model's lead time distribution for backorder class

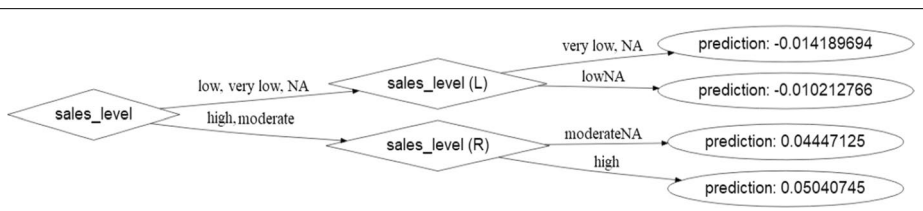

Fig. 12 GBM model's sales distribution for backorder class

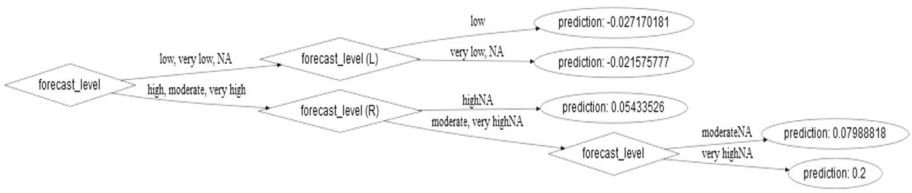

Fig. 13 GBM model's forecast distribution for backorder class

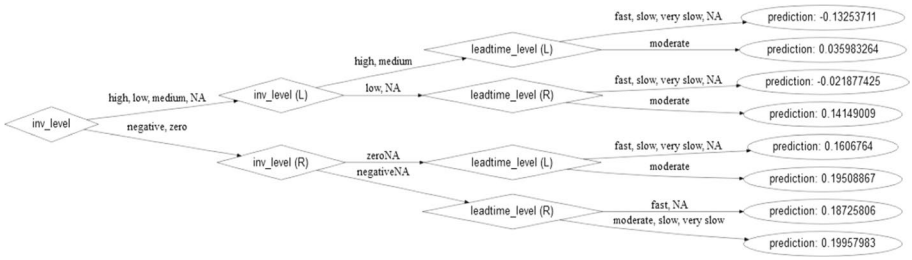

Fig. 14 GBM model's inventory and lead time level distribution for backorder class

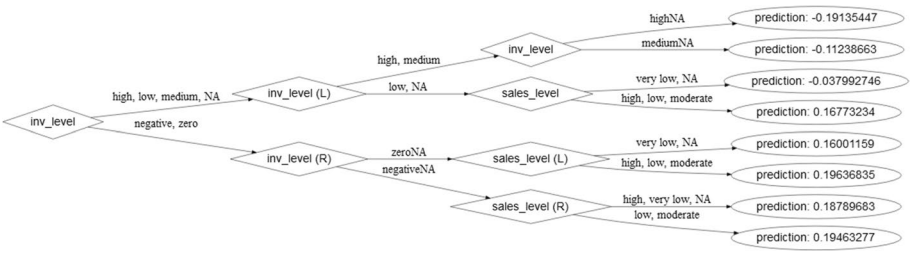

Fig. 15 GBM model's inventory and sales level distribution for backorder class

Figures 10, 11, 12, 13, 14, 15 show how the tree grows with backorder probability for different range levels. For simplicity, we have not included every part. Table 7 includes the probable causes of the backorder for the full tree.

The majority count rank-based data in Table 7 can be used to get more optimal backorder scenarios as depicted in Fig. 16. As an example, consider any specific product item 'A' with a fast lead time. Suppose that the inventory level of 'A' is observed either in negative/zero/medium levels along with the high sales level and high forecast level. In this scenario, there are 20% chances that the product item 'A' will be backorder.

Conclusions

Early prediction of different business issues helps organizations to act in advance and retain the profit along with business goodwill. The focus of this study is on the product backorder. In this research, the probable backorder items have been predicted using two machine learning techniques. Prediction using real-world data is sometimes challenging as it suffers from many problems such as high bias, redundancy, and missing values.

Table 7 Backorder cases where $p \geq 0.2$

inv_level	sales_level	forecast_level	leadtime_level	Prediction > = 0.2
Low	Very low	Moderate	Very slow	0.2
Zero	Very low	very high	Fast	0.2
Zero	Low	Low	Fast	0.2
Negative	Low	High	Fast	0.2
Zero	High	High	Fast	0.2
Negative	Moderate	High	Very slow	0.2
Negative	Moderate	Very low	Fast	0.2
Negative	High	Moderate	Fast	0.2
Zero	High	Very low	Moderate	0.2
Low	Very low	High	Fast	0.2
Medium	High	Moderate	Moderate	0.2
Medium	High	Very low	Moderate	0.2
High	High	Very low	Moderate	0.2
Medium	High	High	Very slow	0.2
Medium	Very low	High	Very slow	0.2

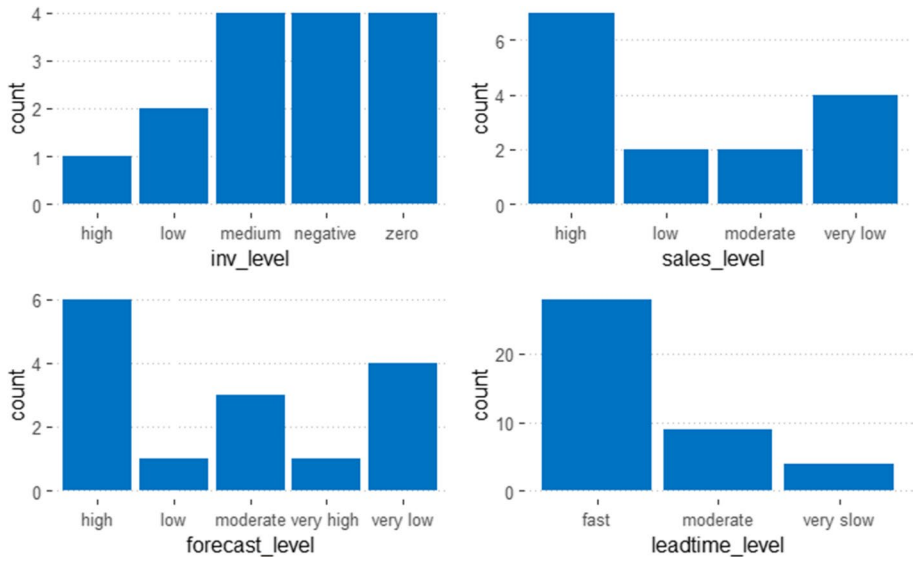

Fig. 16 Distribution of probable backorder cases for 20% backorder probability

More specifically in our case, the impacts of negative values and high variance are well observed. Data pruning can be one of the solutions, but it requires the availability of experts which is costly more often. In this study, a ranged technique is proposed to solve this issue. The actual data and the ranged data have been used in both machine learning models, and their performances have been compared. The comparison result shows that the models' performance increases by approximately 20% when they are trained with the ranged data.

As the market characteristics are varying rapidly based on the customers' demands and expectations, flexibility in inventory control is required to maximize the profit. In this proposed method, the ranges of different inventory, lead time, sales, and forecast levels

are easily tunable. These ranges can be tuned based on business types, business requirements, and goals. The correct identification of backorder probability and tuned range during inventory control may play a critical role in boosting up revenues and profits. This method not only copes with negative inventory data but also minimizes the number of ties in the model building phase as well as in the prediction phase. Tie happens when exact prediction variables' values are found among more than one class. Based on the majority of similar events of a tie, the model decides the class for those events. When the number of ties increases, computational complexity also increases. Decision models produce incorrect results when there are too many ties. Instead, the use of ranged data decreases the number of ties. The ranged method may be time-efficient for big data records. The decision authority of the respective business will have a broader picture of the backorder conditions as reported in Table 7 and may take necessary actions in advance.

In this paper, it has been shown that based on known inventory, lead time, sales, and forecasted sales, we can identify those products that will be backorder products. However, the relational factors such as local buyout quantity, past due stock, suppliers' performances, and different product risk flags have not been considered because of the lack of that information. We may focus on those factors in our future work. As the uncertainty of demand plays a vital role to make the market volatile, the relationship between the predicted demand and the predicted backorder may also need attention. As future research, the mentioned perspectives can be considered to develop an integrated model and to understand more accurate backorder scenarios in advance.

Abbreviations

AUC: area under the curve; DRF: Distributed Random Forest; FN: false negative; FP: false positive; FPR: false positive rates; GBM: Gradient Boosting Machine; ML: machine learning; MAAPE: Mean Arctangent Absolute Percentage Error; MSE: mean square error; NN: neural networks; RBF: radial basis function; RF: random forest; ROC: receiver operating characteristics; RMSE: root mean square error; SVM: support vector machine; SVR: support vector regression; SMOTE: synthetic minority oversampling technique; TN: true negative; TP: true positive; TPR: true positive rates.

Acknowledgements

The authors would like to thank the editor and reviewers for the great comments that improved the quality of the paper significantly. This research has been supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

Authors' contributions

SI is the first author. SHA is the second author. He is the Ph.D. supervisor of SI. Both authors read and approved the final manuscript.

Funding

This research has been supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

Availability of data and materials

The data will be available upon request.

Competing interests

There are no financial and non-financial competing interests.

Received: 7 May 2020 Accepted: 14 August 2020
Published online: 26 August 2020

References

1. Clark KB, Fujimoto T. Product development performance: strategy, organization, and management in the world auto industry. 1991.
2. Guo L, Wang Y, Kong D, Zhang Z, Yang Y. Decisions on spare parts allocation for repairable isolated system with dependent backorders. Comput Ind Eng. 2019;127:8–20.

3. Carter CR, Rogers DS. A framework of sustainable supply chain management: moving toward new theory. Int J Phys Distrib Logistics Manag. 2008;38(5):360–87.
4. Mohebalizadehgashti F, Zolfagharinia H, Amin SH. Designing a green meat supply chain network: a multi-objective approach. Int J Prod Econ. 2020;219:312–27.
5. Simchi-Levi D, Kaminsky P, Simchi-Levi E, Shankar R. Designing and managing the supply chain: concepts, strategies and case studies. New York: Tata McGraw-Hill Education; 2008.
6. Yu L, Duan Y, Fan T. Innovation performance of new products in China's high-technology industry. Int J Prod Econ. 2020;219:204–15.
7. Mitra A. Fundamentals of quality control and improvement. New York: Wiley; 2016.
8. Xu Y, Bisi A, Dada M. A finite-horizon inventory system with partial backorders and inventory holdback. Oper Res Lett. 2017;45(4):315–22.
9. Sarker BR, Mukherjee S, Balan CV. An order-level lot size inventory model with inventory-level dependent demand and deterioration. Int J Prod Econ. 1997;48(3):227–36.
10. Wan X, Sanders NR. The negative impact of product variety: forecast bias, inventory levels, and the role of vertical integration. Int J Prod Econ. 2017;186:123–31.
11. Wan X, Britto R, Zhou Z. In search of the negative relationship between product variety and inventory turnover. Int J Prod Econ. 2019. https://doi.org/10.1016/j.ijpe.2019.09.024.
12. Rodger JA. Application of a fuzzy feasibility Bayesian probabilistic estimation of supply chain backorder aging, unfilled backorders, and customer wait time using stochastic simulation with Markov blankets. Expert Syst Appl. 2014;41(16):7005–222.
13. De Brito MP, Carbone V, Blanquart CM. Towards a sustainable fashion retail supply chain in Europe: organisation and performance. Int J Prod Econ. 2008;114(2):534–53.
14. Tosarkani BM, Amin SH. An environmental optimization model to configure a hybrid forward and reverse supply chain network under uncertainty. Comput Chem Eng. 2019;121:540–55.
15. Srivastav A, Agrawal S. Multi-objective optimization of hybrid backorder inventory model. Expert Syst Appl. 2016;51:76–84.
16. Ridgeway G. gbm: Generalized boosted regression models. R package version. 2006;1(3):55.
17. Torgo L. Data mining with R: learning with case studies. New York: Chapman and Hall/CRC; 2011.
18. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: synthetic minority over-sampling technique. J Artif Intell Res. 2002;16:321–57.
19. Carbonneau R, Vahidov R, Laframboise K. Machine learning-Based Demand forecasting in supply chains. Int J Intell Inf Technol (IJIIT). 2007;3(4):40–57.
20. Hearst MA, Susan TD, Edgar O, John P, Bernhard S. Support vector machines. In: IEEE intelligent systems and their applications. 1998. p. 18–28.
21. Funahashi KI. On the approximate realization of continuous mappings by neural networks. Neural Netw. 1989;2(3):183–92.
22. Carbonneau R, Laframboise K, Vahidov R. Application of machine learning techniques for supply chain demand forecasting. Eur J Oper Res. 2008;184(3):1140–54.
23. Guanghui WANG. Demand forecasting of supply chain based on support vector regression method. Procedia Eng. 2012;29:280–4.
24. Chen S, Cowan CF, Grant PM. Orthogonal least squares learning algorithm for radial basis function networks. IEEE Trans Neural Netw. 1991;2(2):302–9.
25. Shin K, Shin Y, Kwon JH, Kang SH. Development of risk based dynamic backorder replenishment planning framework using Bayesian Belief Network. Comput Ind Eng. 2012;62(3):716–25.
26. Acar Y, Gardner ES Jr. Forecasting method selection in a global supply chain. Int J Forecast. 2012;28(4):842–8.
27. de Santis RB, de Aguiar EP, Goliatt L. Predicting material backorders in inventory management using machine learning. In 2017 IEEE Latin American Conference on Computational Intelligence (LA-CCI). 2017. p. 1–6.
28. Prak D, Teunter R. A general method for addressing forecasting uncertainty in inventory models. Int J Forecast. 2019;35(1):224–38.
29. Dancho M. Use Machine Learning to Predict and Optimize Product Backorders. Business Science Article. Business Science Article. 2017. https://www.business-science.io/business/2017/10/16/sales_backorder_prediction.html. Accessed 15 Feb 2020.
30. Petropoulos F, Wang X, Disney SM. The inventory performance of forecasting methods: evidence from the M3 competition data. Int J Forecast. 2019;35(1):251–65.
31. Zhang GP. Time series forecasting using a hybrid ARIMA and neural network model. Neurocomputing. 2003;50:159–75.
32. Yu L, Zhao Y, Tang L, Yang Z. Online big data-driven oil consumption forecasting with Google trends. Int J Forecast. 2019;35(1):213–23.
33. Hyndman RJ, Koehler AB. Another look at measures of forecast accuracy. Int J Forecast. 2006;22(4):679–88.
34. Kim S, Kim H. A new metric of absolute percentage error for intermittent demand forecasts. Int J Forecast. 2016;32(3):669–79.
35. Martínez A, Schmuck C, Pereverzyev S Jr, Pirker C, Haltmeier M. A machine learning framework for customer purchase prediction in the non-contractual setting. Eur J Oper Res. 2020;281(3):588–96.
36. De Baets S, Harvey N. Forecasting from time series subject to sporadic perturbations: effectiveness of different types of forecasting support. Int J Forecast. 2018;34(2):163–80.
37. Kotsiantis SB. Decision trees: a recent overview. Artif Intell Rev. 2013;39(4):261–83.
38. Najafabadi MM, Villanustre F, Khoshgoftaar TM, Seliya N, Wald R, Muharemagic E. Deep learning applications and challenges in big data analytics. J Big Data. 2015;2(1):1.
39. Khosravi K, Pham BT, Chapi K, Shirzadi A, Shahabi H, Revhaug I, Bui DT. A comparative assessment of decision trees algorithms for flash flood susceptibility modeling at Haraz watershed, northern Iran. Sci Total Environ. 2018;627:744–55.
40. Chiabaut J. U.S. Patent No. 8,761,022. Washington: U.S. Patent and Trademark Office. 2014.

41. Rutkowski L, Jaworski M, Pietruczuk L, Duda P. The CART decision tree for mining data streams. Inf Sci. 2014;266:1–15.
42. Ye Y, Wu Q, Huang JZ, Ng MK, Li X. Stratified sampling for feature subspace selection in random forests for high dimensional data. Pattern Recogn. 2013;46(3):769–87.
43. Alsolami F, Azad M, Chikalov I, Moshkov M. Multi-pruning and Restricted Multi-pruning of Decision Trees. Decision and Inhibitory Trees and Rules for Decision Tables with Many-valued Decisions. Cham: Springer; 2020. p. 153–174.
44. Lee S, Gonzalez J, Wright M. Interpretable few-shot image classification with neural-backed decision trees. 2020.
45. Araz OM, Olson D, Ramirez-Nafarrate A. Predictive analytics for hospital admissions from the emergency department using triage information. Int J Prod Econ. 2019;208:199–207.
46. Biau G, Cadre B, Rouvière L. Accelerated gradient boosting. Machine Learning. 2019;108(6):971–92.
47. Ernst D, Geurts P, Wehenkel L. Tree-based batch mode reinforcement learning. J Mach Learn Res. 2005;6:503–56.
48. Yang Y, Qian W, Zou H. Insurance premium prediction via gradient tree-boosted tweedie compound poisson models. J Bus Econ Stat. 2018;36(3):456–70.
49. Spearman C. The proof and measurement of association between two things. Am J Psychol. 1987;100(3/4):441–71.
50. Ernst R, Powell SG. Manufacturer incentives to improve retail service levels. Eur J Oper Res. 1998;104(3):437–50.
51. Appelqvist P, Gubi E. Postponed variety creation: case study in consumer electronics retail. Int J Retail Distrib Manag. 2005;33(10):734–48.

Publisher's Note