Detailed Project Report - Backorder Prediction

1.Project Overview:

When a customer orders a product, which is not readily available due to lack of unavailability of the product in the store or not in the inventory but can guarantee the delivery of the ordered product a certain date in the future and the customer waits for the same. This scenario is called backorder of that specific product.

Backordering has good and bad faces in the business. So there has to be some balance between the demand and supply. ML Predictive analysis can schedule the production, supply chain and inventory helps to forecast product delivery delay which in return increase the customer satisfaction

If the back orders are not handled promptly it reflects high impacts on the company's revenue, share market price, and customer's trust that leads to losing the customer as well as the order. On the other hand, to satisfy backorders leads to enormous pressure on different stages of the supply chain which may break (exhaust) or can cause extra costs on production, shipment …etc.

Nowadays, most of the company's use Machine learning predictive analysis to predict products back orders to overcome the tangible and intangible costs of backorders.

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.

2.Scope of Detailed Project Report (DPR):

Detailed Project Reports (DPRs) are the outputs of planning and design phase of a project. DPR is a very detailed and elaborate plan for a project indicating overall project. A DPR is a final, detailed appraisal report on the project and a blue print for its execution and eventual operation. It provides details of the basic programme the roles and responsibilities, all the activities to be carried out and the resources required and possible risk with recommended measure to counter them.

3. How Backorders differ from Out of Stock:

An item is out of stock when the seller doesn't have the item in inventory and has no sure date to restock, or the item is seasonal or a limited run. Backordered items are expected to be available in a reasonable timeframe.

3.1. What Causes Backorders?

- **Unusual demand or demand exceeds supply:** Holiday seasons or other unforeseen circumstances, like extreme weather, may lead to unusual purchasing.
- **Inaccurate forecasting:** Less accurate forecasts could lead to low safety stock and a higher chance of backorders.
- **Supplier or manufacturing issue**: Supply chain challenges, such as factory shutdowns or raw material shortages, can lead to items being unexpectedly out of stock.
- **Delayed orders:** Companies that order based on safety stock formulas and require manual review of purchase orders may run into restocking delays, only to experience a sudden surge of orders.
- **Human errors:** An employee may enter an order as a backorder even if the item is available. Or, Warehouse management discrepancies: A glitch in an inventory management system may provide inaccurate data, or a breakdown in data entry may lead to stock being miscounted or misplaced. worse, a retailer may accept a backorder even though the item is out of stock.

3.2. Advantages of Backorders:

- **Offers market insights:** Backorders act like customer surveys, indicating what kinds of products buyers want, and when they're in the highest demand.
- **Improved cash flow:** Companies that avoid holding excess stock, with the associated costs, free up cash for other priorities. In some industries, less inventory also translates to reduced taxes.
- **Minimizing storage** and other inventory costs that come with holding extra stock.

3.3. Disadvantages of Backorders

- **Losing out on business:** Customers may not want to wait, or trust the company to fulfil their orders, causing them to cancel and purchase elsewhere.
- **Loss of market share:** If customers frequently encounter backorders or must wait a long time for fulfilment, their loyalty to your brand may wane, and they could turn to other brands.
- **Increased complexity:** Backorders increase the chances of a company having to resolve customer service issues, such as trying to update expired payment information.

3.4. How to avoid Backorders?

- A well planned Supply Chain Management, Warehouse management and inventory management can avoid or minimize Backorders to some extent.
- Manufacturers or suppliers can also hit peak demand. diversify suppliers or ordering from a wide variety of sources is good to some extent.
- Increasing inventory or stock of products is not a solution as it increases storage costs and extra costs means they have to be included in the product prices which might result in losing business to competitors**.**

4. The Problem Statement:

- Classify the products whether they would go into Backorder (**Yes / No**) based on the available data from inventory, supply chain and sales.
- The task is to classifying whether a product would go to Backorder based on the given input data.
- The target variable to predict consists of two values, if it is "Yes" - the predicted product to be treated as Backorder. If the output is "No"- the predicted product to be not going to Backorder
- So we can have considered this is a **Binary Classification problem**.

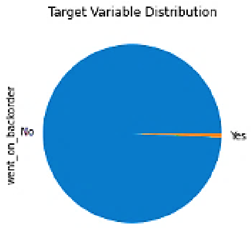
5.Data Exploration:

5.1Data Overview

- The dataset of this project work has been published in Kaggle (Currently the page is not available). The dataset is divided into the training and testing datasets.
  Data Source: https://github.com/rodrigosantis1/backorder_prediction/blob/master/dataset.rar
- The dataset is **highly imbalanced** which should be addressed for accurate predictions by the model

- Each dataset contains **23 attributes with 1,687,862 and 242,077 observations** for the training and testing sets, respectively.

- 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 labelled 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.
- It is observable 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.



- The data samples are distributed among two classes, where 0 indicates ‘No’ class or no backorder items and 1 indicates ‘Yes’ class or backorder items.
- Highly Class imbalance: Most of the samples did not went to back order (99.83%) of the time, while actual back orders occurs (0.17%) of the time.
- The data points which are responsible for extreme positive skewedness may not actually be outliers.

Suppose where the sales of a particular product are very high in the past 12 months. This implies that forecast for coming 6 and 9 months will be high. If forecast and sales is high this implies inventory and in transit quantity for the particular product would be high. Also the amount of sales would be very high for a few products compared to others. Considering these the positive skewedness present in the data may not truly represent outliers.

6.How do we organize our analysis?

Our goal as a Data Scientist is to identify the most important variables and to define the best regression model for predicting out target variable. Hence, this analysis will be divided into five stages

1.Data Preprocessing & Exploratory data analysis (EDA):

- The pre-processing of data is a method for preparing and adapting raw data to a model of learning. This is the first and significant step to construct a machine learning model.

- Data pre-processing needs to be performed in order to purify data and adapt it to the machine learning model of a system which also makes a machine learning model more accurate and efficient.
- Understand and visualize data with the help of libraries like Histograms, Density plots, Scatter plots, Boxplots and Correlation Matrix.
- Analysis and imputation of Missing Variables.
- Short listing of most important features.

2. Feature Engineering & Feature Transformation:

- Feature Engineering is a method to exploit domain data understanding to construct functions that work with machine learning algorithms.
- Feature engineering helps to enhance the predictive capability of machine learning algorithms to building raw data features that simplify the machine learning process.
- Correction of inappropriate values.
- Transforming and scaling features.
- Find out Imitative variables.
- Find out correlated (Interrelated) Variables.

3.Modeling:

- The dataset is now ready to fit a model. The training set is fed into the algorithm in order to learn how to predict values. Testing data is given as input after Model Building a target variable to predict.
- For all models based on the below algorithms, 20-fold cross validation can be used. Essentially cross validation provides an indication of how well a model is generalizing to the unseen results.
- Testing with Linear and Tree-based models.
- Linear based Models like Multi –Linear, Ridge, and Lasso
- Tree-based Models like Decision Tree, Random Forest, Radiant Boosting

4. Hyperparameter tuning:

- Hyperparameter tuning selects an optimal range of hyperparameters for algorithm learning. A hyperparameter for this is a parameter the value of which is set before learning starts. Hyperparameters are not model parameters, and cannot be directly derived from results. Whilst the model parameters specify how input data can be translated to the desired output, the hyperparameters explain how the model is actually being structured. The best way to think of hyperparameters is like an algorithm’s settings which can be modified to maximize performance.
- **Grid Search and 2. Random search**, are the methods used for to get the best performance of the model. Computational methods for both Grid search and Random search tuning take a very long time, from an hour to a day. The **Bayesian Optimization** approach can be used quickest calculation for hyperparameter tuning.

- Adjusting hyperparameaters based on the Gird /Random/search

5. Ensembling:

- Ensemble methods are a machine learning technique that combines several base models in order to produce one optimal predictive model.

- Stacking regressor models with Lasso, Ridge, Random forest can also have a try.

7. Prediction:

Prediction deals with events occurring in the future. The use of Machine learning algorithms improves the intelligence of the system without manual intervention. Machine learning techniques can be applied to all disciplines. Machine learning uses statistics to solve many classification and clustering problems. The ML algorithms are classified in three categories. In this project we are testing with three machine learning algorithms which can be applied to prediction, like Linear Model (LM), Decision Tree (DT) and Gradient Boost Tree (GBT).

7.1. Linear Model:

The most common and simplest statistical approach for predictive modeling is linear regression. Below is the linear regression equation:

Y = _1X1 + _2X2 + :::: _nXn, Where X1, X2,..., Xn are the independent variables, Y is the target variable and all the coefficients are the thetas. The magnitude of a coefficient as compared to the other variables determines the importance of the corresponding independent variable. This algorithm's basic principle is to match a straight line between the chosen training dataset features and a constant target variable, i.e. sales. The algorithm chooses a line which fits better with the data. Linear regression performs the task of predicting a dependent variable value (y) based on a given independent variable (x). This regression technique considers a linear relationship between x (input) and y (output). Some requirements for a successful linear regression model must be fulfilled by data. Some of those is the lack of multicollinearity, i.e. the independent variables should correlate with each other.

7.2. Ridge Regression:

Ridge Regression is a method used where multicollinearity (independent variables are highly correlated) affects outcomes. While the least square estimates (OLS) are objective in multicollinearity, their variances are broad and deviate from the true value. By applying a degree of bias to regression calculations, ridge regression eliminates standard errors. The Linear Regression Loss function is increased in Ridge Regression so as not only to minimize the number of square residuals but also to penalize the estimates of the parameters.

7.3. Random Forest:

Random Forest is a tree-based bootstrapping algorithm that combines a certain number of weak learners (decision trees) to construct a powerful model of prediction. For each person learner, a random set of rows and a few randomly selected variables are used to create a decision tree model. Final prediction may be a function of all the predictions made by the individual learners. In the event of a regression problem, the final prediction may be the mean for all predictions.

7.4. Decision Tree:

Decision tree is a classifier referred as recursive partition of the instant space. It is a powerful form of multiple variable analyses and is a strong data mining tool. Its applications are found in various domains and this approach represents factors involved in achieving a predetermined goal and the corresponding factors to achieve the goal and the ways and means of implementation.

7.5. Gradient Boosted Trees:

Gradient boosting is a machine learning technique for regression and classification problem. This approach could ensemble learning method that combines large number of decision trees to produce final prediction model. This model is built on a principle that a collection of weak learners combined together can produce a strong learner by using boosting process. GBT approach has a strong additive training method, required for adding a new weak learner into the model, the weak learner is the decision tree.

8. Model Deployment:

We will be deploying the model in Heroku cloud platform.

8.1 Heroku:

Heroku is a cloud platform as a service supporting several programming languages. One of the first cloud platforms, Heroku has been in development since June 2007, when it supported only the Ruby programming language, but now supports Java, Node.js, Scala, Clojure, Python, PHP, and Go.

8.2. Flask:

Flask is a micro web framework written in python. Here we use flask to create an API that allows to send data, and receive a prediction as a response. Flask supports extensions that can add application features as if they were implemented in Flask itself.

8.3. Database:

We use Cassandra DB to retrieve, insert, delete and update the customer input database. Flask is a micro web framework written in python. Here we use the flask to create an API that allows sending data, and receive a prediction as a response. Flask API's act as an interface between the user, ML- Model and DB. Users interact with the deployed model on HEROKU and the result will be returned to the user by API's. The Flask API will collect all the customer input data store in Cassandra DB.

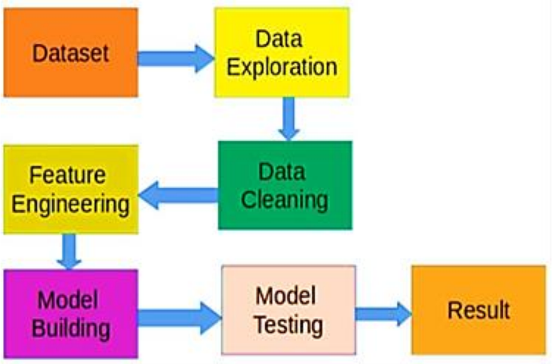


Figure 1.Working Procedure of Proposed Model

9.FAQ's & Answers

Q1) What's the source?

- The dataset of this project work has been published in Kaggle (Currently the page is not available). The dataset is divided into the training and testing datasets.
- Data Source: https://github.com/rodrigosantis1/backorder_prediction/blob/master/dataset.rar

Q2) What was the type of the data?

- Total 23 features given in the dataset 15 are numerical and 8 (including the target variable) are categorical features.

Q3) Describe the overall flow of this project?

- Pls. refer Page 6 Figure.1 also refer architecture design Document.

Q4) After the File validation what you do with incompatible file or files which didn't pass the validation?

- We concatenated the training and test data CSV file as one, used for the prediction

Q5) How Logs managed?

- In production we used different logs to monitor model training log, prediction log etc.

Q6) What techniques were used for data preprocessing?

- Removing unwanted attributes
- Visualizing relation of independent variables with each other and output variables.
- Checking and changing Distribution of continuous values.
- Removing outliers.
- Cleaning data and imputing if null values are present.
- Converting categorical data into numerical values.
- Scaling the data.
- Handling class imbalance, to resolve we can use Near miss Under sampling.
- RandomizedSearchCV used for Hyper parameter tuning is choosing a set of optimal hyper parameters for a learning algorithm

Q7) How was training done? what models were used?

- Training and test sets were split using an 80:20 proportion, generating 50 different combinations. A stratified 2-fold cross validation scheme is adopted in training to avoid overfitting in training.
- Random search of parameters, using 2-fold cross validation, search across 50 different combinations.

- Total 11 Algorithms applied including Random Forest, Naïve Bayes, Logistic Regression, XGBoost, Decision Tree, Light GBM.

Q8) How did the prediction come about?

- The testing files are shared by the client. We pass this data to a saved model, then we get the prediction, which is displayed in the prediction page and also those data are stored in Casendra Database
- Light GBN was the highest –Model Accuracy:96.63

Q9) What are the different stages of deployment?

- When the model is ready we deploy it using Flash framework, where SIT and UAT is performed over it. Once our web App running in local successfully, we deployed in Heroku and UAT is performed over it.