A Project Report
on

Predicting Material Backorders in Inventory Management using Machine Learning

Work carried out for
Technocolabs Softwares Pvt. Ltd., Indore, M.P.

Submitted to Pondicherry University in partial fulfillment of the requirement for the Award of the Degree of

Master of Business Administration
in
Data Analytics

By

RAMESH KUMAR M
(Reg. No. 21401031)

Under the Supervision of

Dr. B. RAJESWARI, ***Associate Professor***
Department of Management Studies, Pondicherry University

&

Mr. YASIN SHAH, ***CEO***,
Technocolabs Softwares Pvt. Ltd.

Department of Management Studies
Pondicherry University, Pondicherry, INDIA – 605 014

July – August 2022

DEPARTMENT OF MANAGEMENT STUDIES
SCHOOL OF MANAGEMENT
PONDICHERRY UNIVERSITY
PONDICHERRY-605014

CERTIFICATE

This is to certify that this project report entitled **"Predicting Material Backorders in Inventory Management"** done for **Technocolabs Softwares Private Limited** is submitted by **Ramesh kumar M (Reg.No:21401031),** II MBA (DA) to the **DEPARTMENT OF MANAGEMENT STUDIES, SCHOOL OF MANAGEMENT, PONDICHERRY UNIVERSITY** in partial fulfilment of the requirements for the award of the degree of **MASTER OF BUSINESS ADMINISTRATION IN DATA ANALYTICS** and is a record of an original and bonafide work done under the guidance of **Dr. B. Rajeswari**, Associate Professor, Department of Management Studies, Pondicherry University. This report has not formed the basis for the award of any degree, diploma, associateship, fellowship or other similar title to the candidate and that the report represents an independent and original work on the part of the candidate.

Dr. B. RAJESWARI
Associate Professor
Department of Management Studies

Dr. B. CHARUMATHI
Professor and Head
Department of Management Studies

Date:
Place: Pondicherry 605 014

DECLARATION

I hereby declare that the project titled, **“Predictive Material Backorders in Inventory Management using Machine Learning”** is original work done by me under the guidance of Dr. B. Rajeswari, Associate Professor, Department of Management Studies, Pondicherry University, and Yasin Shah, CEO, Technocolabs Pvt. Ltd. This project or any part thereof has not been submitted for any Degree / Diploma / Associateship / Fellowship / any other similar title or recognition to this University or any other University.

I take full responsibility for the originality of this report. I am aware that I may have to forfeit the degree if plagiarism has been detected after the award of the degree. Notwithstanding the supervision provided to me by the Faculty Guide, I warrant that any alleged act(s) of plagiarism in this project report are entirely my responsibility. Pondicherry University and/or its employees shall under no circumstances whatsoever be under any liability of any kind in respect of the aforesaid act(s) of plagiarism.

Ramesh kumar m
21401031
II MBA
Pondicherry University

Place: Pondicherry

Date:

Acknowledgements

I thank my research guide ***Dr. B. Rajeswari****, Department of Management Studies, Pondicherry University, Pondicherry for his incessant encouragement and support extended throughout the research period.*

Thanks to ***Mr. Yasin Shah****, Technocolabs Softwares, for sharing and discussion of various research related matters.*

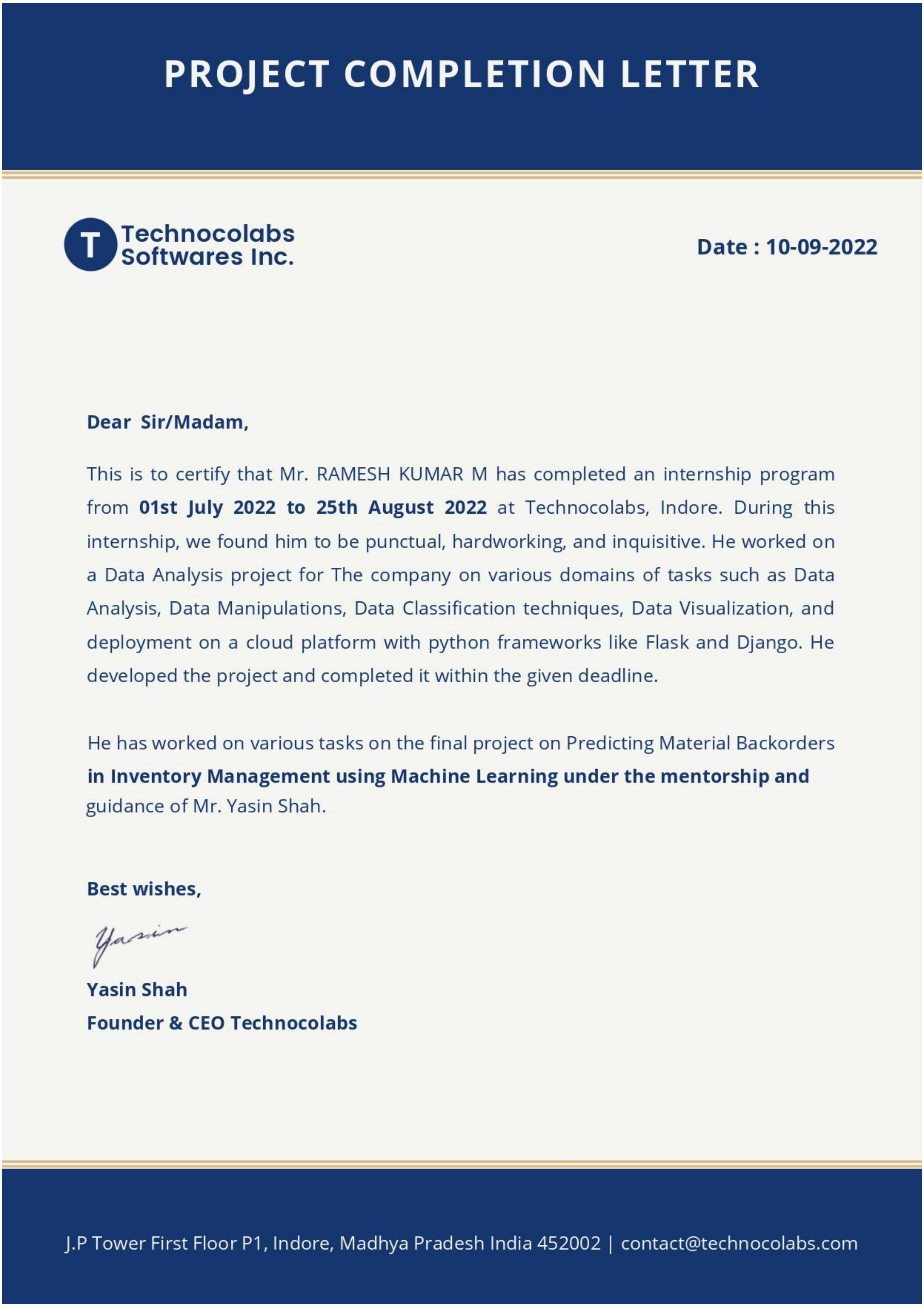
Many thanks to those faculty members who helped me in sharpening my thinking by cheerfully providing challenging comments and questions.

Non-teaching staff of the Department of Management Studies, Pondicherry University was extremely helpful to me during the research period and therefore, I am thankful to them.

I thank my classmates of Pondicherry University who had given me moral support.

Foremost, I am grateful to my family who had to tolerate late night work, and curtailed weekends & vacations. The support and encouragement of my parents gave me the energy, stamina and inspiration to complete the project.

Ramesh kumar m



PROJECT COMPLETION LETTER

Technocolabs
Softwares Inc.

Date : 10-09-2022

Dear Sir/Madam,

This is to certify that Mr. RAMESH KUMAR M has completed an internship program from **01st July 2022 to 25th August 2022** at Technocolabs, Indore. During this internship, we found him to be punctual, hardworking, and inquisitive. He worked on a Data Analysis project for The company on various domains of tasks such as Data Analysis, Data Manipulations, Data Classification techniques, Data Visualization, and deployment on a cloud platform with python frameworks like Flask and Django. He developed the project and completed it within the given deadline.

He has worked on various tasks on the final project on Predicting Material Backorders **in Inventory Management using Machine Learning under the mentorship and** guidance of Mr. Yasin Shah.

Best wishes,

Yasin

Yasin Shah
Founder & CEO Technocolabs

J.P Tower First Floor P1, Indore, Madhya Pradesh India 452002 | contact@technocolabs.com

Executive Summary

Material backorder is a common supply chain problem, impacting an inventory system service level and effectiveness. Identifying parts with the highest chances of shortage prior its occurrence can present a high opportunity to improve an overall company's performance.

In this project, machine learning classifiers are investigated in order to propose a predictive model for this imbalanced class problem, where the relative frequency of items that goes into backorder is rare when compared to items that do not.

Specific metrics such as area under the Receiver Operator Characteristic and precision-recall curves, sampling techniques and ensemble learning are employed in this particular task.

Table of Contents

Chapter – II

INTRODUCTION

INTRODUCTION:

Artificial intelligence and big data has disruptively changed the industry, as the barriers of its implementation (cost, computing power, open-source platforms, etc) disappear.

In this context, machine learning is applied on the design and development of predictive models which assess all areas of management, providing essential insights for companies to understand and react to changes in its operation.

A subject profoundly discussed in supply chain management is the inventory planning, which is an essential activity for any enterprise which tries to determine the decision about when to order and how much should order, considering different mechanisms of control.

Most of the approaches proposed so far formulate the problem as a multi-objective optimization one: ordering and storages costs must be held to a minimum, while service level is leverage as higher as possible.

A different approach for managing the inventory more efficiently - and complementary to the models developed in literature - is to identify the materials at risk of backorder before the event occurs, conferring the business a suitable time to react. A complication uprises in this particular kind of supervised learning application, since in regular inventory system the number of items which goes on backorder (positive or majority class) is utterly inferior to the amount of active items (negative or minority class).

This case is known as the class imbalance problem and it is common in many other real problems from telecommunications, web, finance-world, ecology, biology, medicine, among others, and requires appropriate techniques for handling the construction of the prediction model desired.

About the Company:

Technocolabs is an Indore, Madhya Pradesh based Start-up company. The company was established in 2019.The primary area of focus of the company is Machine Learning, Data Science and Artificial Intelligence based product development.

The Chief Executive Officer of Technocolabs is ***Mr. Yasin Shah***. The technologies that they utilise are Django, Heroku, Java, Node JS, JS, Python, C, C++, C#, Android, React, SwiftUI, VUE.js, Angular, CSS and GIT.

The services that they are offer are in the domain of Machine Learning, Computer Vision, Mobile Application Development, Voice enabled Skills, Web Application Development, Big Data and Data Science.

Mission – "Our Mission is to enhance business growth of our customers with creative design, development and to deliver market defining high quality solutions that create value and reliable competitive advantage to customers around the globe."

Plan – "Our plan is to setup requirements according to our clients and customers satisfaction and proper understanding."

Vision – "We believe that we are on the face of the earth to make great products and that's not changing. We are constantly focusing on innovating. We believe in the simple not the complex. We believe that we need to own and control the primary technologies behind the products that we make, and participate only in markets where we can make a significant contribution."

Care – "Our value defines us as a company. They are a source of inspiration as we lead the way to a brighter future for our company and all who depend on it. They support our mission of making the lives better for each and every client and patient we care for."

Chapter – III

Literature Review

A. The Imbalanced Classes Problem

In supervised learning, a dataset is said to be imbalanced when the number of instances of a given class of interest is rare when compared to the other (or others, in the case of multi-class problems). This is a problem of interest to research since there are many of classification problems of this nature in real-world, such as remote-sensing, pollution detection, risk management, fraud detection and medical diagnosis.

The balance ratio between the minority class and majority class in such applications may achieve distributions on the order of 1:100, 1:1,000 and 1:10,000. Standard learning classifiers trained using accuracy commonly perform poor results, ignoring minority classes which are treated as noise.

Several factors can increase the complexity of the imbalanced problem: the presence of small disjuncts groups of positive samples, classes overlapping hardening the induction of discriminative rules and the insufficiency of minority class examples.

To deal with these particular circumstances, several techniques have been developed and categorized into three groups according to how they address the problem.

Internal approaches provide modifications to existing classifier learning algorithms to favour the learning of positive classes; external approaches are applied in data level to adjust classes distributions prior the application of the classifiers; cost-sensitive learning framework lies between internal and external approaches, since applies both data transformations (establishing misclassification costs to instances) and algorithm level adaptation by considering costs during the training process.

To evaluate the predictive learning systems developed adopting the proposed imbalanced methods framework, specific assessment metrics are necessary, further explored later in this section.

B. Assessment Metrics in Imbalanced Domains:

Selecting the right evaluation metrics is a key determinant for guiding the construction of a predictive model. In a binary classification problem, the confusion matrix (shown in Table I) records the results of correctly and incorrectly recognized samples of each class.

TABLE I

CONFUSION MATRIX IN A BINARY CLASSIFICATION PROBLEM

	Positive prediction	Negative prediction
Positive class	True Positive (TP)	False Negative (FN)
Negative class	False Positive (FP)	True Negative (TN

$Acc = Tp + Tn / Tp + Fn + Fp + Tn$ ....(1)

Several specific metrics are proposed within imbalanced problems domain in order to take into account the class distribution: precision, defined by (2), express the accuracy of an estimator when predicting the positive class, while recall (3), also known as true positive rate or sensitivity, indicates its ability of finding all the positive samples

$P = Tp / Tp + Fp$ ....(2)

$R = Tp / Tp + Fn$ ....(3)

Precision-recall curves represent the conflict existing between both metrics and are commonly used in binary classification to understand the output of a classifier and aid the choice of the decision function threshold.

Another metric of interest obtained from the confusion matrix analysis is the fall-out (4), or the false positive rate, which is the number of false positives divided by the total number of negatives.

$F = Fp / Fp + Tn$ ....(4)

A standard approach used to evaluate classification models in imbalanced problems is to use the Receiver Operating Characteristic (ROC). Likewise, precision-recall curve, this graphic allows the visualization of the trade-off between the precision and fall-out, as it evidences that any classifier cannot increase the number of true positives without also increasing the false positives.

The Area Under the ROC Curve (AUC) corresponds to the probability of correctly identifying which one of the two stimuli is noise and which one is signal plus noise.

AUC provides a single measure of a classifier’s capability of evaluating which model is better on average and can be computed by:

$AUC = 1 + P - F / 2$ ....(5)

Other metrics can be used in classifiers evaluation, although AUC has been one of the most applied in literature for assessment and benchmark reference.

C. Sampling Methods and Ensembles in Imbalanced Datasets:

Essentially, the idea of transforming an uneven set of classes into a balanced distribution may seem a reasonable solution for the imbalanced problem.

This general idea led to the development of several techniques, known as sampling methods, which are grouped into two major groups: under-sampling, in which instances of the majority class are eliminated to adjust balance; or over-sampling, in which instances of the minority class are replicated to meet the majority one.

The use of this method is justified by a verified improvement in overall classification performance in balanced datasets when compared to imbalance datasets.

The major advantage of using these techniques is that they can be combined with any desired classifier.

Several sampling approaches have been employed so far: random replicating (or eliminating) instances from the classes, informed under-sampling intend to overcome the deficiency of information loss by deciding specific rules to determine what instances of majority class are going to be abandon, synthetic sampling seeks to create artificial data based on the similarities between the existing minority examples, data cleaning techniques are applied to remove classes overlapping prior the estimator fitting and cluster-based techniques creates synthetic instances for each class of the problem based on the cluster means of each.

The combination of sampling strategies with ensemble learning techniques has been broadly discussed in the community, given that the use of these techniques has presented higher quality results when compared to the application of the techniques apart.

Chapter – IV

OBJECTIVE

OBJECTIVE:

This project proposes the application of a supervised learning model for backorder prediction in inventory control. The main goal is to predict if a product has gone into backorder or not based on the above features. This can be posed as a binary class classification problem in machine learning.

The major contributions of this project are stated as follows:

- To compare the different learning classifiers algorithms, based on specific techniques to tackle the class im-balanced problem, such as sampling and ensembles of classifiers;

- To provide a common framework for model development, testing and evaluation, in the considered detection system design;

- Achievement of **0.9259 ±0.0025 AUC score**, adopting random forest model.

Chapter – V

OVERVIEW OF DATA

DATASET:

In this project, a real-world imbalanced dataset available on Kaggle's competition, Can You Predict Product Backorders? *Table II* summarizes the properties of the dataset: the number of attributes, positive and negative classes, samples and imbalance ratio.

The current service level of this inventory system is around 99,27%, and it is company's interest identifying parts with highest shortage risk prior the event, so short-term actions can be carried out to mitigate those risks and improve the general system performance.

TABLE II

DATASET SUMMARY

Dataset	**#Atts**	**#Pos.**	**#Neg.**	**#Total**	**Imb. Ratio**
bopredict	22	13,981	1,915,954	1,929,936	1:137

The dataset contains the historical data for the 8 weeks prior to the week we are trying to predict, taken as a weekly snapshot at the start of the week.

The dataset has the following columns:

- ***sku:*** *Stock Keeping Unit;*
- ***national_inv:*** *Current inventory level of component;*
- ***lead_time:*** *Registered transit time;*
- ***in_transit_qty:*** *In transit quantity;*
- ***forecast_3_month:*** *Forecast sales for the next 3 months;*
- ***forecast_6_month:*** *Forecast sales for the next 6 months;*
- ***forecast_9_month:*** *Forecast sales for the next 9 months;*
- ***sales_1_month:*** *Sales quantity for the prior 1 month;*
- ***sales_3_month:*** *Sales quantity for the prior 3 months;*
- ***sales_6_month:*** *Sales quantity for the prior 6 months;*
- ***sales_9_month:*** *Sales quantity for the prior 9 months;*
- ***min_bank:*** *Minimum recommended amount in stock;*
- ***potential_issue:*** *Indictor variable noting potential issue with item;*
- ***pieces_past_due:*** *Parts overdue from source;*
- ***perf_6_month_avg:*** *Source performance in last 6 months;*
- ***perf_12_month_avg:*** *Source performance in last 12 months;*
- ***local_bo_qty:*** *Amount of stock orders overdue;*
- ***deck_risk:*** *General risk flag;*
- ***oe_constraint:*** *General risk flag;*
- ***ppap_risk:*** *General risk flag;*
- ***stop_auto_buy:*** *General risk flag;*
- ***rev_stop:*** *General risk flag;*
- ***went_on_backorder:*** *Product went on backorder.*

Chapter – VI

DATA SELECTION AND CLEANING

OBSERVATIONS:

- At the first look of both the train and test data, there are **23** features including the class label (went_on_backorder).

- It is highly imbalanced dataset with **positive** classes **(11293)** being very less compared to the **negative** classes **(1676567**) in the training set.

- The ration of the positive class and negative class in the train dataset is **1:148**.

- It is observed that the feature "lead_time" has a few missing values.

- There are about **5.97%** of data point containing null values in the train set and about **6.08%** of data points containing null values in the test set.

- Among all the features, ***'sku', 'potential_issue', 'deck_risk', 'oe_constraint', 'ppap_risk', 'stop_auto_buy', 'rev_stop'*** and ***'went_on_backorder'*** are considered as categorical features.

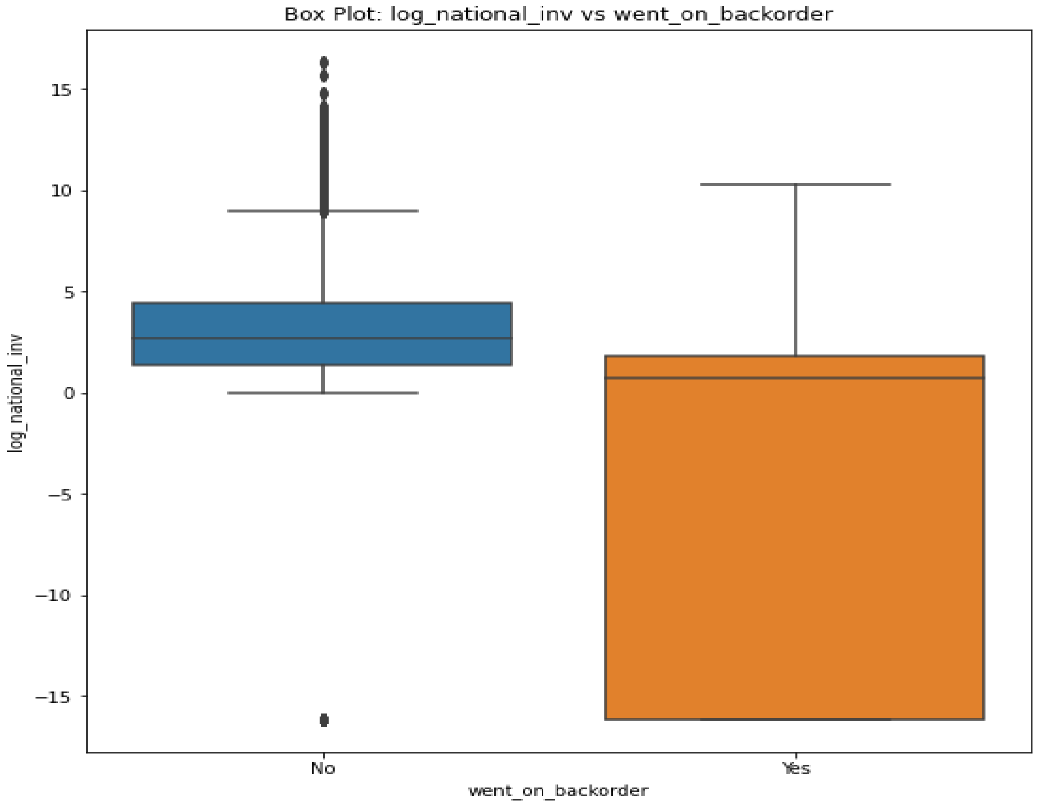
- However, ***'sku'*** is supposed to be the identifier and ***'went_on_backorder'*** is the class label. Therefore, drop them both.

Chapter – VII

EXPLORATORY DATA ANALYSIS

Let's start with univariate analysis of the features. It will be switching to bivariate or multivariate analysis whenever it is necessary.

1. national_inv vs went_on_backorder:



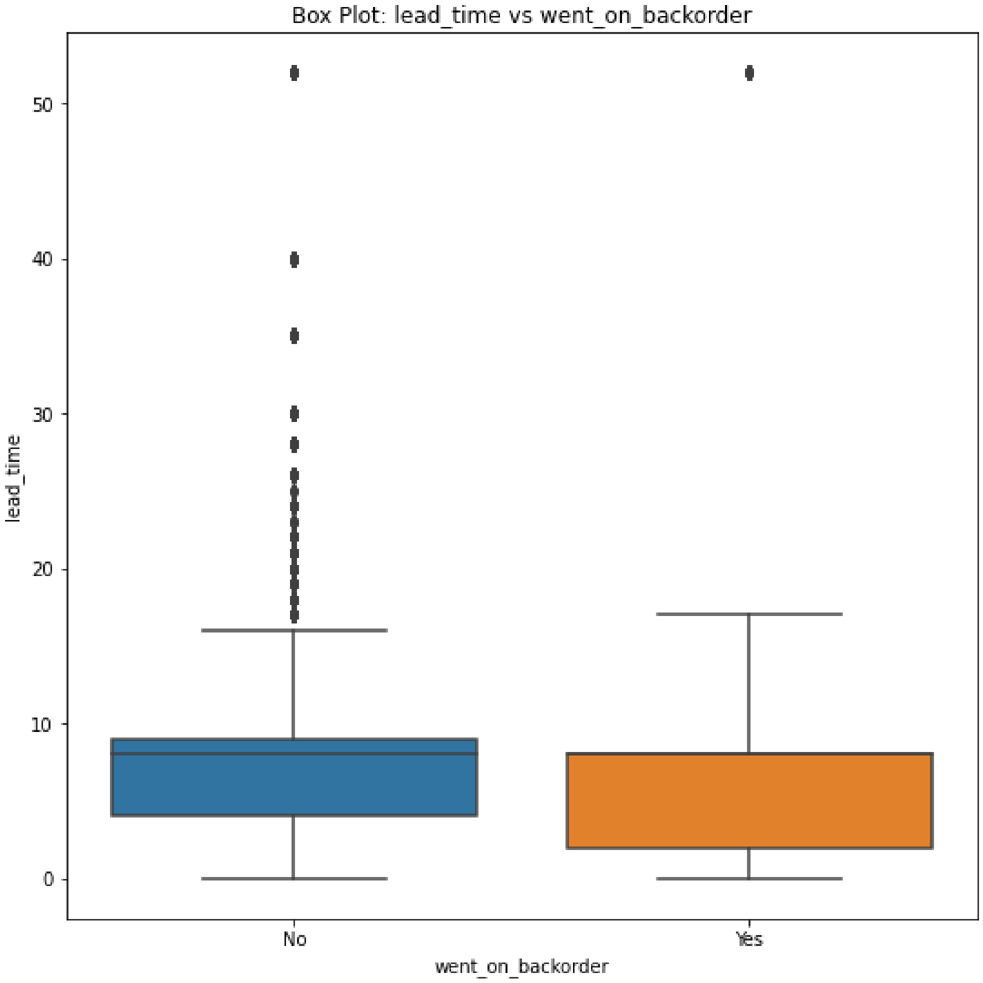
Observations:

From the initial plots, it is evident that there are a lot of outliers and the distribution is extremely skewed towards the positive side. However, it is unable to properly see that the Inter Quartile Range (IQR) for both the box plots. Therefore, this modified the national_inv to show its log values. And since there are zero values in the feature, a small value 'epsilon' is added which is 1e-7, to avoid infinity.

From the box plot of the logarithm of national_inv, it is clear that the IQRs are now visible. The median and the maximums for both the classes seems to be similar but the IQRs themselves vary a lot. Still there are outliers for the feature, especially for the negative class label.

With regard to the positive class, quickly observe that there is no separate minimum. The minimum seems to be same as the 25th percentile. And the number of points lying between the 25th percentile and the median is quite large compared to the median and the 75th percentile.

2. lead_time vs went_on_backorder:



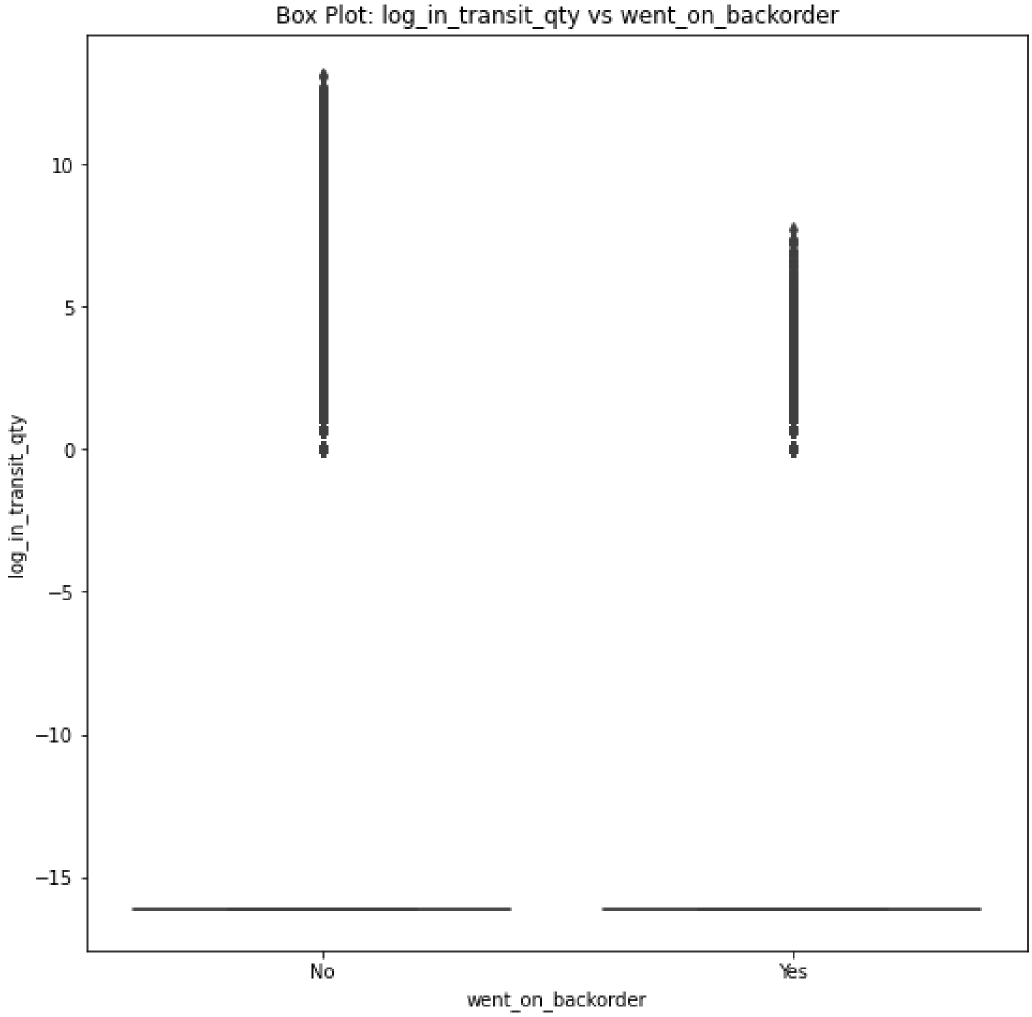
Observations:

To analyse this feature, all NaN values have been dropped. It is clear that the feature is not normally distributed as per the first pdf plot. There is a lot of overlap and we see that they are a lot of datapoints spread towards the right side of the graph which means skewness. The feature 'lead_time' is extremely skewed towards the positive side.

Look at the box plot, there is no distinct median for the positive class. The median seems to have been merged into the Q1 value. Therefore, the most of the datapoints in the feature is that one value at Q1 for the positive class. However, for the negative class we see the median but it is closer to the Q3 value. Here as well, there is skewness but due to outliers.

The minimum for both the classes seems to be similar. There are many outliers here, especially for the negative class.

3. in_transit_qty vs went_on_backorder:



Observations:

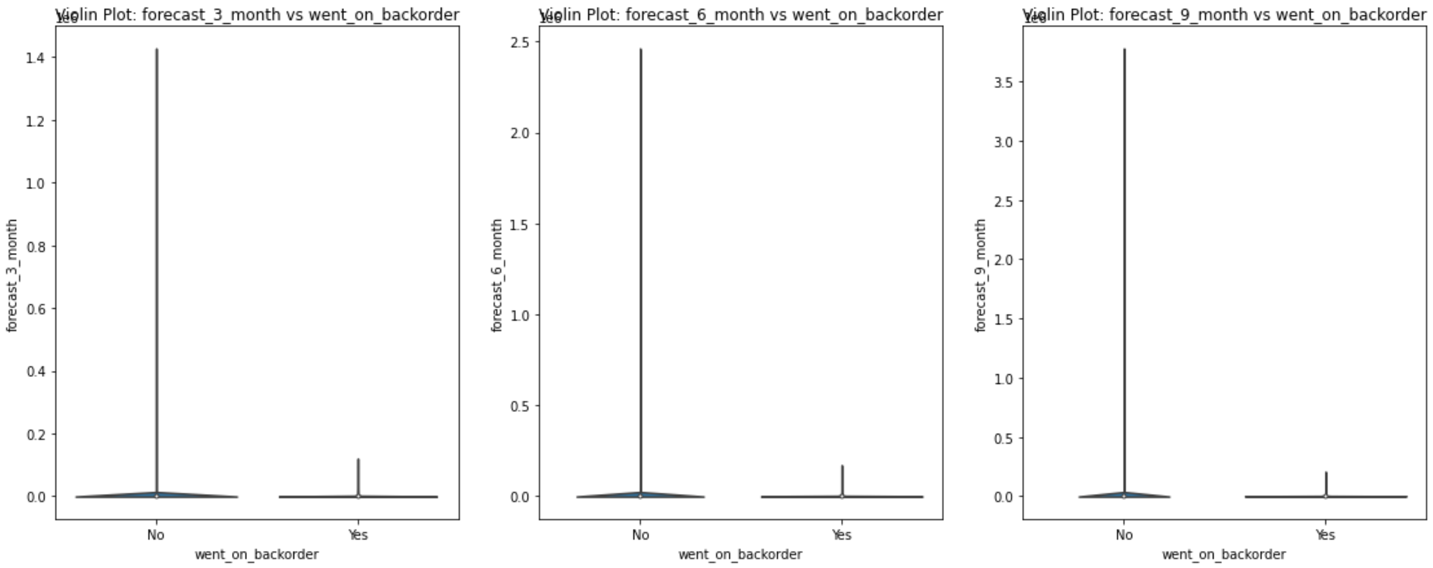
From the above plots 'PDF: in_transit_qty vs went_on_backorder' and 'Box Plot: in_transit_qty vs went_on_backorder' that the distribution of in_transit_quantity is a bit similar to national_inv. Hence, to further investigate that, we have plotted an additional plot for comparing PDFs of 'in_transit_qty' and 'national_inv'. The assumption is partially right. Both the features are positively skewed.

On the initial box plot, there are a bunch of outliers and unable to see the IQR properly, the same alternative step has employed as national_inv. a small epsilon value has added to not get any infinity values while converting to the log scale. The box plot of the logarithm of 'in_transit_qty' clearly shows the impact of outliers to be very large. Still unable to spot the IQR of 'in_transit_qty' properly.

Therefore, the mean, median and quantiles have been computed manually to better understand the data. The mean of in_transit_qty is 44.05202208713993 while the median is 0.0. As median is robust to outliers while the mean is susceptible to outliers, it is certain that outliers have a high impact on the feature. In addition, if the quantiles, the 25th, 50th and 75th percentiles are all zero. It is observed that 75% of the datapoints are equal to zero. And 90% of the point are less than or equal to 16 while the maximum value is 489408. That is a very large margin for the other 10% of the points.

Finally, it is clear that no quantity of products is in transit 75% percent of the time.

4. forecast_3_month, forecast_6_month and forecast_9_month vs went_on_backorder:



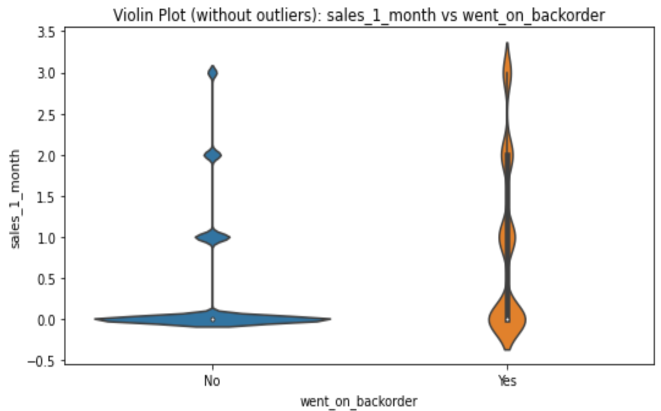
Observations:

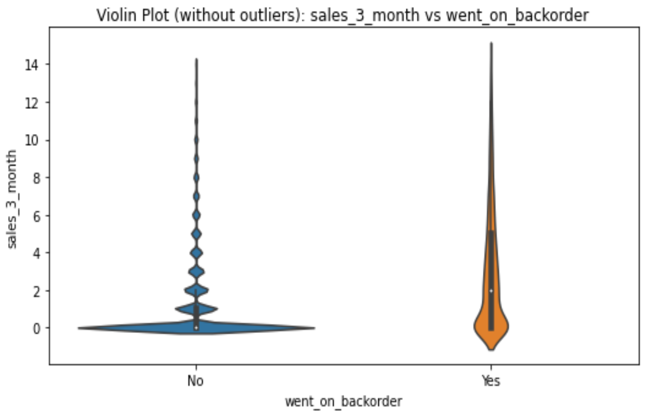
The bar plots represent an estimate of central tendency (in this case mean). Therefore, from the set of bar plots, it says that the over a span of 3, 6 and 9 months, the mean forecast sales are decreasing as a whole for the positive class while the mean forecast sales seem to be constant for the negative class.

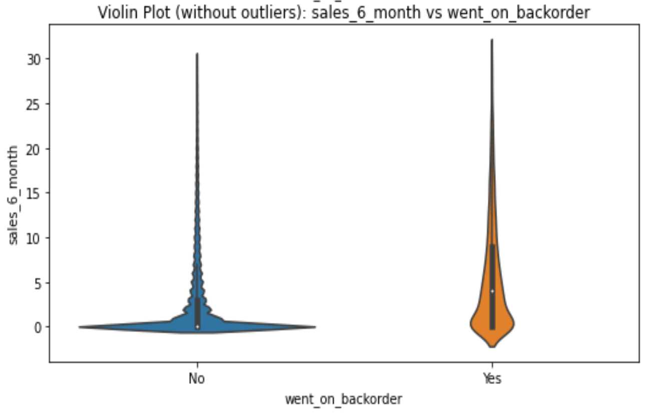
To understand the distributions and IQRs, the box plots and violin plots have plotted. It shows the IQRs are not visible here as well. And there are a lot of outliers especially for the negative class for all the 3 features. And the range of the forecast of outliers only seems to increase for the future months. The is kind of expected as the number of orders increase with time. From the violin plot, it shows that the distributions of all the three features are similar, with all being positively skewed extremely.

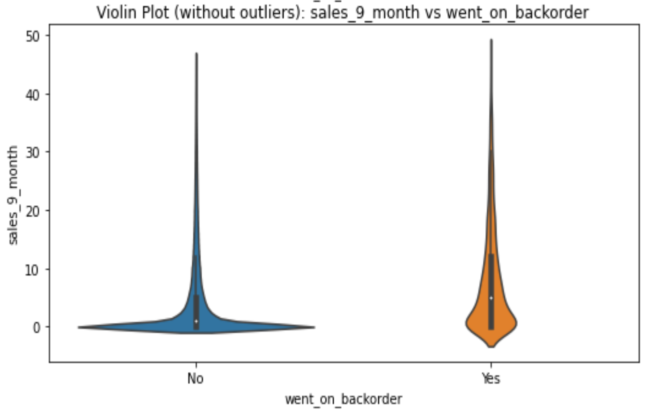
It also shows that at least 60th percentile of the datapoints is equal to zero for all the three features 'forecast_3_month', 'forecast_3_month' and 'forecast_9_month'. And there is a large margin between 90% percentile and the maximum values for the three features which again indicates outliers.

5. sales_1_month, sales_3_month, sales_6_month and sales_9_month vs went_on_backorder


Violin Plot (without outliers): sales_1_month vs went_on_backorder


Violin Plot (without outliers): sales_3_month vs went_on_backorder


Violin Plot (without outliers): sales_6_month vs went_on_backorder


Violin Plot (without outliers): sales_9_month vs went_on_backorder

Observations:

From the first set of bar plots, we understand that the mean number of orders that went into backorder over a span of a few months decreases as the number of orders increase. The violin plots indicate that the distributions are skewed.

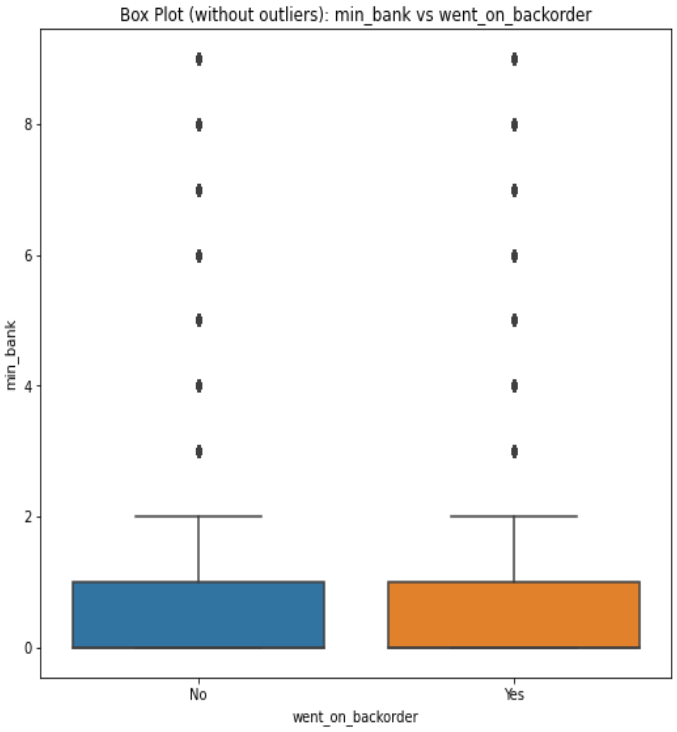
When look at the percentiles, it shows that at least 25% of the datapoints are equal to zero for all the four features and the 90th percentiles seem to have very high values compared to the rest.

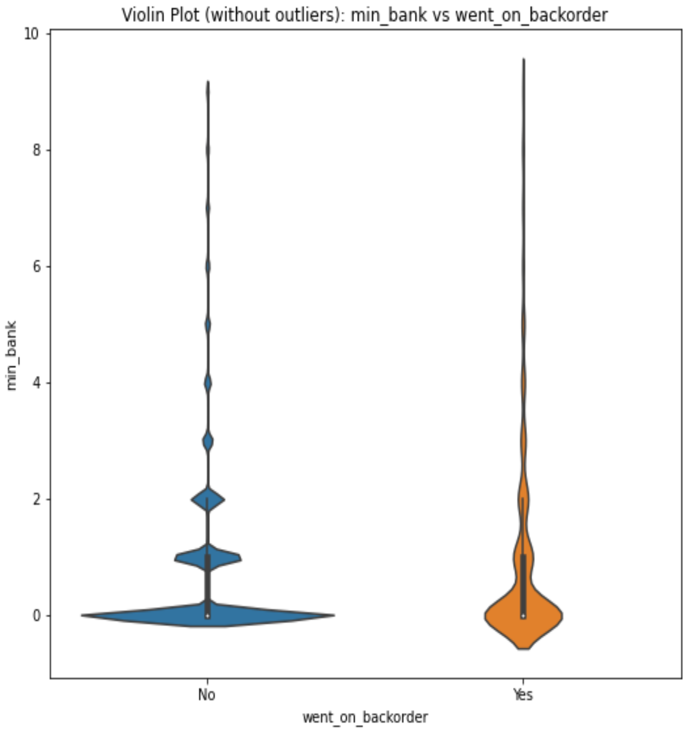
To understand that data better, the entire Q4 has been removed for all the four features and have plotted violin plots. Now it’s clear that the distributions are all skewed towards the positive side and all the data points seems to be positive integers only. Therefore, a count plot also plotted to understand the relationship between the count of sales quantity for all the four features.

It shows that there are a lot of products with zero number of units sold in all the prior months. Datapoints with at least one unit sold are more compared to datapoints with at least 3 units sold for the feature 'sales_1_month'. An extended version of this is true for all the other features i.e., datapoints with at least one unit sold are more compared to datapoints with at least 3 or more units sold.

By looking at sales quantity the prior 9 months, It shows that the number of units sold are greater than the sales quantity for the prior 3 or 6 months, which is ideal.

6. min_bank vs went_on_backorder:



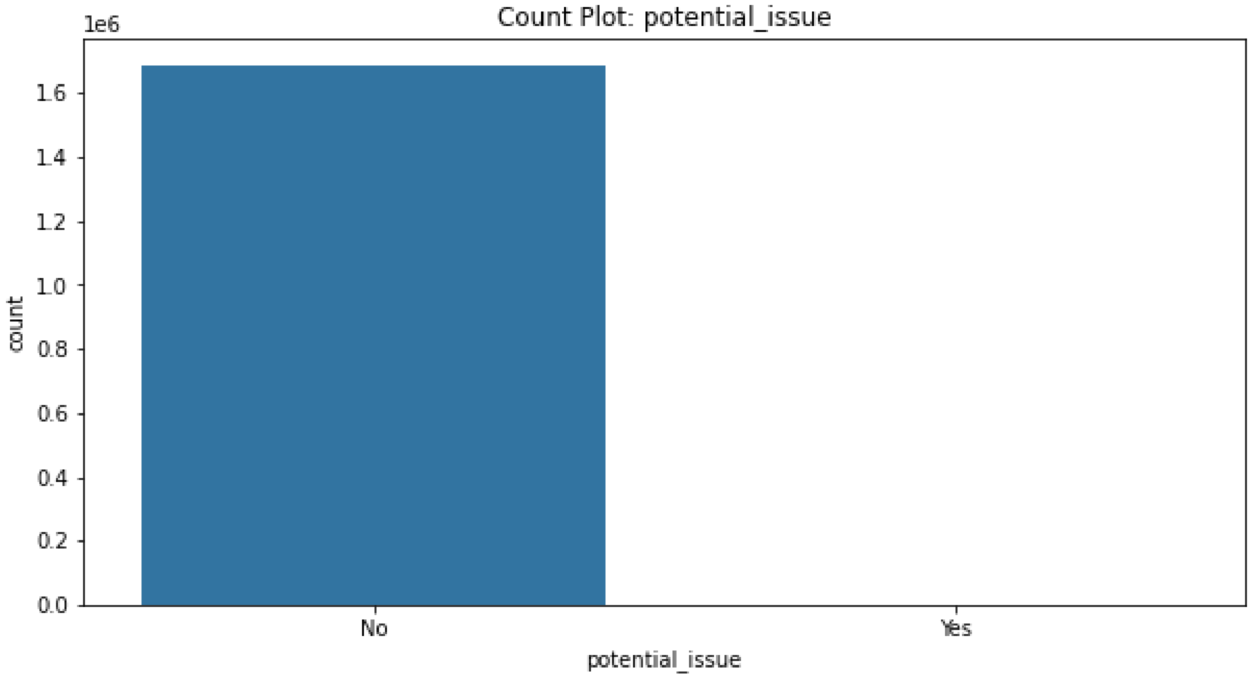


Observations:

From the box plot, most of the values tend to be zero. This statement is true is we check the quartiles. At least 50% of the data points are zero which means the median value of the feature is zero. We have tried to remove the datapoints above 80% percentile and have plotted box and a violin plot. It is observed from these plots that the values are positive integers and the maximum value that is not considered an outlier is 2.

From the count plot also, make the same deductions that most of the values tend to be zero and there are very less data points with a min_bank value of 3 or more.

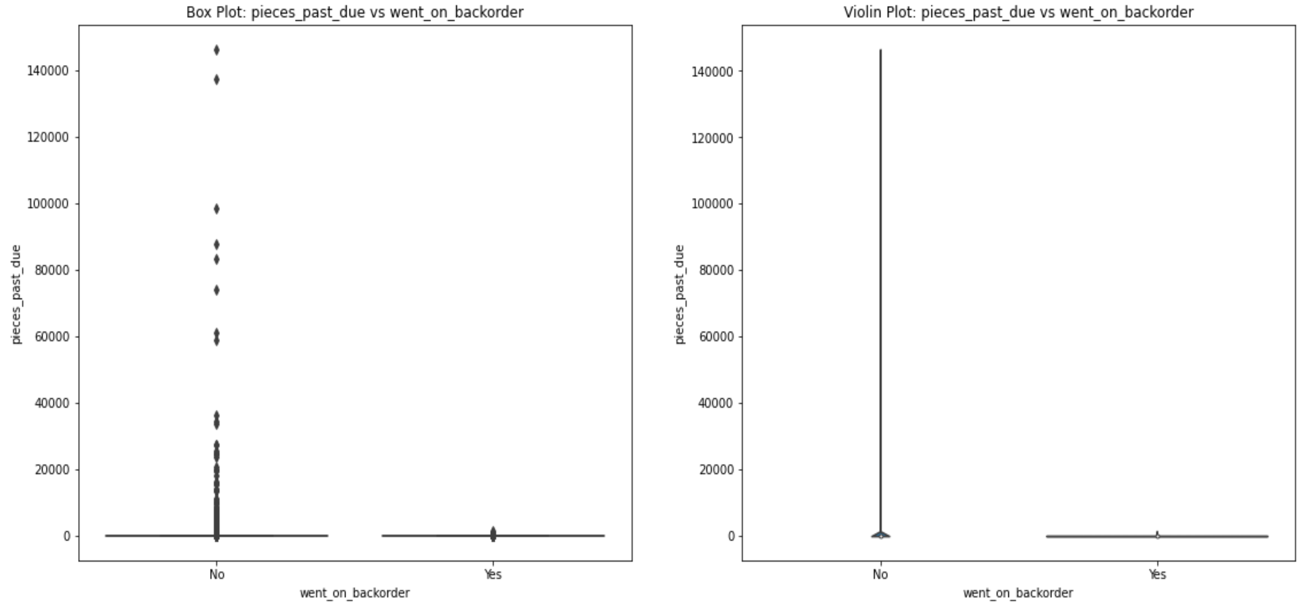
7. potential_issue vs went_on_backorder:



Observations:

It clear that the feature potential_issue is a categorical feature. From the count plot, the count of datapoints which have a potential issue is far less that the count of datapoints which do not have any potential issue.

8. pieces_past_due vs went_on_backorder:

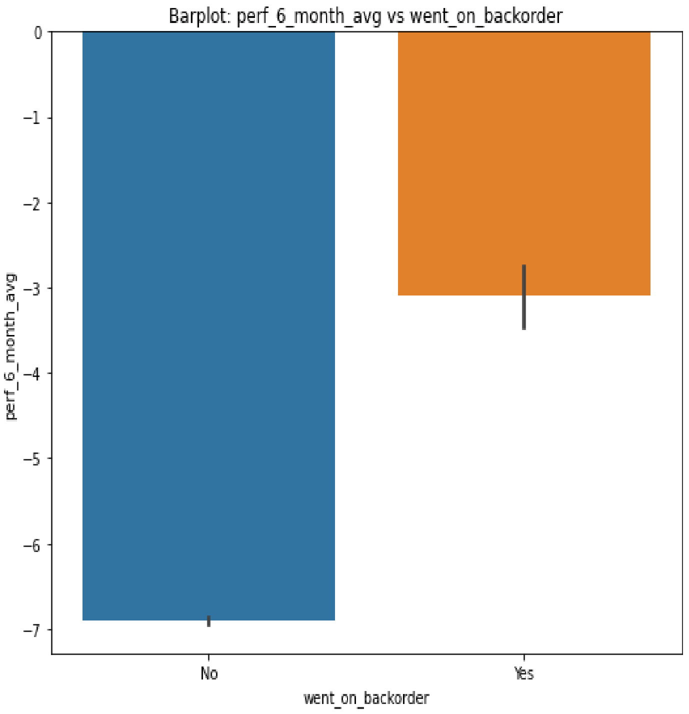


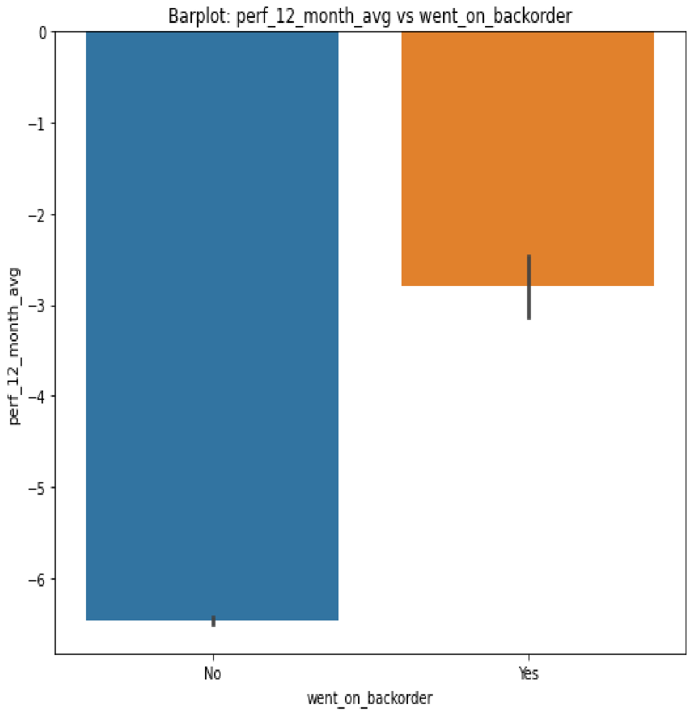
Observations:

From all the above plots, the feature is a large number of instances as zero. By looking at the quartiles, at least 98% of the datapoints are zero. If try to remove the outliers in this feature, let’s say around 1-2% of the datapoints, it probably would end up with all the instances in the feature being 0. It says that this feature is a sparse feature. Check the correlation matrix for all the features later in this process to see if this feature is correlated.

In addition, it needs to perform some feature engineering techniques for this feature and for all similar feature to create more meaningful feature for our model.

9. perf_6_month_avg and perf_12_month_avg vs went_on_backorder:



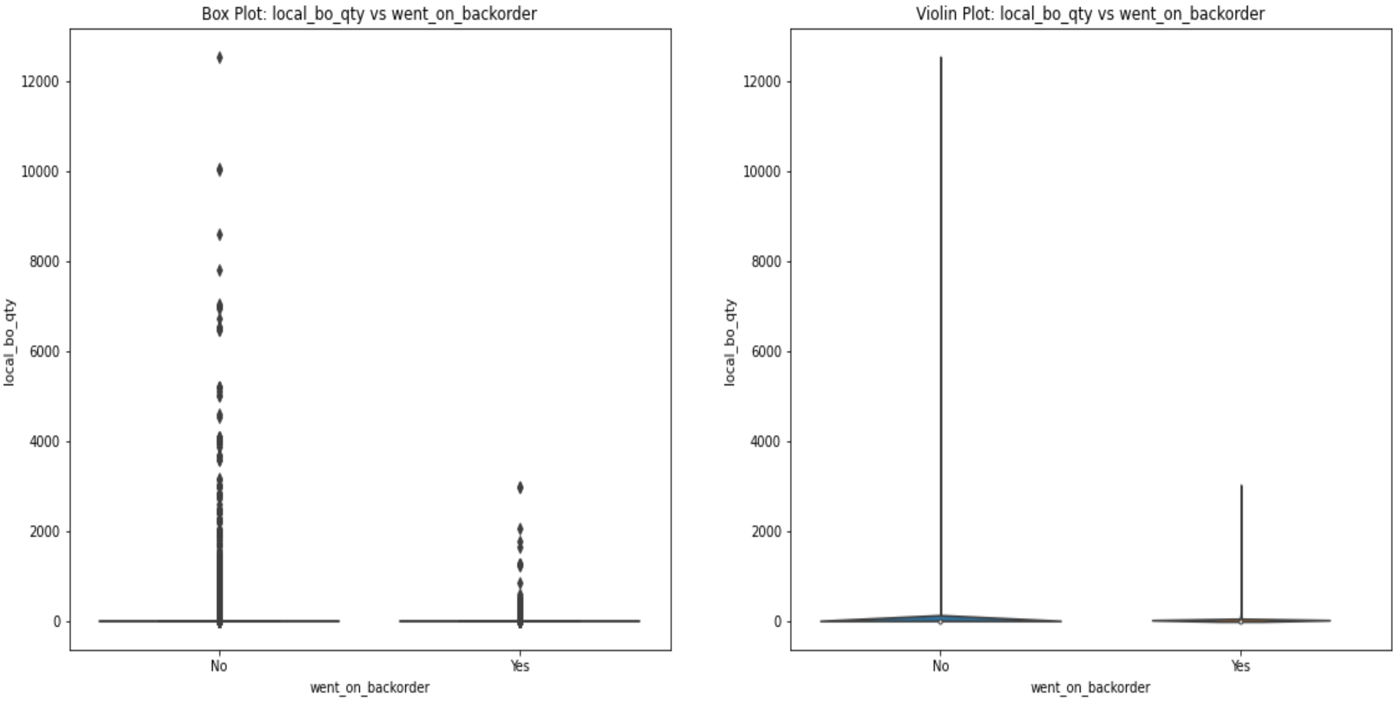


Observations:

It shows that the pdf for the two features ‘perf_6_month_avg’ and ‘perf_12_month_avg’ are very similar. It shows a gaussian-like distribution for both the features around zero. However, the curve extends extremely towards the negative axis indicating negative skewness. From the bar plots, the average source performance over 6 and 12 months is around -3 for the orders that went into backorder and around -6 to -7 for the orders which did not go into backorder.

The box and violin plots also indicate that the distribution in negatively skewed and there are a few outliers for both the classes. The median value for ‘perf_6_month_avg’ and ‘perf_12_month_avg’ is 0.82 and 0.81 respectively and 90% percent of the points are less than 0.99 for both the features.

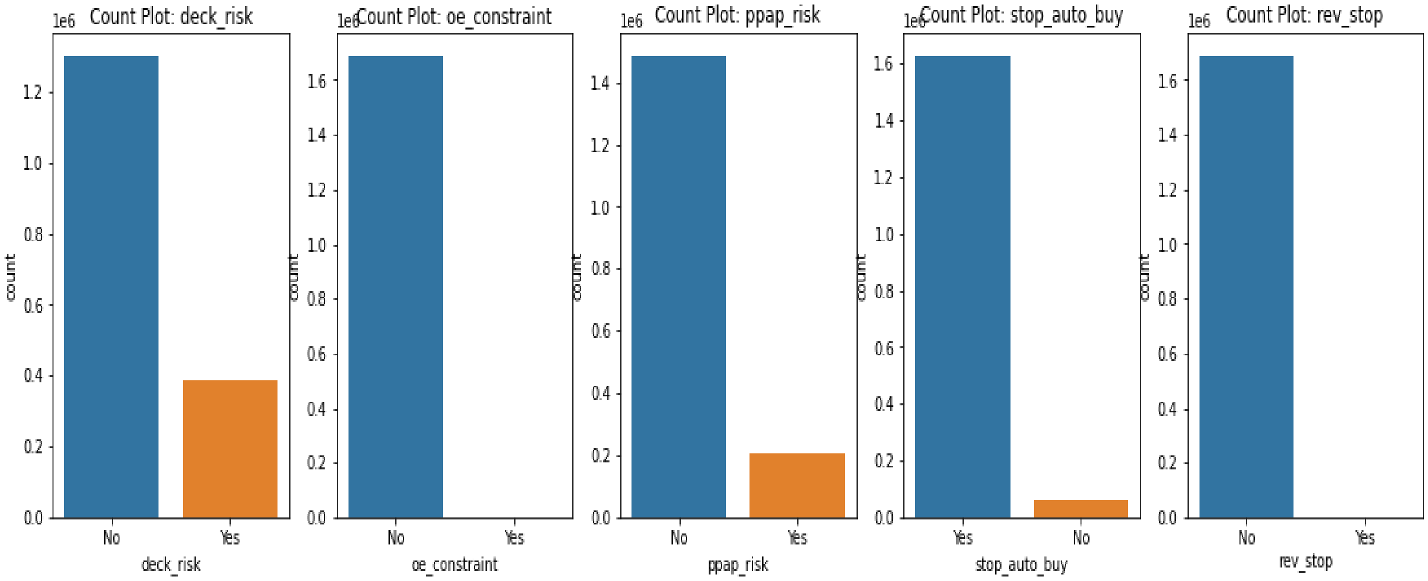
10. local_bo_qty vs went_on_backorder:



Observations:

By looking at the pdf for the feature, it shows that the majority of datapoints are at zero. This is further confirmed with the box and violin plots. To find the exact values, the percentiles have calculated. It shows that 98% percent of the datapoints are equal to zero and 99% of the datapoint are less than or equal to 1. That makes this feature a sparse feature. By looking forward at correlation matrices further in our EDA process to better understand the impact each feature has with each other and with the target.

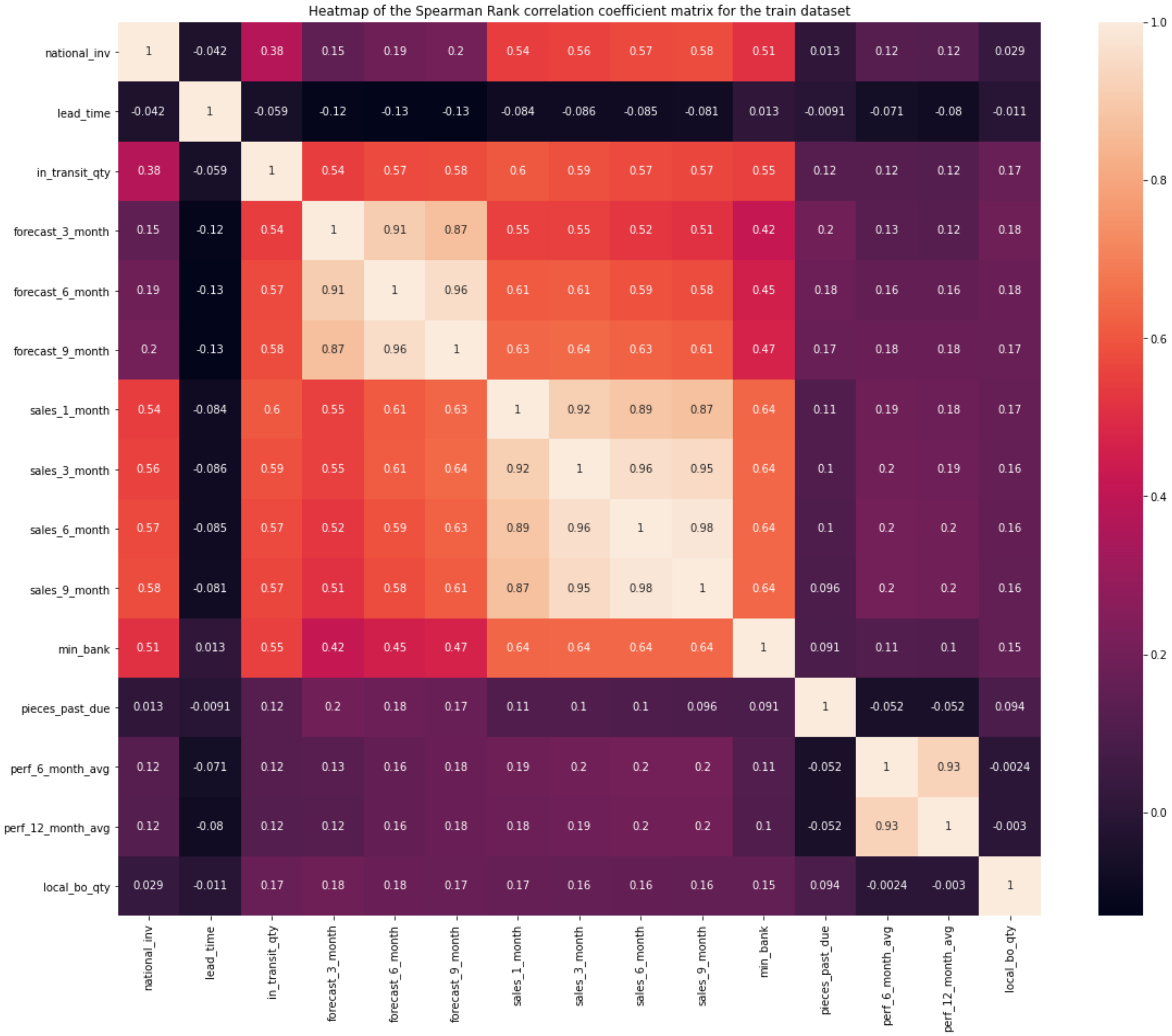
11. deck_risk, oe_constraint, ppap_risk, stop_auto_buy and rev_stop vs went_on_backorder:



Observations:

From the count plots above, it is clear that there are very less number of datapoints with the risk flags 'oe_constraint' and 'rev_stop'. There are a decent number of datapoints with 'deck_risk' as 'Yes'. And, a considerable amount of datapoints with 'ppap_risk' and 'stop_auto_buy' as 'Yes'. The majority of the datapoints do not have any risk flags in the train set.

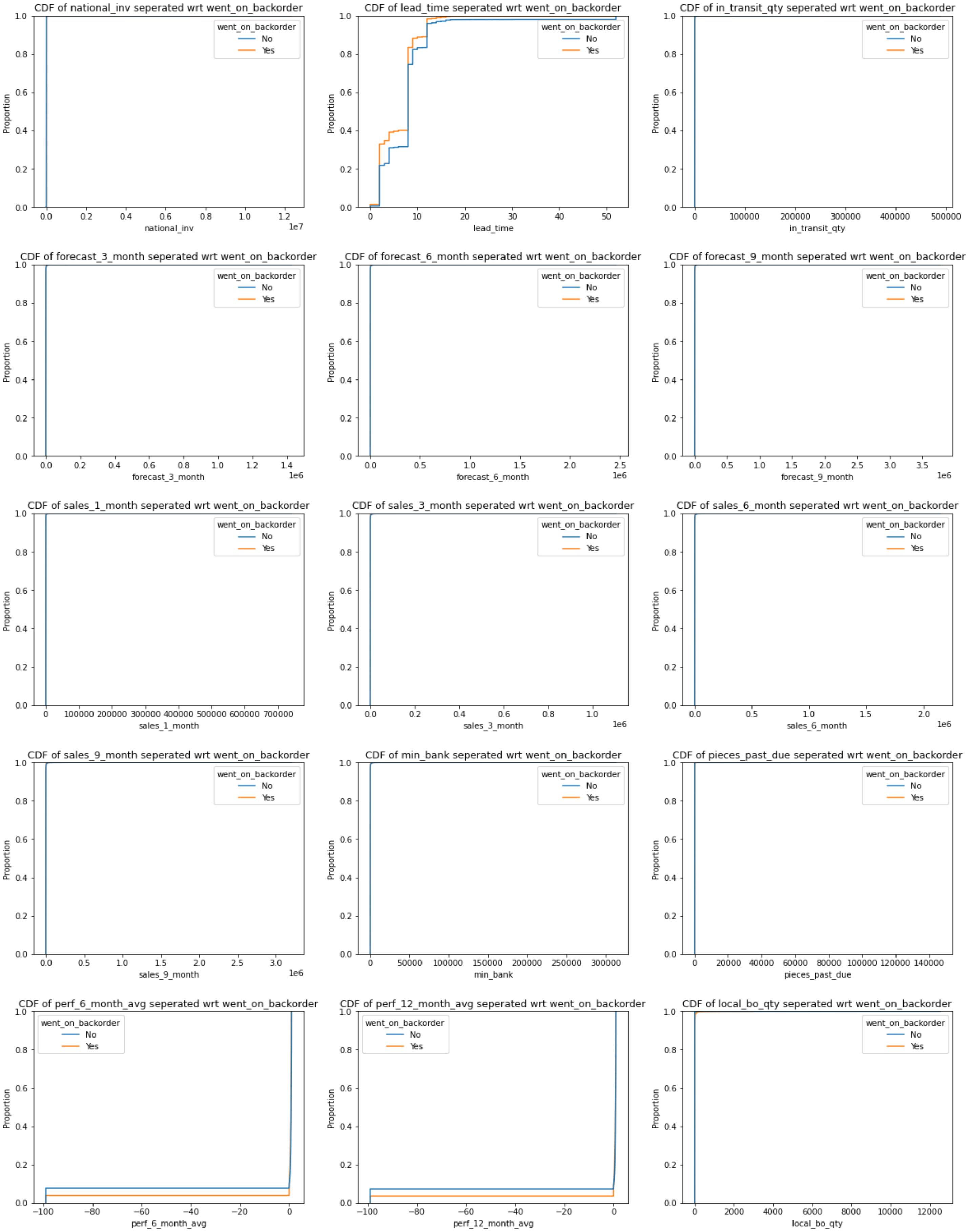
a. Spearman Rank Correlation Coefficient:


Heatmap of the Spearman Rank correlation coefficient matrix for the train dataset

Observations:

Here, the heatmaps of spearman rank correlation coefficient have been plotted. It shows that the 'in_transit_qty', 'forecast_3_month', 'forecast_6_month', 'forecast_9_month', 'sales_1_month', 'sales_3_month', 'sales_6_month', 'sales_9_month' and 'min_bank' are highly correlated with each other. Among them, 'forecast_3_month', 'forecast_6_month' and 'forecast_9_month' are more correlated with each other compared to the rest. Similarly, 'sales_1_month', 'sales_3_month', 'sales_6_month' and 'sales_9_month' are more correlated with each other than any other feature. Furthermore, it shows that the 'perf_6_month_avg' and 'perf_12_month_avg' are highly correlated with each other.

b. Kolmogorov–Smirnov test for numerical features:

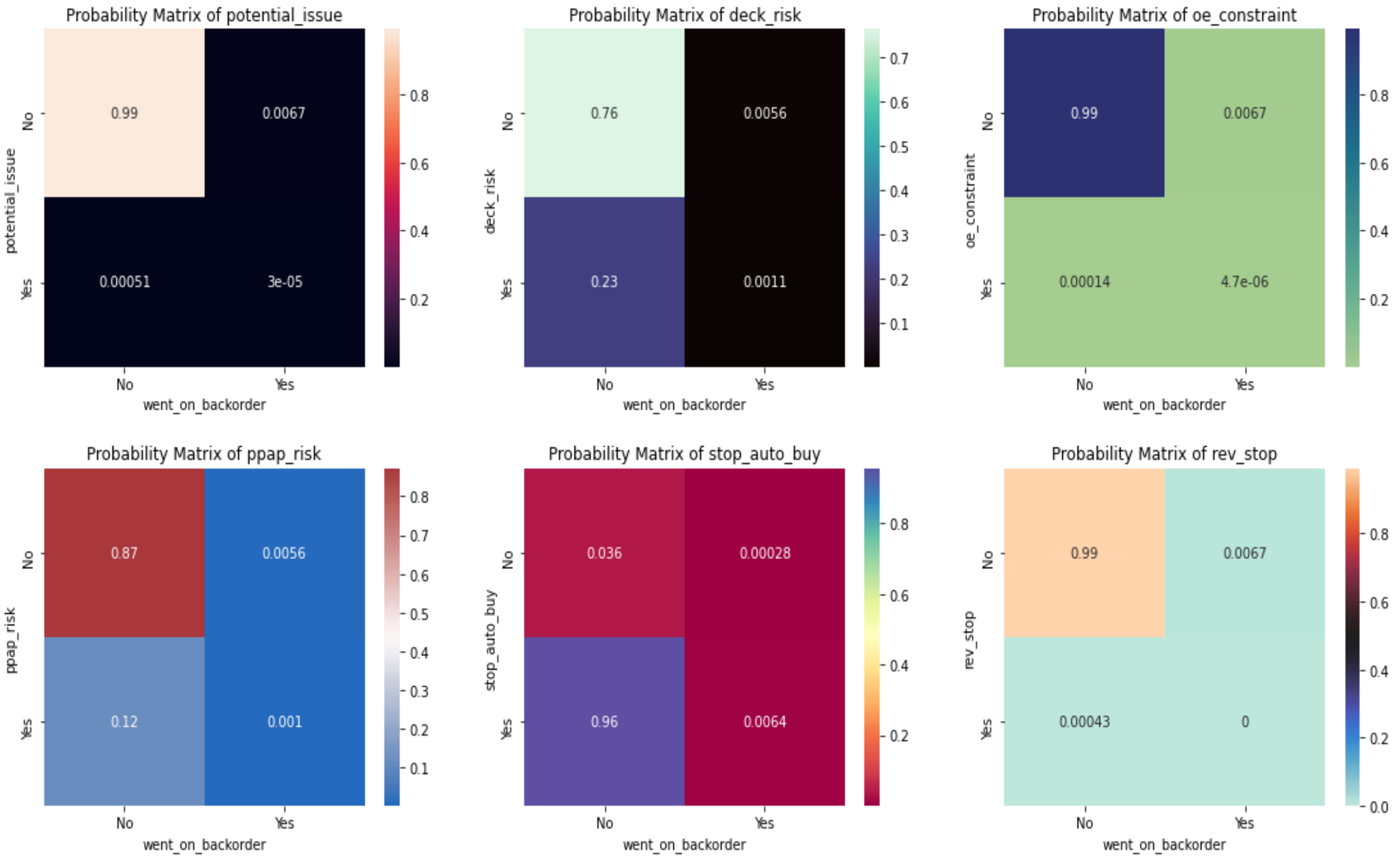


Observations:

It is clearly shows that most of the feature have very high number of datapoints at 0. From the KS test for all the numerical feature we can say most of the features do not have a very good p values and thus the null hypothesis has to be rejected. Therefore, these distributions are not similar are do not show much correlation with the target variable.

However, some features like lead_time, perf_6_month_avg, perf_12_month_avg show good enough correlation with the target variable.

c. Stochastic/Probability Matrix for categorical features:

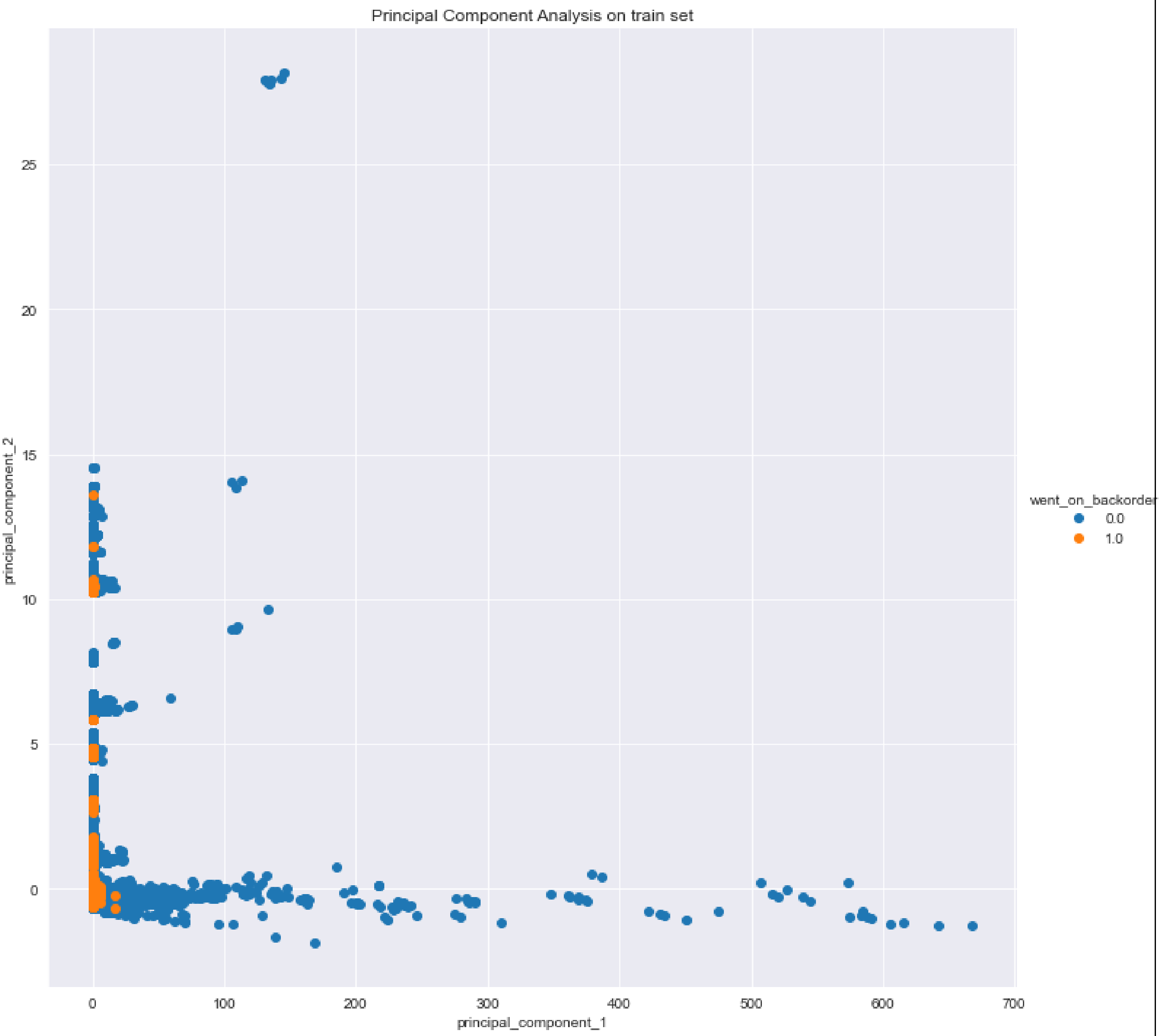


Observations:

From the above set of probability matrices for all the categorical features, it shows that most of these categorical features have a very high probability of having a negative flag when the product did not go into backorder. Therefore, it says that when a product does not go into backorder, most of the general risk flag are negative.

Dimensionality Reduction:

Principal Component Analysis:



Observations:

Dimensionality reduction techniques is uesd, in this case Principal Component Analysis to capture the essence of the data. From the above plot, it shows that most of the datapoints lie alongside 0. This deduction is true because many features with mostly 0 values in our EDA. There are outliers in the data but those datapoints does not have to be outlier per se. Furthermore, these potential outliers are more of the negative class compared to the positive class. And, for the positive class, almost all of the datapoints lie alongside 0.

Chapter – VIII

FEATURE ENGINEERING

Observations:

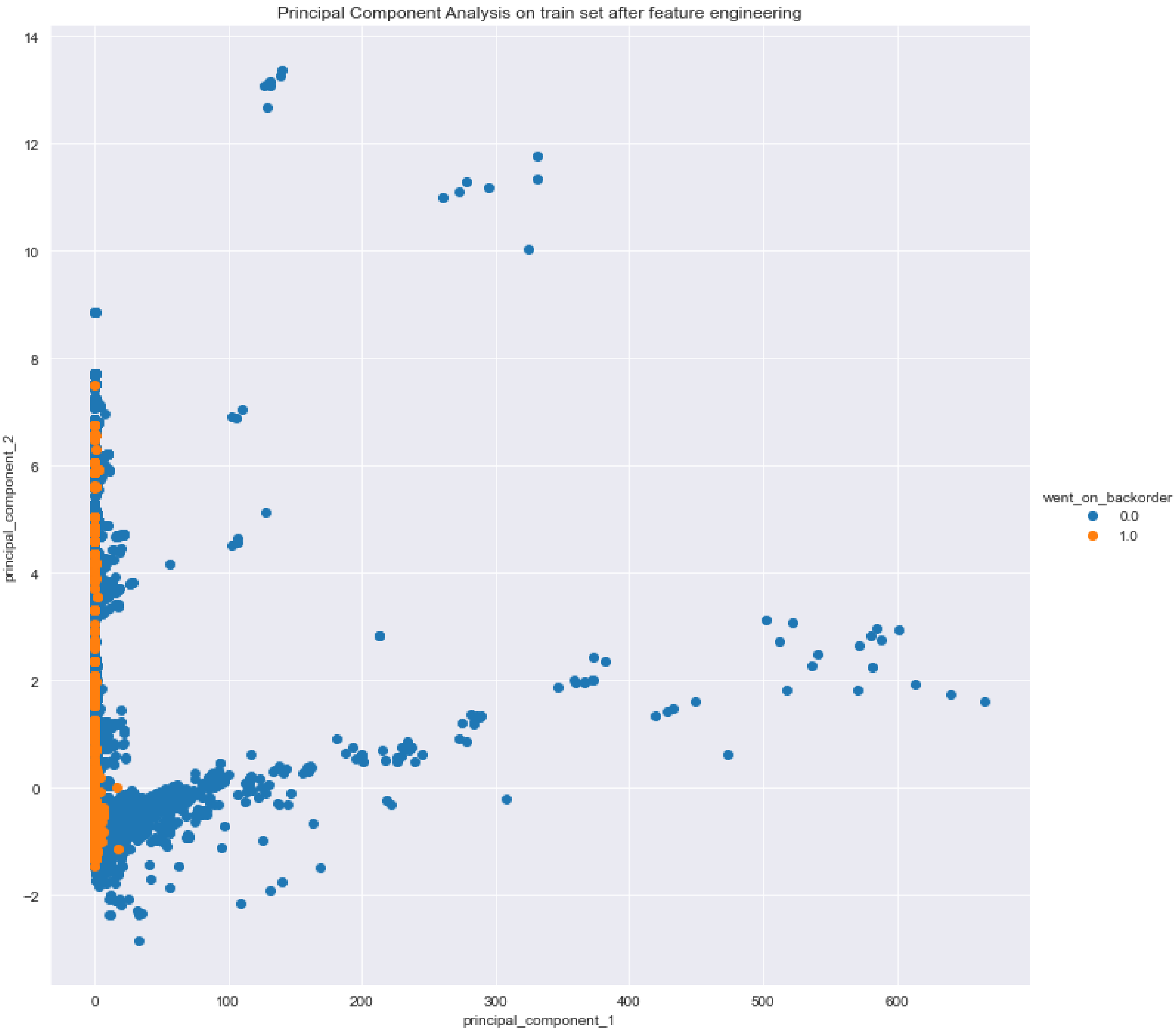
Mean imputations for the feature lead_time have been performed. Furthermore, it showed that the feature pieces_past_due and local_bo_quantity has more than 95% of values as 0. Therefore, as a feature engineering process, add another feature which shows if each datapoint in the two features is zero or non-zero.

Two new features have been added which show us if the datapoint in pieces_past_due and local_bo_quantity is a zero value or a non-zero value respectively. For further feature engineering, impute the zero values in all categorical features with the respective probability values from the probability matrices we calculated above.

Now perform the same preprocessing and feature engineering steps for the test dataset. Make sure the all the values imputed the test set are calculated from the train set to ensure there is no data leakage.

The final dataset which will be used to build a machine learning model, where the column **'went_on_backorder'** is our target label.

Principal Component Analysis on train after Feature Engineering:



Observations:

It shows some separation and also overlap between the positive class and the negative class. This means that the model we build should be able to fairly distinguish between a product that went backorder versus a product that did not go into backorder.

Chapter – IX

MODEL DEVELOPMENT AND SELECTION

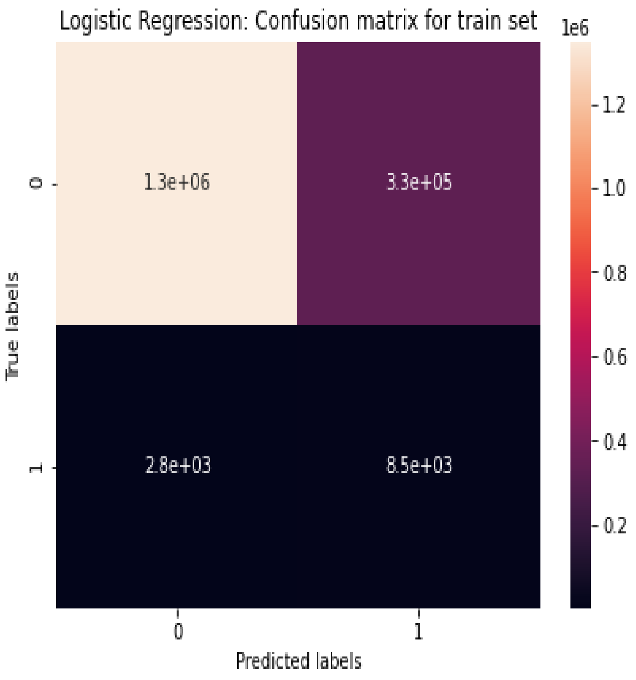
.

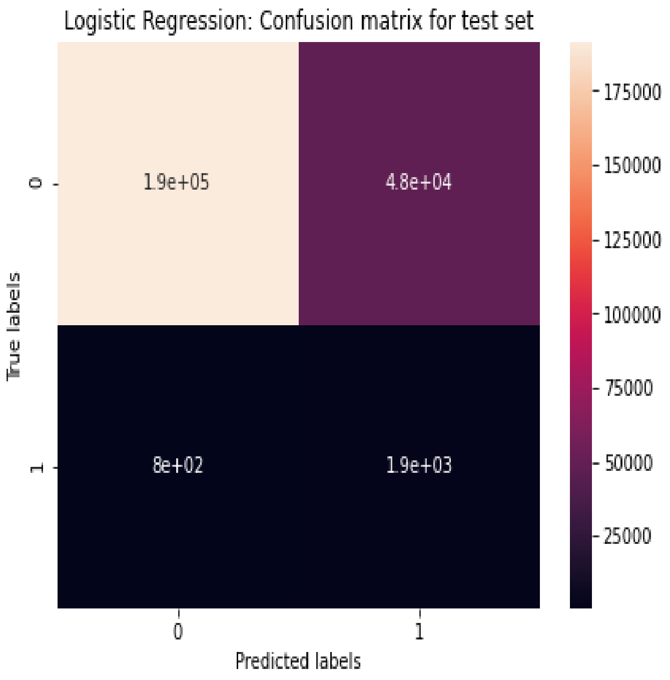
After preprocessing and feature engineering, comes model building. Here, different machine learning models like Logistic Regression, Decision Trees, Random Forests and more have been tested. All the results are documents and plotted wherever possible.

I. Logistic Regression:

The accuracy score of the logistic regression model on train set is: 0.8023408339554229

The accuracy score of the logistic regression model on test set is: 0.798430238562429





The precision score of the best logistic regression model on train set is: 0.5114060281441166

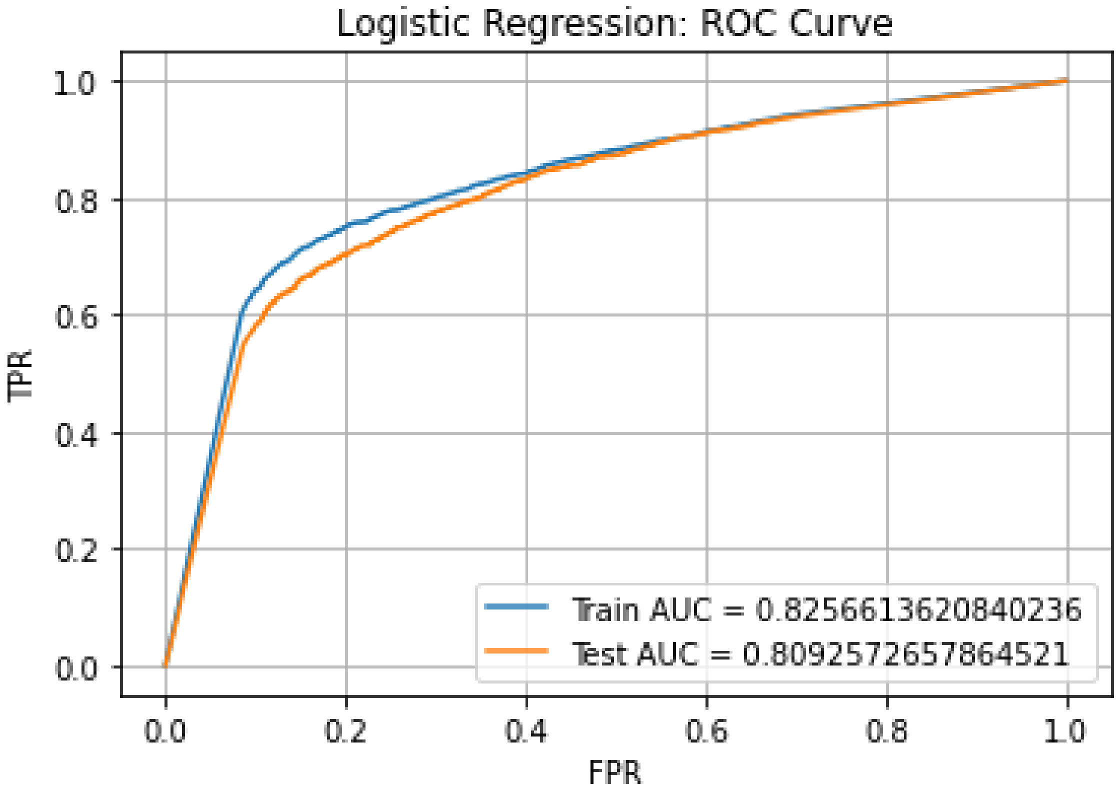
The precision score of the best logistic regression model on test set is: 0.5168904036883744

The recall score of the best logistic regression model on train set is: 0.7756100808401734

The recall score of the best logistic regression model on test set is: 0.7516805417637151

The AUC score of the best logistic regression model on train set is: 0.7756100808401734

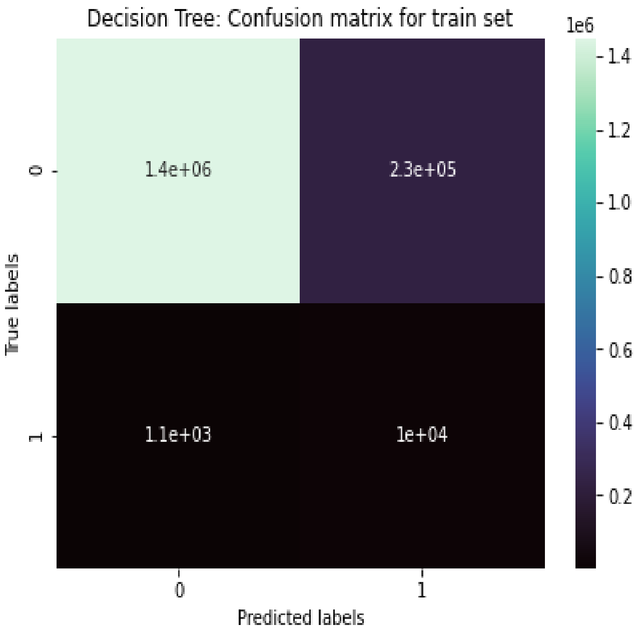
The AUC score of the best logistic regression model on test set is: 0.7516805417637152

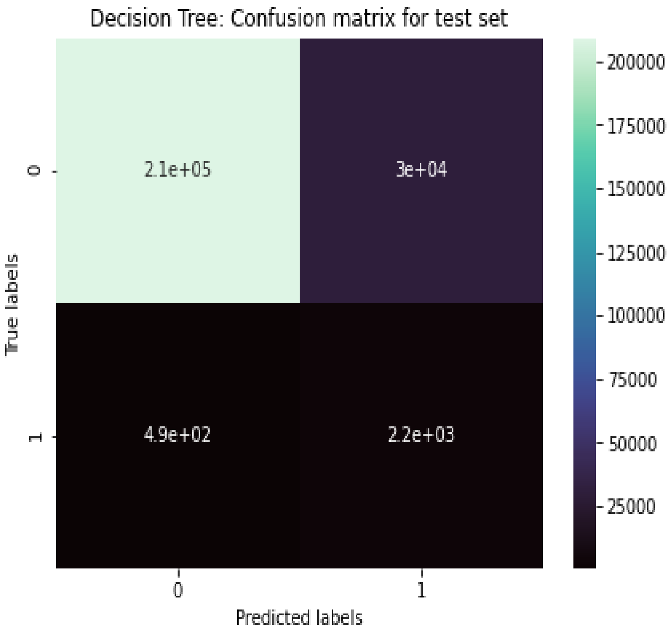


II. Decision Tree:

The accuracy score of the decision tree model on train set is: 0.8645657815221641

The accuracy score of the decision tree model on test set is: 0.8729980377982031





The precision score of the best decision tree model on train set is: 0.5211217424892802

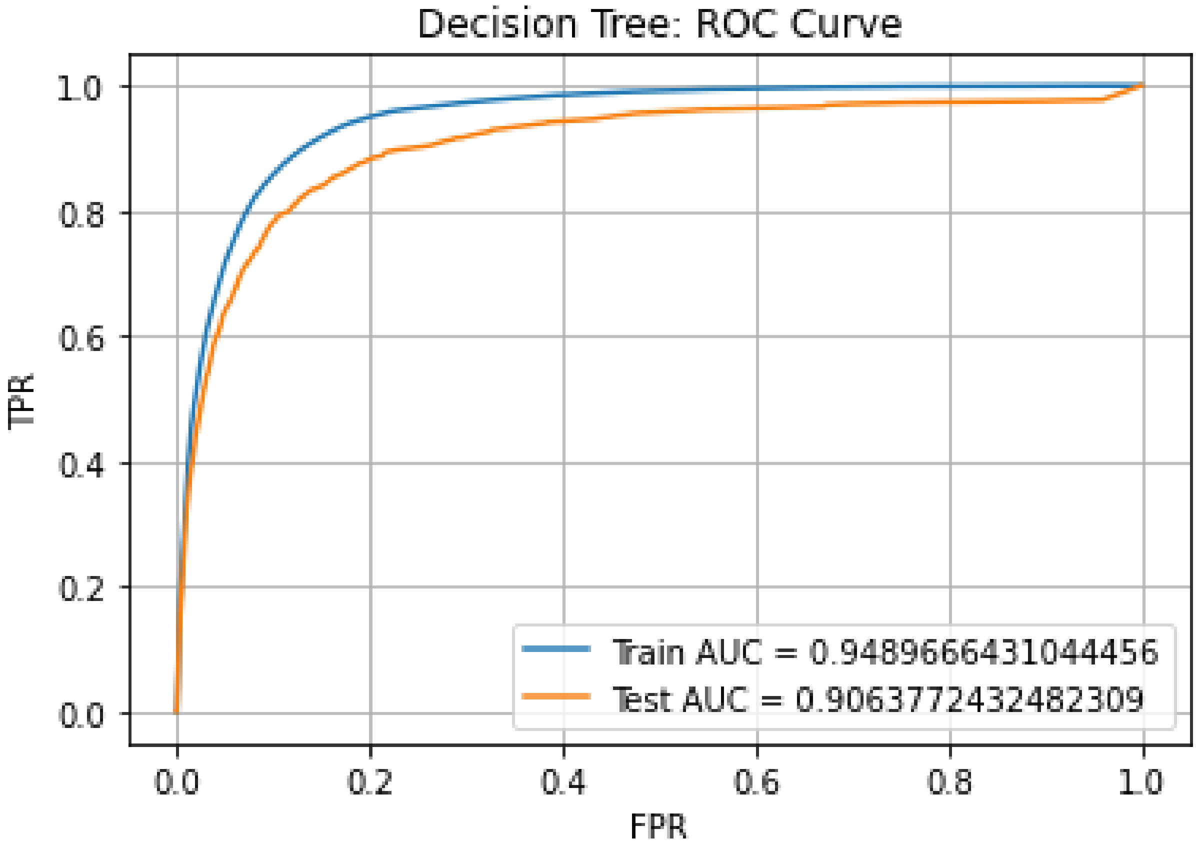
The precision score of the best decision tree model on test set is: 0.5327123298788503

The recall score of the best decision tree model on train set is: 0.8845954794088201

The recall score of the best decision tree model on test set is: 0.8458475239995429

The AUC score the best decision tree model on train set is: 0.8845954794088201

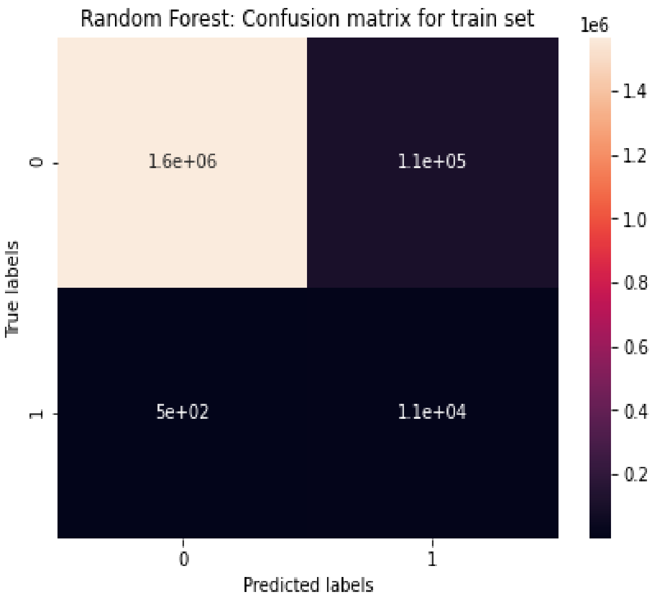
The AUC score the best decision tree model on test set is: 0.8458475239995429

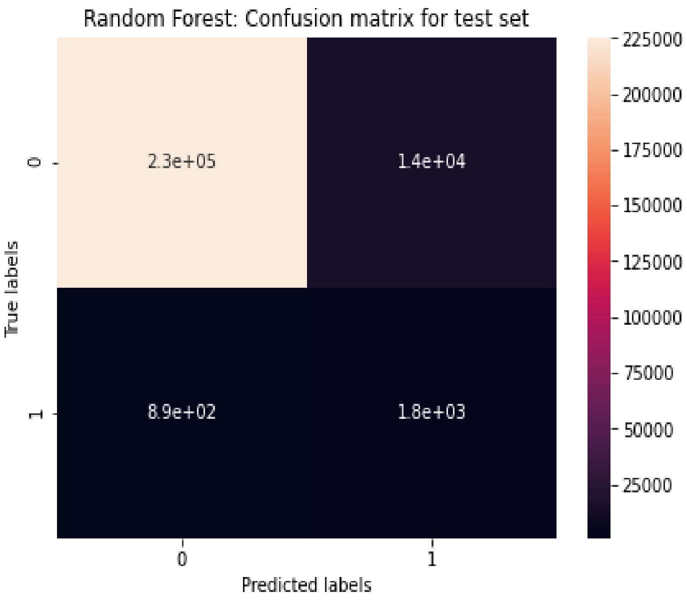


III. Random Forest:

The accuracy score of the random forest model on train set is: 0.934951950991196

The accuracy score of the random forest model on test set is: 0.9381927088712176





The precision score of the best random forest model on train set is: 0.5447832738107107

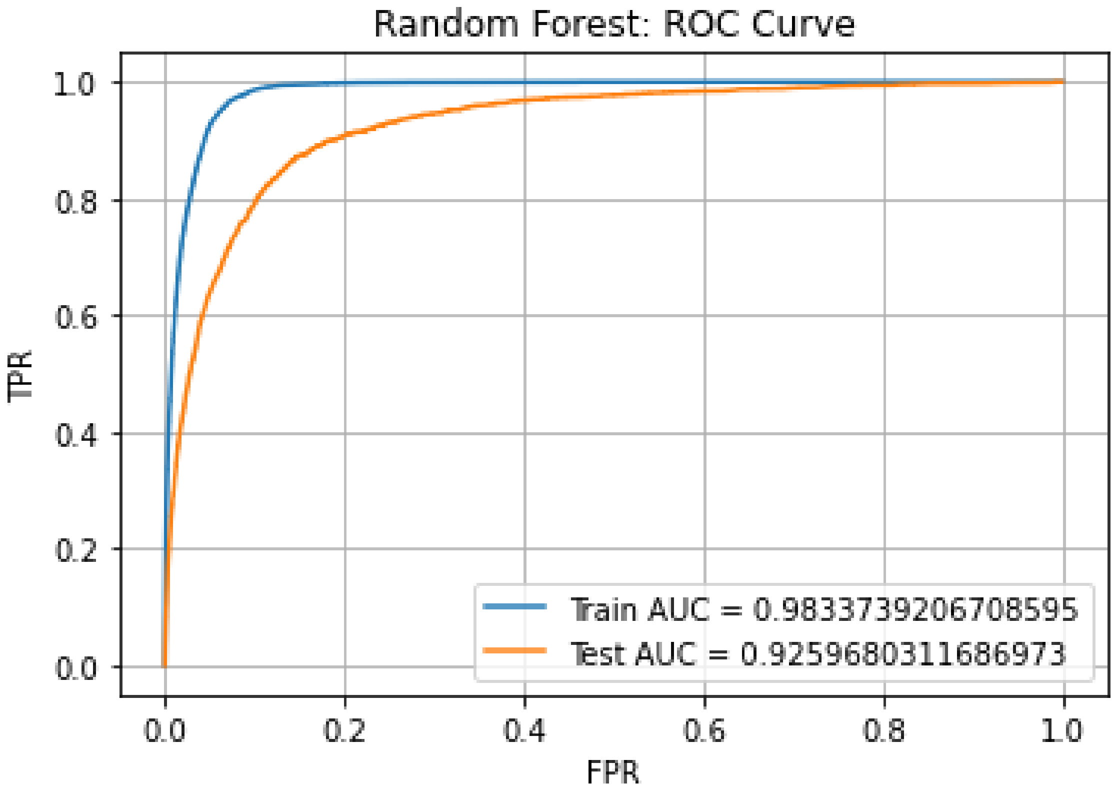
The precision score of the best random forest model on test set is: 0.5547867121834514

The recall score of the best random forest model on train set is: 0.9453123844098146

The recall score of the best random forest model on test set is: 0.8057933596440869

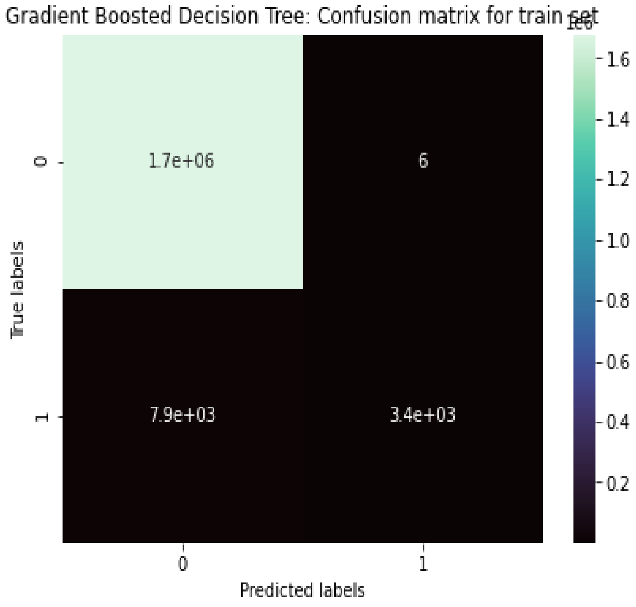
The AUC score of the best random forest model on train set is: 0.9453123844098146

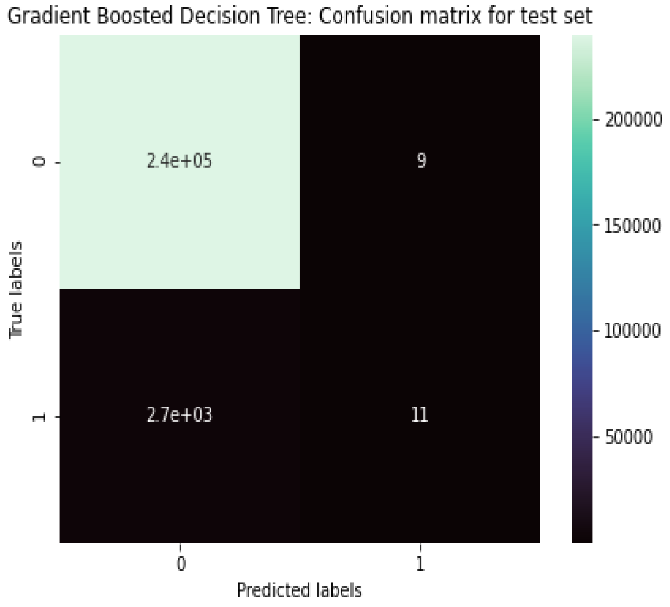
The AUC score of the best random forest model on test set is: 0.8057933596440869



IV. Gradient Boosted Decision Tree:

The accuracy score of the gradient boosted decision tree model on train set is: 0.9953218868863531





The accuracy score of the gradient boosted decision tree model on test set is: 0.988904265207064

The precision score of the best gradient boosted decision tree model on train set is: 0.9967779671777308

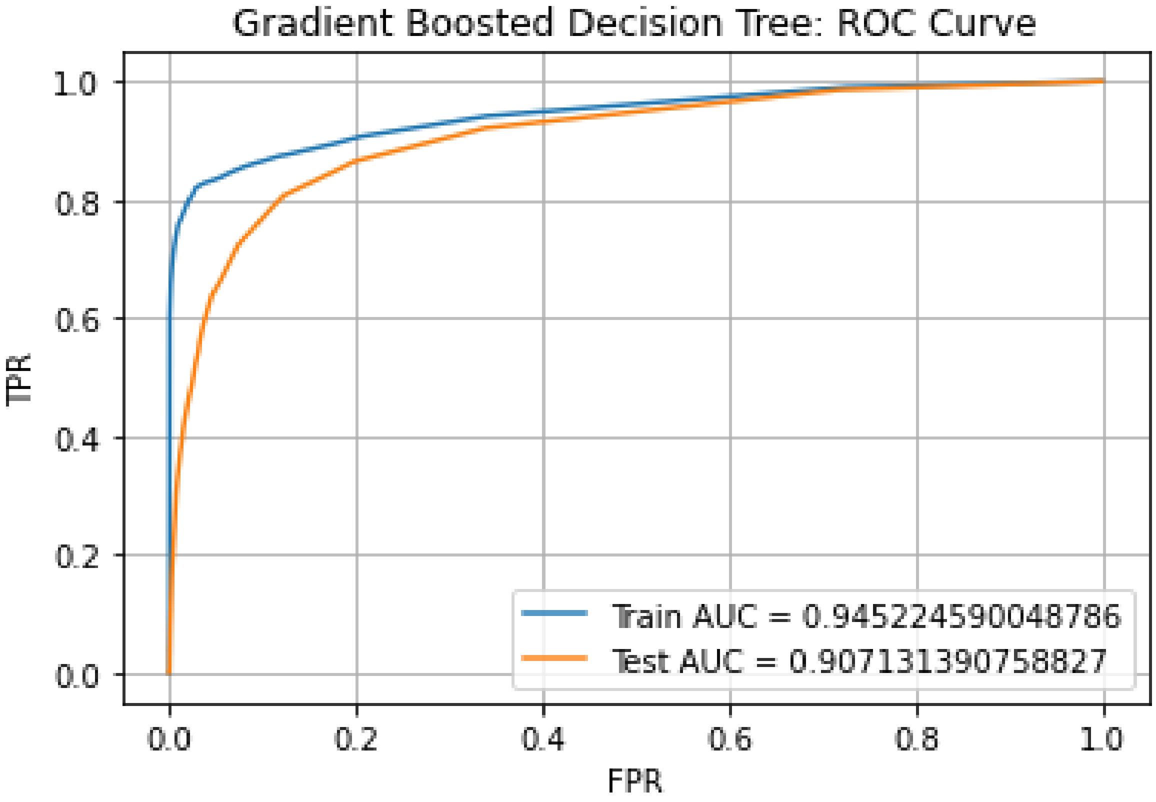
The precision score of the best gradient boosted decision tree model on test set is: 0.7694702650224123

The recall score of the best gradient boosted decision tree model on train set is: 0.6506667663717156

The recall score of the best gradient boosted decision tree model on test set is: 0.502027332939122

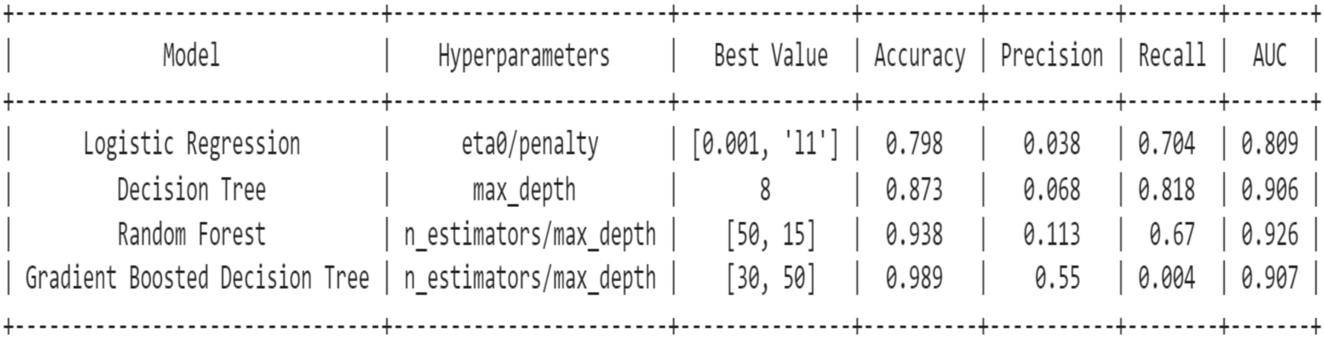
The AUC score of the best gradient boosted decision tree model on train set is: 0.6506667663717157

The AUC score of the best gradient boosted decision tree model on test set is: 0.502027332939122



Saving best model:

4 models have been built for this project i.e., Logistic Regression, Decision Tree, Random Forest and Gradient Boosted Decision Tree. Among them it says that Random Forest and Gradient Boosted Decision Tree are the best performing models. Both of them are giving a perfect score on all the metrics. Therefore, finalize either of the models as the best model. For this project, Random Forest is the best model.

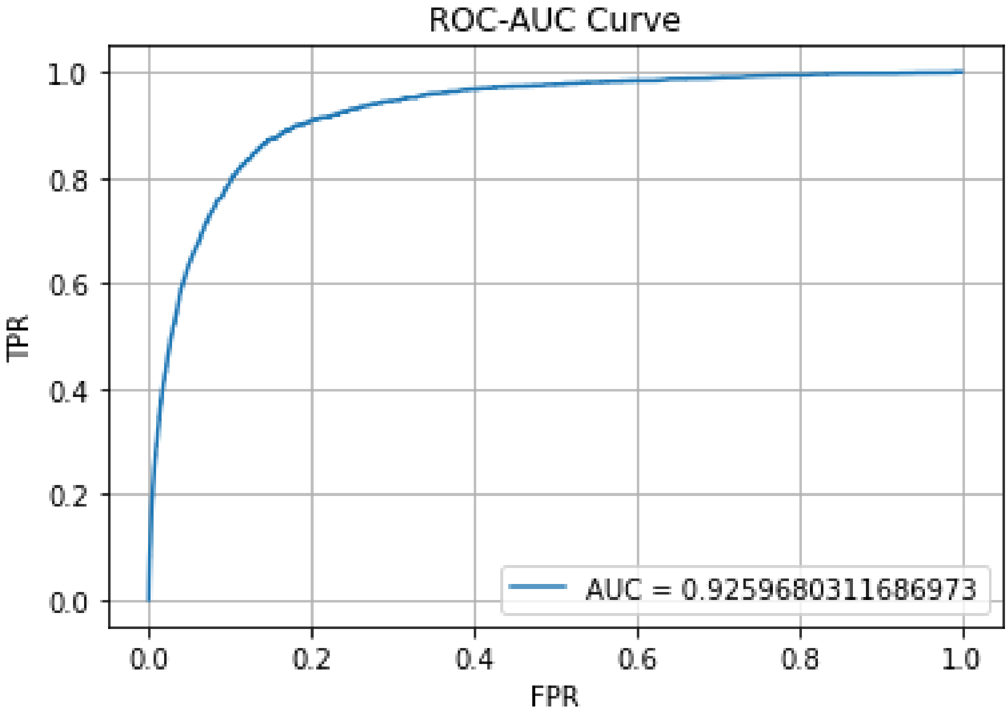


```
+-------------------------------+----------------------+--------------+----------+-----------+--------+-------+
|             Model             |   Hyperparameters    |  Best Value  | Accuracy | Precision | Recall |  AUC  |
+-------------------------------+----------------------+--------------+----------+-----------+--------+-------+
|      Logistic Regression      |     eta0/penalty     | [0.001, 'l1'] |  0.798   |   0.038   | 0.704  | 0.809 |
|         Decision Tree         |      max_depth       |      8       |  0.873   |   0.068   | 0.818  | 0.906 |
|         Random Forest         | n_estimators/max_depth |   [50, 15]   |  0.938   |   0.113   |  0.67  | 0.926 |
| Gradient Boosted Decision Tree | n_estimators/max_depth |   [30, 50]   |  0.989   |    0.55   | 0.004  | 0.907 |
+-------------------------------+----------------------+--------------+----------+-----------+--------+-------+
```

It shows that tree-based models perform way better than linear models. Ensemble models like Random Forest and Gradient Boosted Decision Trees performed the best. It shows that the best model is **Random Forest model** with an AUC of **0.926**

Accuracy: 0.9381927088712176

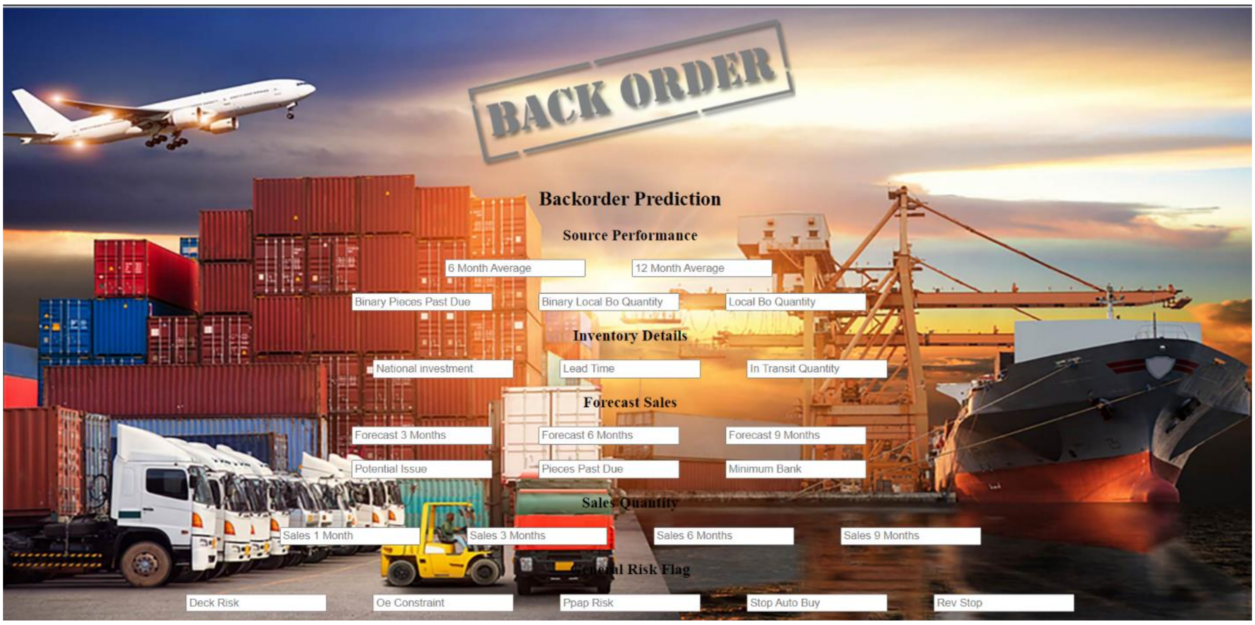
AUC: 0.9259680311686973

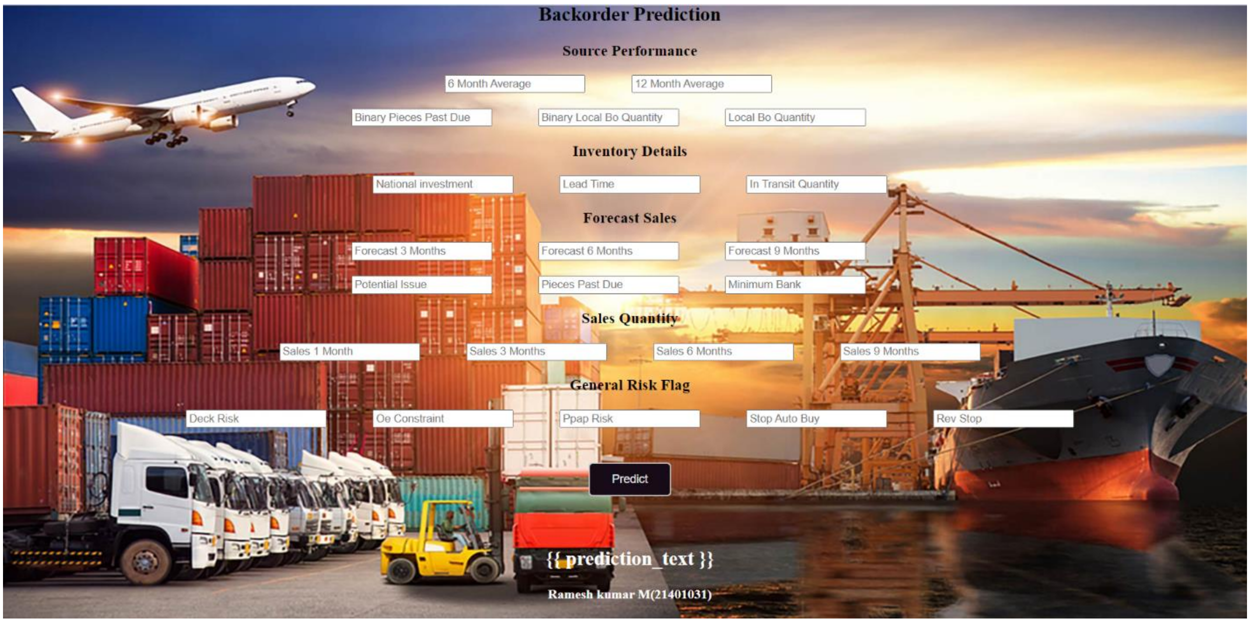


Chapter – X

Model Deployment

Link : Backorder prediction





Chapter – XI

CONCLUSION

Conclusion:

The current project presented the results of machine learning classifiers application within a predictive system design for inventory control, in extension of inventory planning models customarily discussed in literature.

A company’s overall service level can be extent by adopting a system such as this.

Since the items which goes on backorder (positive class) are rare compared to items which does not (negative class), some particular methods and metrics are employed either in design, development, and evaluation of the models in the considered imbalanced class problem. RANDOM FOREST has shown the best AUC score, although GBOOST performed preferably when taking into account precision-recall curves, computational costs, and enhancement capability.

The major conclusions are as follows:

- Ensemble learning achieved greater scores than single classifiers, whilst sampling techniques usage did not generate benefits except when combined with ensembles;
- The proposed predictive machine exhibited high-potential of increasing service level in real inventory management systems;
- Blagging, a combination of under-sampling and tree ensemble, has shown more likelihood of being adopted in practice application considered its precision-recall curve and potential of enhancement.

Chapter – XII

REFERENCES

REFERENCES:

- C. Tsou, “Multi-objective inventory planning using MOPSO and TOP-SIS,” Expert Systems with Applications, vol. 35, pp. 136-142, 2008.

- V. Lopez, A. Fernandez, S. Garcia, V. Palade, F. Herrera, “An insight into classification with imbalanced data: Empirical results and currenttrends on using data intrinsic characteristics,” Information Sciences, vol. 250, pp.113-141, 2013.

- M. Galar, A. Fernandez, E. Barrenechea, H. Bustince and F. Herrera, “A review on ensembles for class imbalance problem: bagging-, boosting- , and hybrid-based approaches,” IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews, vol. 42, no. 4, Jul. 2012.

- H. He and E. A. Garcia, “Learning from imbalanced data,” IEEE Transactions on Knowledgment and Data Engineering, vol. 21, no. 9, pp.1263-1284, Sep. 2009.

- F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, M. and E. Duchesnay, “Scikit-learn: machine learning in python,” Journal of Machine Learning Research, vol. 12, pp. 2825-2830, 2011.

- D. W. Hosmer Jr., S. Lemeshow and R. X. Sturdivant, “Applied logistic regression,” John Wiley & Sons, vol. 398, 2013.

- L. Breiman, J. Friedman, R. Olshen, and C. Stone, “Classification and regression trees,” Wadsworth, Belmont, CA, 1984.

- N. V. Chawla, K. W. Bowyer, L. O.Hall and W. P. Kegelmeyer, ”SMOTE: Synthetic minority over-sampling technique,” Journal of Artificial Intelligence Research, pp. 321-357, 2002.

- G. Lemaitre, F. Nogueira and C. K. Aridas, “Imbalanced-learn: a python toolbox to tackle the curse of imbalanced datasets in machine learning,” Journal of Machine Learning Research, vol.18, no. 17, pp. 1-5, 2017