University of Pittsburgh

PITT | SWANSON ENGINEERING

INDUSTRIAL ENGINEERING
SPRING 2021

DATA SCIENCE
PROF OLIVER HINDER HAMILTON

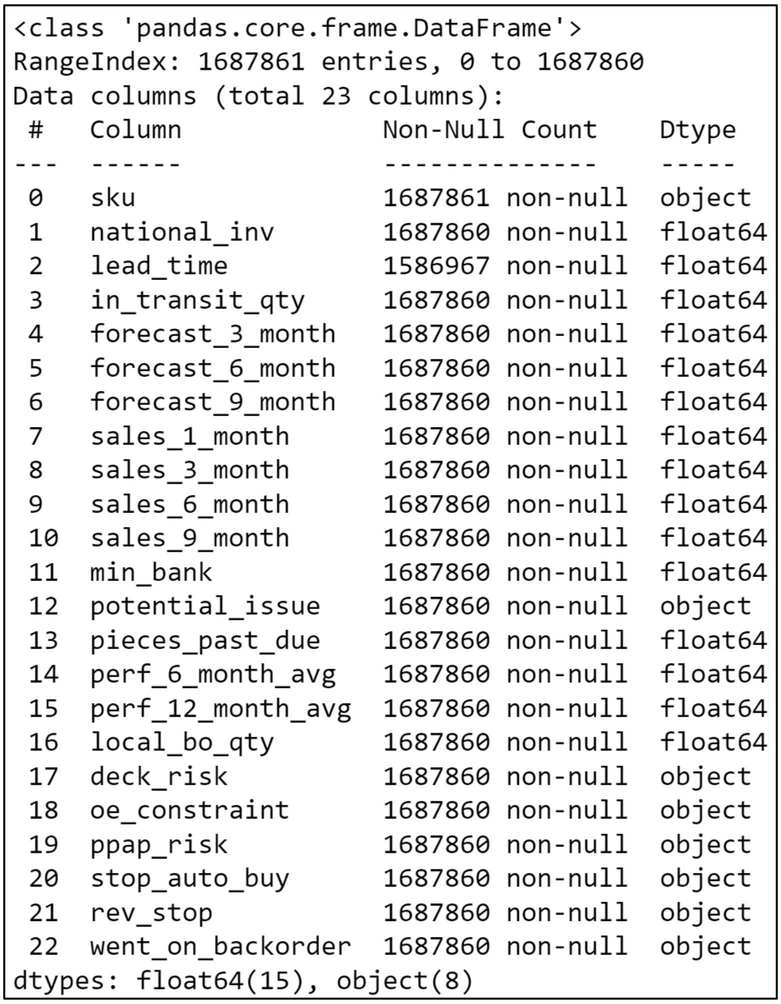
BACK-ORDERS PREDICTION
SUPPLY CHAIN

SUBMITTED BY
SABBELLA PRASANNA
4450873
prs98@pitt.edu

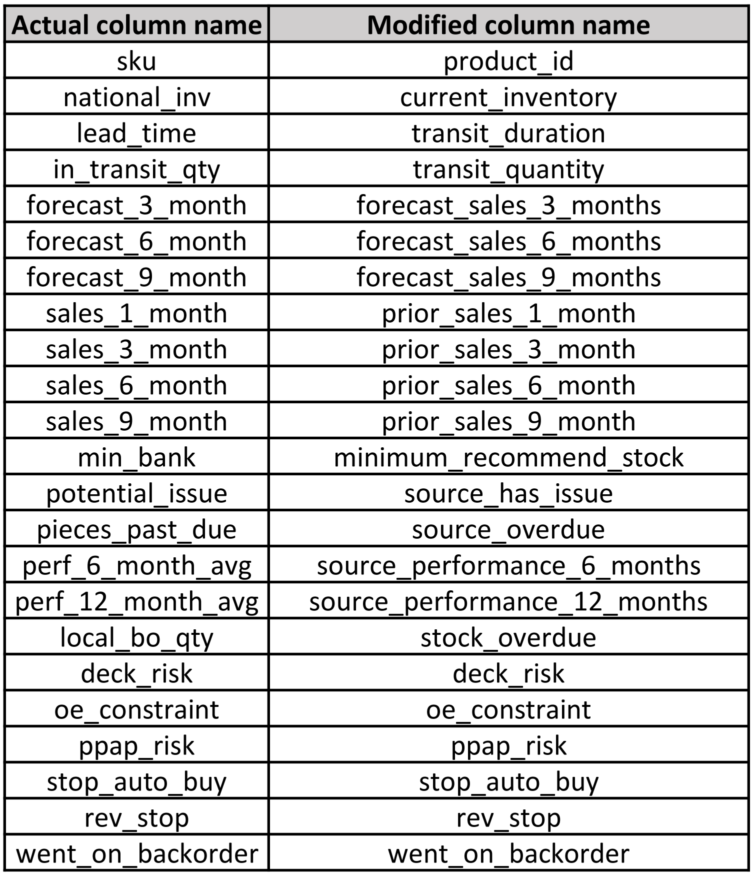
Data Wrangling

The data set used was uploaded during a Kaggle competition.
The data set consists of 23 different columns/features and 1687861 rows/observations.

- Sku (Discrete numeric): Product ID
- national_inv (Discrete numeric): Current inventory level for the part
- lead_time (Continuous numeric ~ Discrete numeric): Transit time for product (if available)
- in_transit_qty (Discrete numeric): Amount of product in transit from source
- forecast_3_month (Discrete numeric): Forecast sales for the next 3 months
- forecast_6_month (Discrete numeric): Forecast sales for the next 6 months
- forecast_9_month (Discrete numeric): Forecast sales for the next 9 months
- sales_1_month (Discrete numeric): Sales quantity during the last 1-month time period
- sales_3_month (Discrete numeric): Sales quantity during the last 3-month time period
- sales_6_month (Discrete numeric): Sales quantity during the last 6-month time period
- sales_9_month (Discrete numeric): Sales quantity during the last 9-month time period
- min_bank (Discrete numeric): Minimum recommend amount to stock
- potential_issue (Binary): Source issue identified for part
- pieces_past_due (Discrete numeric): Parts overdue from source
- perf_6_month_avg (Discrete numeric): Average source performance for the last 6-month period
- perf_12_month_avg (Discrete numeric): Average source performance for the last 12-month period
- local_bo_qty (Discrete numeric): Amount of stock orders overdue
- deck_risk (Binary): - Part risk flag
- oe_constraint (Binary): Part risk flag
- ppap_risk (Binary): Part risk flag
- stop_auto_buy (Binary): Part risk flag
- rev_stop (Binary): Part risk flag
- went_on_backorder (Binary): - Product went on backorder. This is the target value.

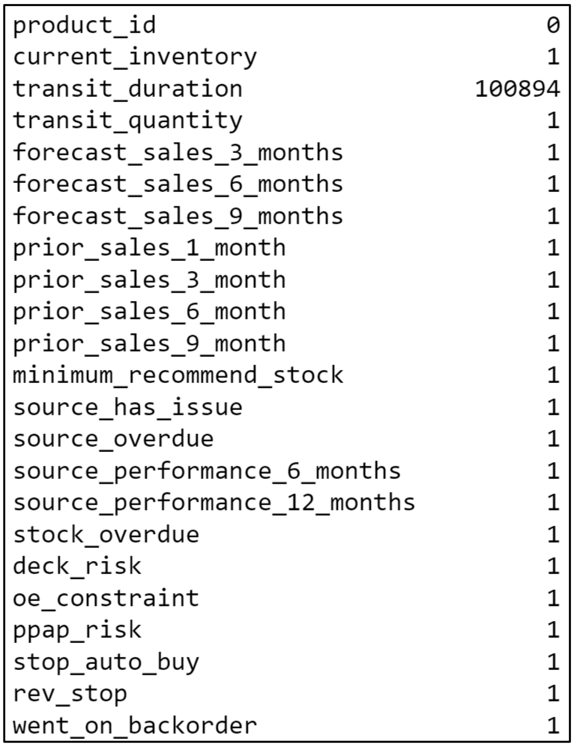


```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1687861 entries, 0 to 1687860
Data columns (total 23 columns):
 #   Column             Non-Null Count    Dtype
---  ------             --------------    -----
 0   sku                1687861 non-null  object
 1   national_inv       1687860 non-null  float64
 2   lead_time          1586967 non-null  float64
 3   in_transit_qty     1687860 non-null  float64
 4   forecast_3_month   1687860 non-null  float64
 5   forecast_6_month   1687860 non-null  float64
 6   forecast_9_month   1687860 non-null  float64
 7   sales_1_month      1687860 non-null  float64
 8   sales_3_month      1687860 non-null  float64
 9   sales_6_month      1687860 non-null  float64
 10  sales_9_month      1687860 non-null  float64
 11  min_bank           1687860 non-null  float64
 12  potential_issue    1687860 non-null  object
 13  pieces_past_due    1687860 non-null  float64
 14  perf_6_month_avg   1687860 non-null  float64
 15  perf_12_month_avg  1687860 non-null  float64
 16  local_bo_qty       1687860 non-null  float64
 17  deck_risk          1687860 non-null  object
 18  oe_constraint      1687860 non-null  object
 19  ppap_risk          1687860 non-null  object
 20  stop_auto_buy      1687860 non-null  object
 21  rev_stop           1687860 non-null  object
 22  went_on_backorder  1687860 non-null  object
dtypes: float64(15), object(8)
```



Actual column name	Modified column name
sku	product_id
national_inv	current_inventory
lead_time	transit_duration
in_transit_qty	transit_quantity
forecast_3_month	forecast_sales_3_months
forecast_6_month	forecast_sales_6_months
forecast_9_month	forecast_sales_9_months
sales_1_month	prior_sales_1_month
sales_3_month	prior_sales_3_month
sales_6_month	prior_sales_6_month
sales_9_month	prior_sales_9_month
min_bank	minimum_recommend_stock
potential_issue	source_has_issue
pieces_past_due	source_overdue
perf_6_month_avg	source_performance_6_months
perf_12_month_avg	source_performance_12_months
local_bo_qty	stock_overdue
deck_risk	deck_risk
oe_constraint	oe_constraint
ppap_risk	ppap_risk
stop_auto_buy	stop_auto_buy
rev_stop	rev_stop
went_on_backorder	went_on_backorder

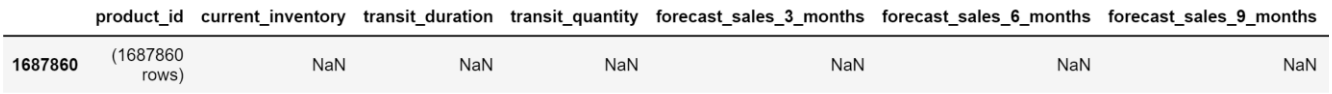
The column names are modified based on ease of use and understanding.



```
product_id                           0
current_inventory                    1
transit_duration                100894
transit_quantity                     1
forecast_sales_3_months              1
forecast_sales_6_months              1
forecast_sales_9_months              1
prior_sales_1_month                  1
prior_sales_3_month                  1
prior_sales_6_month                  1
prior_sales_9_month                  1
minimum_recommend_stock              1
source_has_issue                     1
source_overdue                       1
source_performance_6_months          1
source_performance_12_months         1
stock_overdue                        1
deck_risk                            1
oe_constraint                        1
ppap_risk                            1
stop_auto_buy                        1
rev_stop                             1
went_on_backorder                    1
```

Most of the columns have just one entry as null. Quite possible that all these NaN's belong to same row. If we remove this row, it will also solve the mix datatypes issue which we saw during data load.

There are nulls and lots of 0's in each column. During the EDA/Feature engineering step we can determine if we want to omit or keep this column. But for now, we will look for an approach to fill these missing values.



	product_id	current_inventory	transit_duration	transit_quantity	forecast_sales_3_months	forecast_sales_6_months	forecast_sales_9_months
1687860	(1687860 rows)	NaN	NaN	NaN	NaN	NaN	NaN

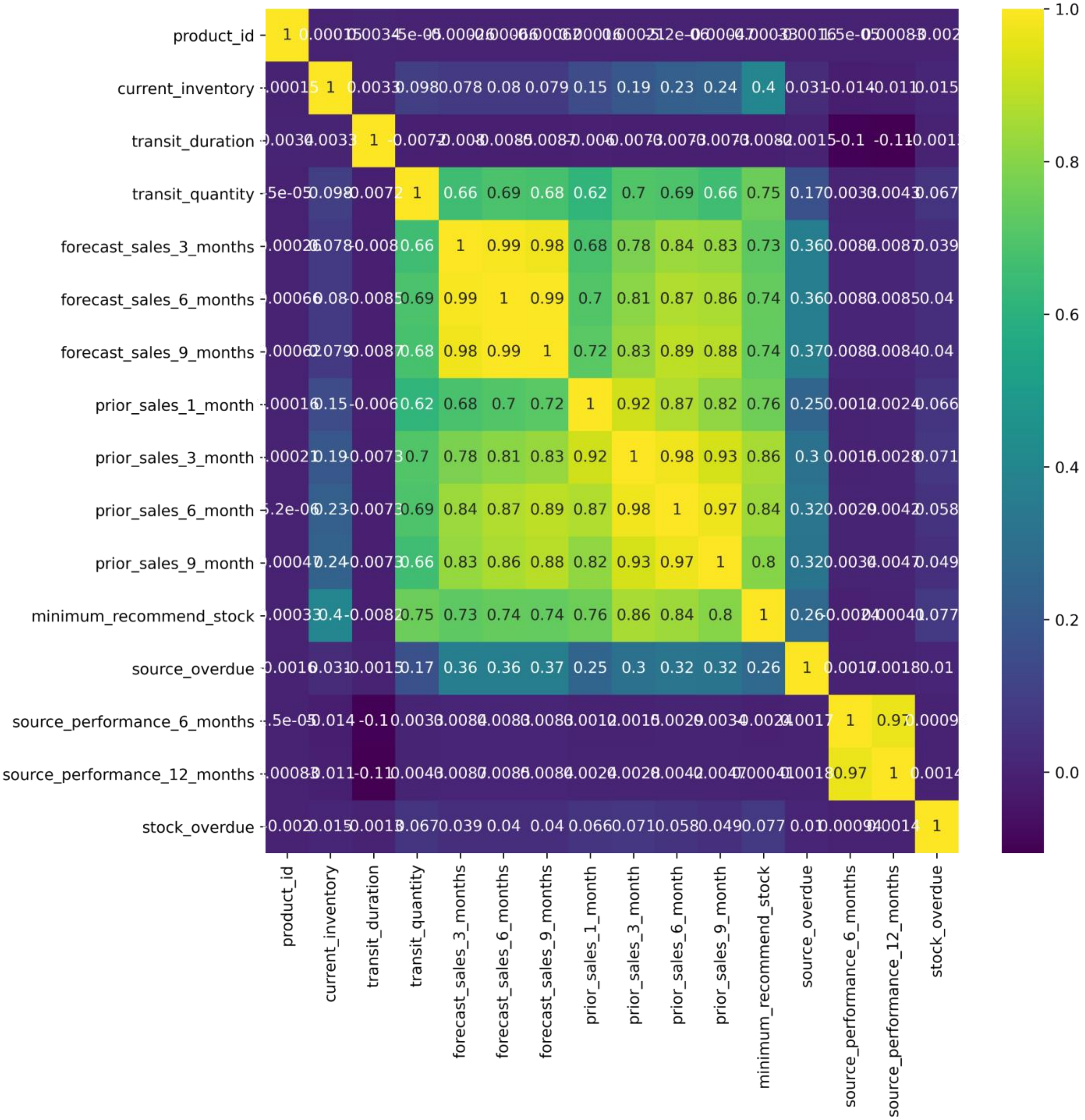
1 rows × 23 columns

Cont.

As expected, the observation with product ID 1687860 holds the null values in every column. This row has been eliminated.

Handling NAN and Missing values

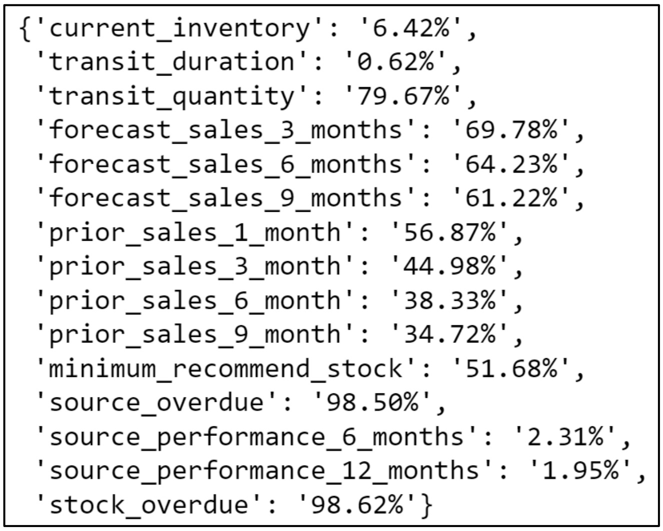
- "transit_duration" has 100893 null values which is 6% of the total data.
- Missing values in columns source_performance_6_months and source_performance_12_months are represented with '-99’.
- "source_performance_6_months" has 129478 null values which is 8% of the total data.
- "source_performance_12_months" has 122050 null values which is 8% of the total data.
- Heatmap shows a strong co-relation (0.97) between "source_performance_6_months" and "source_performance_12_months".



- Given that we do have cells with -99 in the sourec_performance_12_months and source_performance_6_months of Data, we need to think of a feature engineering technique appropriately to replace these values with most desirable one's.
- Heat map strongly indicates that there is strong correlation between these two. Hence, we can use the linear regression to estimate the missing values. However, there are quiet a lot of missing values, it is strongly encouraged to replace the missing values with the alternate approach. In this case, we are trying to replace the missing values with the median of their respective fields.

Handling the repetitive values

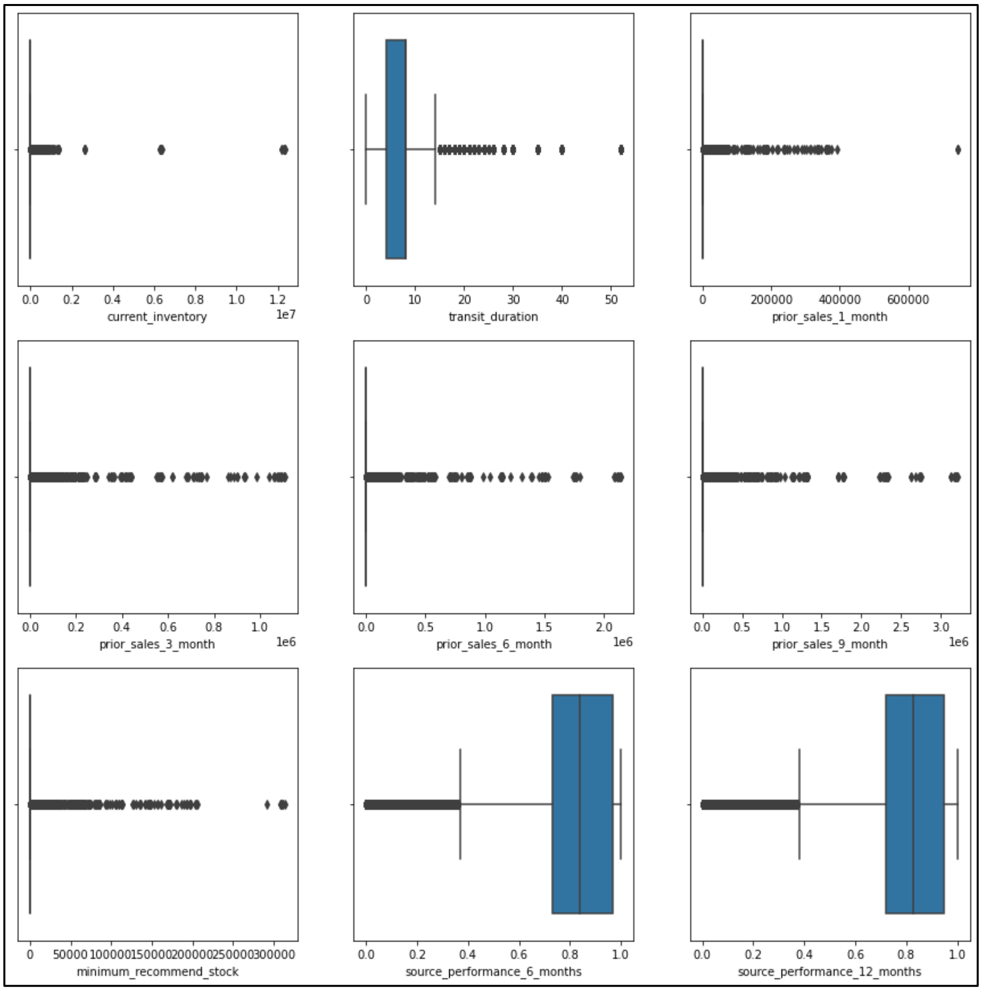
- It can be seen that there are 0's in our Data. Let us find out the percentage of repetitions of 0's. If any of the columns has more than the 60%, we can safely drop that column.
- 'transit_quantity', 'forecast_sales_3_months', 'forecast_sales_6_months','forecast_sales_9_months', 'source_overdue', 'stock_overdue’ were dropped.



```
{'current_inventory': '6.42%',
 'transit_duration': '0.62%',
 'transit_quantity': '79.67%',
 'forecast_sales_3_months': '69.78%',
 'forecast_sales_6_months': '64.23%',
 'forecast_sales_9_months': '61.22%',
 'prior_sales_1_month': '56.87%',
 'prior_sales_3_month': '44.98%',
 'prior_sales_6_month': '38.33%',
 'prior_sales_9_month': '34.72%',
 'minimum_recommend_stock': '51.68%',
 'source_overdue': '98.50%',
 'source_performance_6_months': '2.31%',
 'source_performance_12_months': '1.95%',
 'stock_overdue': '98.62%'}
```

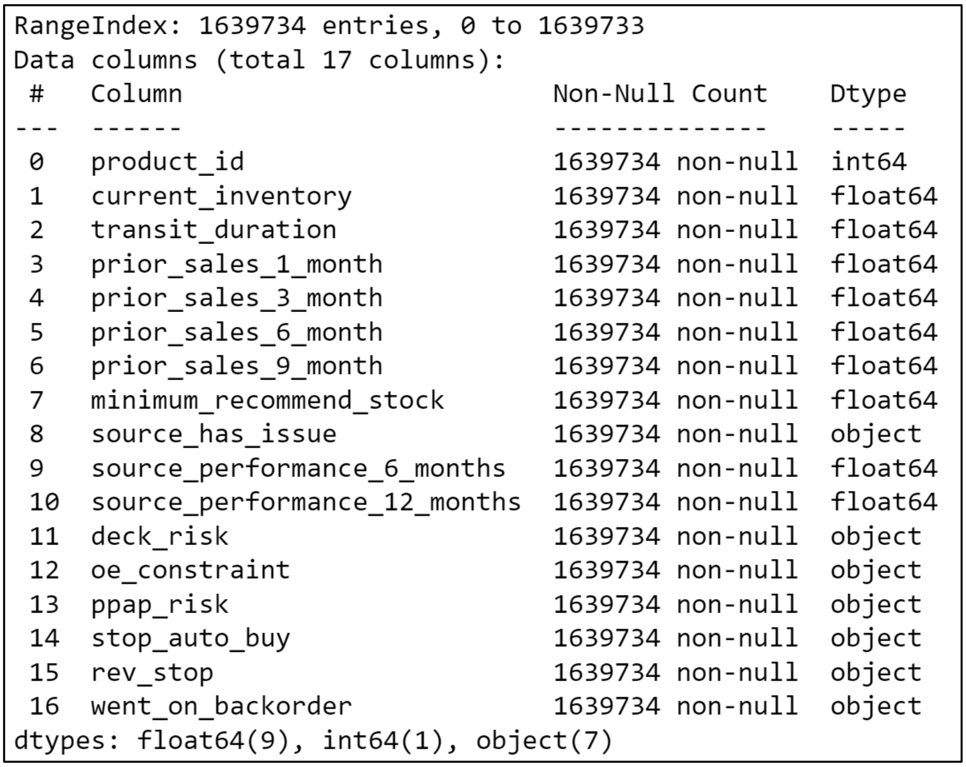
Handling the outliers

- Certain columns are identified with outliers that are way beyond the frequency of 90% of the values.
- These columns were identified and the entire row corresponding to those outliers were deleted.
- With these eliminations about 1.09 million rows remained in the data.



The final dataset

- The final dataset consists of 1639743 observations and 17 attributes.

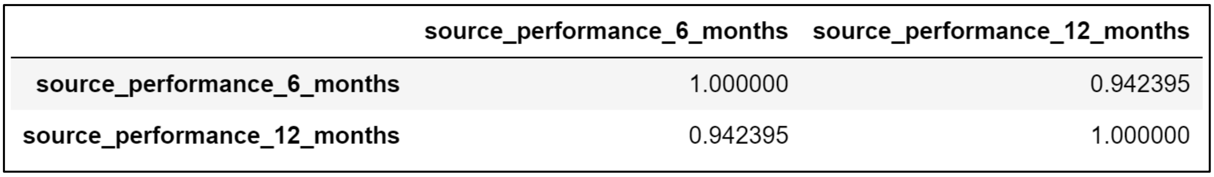


```
RangeIndex: 1639734 entries, 0 to 1639733
Data columns (total 17 columns):
 #   Column                        Non-Null Count    Dtype  
---  ------                        --------------    -----  
 0   product_id                    1639734 non-null  int64  
 1   current_inventory             1639734 non-null  float64
 2   transit_duration              1639734 non-null  float64
 3   prior_sales_1_month           1639734 non-null  float64
 4   prior_sales_3_month           1639734 non-null  float64
 5   prior_sales_6_month           1639734 non-null  float64
 6   prior_sales_9_month           1639734 non-null  float64
 7   minimum_recommend_stock       1639734 non-null  float64
 8   source_has_issue              1639734 non-null  object 
 9   source_performance_6_months   1639734 non-null  float64
 10  source_performance_12_months  1639734 non-null  float64
 11  deck_risk                     1639734 non-null  object 
 12  oe_constraint                 1639734 non-null  object 
 13  ppap_risk                     1639734 non-null  object 
 14  stop_auto_buy                 1639734 non-null  object 
 15  rev_stop                      1639734 non-null  object 
 16  went_on_backorder             1639734 non-null  object 
dtypes: float64(9), int64(1), object(7)
```

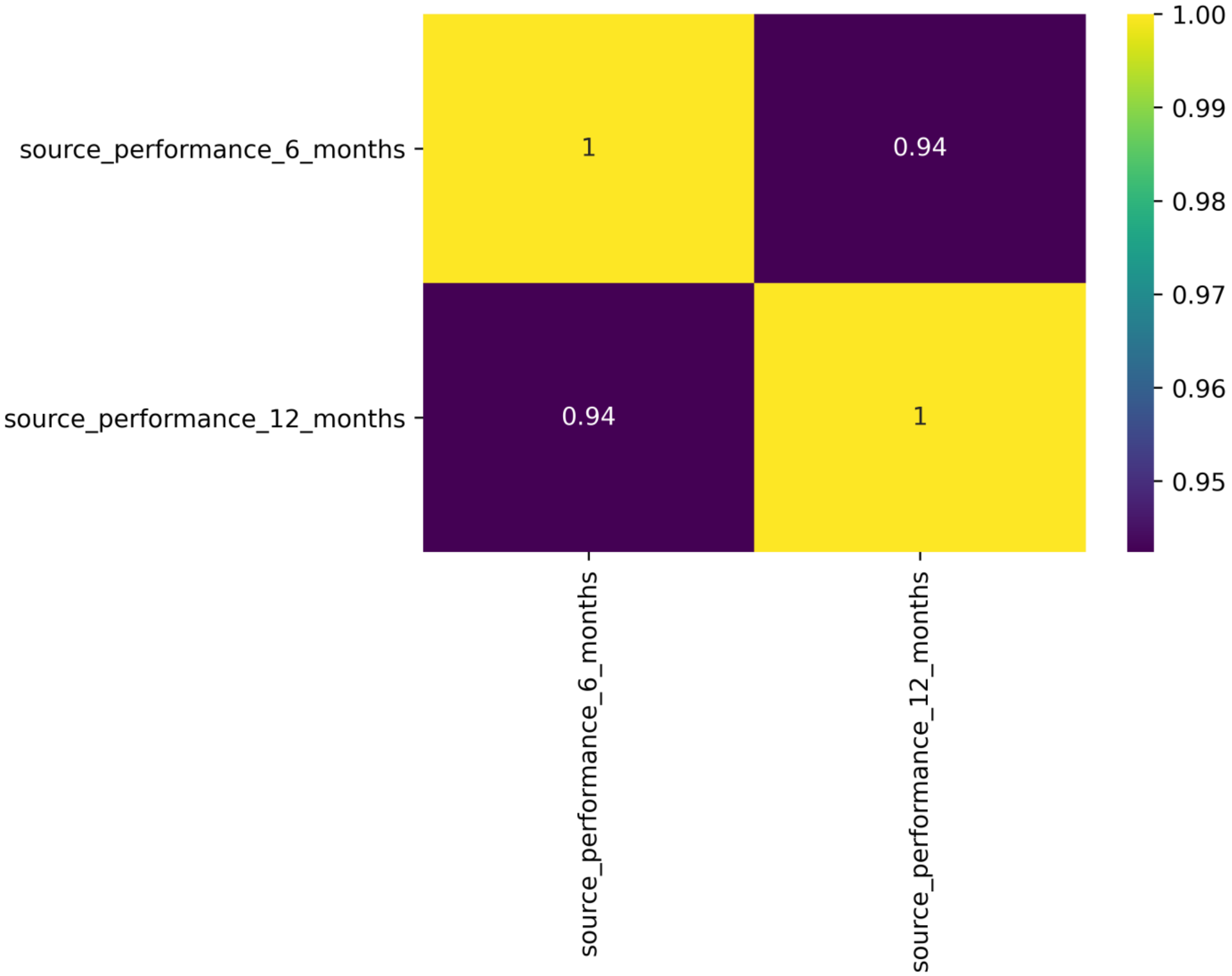
Exploratory Data Analysis

Correlations

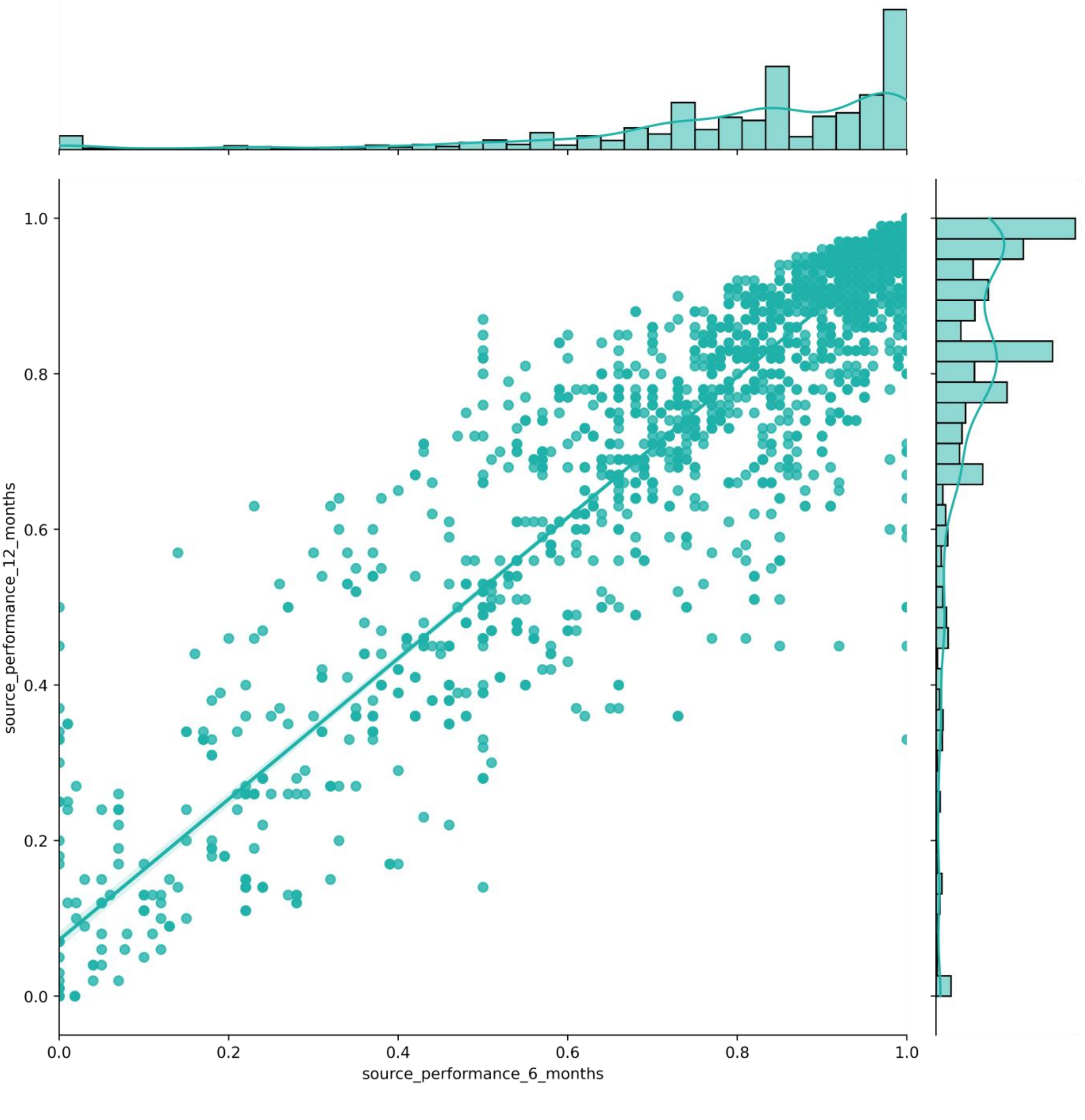
- Correlations have been plotted between various attributes as shown in the figure below.
- One among the highly correlated variables can be retained and the rest can be eliminated for the purposes saving computation time and memory.

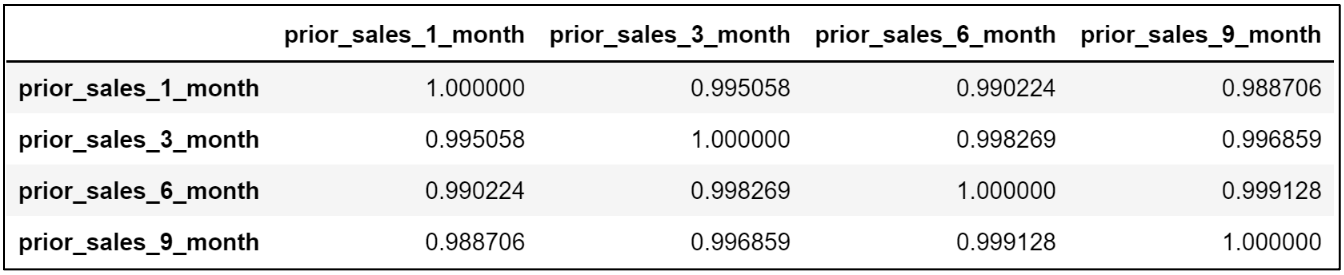


	source_performance_6_months	source_performance_12_months
source_performance_6_months	1.000000	0.942395
source_performance_12_months	0.942395	1.000000

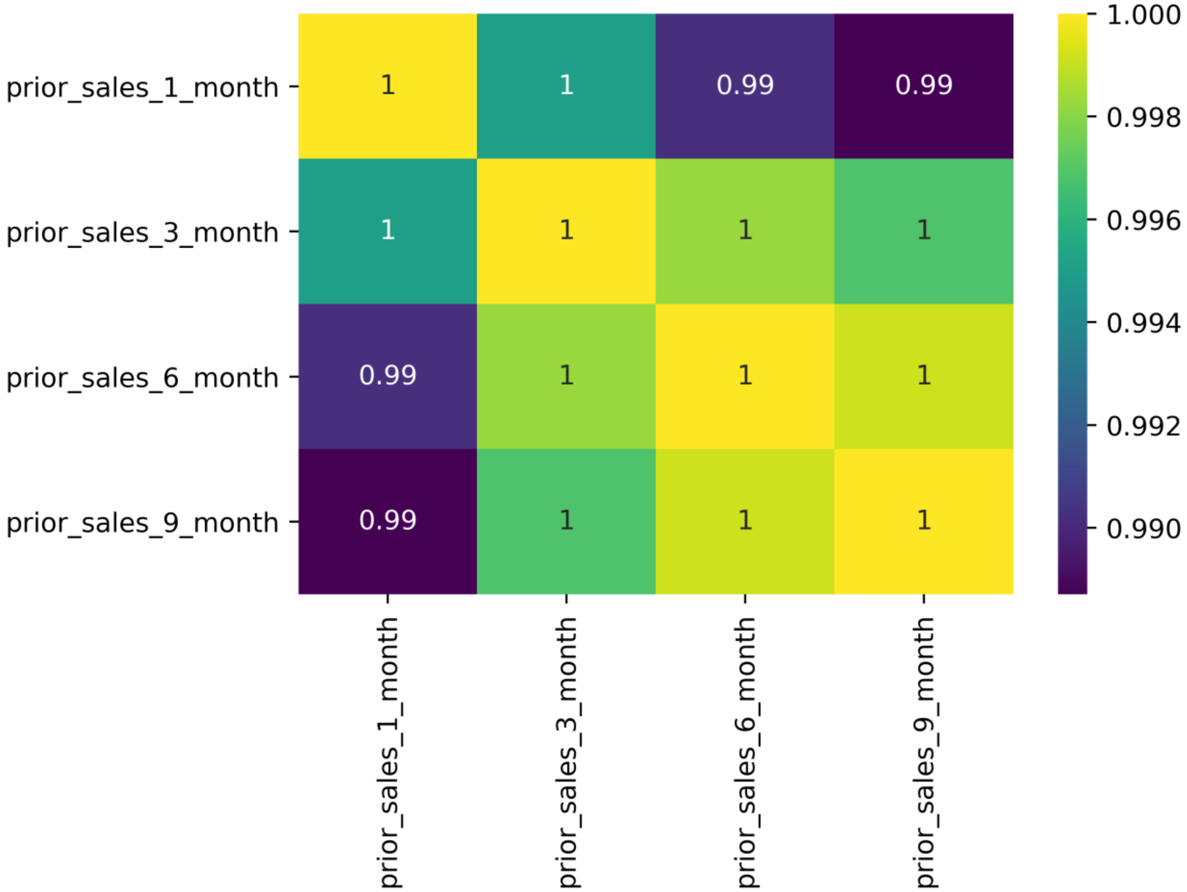


Scatter Plot between source_performance_6_months and source_performance_12_months with a linear regression curve

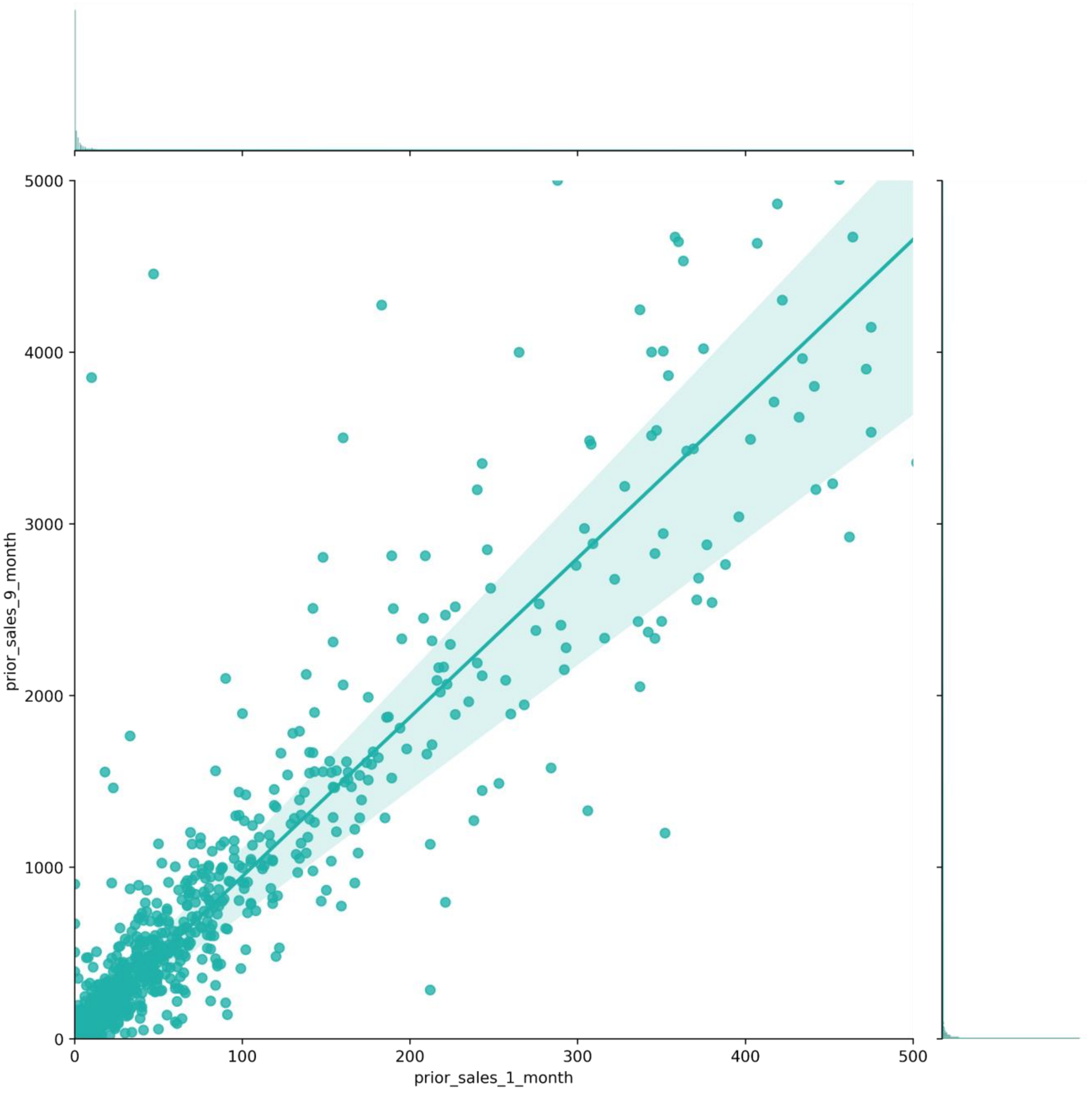




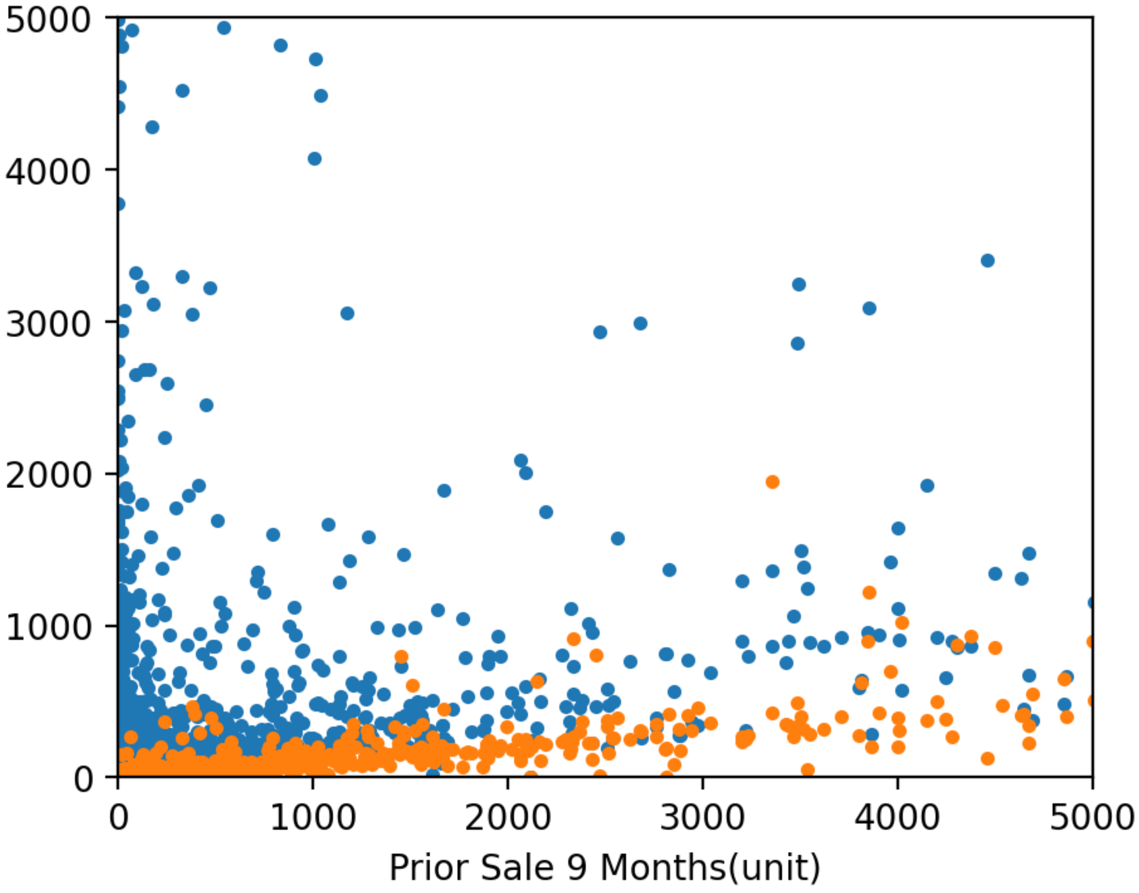
	prior_sales_1_month	prior_sales_3_month	prior_sales_6_month	prior_sales_9_month
prior_sales_1_month	1.000000	0.995058	0.990224	0.988706
prior_sales_3_month	0.995058	1.000000	0.998269	0.996859
prior_sales_6_month	0.990224	0.998269	1.000000	0.999128
prior_sales_9_month	0.988706	0.996859	0.999128	1.000000



Scatter Plot between prior_sales_1_month and prior_sales_9_month with a linear regression curve

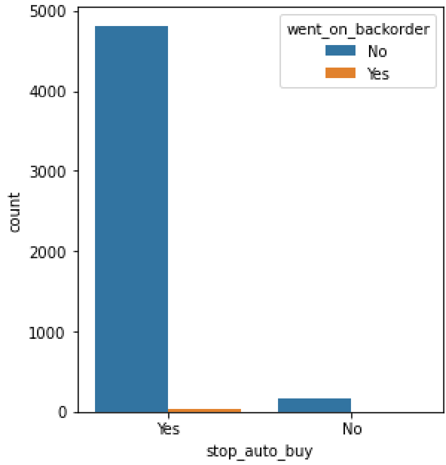


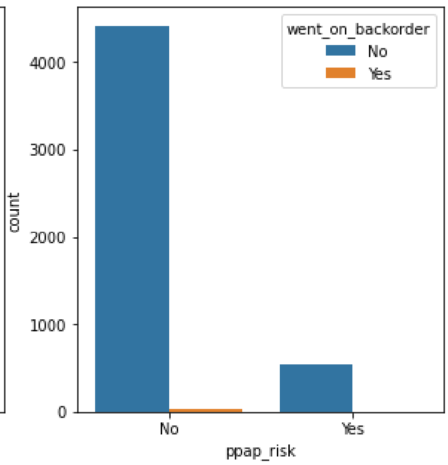
Scatter Plot between prior_sales_9_month vs Current Inventory in blue
Scatter Plot between prior_sales_9_month vs Minimum recommended/Inventory in orange

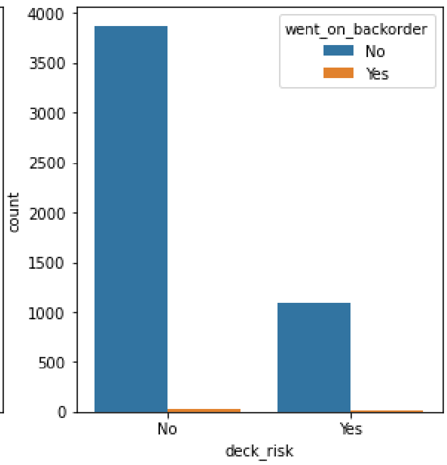


<u>**Here are a couple of observations that can be drawn:**</u>

- There are few sku's with the current inventory as high as 1.4 million even when there are no prior 9 months sales.
- Similarly, the recommended the stock for few sku's are high as 150k even when there are no prior 9 months sale for these products.
- This is an interesting observation that can be presented to the client.
- This could probably be a recording issue.





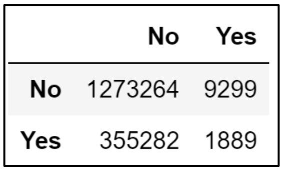


Realtionship with deck_risk, ppap_risk and stop_auto_buy with the outcome variable went_backorder

In order to further understand the relationship between the categorical variable with the outcome variable, we can start using the crosstabulation and chi-square test.

Ho ----> Feature are independent, no association between the variables exists
H1 ----> Feature are not independent; there is an association between the variable exists.

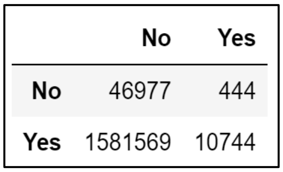
contingency_table: deck_risk, went_on_backorder



	No	Yes
No	1273264	9299
Yes	355282	1889

- Chi-Square Critical value: 3.841458820694124
- chi_deck_risk: 158.33558932833364
- p_val_deck_risk 2.6140480466303405e-36
- degree of freedom 1

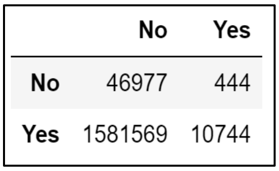
contingency_table: ppap_risk, went_on_backorder



	No	Yes
No	46977	444
Yes	1581569	10744

- Chi-Square Critical value: 3.841458820694124
- chi_ppap_risk: 139.72726876786075
- p_val_ppap_risk 3.0538954261451925e-32
- degree of freedom 1

contingency_table: stop_auto_buy, went_on_backorder



	No	Yes
No	46977	444
Yes	1581569	10744

- Chi-Square Critical value: 3.841458820694124
- chi_stop_auto_buy: 46.102446623300395
- p_val_stop_auto_buy 1.1222821596399783e-11
- degree of freedom 1

- **We used cross tabulation and chi-square to find the relation between target variable with other categorical variables. All the relations has p-values is less than 0.05 and we also have chi-square calculated value greater than the chi-square critical value.**

- **Based on these two evidence we can reject the null hypothesis and can go with the alternate hypothesis.**

- **Here we can say that went_on_backorder is related to deck_risk, ppap_risk and stop_auto_buy, so we will keep all these features for modeling.**

Preliminary Data Analysis

Imbalance in the dataset

As we can notice that there are about ~1.6 million rows of data out of which only 11,188 rows are under the “yes” category of “went_on_backorder”.

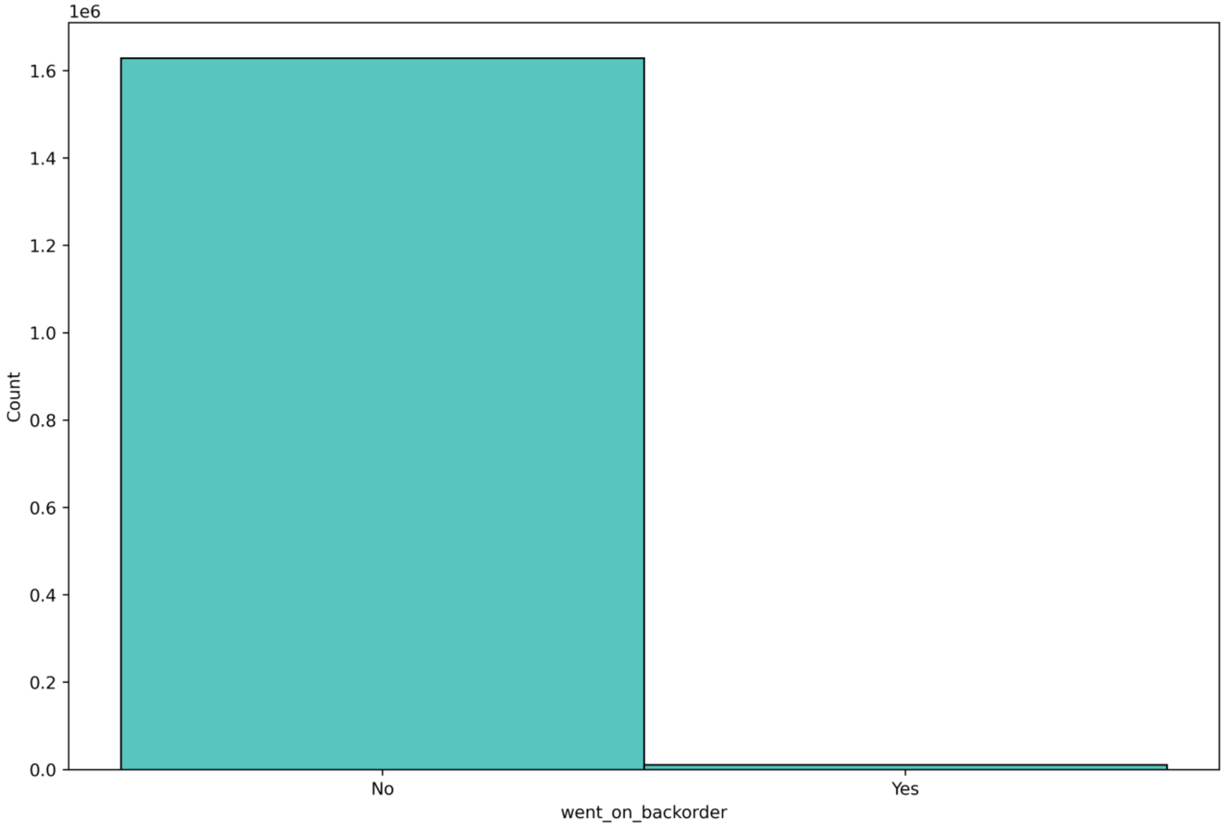
There are two ways of dealing with this imbalance.

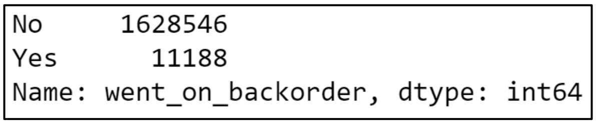
- Up-sampling
- Down-sampling

Since there are about 11,000 rows of data under “yes” category, which by itself is a huge number we have decided to add to it another 11,000 rows of data randomly from the “No” category and perform the training and testing. In short, we have decided to do Down-sampling.

Intuition: Randomly picking 11,000 rows of “No” category out of ~1.6 million rows of data may seem skeptic to the fact that there might be many attributes corresponding to “No” missing in the picked 11,000 rows. To overcome this fact the next good way of approach is to cross validate this random process with different seeds and perform the analysis. If anything suspicious to be found like a sudden accuracy drift, then the Down sampling method can be considered unreliable.

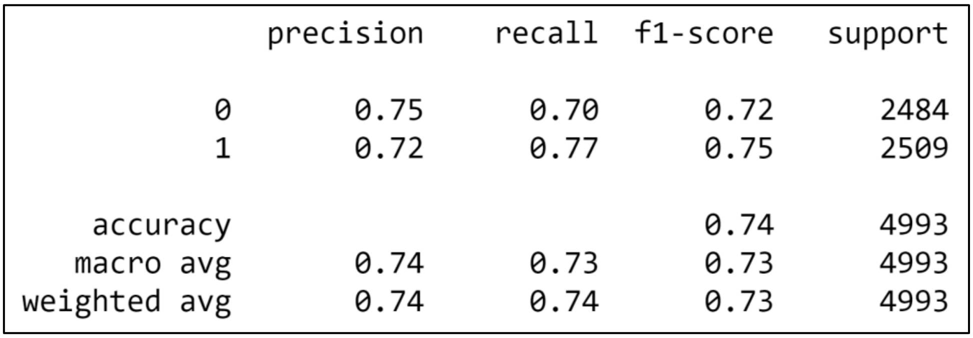
Pictorial representation of imbalance in the dataset





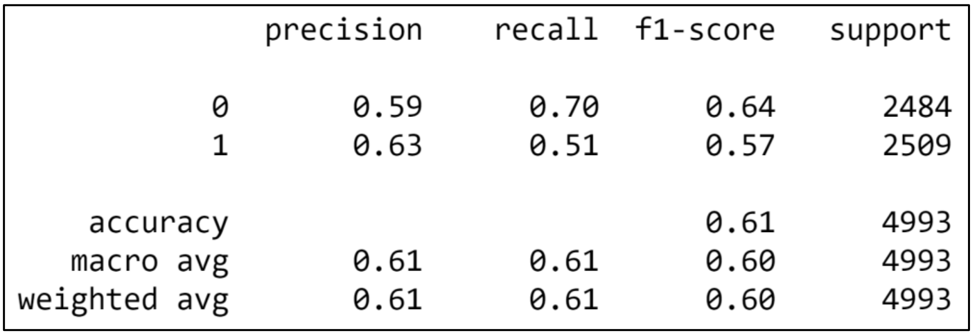
```
No     1628546
Yes      11188
Name: went_on_backorder, dtype: int64
```

Training the KNN Model Classifier



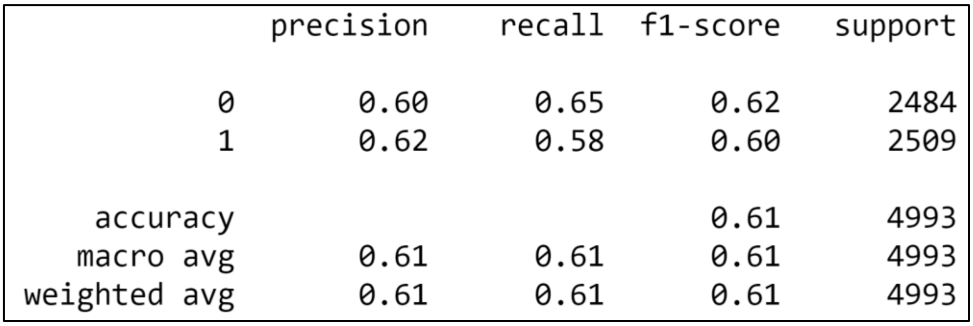
```
              precision    recall  f1-score   support

           0       0.75      0.70      0.72      2484
           1       0.72      0.77      0.75      2509

    accuracy                           0.74      4993
   macro avg       0.74      0.73      0.73      4993
weighted avg       0.74      0.74      0.73      4993
```

Training the SVC Model Classifier



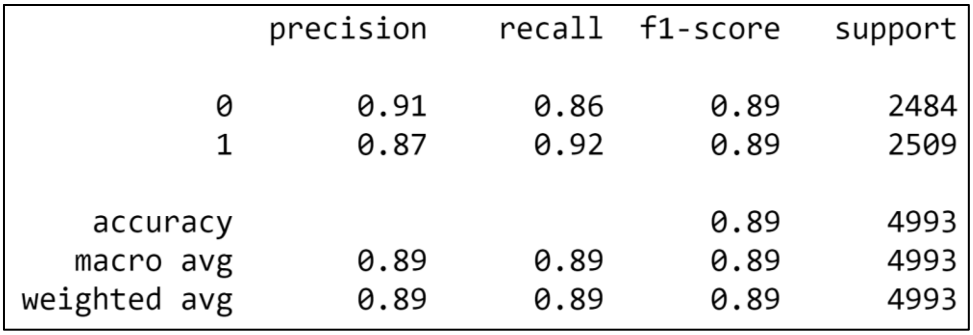
```
              precision    recall  f1-score   support

           0       0.59      0.70      0.64      2484
           1       0.63      0.51      0.57      2509

    accuracy                           0.61      4993
   macro avg       0.61      0.61      0.60      4993
weighted avg       0.61      0.61      0.60      4993
```

Training the Logistic Regression model

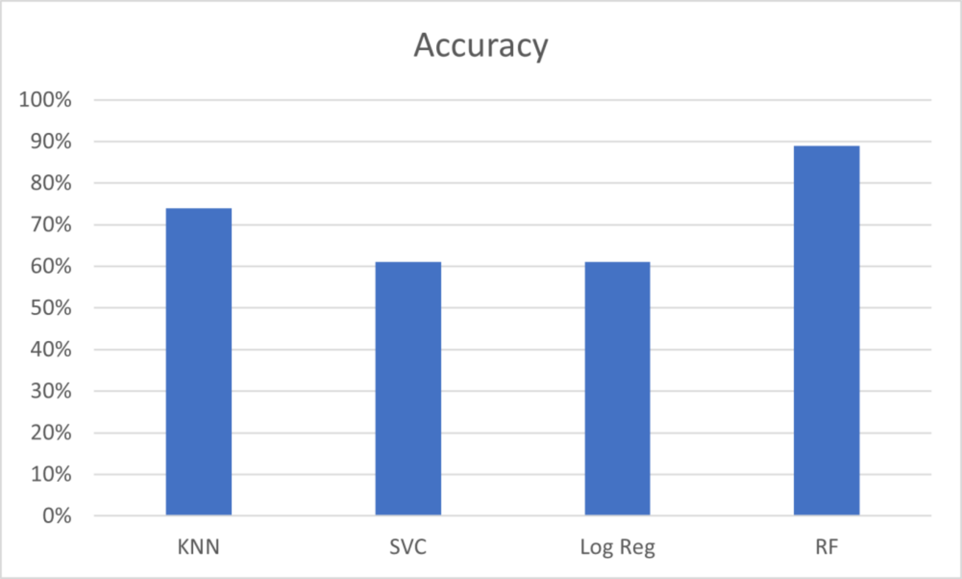


```
              precision    recall  f1-score   support

           0       0.60      0.65      0.62      2484
           1       0.62      0.58      0.60      2509

    accuracy                           0.61      4993
   macro avg       0.61      0.61      0.61      4993
weighted avg       0.61      0.61      0.61      4993
```

Training the Random Forest Classifier



```
              precision    recall  f1-score   support

           0       0.91      0.86      0.89      2484
           1       0.87      0.92      0.89      2509

    accuracy                           0.89      4993
   macro avg       0.89      0.89      0.89      4993
weighted avg       0.89      0.89      0.89      4993
```

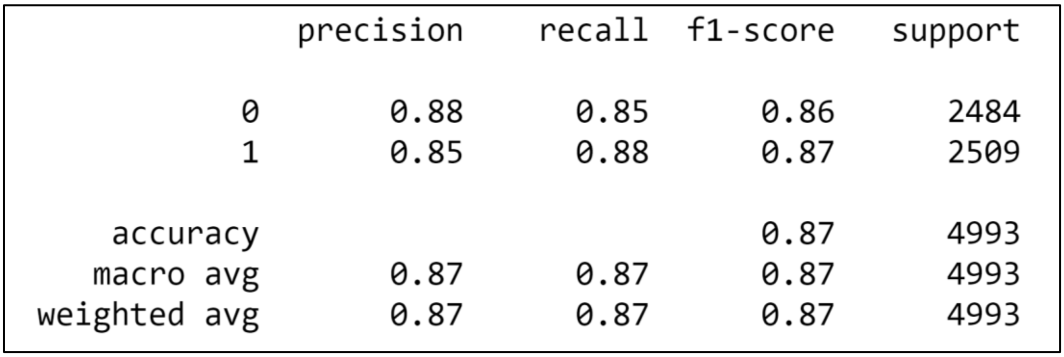


Choosing the model based on accuracy

- Four different models have been chosen to predict the outcome variable.
- **Random forests** shows promising results with an accuracy of 89%.
- Further model building can be done in the final stage using boosting methods.

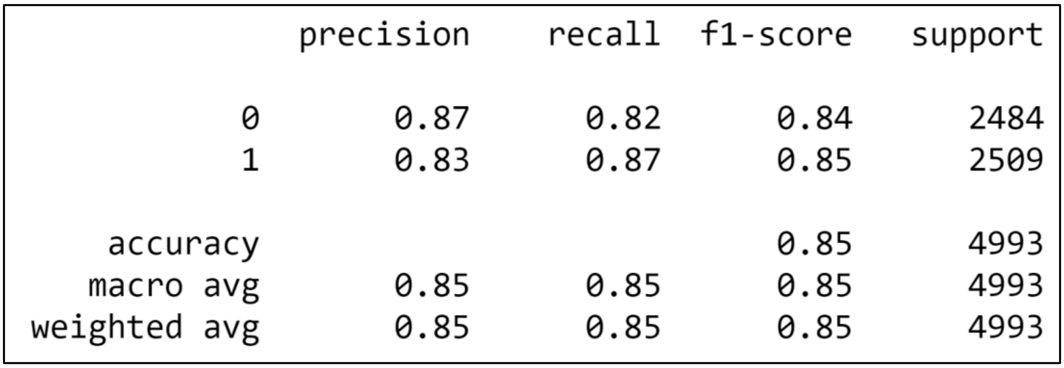
Final Data Analysis

Training the Gradient Boosting Classifier



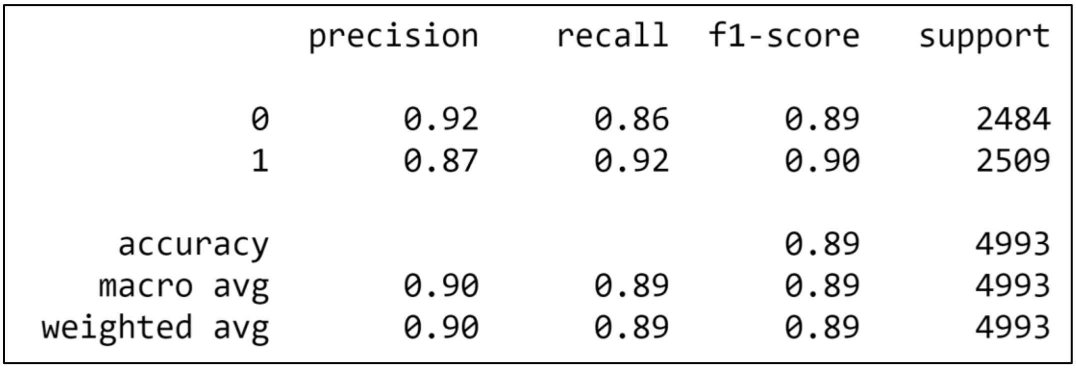
```
              precision    recall  f1-score   support

           0       0.88      0.85      0.86      2484
           1       0.85      0.88      0.87      2509

    accuracy                           0.87      4993
   macro avg       0.87      0.87      0.87      4993
weighted avg       0.87      0.87      0.87      4993
```

Training the Adaptive Boosting Classifier



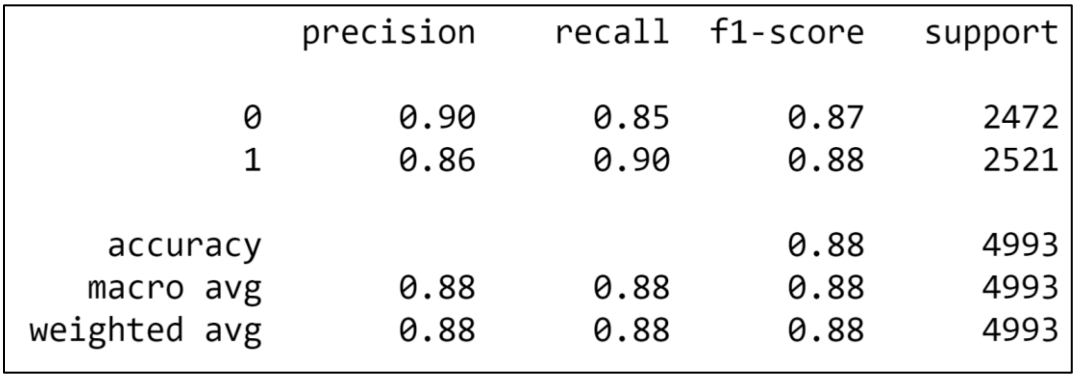
```
              precision    recall  f1-score   support

           0       0.87      0.82      0.84      2484
           1       0.83      0.87      0.85      2509

    accuracy                           0.85      4993
   macro avg       0.85      0.85      0.85      4993
weighted avg       0.85      0.85      0.85      4993
```

Training the Adaptive Boosting Classifier with Random Forest estimator

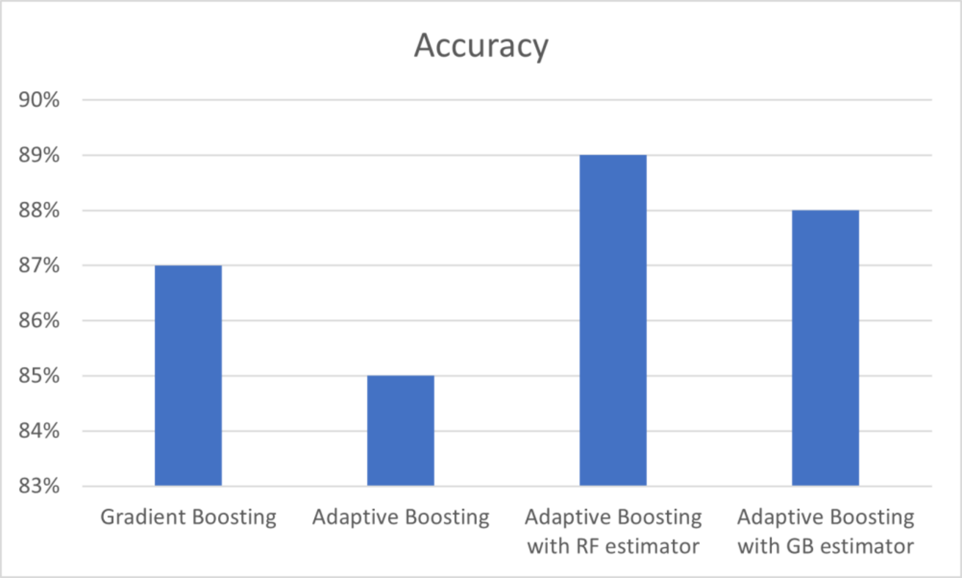


```
              precision    recall  f1-score   support

           0       0.92      0.86      0.89      2484
           1       0.87      0.92      0.90      2509

    accuracy                           0.89      4993
   macro avg       0.90      0.89      0.89      4993
weighted avg       0.90      0.89      0.89      4993
```

Training the Adaptive Boosting Classifier with Gradient Boosting estimator



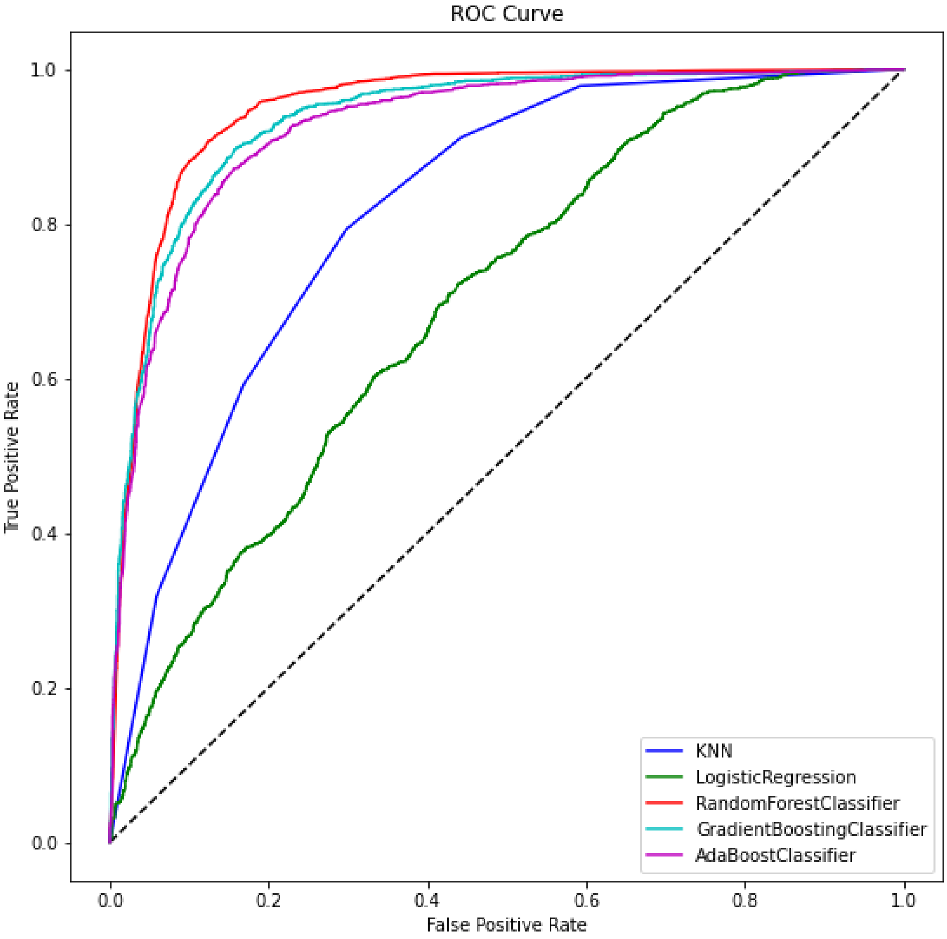
```
              precision    recall  f1-score   support

           0       0.90      0.85      0.87      2472
           1       0.86      0.90      0.88      2521

    accuracy                           0.88      4993
   macro avg       0.88      0.88      0.88      4993
weighted avg       0.88      0.88      0.88      4993
```



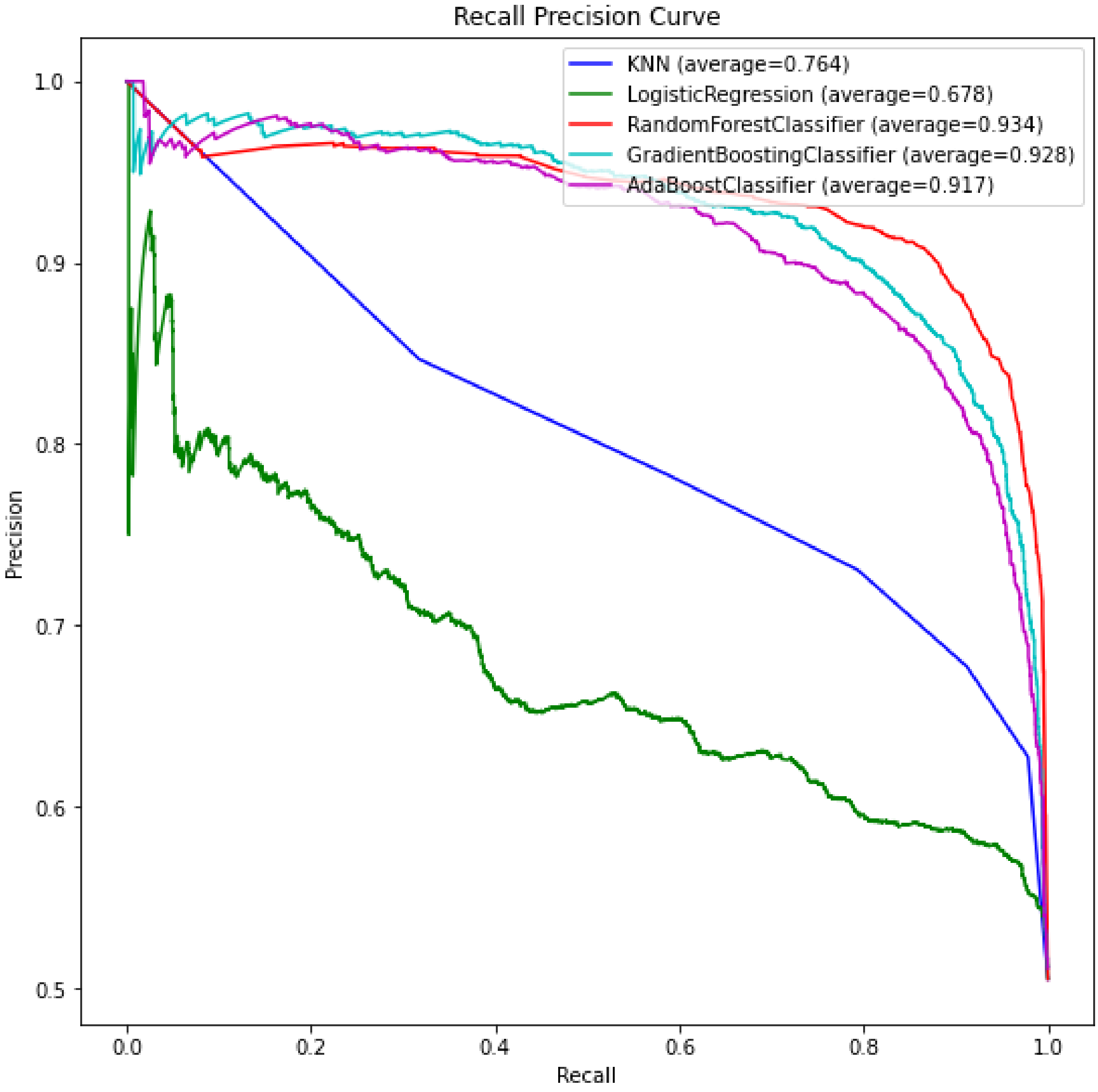
Choosing the model based on accuracy

- **Adaptive Boosting with Random forest estimator** shows promising results with an accuracy of 89%.

The ROC curve



The Recall Precision curve



Project description

1. Proposal and Exploratory data analysis
2. Preliminary data analysis
3. Final report and presentation

Exploratory Data Analysis

1. Identify a data set. Discuss the collection of the data. Who organized the collection, who actually performed the collection, for what purposes.
2. Describe all of the fields in the data set. Note: you may group fields if it makes sense. If there are a very large number of fields, pick 15-20.
   i. Name
   ii. Short, one sentence description
   iii. Description of potential values (include data type and units)
   iv. summary statistics: mean, median, quartiles, range if applicable
   v. Is this an outcome or a potential predictor?
   (dependent or independent variable)
3. Write a brief CoNVO statement for the dataset (200-400 words).
4. Create a pairs plot. Identify which independent variables seem to produce variation in the outcome variables. If there are more than 7 variables in the data set you should select (with justification) 7 variables to include in the pairs plot.
5. Identify an outcome and three independent variables based on your CoNVO statement and the pairs plot.
6. Explore the variables by applying at least five plot types. Note: the easy way to do this is to take from the five named plot types, but you may find more complex plots are more useful.
7. Design and create a single plot that shows the effects of all three independent variables in question 5.
8. Suggest possible reasons for relationships between variables.Use causal diagrams in your discussion.
9. What were the difficulties while preparing the data for analysis?
10. Are there other datasets you would like to combine with this dataset to improve your analysis?

Preliminary data analysis

1. Propose a question that you will explore for the final project. What makes it interesting? Difficult?
2. Provide a more detailed CoNVO statement (400-600 words)
3. Preliminary analysis including data visualization.
4. Apply methods using set of methods following the same rules as the applicable homework assignment (e.g. is this a regression problem or a classification problem. Pick one from each class of methods)
6. Plan of work. Identify what class of methods you will be comparing in the final project.

Final project

This is a suggested structure. You may use a different structure if you wish but all content should appear in the final report.

1. Introduction. Motivation, background and goals.
2. About the dataset. Discuss origin and structure of data. What is missing? Why was it collected? How was it collected? What makes it interesting?
3. Obstacles. What were obstacles you faced while analyzing the data? How did you deal with data that you would have wanted but was not available?
4. Analysis.
   i. Describe and justify any preprocessing that you did.
   ii. Compare several predictive analytics methods from different families and discuss how they perform? Why do some methods work better than others?
   iii. For the best performing model, discuss in depth the quality of its predictions and try to build some intuition for the reader of why it makes its decisions.
5. Discussion
   i. Any interesting relationships in the data you observed and potential reasons for them.
   ii. Based on your data analysis, do you think you will be able to achieve your goal? Explain why/why not.
   iii. Future directions for the project.

Note: It is not necessary for there to be a 'successful' application of data mining methods that 'solves' the problem. But you should explain how the methodology you applied failed (i.e. it may imply that the relationship hypothesized does not exist, although if you say this, you should probably have tried either additional methodologies or an additional hypothesis before reaching this conclusion).