upGrad

E-Commerce and Retail B2B Case Study

by

Shagun Deora

Problem Statement

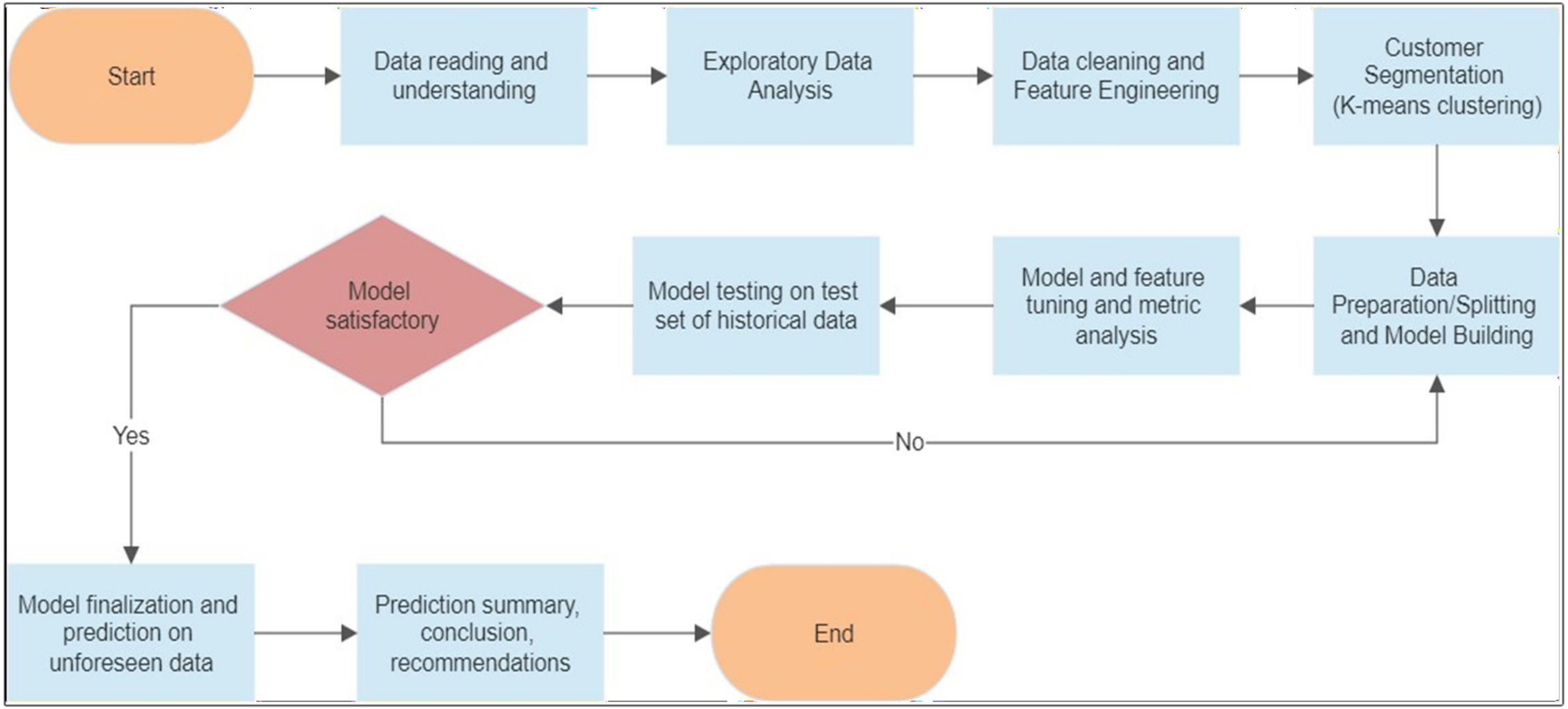
Schuster is a multinational retail company dealing in sports goods and accessories. Schuster conducts significant business with hundreds of its vendors, with whom it has credit arrangements. Unfortunately, not all vendors respect credit terms and some of them tend to make payments late. Schuster levies heavy late payment fees, although this procedure is not beneficial to either party in a long-term business relationship. The company has some employees who keep chasing vendors to get the payment on time; this procedure nevertheless also results in non-value-added activities, loss of time and financial impact. Schuster would thus try to understand its customers’ payment behavior and predict the likelihood of late payments against open invoices.

Assignment Objective

- Analyze the customer transactions data to find different payment behaviors
- Segregate the customers based on their previous payment patterns/behaviors
- Predict the likelihood of delayed payment against open invoices from the customers based on the historical data
- Draw some business insights based on the developed model

upGrad

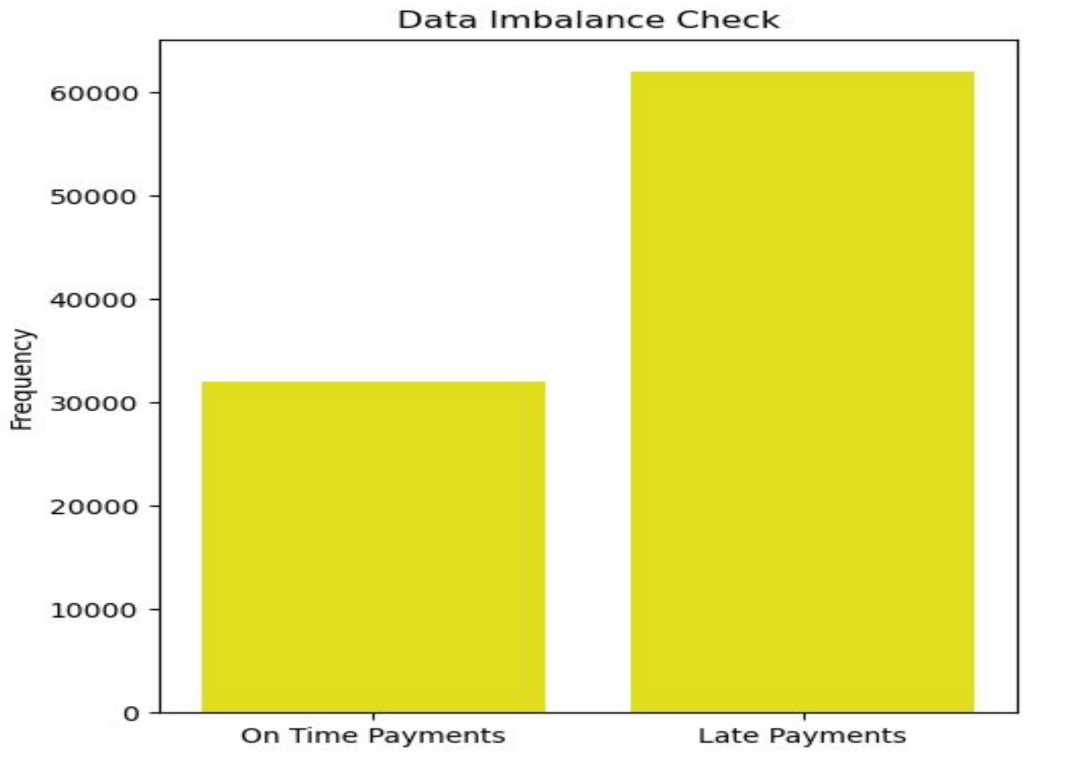
Strategic Approach to the Problem

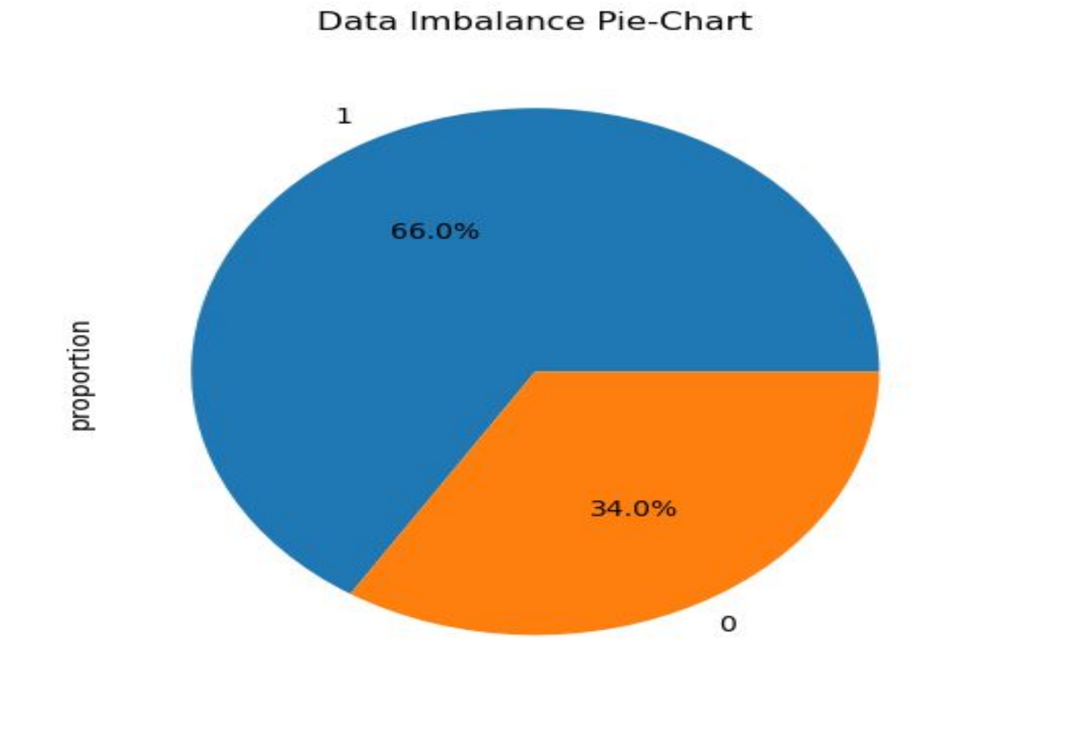

Start
Data reading and understanding
Exploratory Data Analysis
Data cleaning and Feature Engineering
Customer Segmentation (K-means clustering)
Data Preparation/Splitting and Model Building
Model and feature tuning and metric analysis
Model testing on test set of historical data
Model satisfactory
No
Yes
Model finalization and prediction on unforeseen data
Prediction summary, conclusion, recommendations
End

Data reading and understanding

The target variable is derived from the dataset taking the due date and receipt date into consideration.

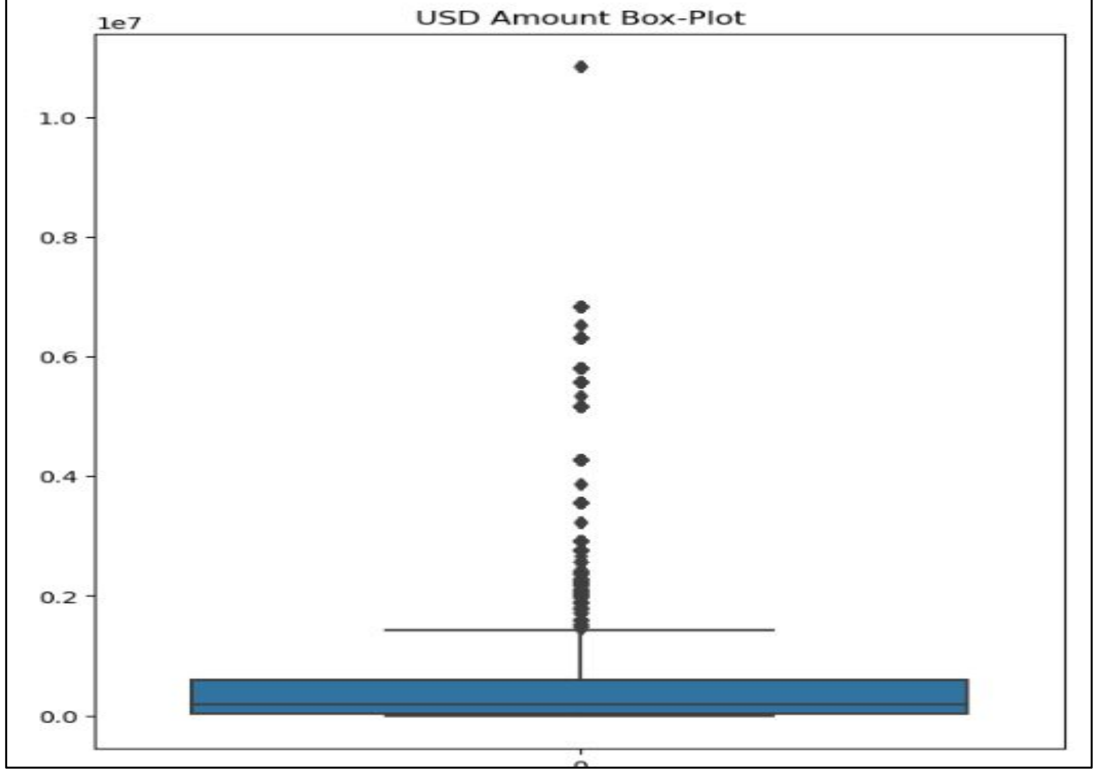
The class imbalance is also checked to avoid model biasness and it is found to be in the ratio of 1:2



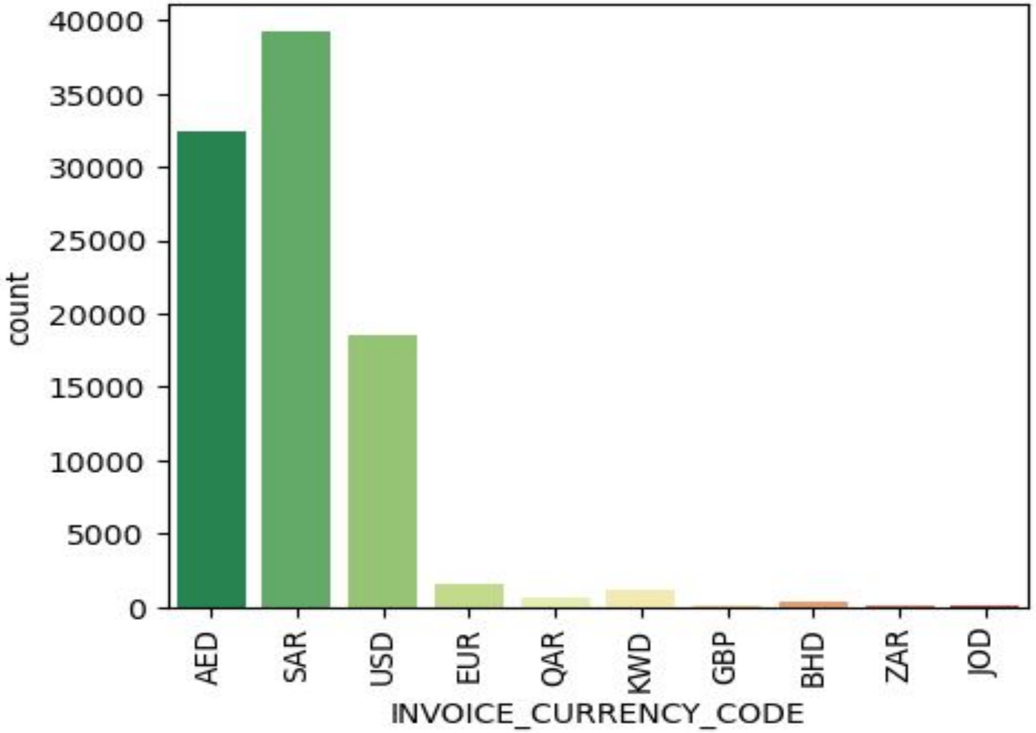


Exploratory Data Analysis

Univariate Analysis – USD Amount

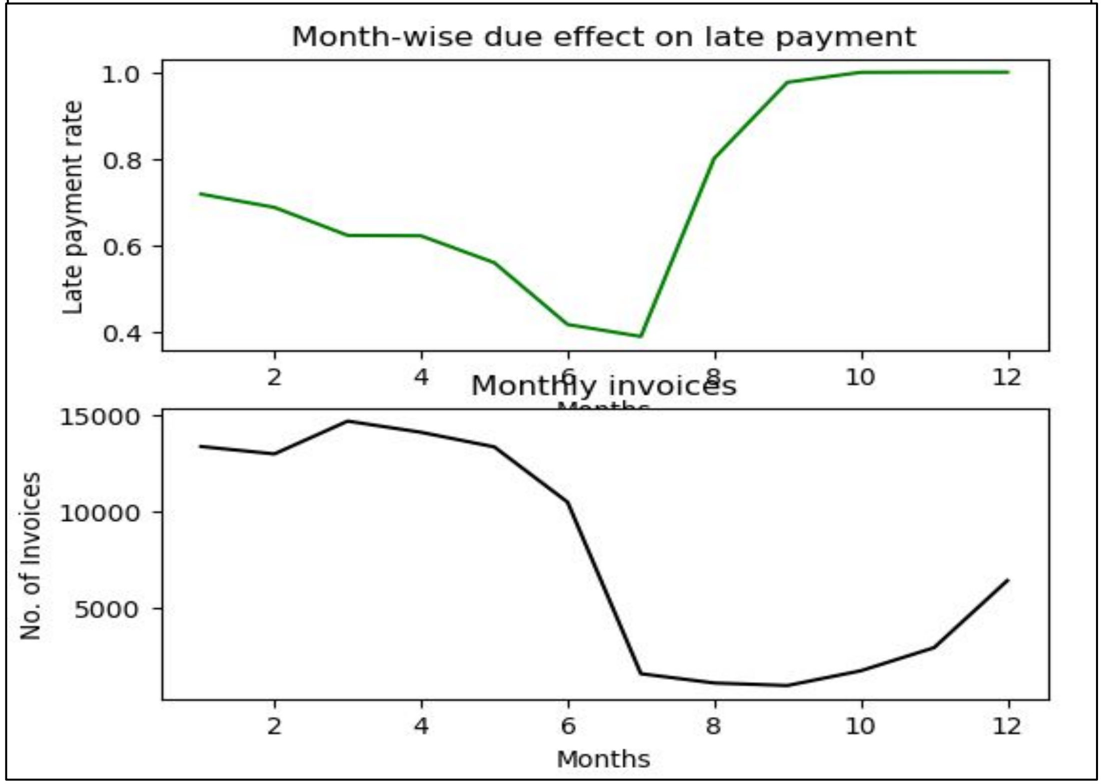


Univariate Analysis - Invoice_Currency_Code

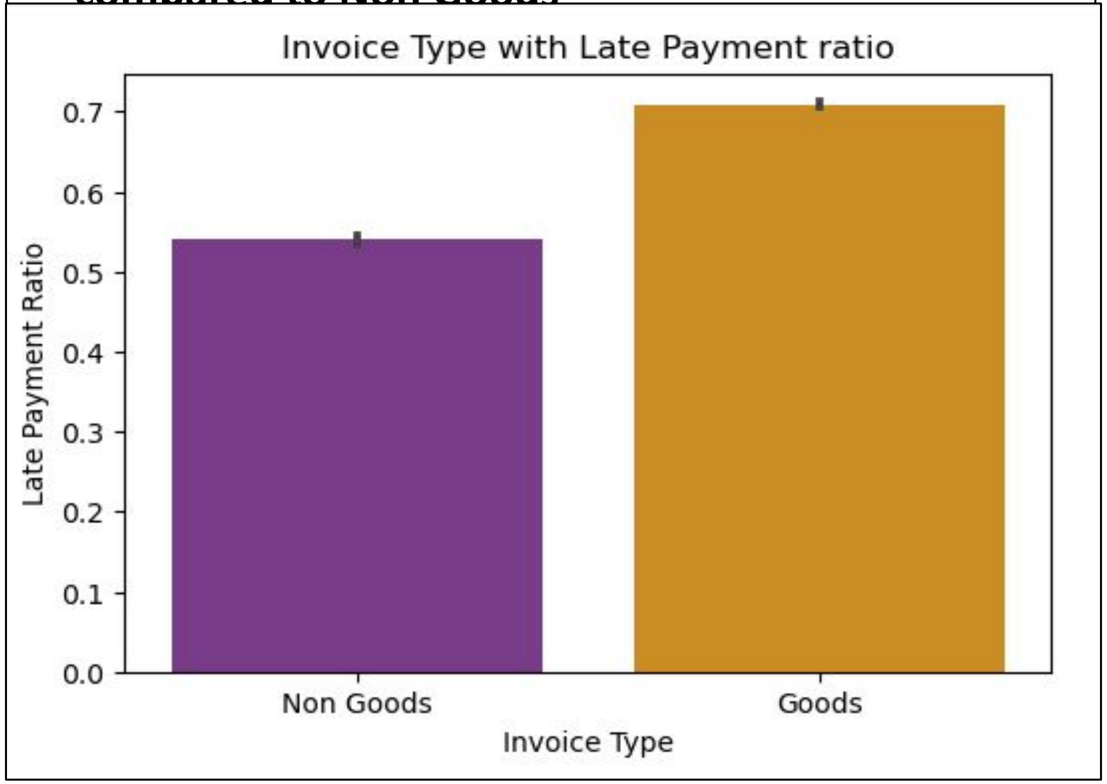


Exploratory Data Analysis

Bivariate Analysis – Month-wise Invoice and its impact on late payment – Here we can clearly see that there is a surge in delayed payment after July



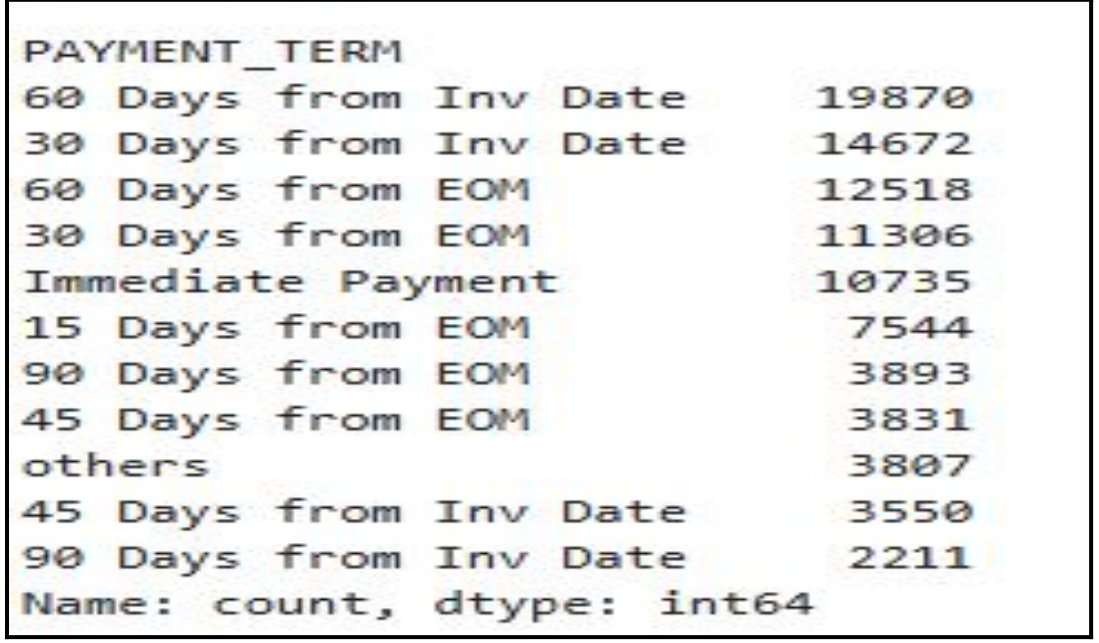
Bivariate Analysis of Invoice Type shows delayed payment is a common scenario for Goods compared to Non Goods



Data cleaning and Feature Engineering

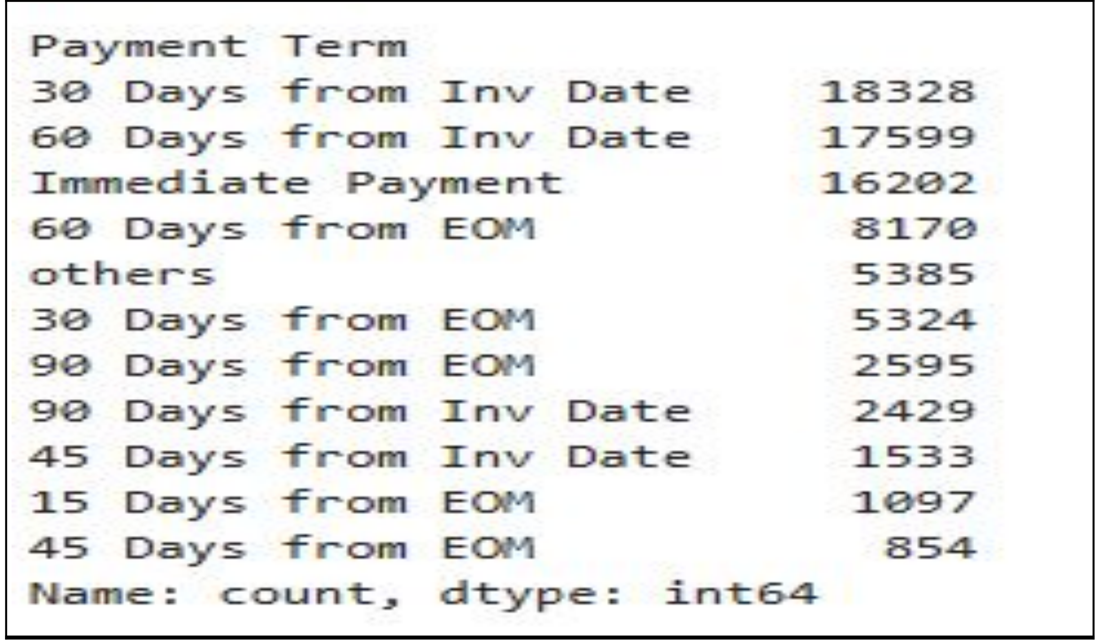
In PAYMENT_TERM feature there were 42 variables which is not conducive for modelling so using clubbing method it is reduced to 11 for both the datasets

Received_Payments_Data Dataset



```
PAYMENT_TERM
60 Days from Inv Date      19870
30 Days from Inv Date      14672
60 Days from EOM           12518
30 Days from EOM           11306
Immediate Payment          10735
15 Days from EOM            7544
90 Days from EOM            3893
45 Days from EOM            3831
others                      3807
45 Days from Inv Date       3550
90 Days from Inv Date       2211
Name: count, dtype: int64
```

Open_Invoice_data Dataset

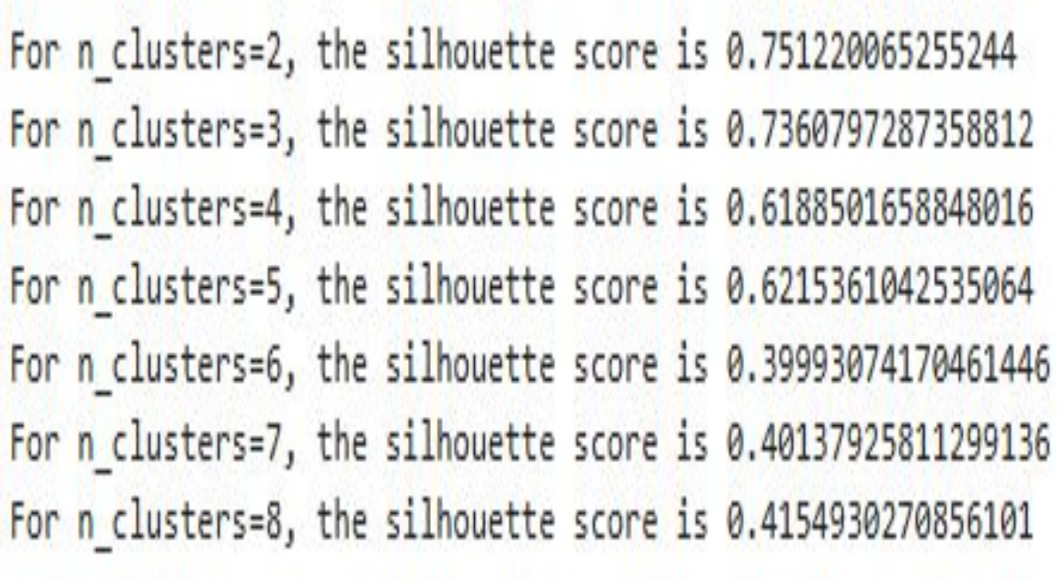


```
Payment Term
30 Days from Inv Date      18328
60 Days from Inv Date      17599
Immediate Payment          16202
60 Days from EOM            8170
others                      5385
30 Days from EOM            5324
90 Days from EOM            2595
90 Days from Inv Date       2429
45 Days from Inv Date       1533
15 Days from EOM            1097
45 Days from EOM             854
Name: count, dtype: int64
```

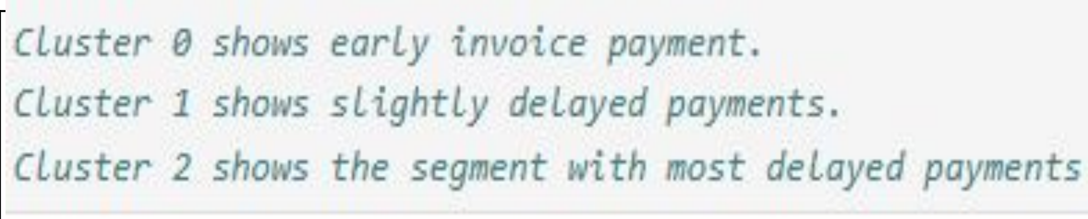
Customer Segmentation (K-Means Clustering)

One of the objectives was to categorize customers to understand payment behaviors which was achieved by K-means clustering using average and standard deviation of number of days it took for the vendor to make payment

The number of clusters were decided to be 3 since with increase in clusters post 3, there was a significant decrease in silhouette score



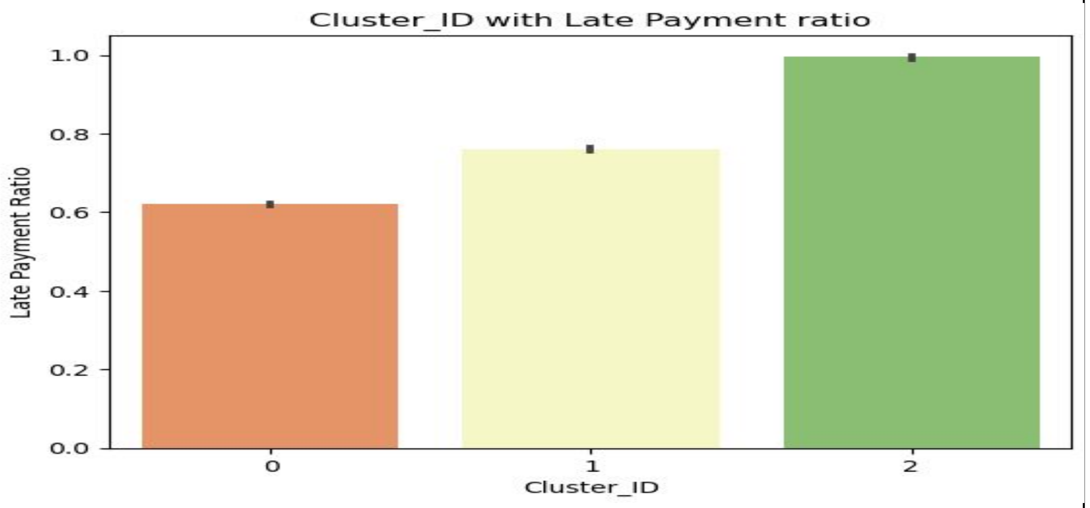
```
For n_clusters=2, the silhouette score is 0.751220065255244
For n_clusters=3, the silhouette score is 0.7360797287358812
For n_clusters=4, the silhouette score is 0.6188501658848016
For n_clusters=5, the silhouette score is 0.6215361042535064
For n_clusters=6, the silhouette score is 0.39993074170461446
For n_clusters=7, the silhouette score is 0.40137925811299136
For n_clusters=8, the silhouette score is 0.4154930270856101
```



Cluster 0 shows early invoice payment.

Cluster 1 shows slightly delayed payments.

Cluster 2 shows the segment with most delayed payments

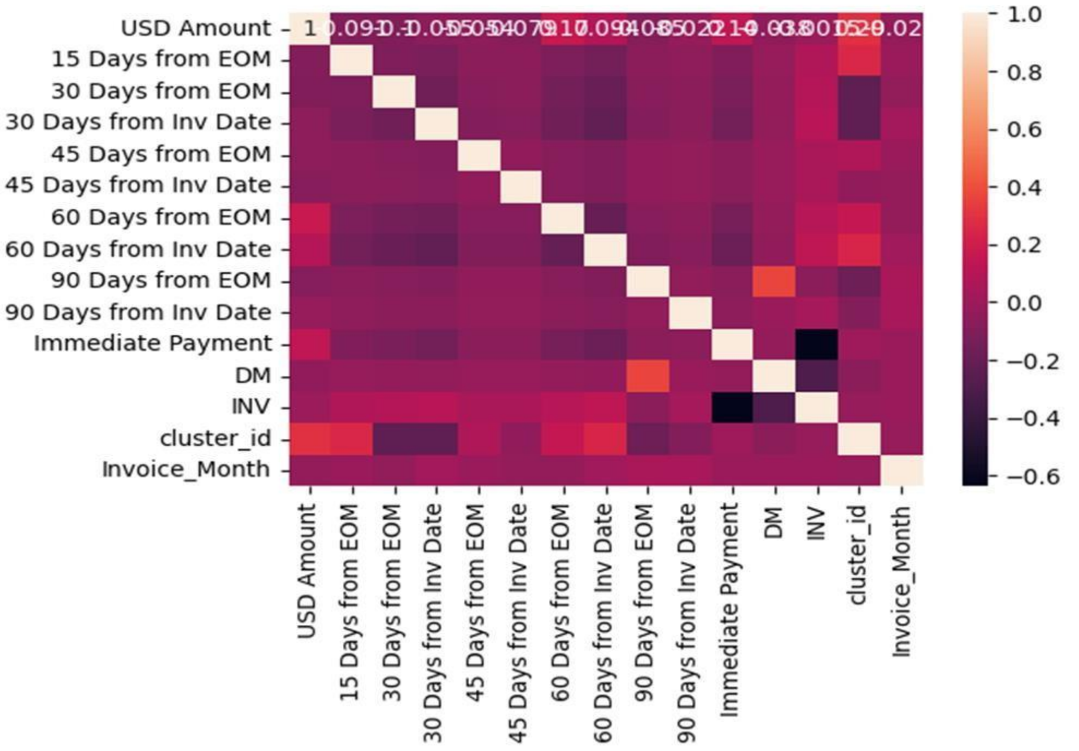


Data Preparation/Splitting and Model Building

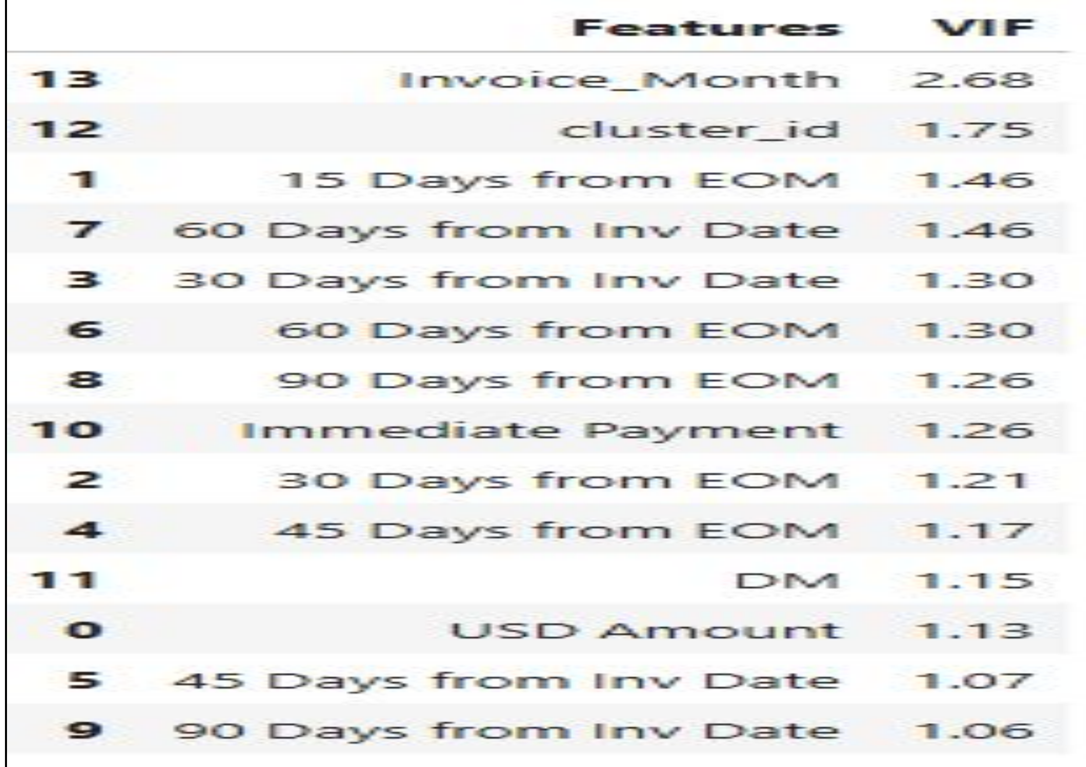
- The merged dataset have some Datetime Columns which are first converted to month columns namely **Invoice_Month & Due_Month** for easy computation.
- Then the dataset is passed through **Train-Test Split** in 7:3 ratio
- Followed by **Feature Scaling** using StandardScaler to bring the variables in the same scale

Model and Feature tuning with Metric Analysis

Testing Correlation among different independent variables of train dataset



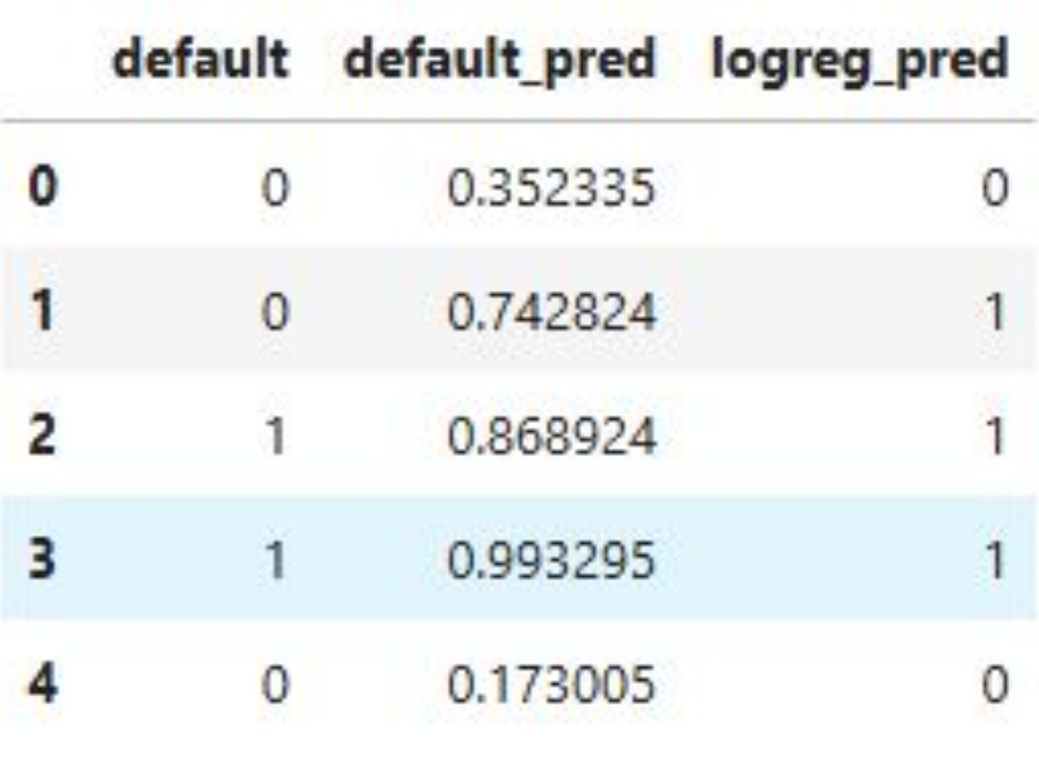
Relevant features which are finally incorporated in the model for prediction



	Features	VIF
13	Invoice_Month	2.68
12	cluster_id	1.75
1	15 Days from EOM	1.46
7	60 Days from Inv Date	1.46
3	30 Days from Inv Date	1.30
6	60 Days from EOM	1.30
8	90 Days from EOM	1.26
10	Immediate Payment	1.26
2	30 Days from EOM	1.21
4	45 Days from EOM	1.17
11	DM	1.15
0	USD Amount	1.13
5	45 Days from Inv Date	1.07
9	90 Days from Inv Date	1.06

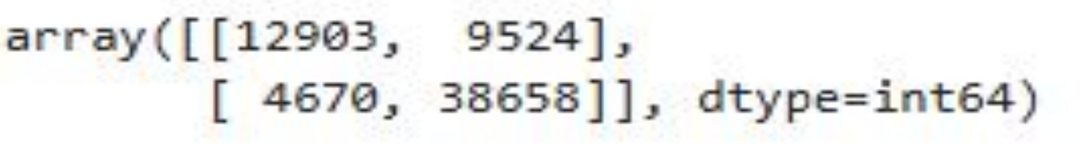
Validatingthe Model on set of historical data

Validating Model Prediction taking random cut-off as 0.5



	default	default_pred	logreg_pred
0	0	0.352335	0
1	0	0.742824	1
2	1	0.868924	1
3	1	0.993295	1
4	0	0.173005	0

Checking Scores as per confusion matrix



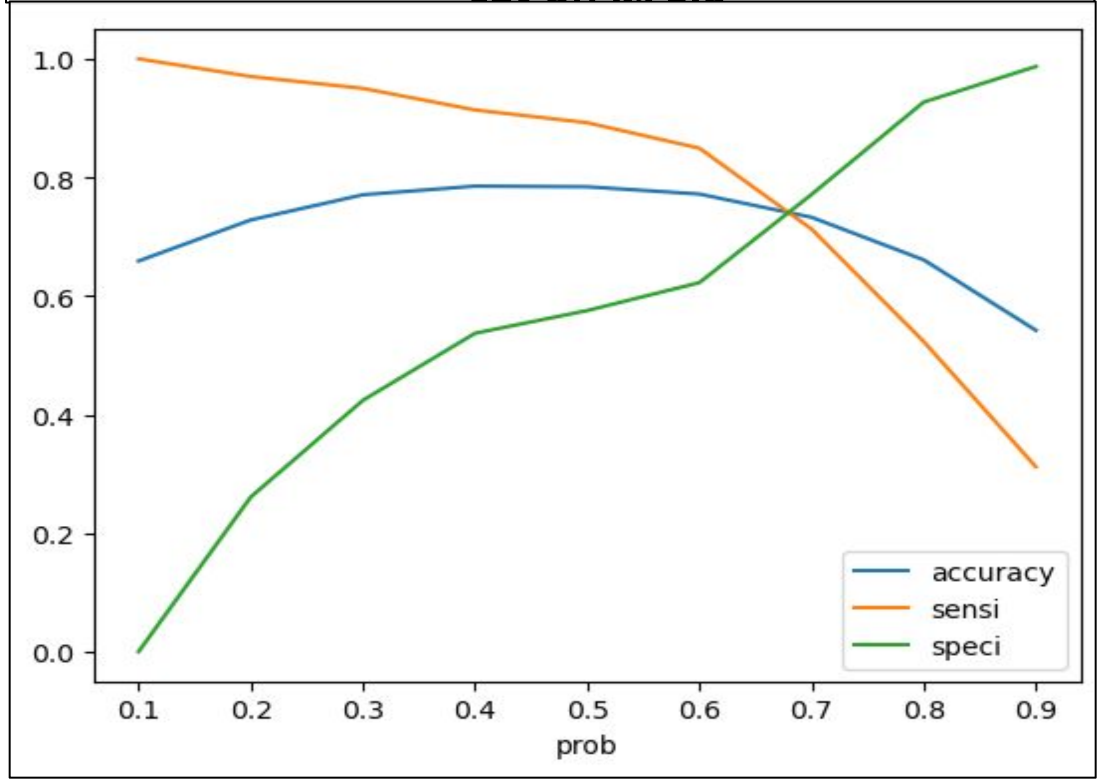
```
array([[12903,  9524],
       [ 4670, 38658]], dtype=int64)
```

- Accuracy Score - 0.78
- Recall Score – 0.89
- Specificity – 0.58
- False Positive Rate – 0.42
- Positive predictive value – 0.80
- Negative predictive value – 0.73

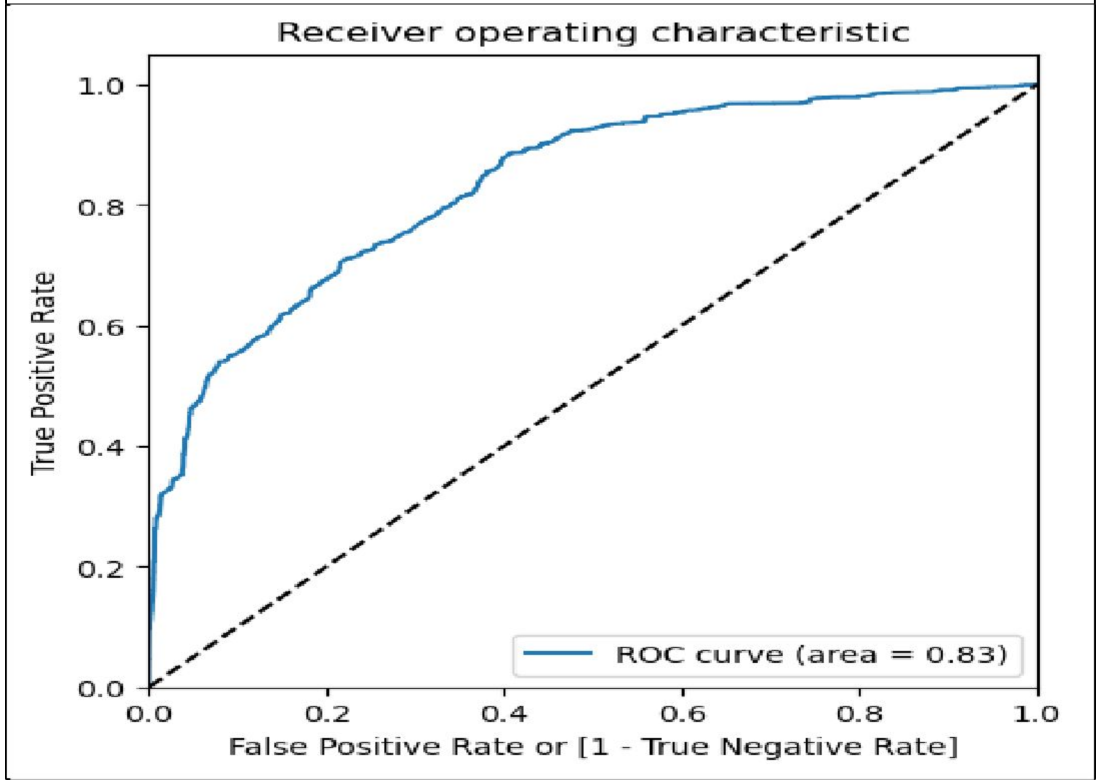
upGrad

Model Optimization

Plotting the accuracy sensitivity and specificity for the different probabilities to obtain optimum cut-off as 0.6

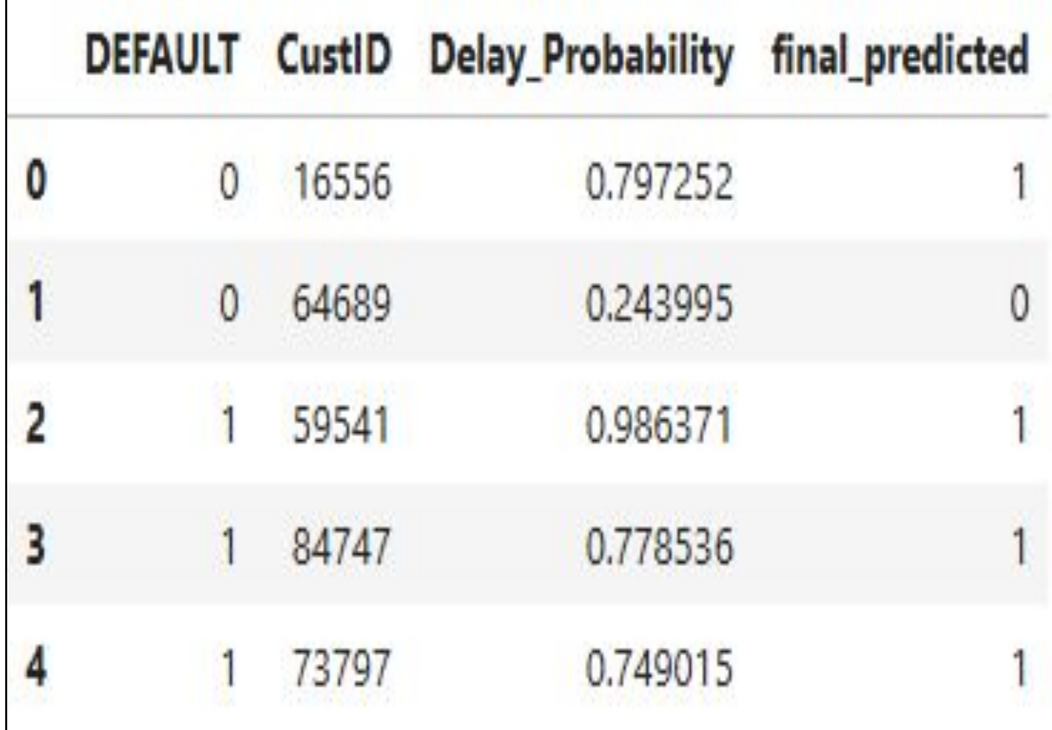


AUC of 0.83 proves that the model is good



Model testing on Test set of historical data

Testing Model Prediction taking optimum cut-off as 0.6



	DEFAULT	CustID	Delay_Probability	final_predicted
0	0	16556	0.797252	1
1	0	64689	0.243995	0
2	1	59541	0.986371	1
3	1	84747	0.778536	1
4	1	73797	0.749015	1

Checking Scores as per confusion matrix

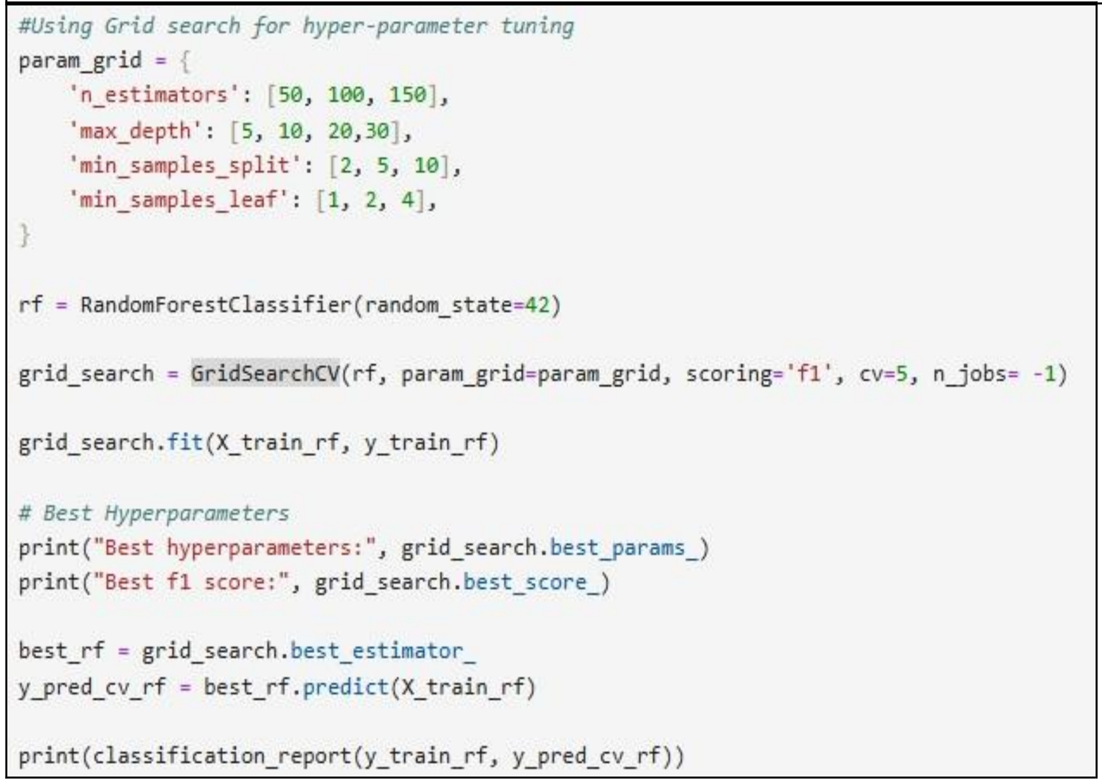
Train and test metrics are almost the same. Model is good to go.

- Accuracy Score - 0.77
- Precision Score – 0.82
- Recall Score – 0.85

upGrad

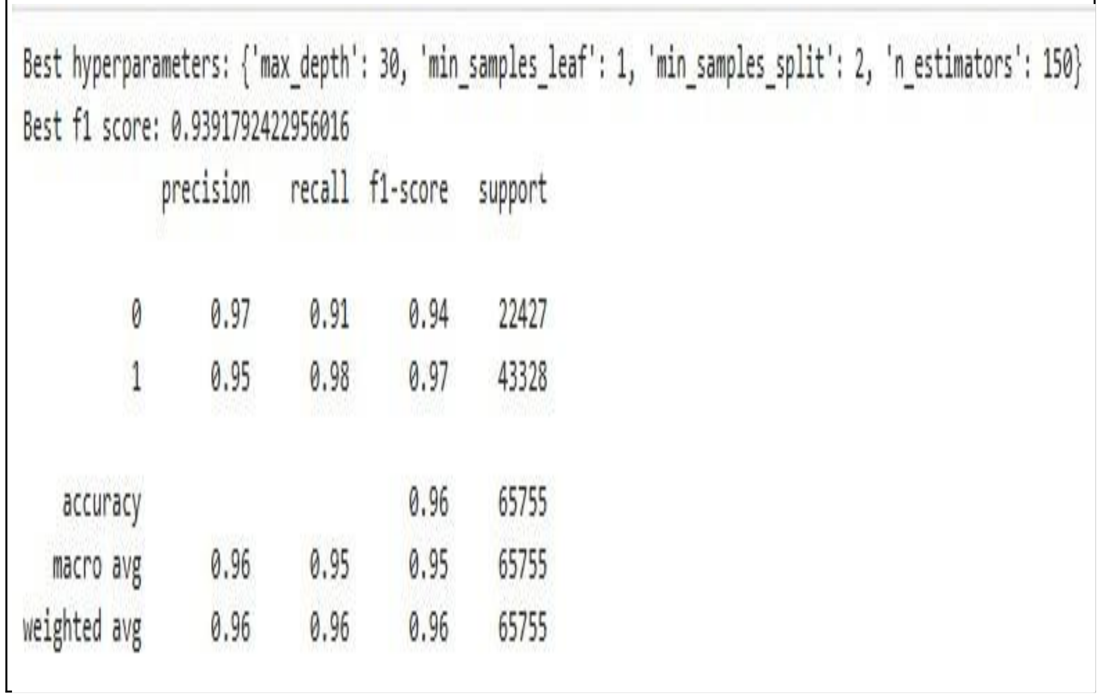
Model Satisfaction and Exploring other options

Making a Random Forest model for predictions on the Open Invoice set and using GridSearchCV for Hyper-Parameter Tuning



```
#Using Grid search for hyper-parameter tuning
param_grid = {
    'n_estimators': [50, 100, 150],
    'max_depth': [5, 10, 20,30],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4],
}

rf = RandomForestClassifier(random_state=42)

grid_search = GridSearchCV(rf, param_grid=param_grid, scoring='f1', cv=5, n_jobs= -1)

grid_search.fit(X_train_rf, y_train_rf)

# Best Hyperparameters
print("Best hyperparameters:", grid_search.best_params_)
print("Best f1 score:", grid_search.best_score_)

best_rf = grid_search.best_estimator_
y_pred_cv_rf = best_rf.predict(X_train_rf)

print(classification_report(y_train_rf, y_pred_cv_rf))
```

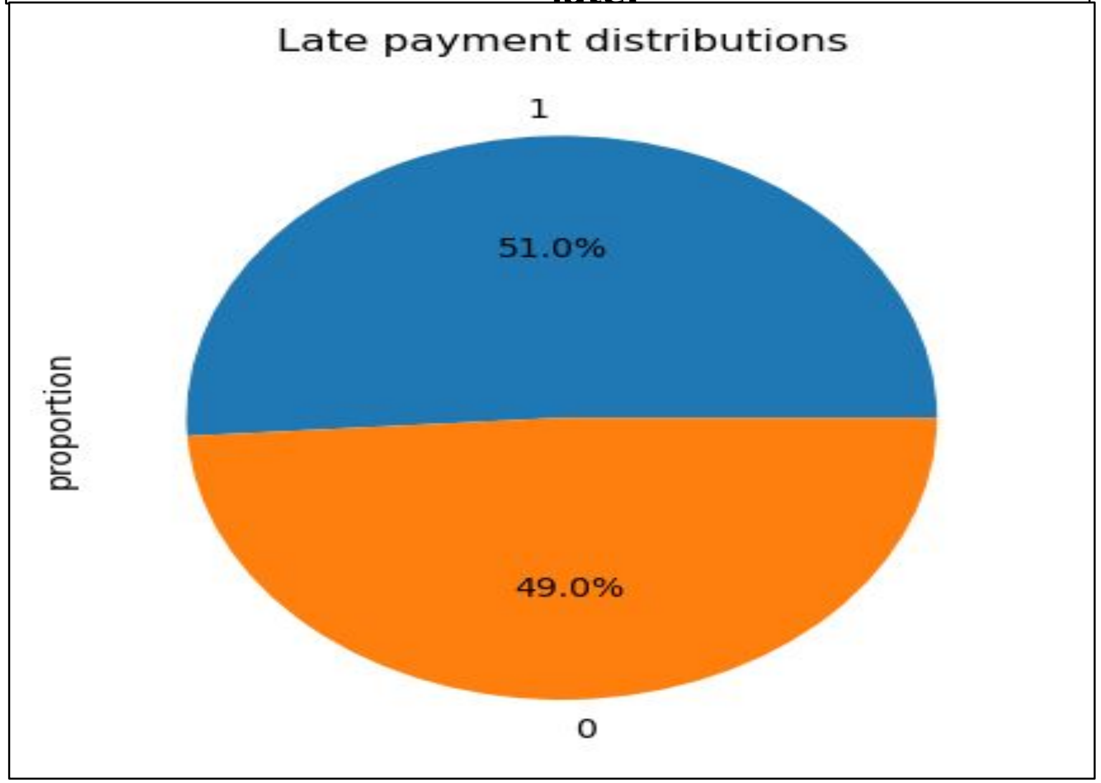
Best hyperparameters



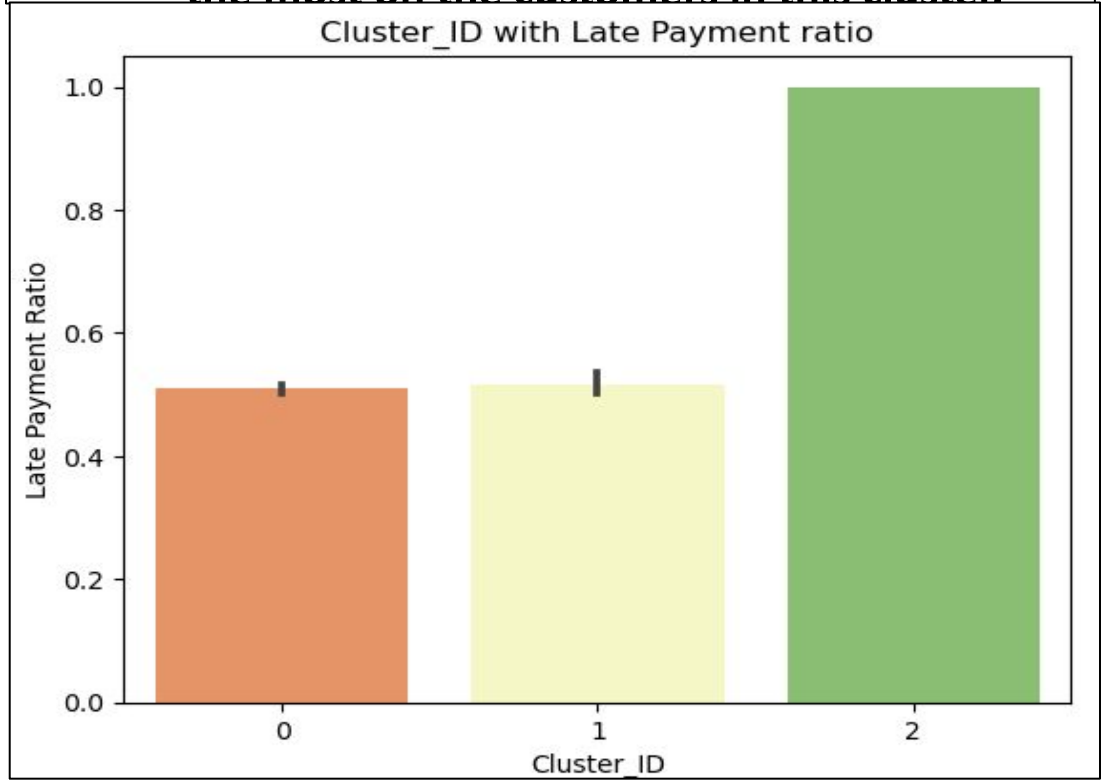
```
Best hyperparameters: {'max_depth': 30, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 150}
Best f1 score: 0.9391792422956016
              precision    recall  f1-score   support

           0       0.97      0.91      0.94     22427
           1       0.95      0.98      0.97     43328

    accuracy                           0.96     65755
   macro avg       0.96      0.95      0.95     65755
weighted avg       0.96      0.96      0.96     65755
```

Model finalization and prediction on unforeseen data

As per the model prediction 51% of the vendors will pay on time but 49% will default and pay late.

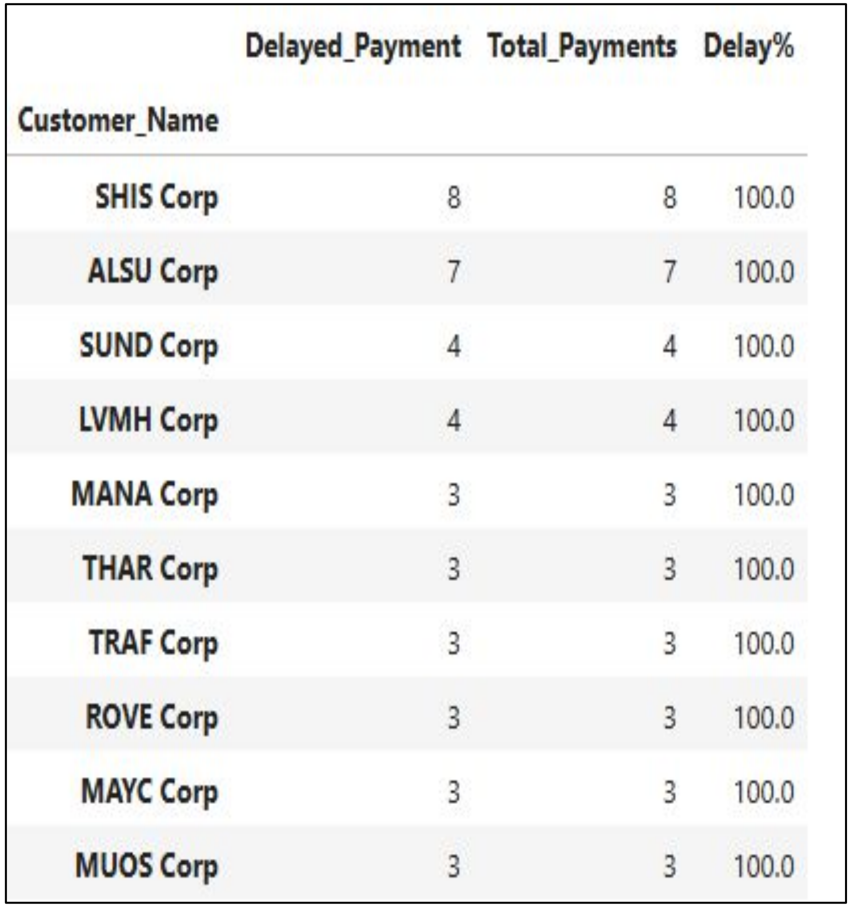

Late payment distributions

Cluster ID 2 has severely higher chances of payment default, the organization must focus the most on the customers in this cluster.


Cluster_ID with Late Payment ratio

Prediction Summary and Recommendations

Top 10 customers with highest delay rates



	Delayed_Payment	Total_Payments	Delay%
Customer_Name			
SHIS Corp	8	8	100.0
ALSU Corp	7	7	100.0
SUND Corp	4	4	100.0
LVMH Corp	4	4	100.0
MANA Corp	3	3	100.0
THAR Corp	3	3	100.0
TRAF Corp	3	3	100.0
ROVE Corp	3	3	100.0
MAYC Corp	3	3	100.0
MUOS Corp	3	3	100.0

Second half of the year has low number of invoices but higher ratio of late payments and first half of the year has higher number of invoices but lower late payment ratio. . The mean and median invoice value for on time payments is more that that of late payments. Which means that smaller orders show a tendency of delayed payments compared to larger orders. . Late payment rate is the lowest for INV and highest for CM invoice classes. . Goods sales have a higher late payment ratio than Non goods sales. . Clustered the customers into three distinct cluster 0,1 and 2. . Customers belonging in cluster 2 have the highest chances of delayed payments and should be handled with proper precautions. Extensive focus must be paid to these customers to make them pay on time. . Top 50 customer names mentioned above have the highest chances of delayed payments as predicted by the models and should be taken into consideration.

Thank you