Case Study: ECommerce & Retail B2B

- How Company focused only on customers who tend to delay payment

- Submitted by: Rajesh Thote, Rohit Ramachandran, Praveen Rathi
- Batch: DS C52 23

The Problem

Company

- Schuster is a multinational retail company dealing in sports goods and accessories with many vendors in B2B segment

Context

- In B2B segment it is business nature that many vendors payments are not as per payment terms and there are many cases of delayed payments

Problem statement

- To get timely payment follow up with all vendors costs high and loss of time. Company wants to know in advance likely to delay in payment so that focus on those vendors would result timely clearance of invoices

Challenges

01

Domain knowledge is very important for EDA of the data, thus as of now this is a challenge how we can get some useful insights from the data which has real world relevance

02

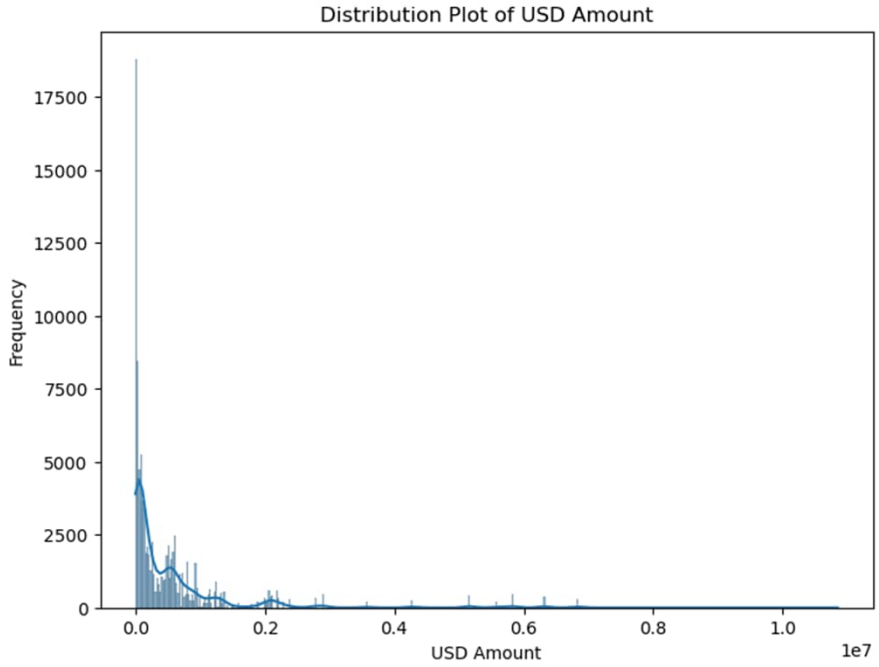
To identify the trend of late payment for various customers, it will be challenging to segment customer so as to meaning full for the model

03

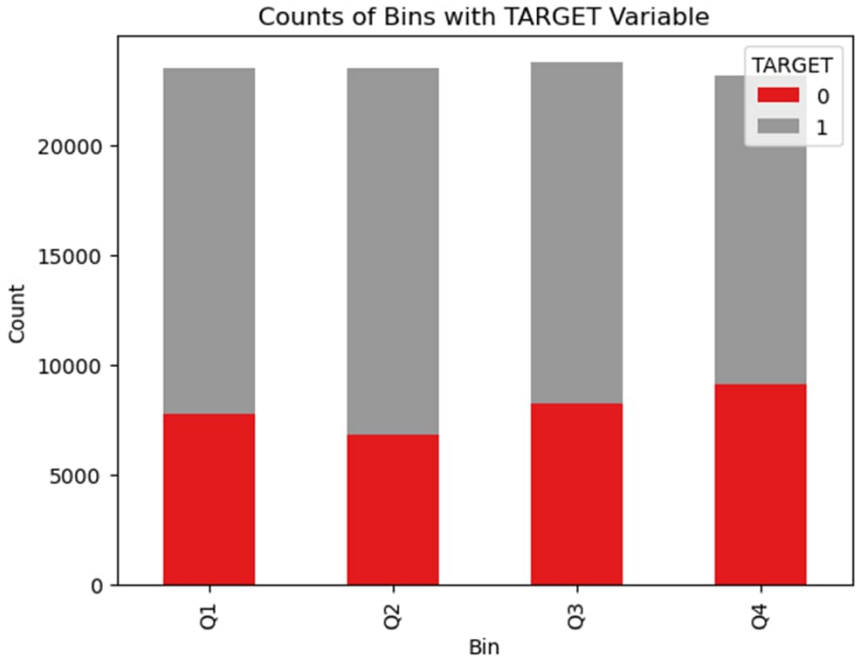
As this is a classification problem hence Model selection will be challenging as there are so many different models with pros and cons, we have to create a best model considering all limitations of a model

USD amount Analysis

We can see that major of the transaction are with value less than 1.5 million USD.

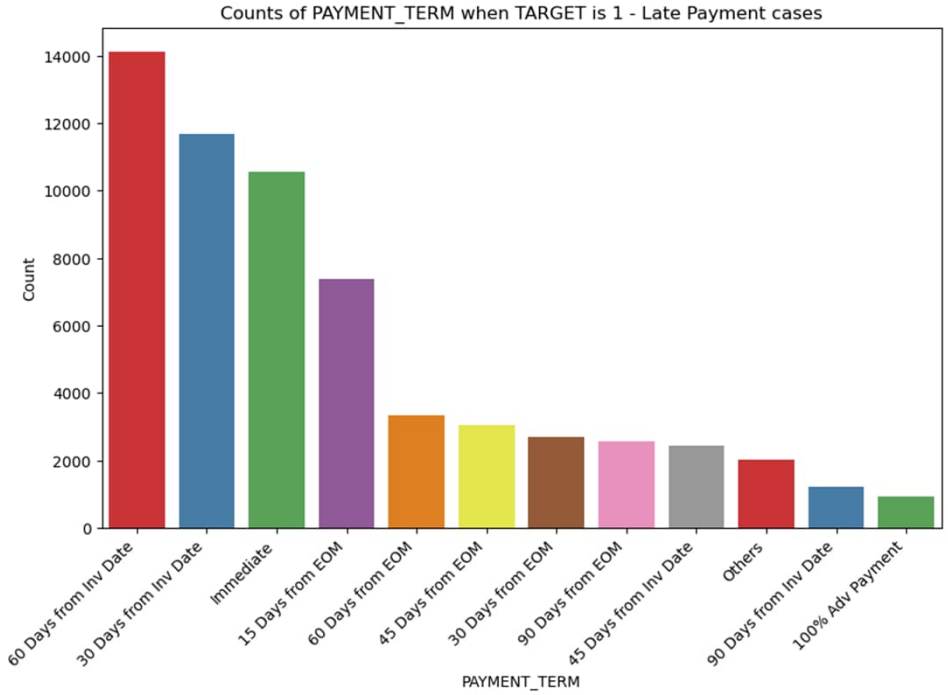


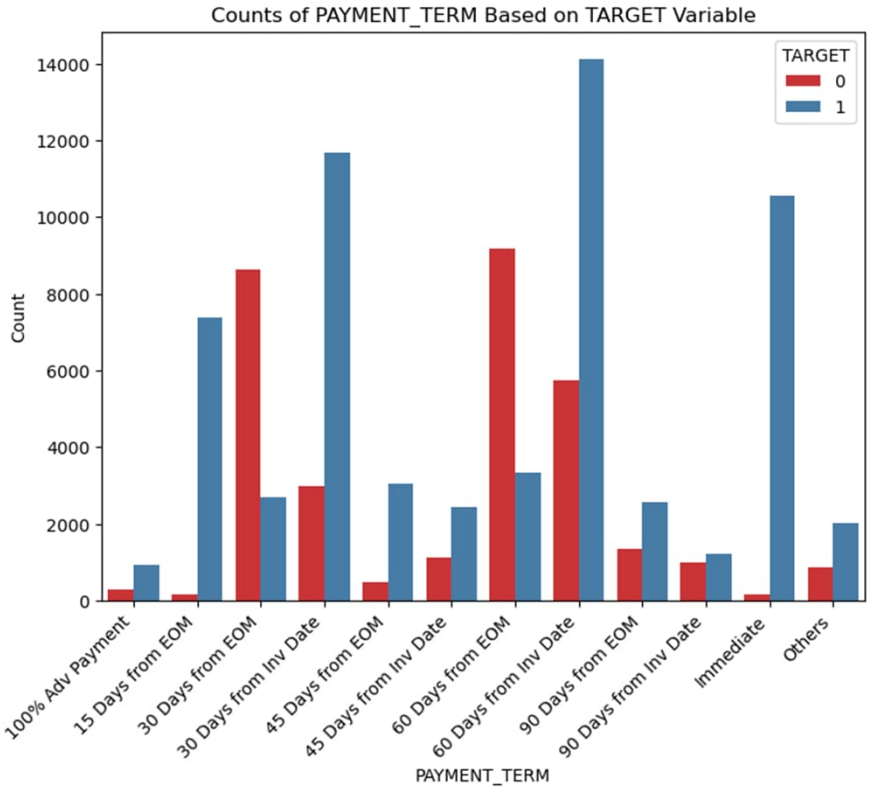
We can see that Q4 has slightly higher prone to late payment.



Set Priority

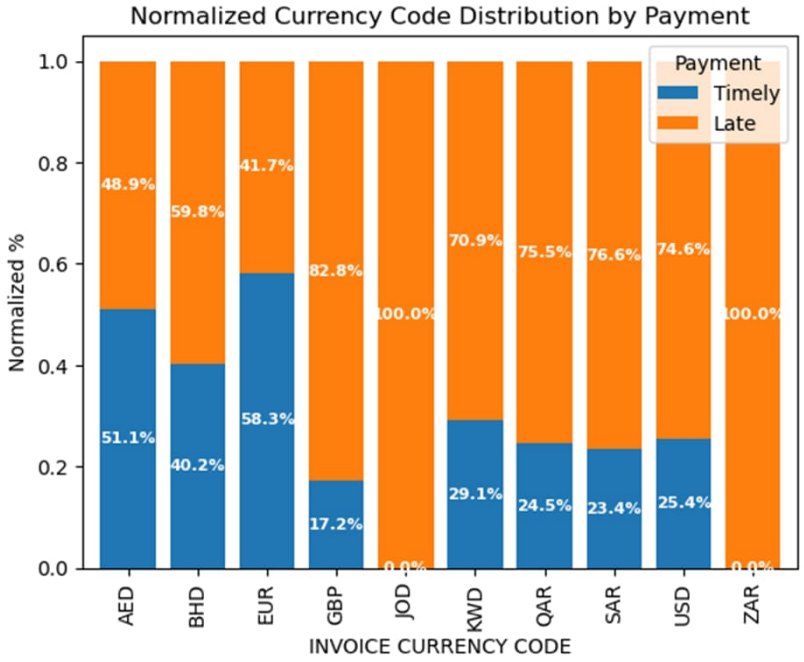
We can see from below bar plot, 60 days, 30 days and immediate payment terms are with most delays and can be targeted.



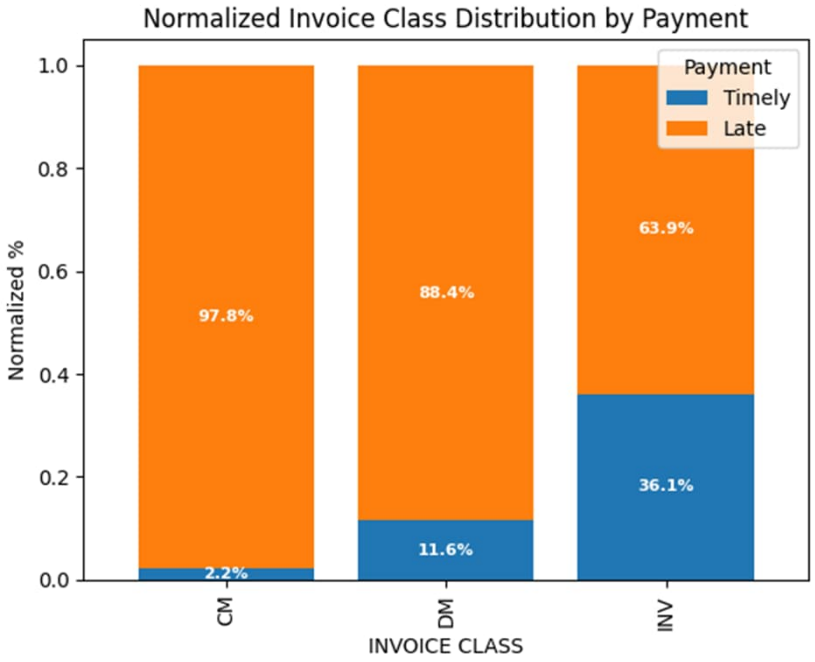


Finding Good and Bad pay master type

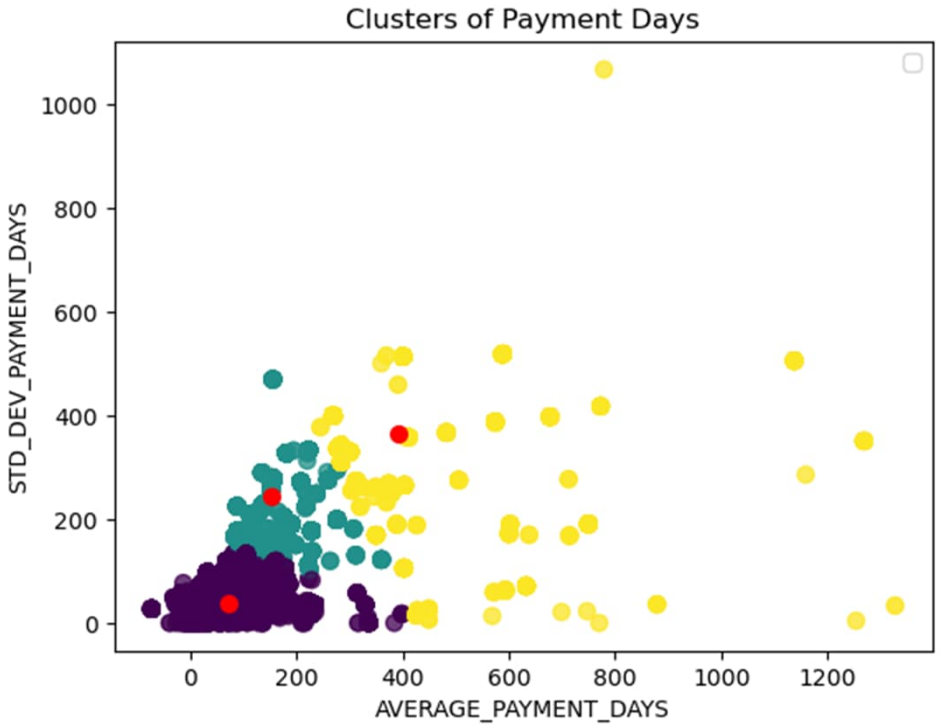
Customer paying into EUR are with good track record of payment against all other.



Customer whose payment method is CM are risky with high rate of delayed payment.



Clusters of Customer on Basis of Payment Days



We found with Elbow curve method/SSD and Silhouette analysis, 3 are the best way to form a cluster.

We segmented customers based on their average payment days and standard deviation of payment days.

Among these we can see as average payment days are higher the segment is less clustered.

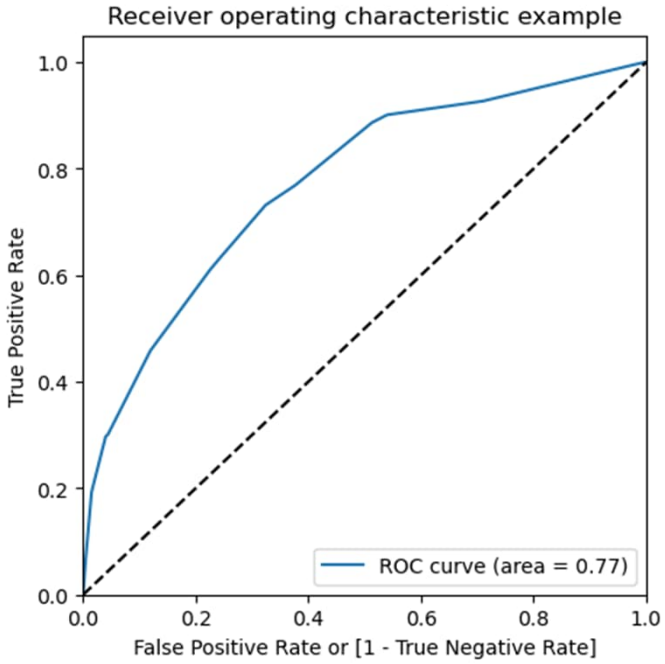
This shows that lesser the payment days less is the deviation.

Model Evaluation Logistic Regression

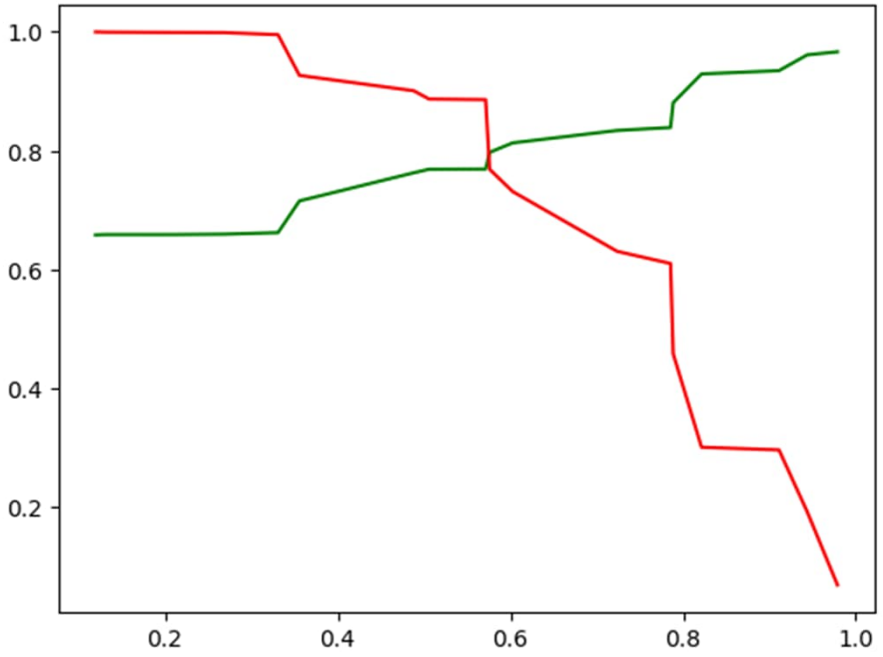
Our model is trained with ~75% accuracy has resulted on test data with ~74% accuracy

Logistic Regression

Our model is covering 77% of AUC on test data.



0.6 is the optimum point to take it as a cutoff probability on test data.



Logistic Regression

Summary of model

	Train	Test
# Accuracy	75%	72%
# Sensitivity	88%	74%
# Specificity	48%	68%
# Precision	77%	82%
# Recall	73%	73%

Example of final results

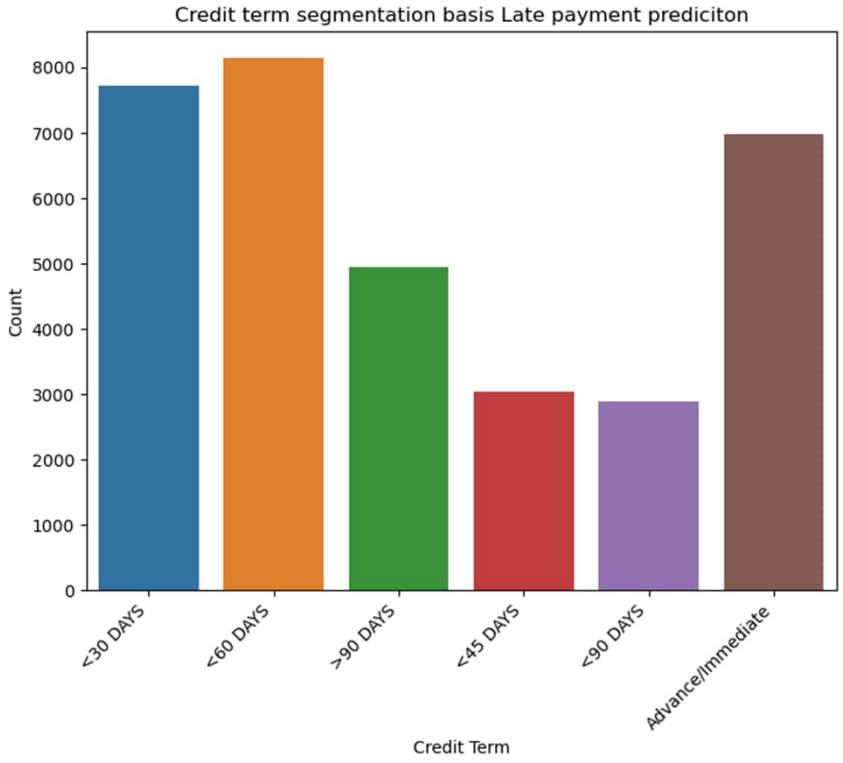
	CustID	predicted
0	1044	1
1	1076	1
2	1146	0
3	1154	1
4	1192	0

Open Invoice Predictions

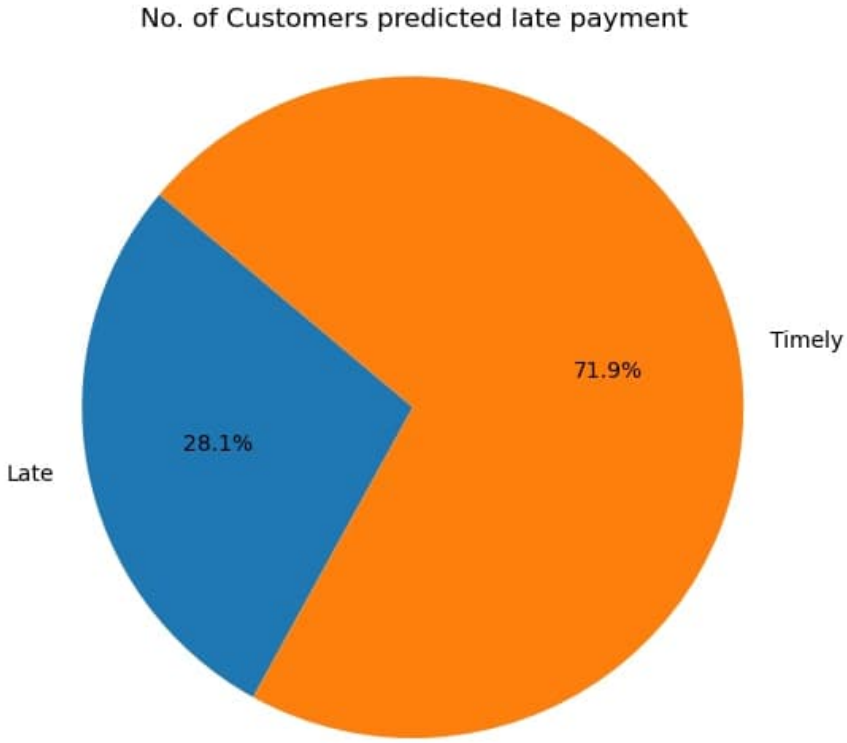
- First, we cleaned the data.
- Then prepared it for predictions making it similar to test dataset.
- We perform all basic cleaning techniques on this dataset too like renaming columns, deriving new features, creating dummies, dropping unwanted columns, etc.
- Then we start making predictions on it.
- We Set a cut-off of 60% probability which we got from Precision Recall trade off from train model.

Making Predictions on open invoice data

- Need to focus more on Advance or Immediate payment terms as they are likely to delay along with 30 Days and 60 Days credit term also.
- We had got similar results while training the model also.

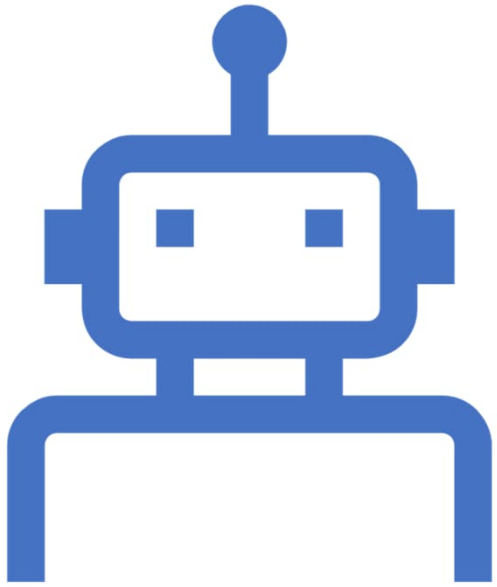


- From the model we can see that around 28% customer are going to make late payment.
- By number it is 393 customers out of 1399 customers.



Decision Tree Model

Accuracy score is 95% through Grid Search



DT Model Building and Testing

- First, we build the model without tunning any hyper parameter and we got r2 score of 70% on train dataset and 71% on test dataset.
- After hyper parameter tuning, we got accuracy score of 95% for both train and test dataset hence this model is seeming to be **overfitting**.
- We also tried 5th best estimator where min_samples_leaf=1000, there also we are getting accuracy score of 95%.

Random Forest Model

Accuracy score is 95% through Hyper Parameter tuning

Random Forest Model Building and Testing

- First, we build the model without tunning any hyper parameter and we got r2 score of 77% on train and test dataset.
- OOB score is 85% with this model.
- When we plot the ROC curve we got 96% of AUC which is very high.
- After hyper parameter tuning, we got grid_search.best_score_ of 91.7% and ROC of 98%.
- We also tried 5^{th} best estimator where min_samples_leaf=1000, there also we are getting accuracy score of 95%.
- We got 93% of accuracy score both in train and test dataset which also seem to be very high hence we are not using this model for making predictions on open invoices dataset.

Conclusions

- We tried total three model
  - Logistic Regression: This model is fairly stable with 75% of accuracy score in train and 74% in test dataset.
  - Decision Tree: This model was showing results with very high accuracy score of 95% hence it is overfitting
  - Random Forest: This model also was giving accuracy score of 95% hence we are also not considering this model.
- From above results we can see that Logistic Regression model is best to use.
- After running LR model on open invoices dataset finally we saw that nearly 28% of customer are in high risk of late payment.
- Company can focus on these customer and can allocate their resources accordingly.
- This will also improve companies working capital management