Assignment 2 – sxb180048 – Sai Pratheek Banda

Loan Status Classification

Classification Problem:

The aim of the problem is to predict the classification of the loan status to **'Fully Paid'** and '**Charged OFF'**. The interesting part of this classification is that it will help us understand the patterns for customers if they are going to get "**Charged OFF'**. **Class 0(Fully Paid)**: 28972 Observations| **Class 1 (Charged OFF)**: 7451 Observations

EDA*:*

1) Features 'Customer ID', 'Loan ID' were insignificant due to its inability to explain the target variable therefore dropped
2) Though the Dataset contains 100000 Records, after dropping the not available rows, we get consistent data of 26423.
3) The Features Term, Home Ownership, Purpose and Target Variable were converted into dummy variables.
4) In the process I have chosen to use F-1 Score as metric because its of utmost importance that we get the Class 1 classification better than other measures or accuracy.

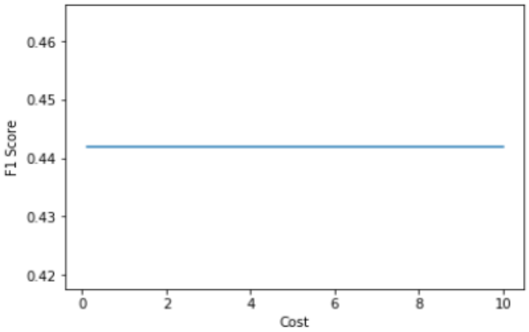
Data Sampling:

Since the data is heavily unbalanced in terms of the target classification '**Charged OFF',** The consistent data was split into train and test. Due to heavy imbalance and for the quest of a better model I have under sampled the train data to a **Class 0(Fully Paid)**: 7000 Observations| **Class 1 (Charged OFF)**: 5194 Observations.
It is a tradeoff I have chosen to for training the model to the information lost in the process.

Support Vector Machine (Linear):

Parameters:

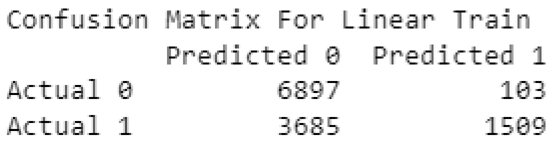
To linearly separate the data the **Regularization parameter** 'C' which, for a high value chooses a smaller-margin hyperplane resulting in better data point classification.
Implementing the **Cross Validation with 3-folds** I found that there is not significant change in the F-1 Score Metric. Even with scores as high as 200 and 300 there was no change in the Metric.



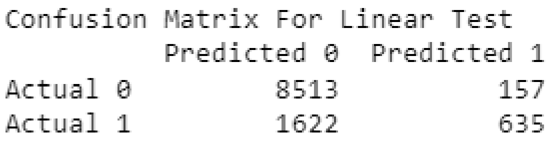
Train and Test Data:

With the default regularization Parameter SVM is fit and predicted on train data without Cross Validation to compare the Accuracy metric and have a base line to check with the test data.

As shown below in the metric, the Prediction of Class 1 is 44% which is a bad model to train and the test results reflect the same. This issue is due to inconsistent and low number of records to train on.



```
Confusion Matrix For Linear Train
          Predicted 0  Predicted 1
Actual 0         6897          103
Actual 1         3685         1509
```



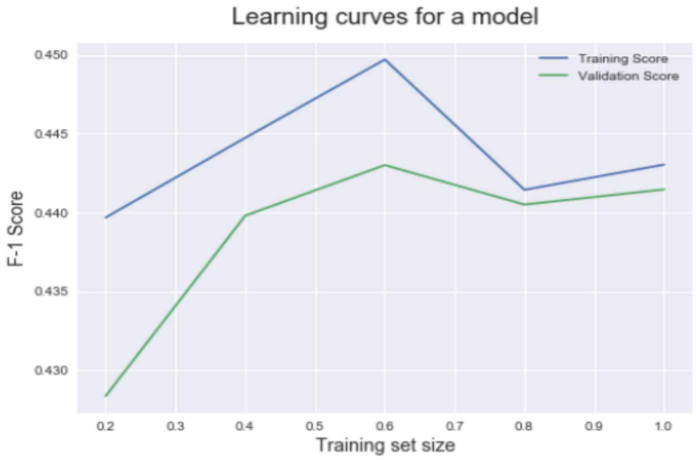
```
Confusion Matrix For Linear Test
          Predicted 0  Predicted 1
Actual 0         8513          157
Actual 1         1622          635
```

Learning Curve:

Though the learning curve performs well and coverages with the influx of increase of data. With a train F-1 Score of 44.3% and test of 44.14% but the graph metric is at 45% which is worse than a coin toss of 50%.

But when experimented with the unsampled data, the results were similar or even worse.

Yes the model metrics increase with data influx but more records of Class 1 are required.



Support Vector Machine (Polynomial):

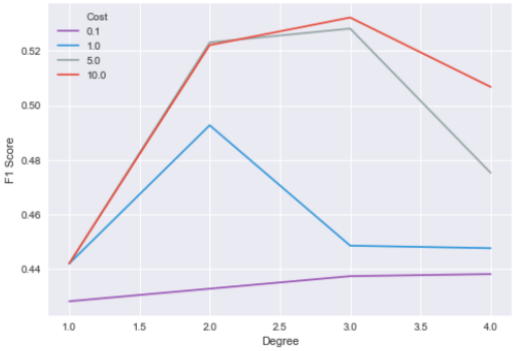
Parameters:

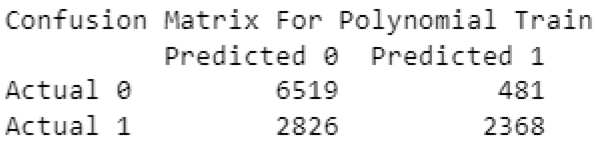
Polynomial Seperation of the data points is experimented with multiple Regularization Parameter and multiple Degree of the polynomial to find the the combination of the best parameter.
Cross Validation is implemented to find the best results which were found with C =20 & Degree of the Polynomial 3

Train and Test:

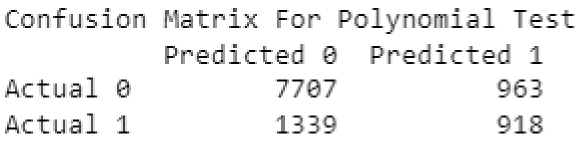
With the best parameters for polynomial separation, SVM is fit and predicted on train data without Cross Validation to compare the Accuracy metric and have a base line to check with the test data.

We can observe an f-1 Score of 59.7% on the train data and 44% on test data which is not a significant improvement from the linearly separated model. Hinting that the data may be explained better by neither of them.





```
Confusion Matrix For Polynomial Train
           Predicted 0  Predicted 1
Actual 0          6519          481
Actual 1          2826         2368
```

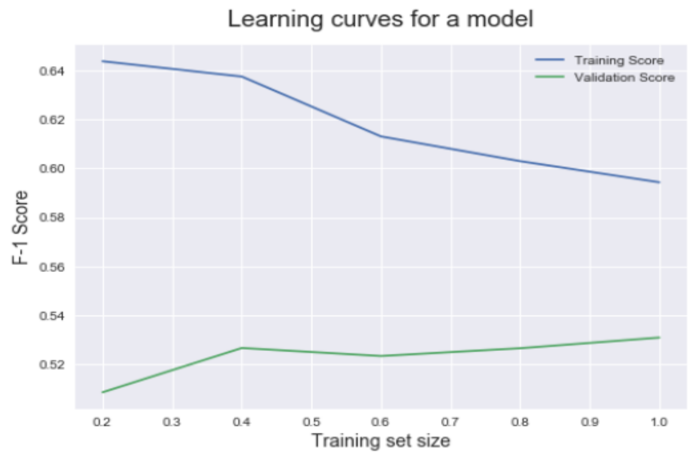


```
Confusion Matrix For Polynomial Test
           Predicted 0  Predicted 1
Actual 0          7707          963
Actual 1          1339          918
```

Learning Curve:

For the model we plot a training curve which results in a gradual decrease in accuracy with the train data but a simultaneous gradual increase in the validation data with increase of influx training data.

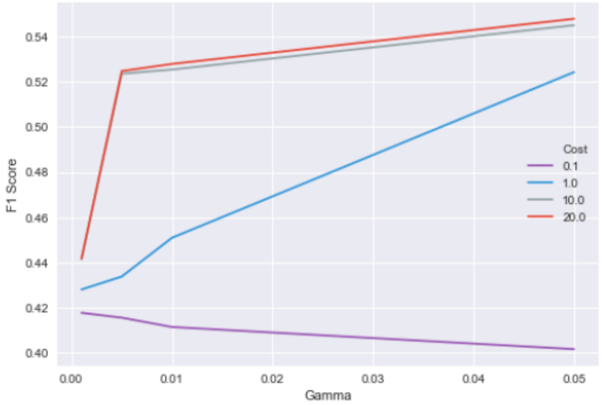
As learning curve implements cross validation, we can conclude that the model is learning better with increase in data and but is not generalizing well with 59.4% on train and 53.4 % on validation data.



Support Vector Machine (Radial):

Parameters:

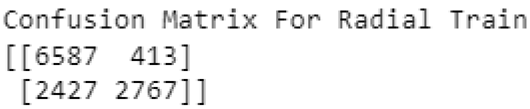
Non- Linear Seperation of the data points is experimented with the combination of multiple Regularization Parameter and multiple Gamma values which with increase in its value data points closer to the seperation line are used and vice versa.
Cross Validation is implemented to find the best best parameters on the metric of accuracy which were found with **C =20 & Gamma = 0.05**



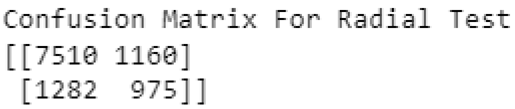
Train and Test Data:

With the best parameters for non-linear separation, SVM is fit and predicted on train data without Cross Validation to compare the Accuracy metric and have a base line to check with the test data.

We can observe an **F-1 Score of 66.3% on the train data**, and 44% on test which is the same from the polynomial separated model or Linear kernel. Hinting that the data may be **explained better by neither of the three kernels**



```
Confusion Matrix For Radial Train
[[6587  413]
 [2427 2767]]
```



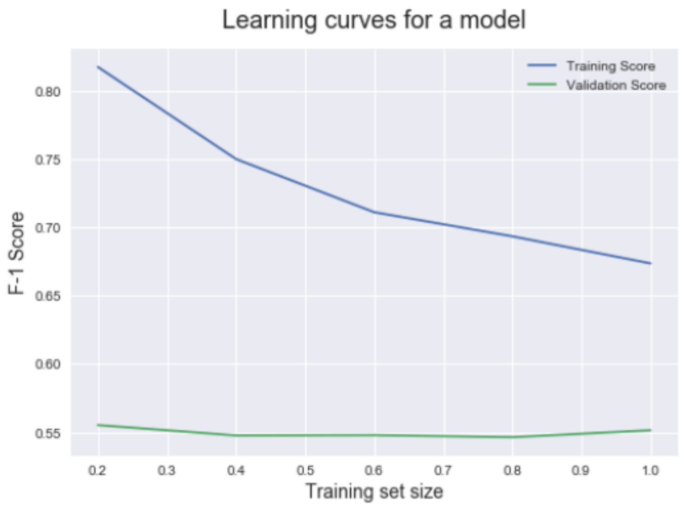
```
Confusion Matrix For Radial Test
[[7510 1160]
 [1282  975]]
```

Learning Curve:

With increase of influx of data there is a gradual decrease in accuracy with the train data but a simultaneous and significant increase in the validation data.

As learning curve implements cross validation, we can conclude that the model is learning better than any other Kernels of SVM.

With increase in data the model converges andbut does not generalizing well with an **59.43 %** on train and **53%** on validation data.



Decision Tree:

Parameters:

The feature selection is not significant in case of Decision Trees as Entropy is calculated for each feature and thus selects the best ones to split on
The 2 sets of Parameters which were experimented on are:

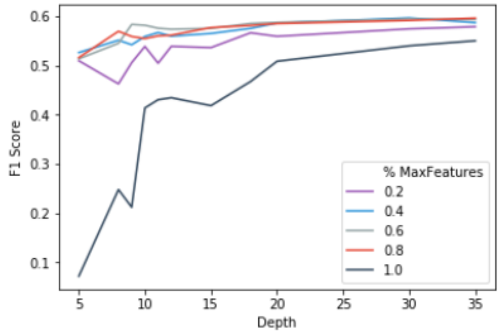
- Depth and Maximum number of Leaf Nodes
- Depth and Percentage of Maximum Features

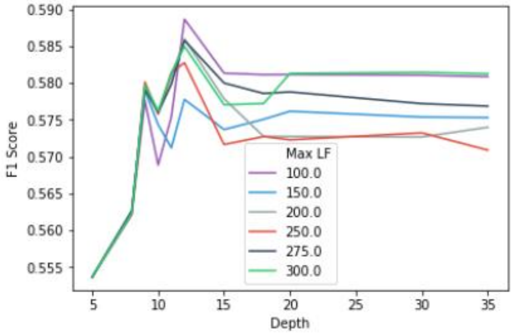
Cross Validation is implemented to find the best parameters.

In the first set of parameters the Depth = 15 and Maximum number of Leaf nodes = 18

In the Second Set of parameters the Depth =18 and Percentage of Maximum Features = 0.8

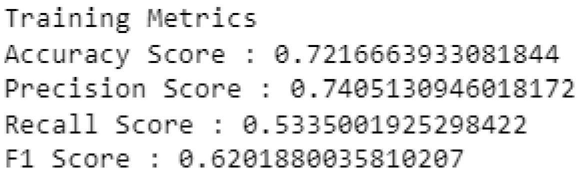
A combination of these parameter is used to find the best decision tree.



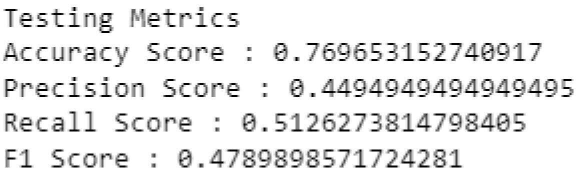


Train and Test:

With the best parameters of a decision tree is done without Cross Validation to compare the metric and have a base line to check with the test data. As shown below in the metric, we can observe an F-1 Score of 62.01% on the train and 47% on the Test which when compared is better than all svm kernels but is still not a model which is of good use.

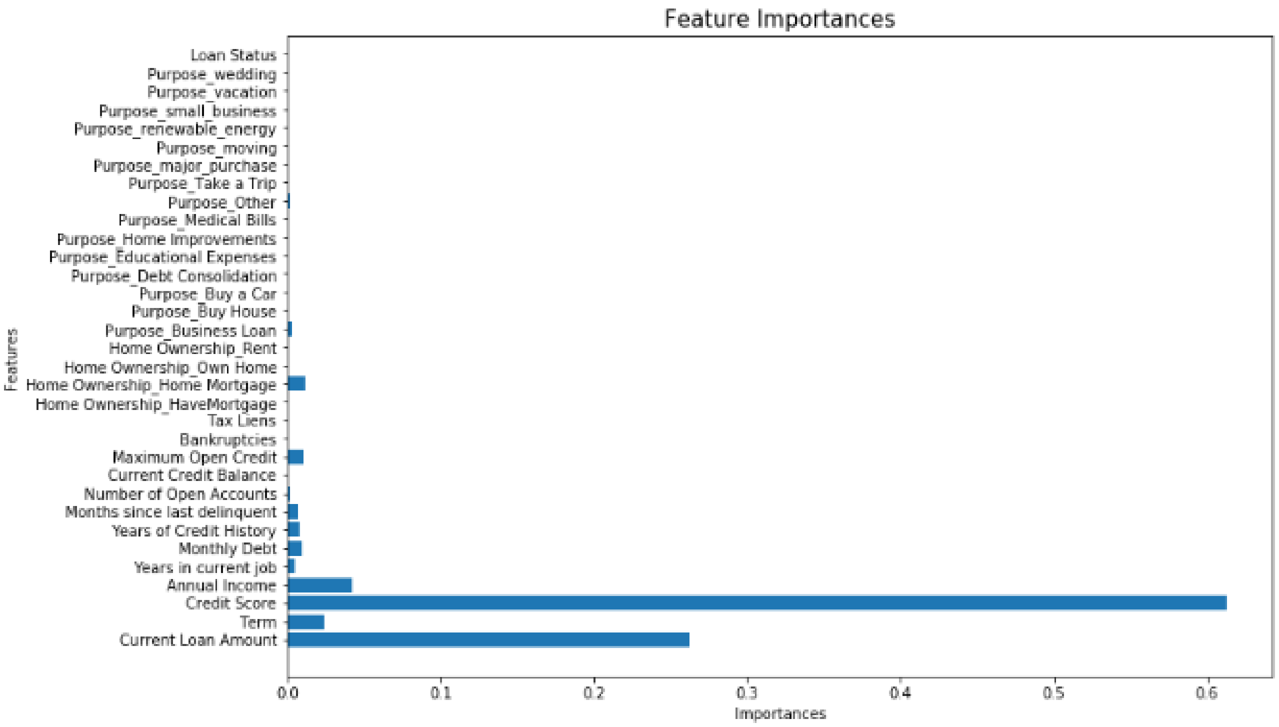


```
Training Metrics
Accuracy Score : 0.7216663933081844
Precision Score : 0.7405130946018172
Recall Score : 0.5335001925298422
F1 Score : 0.6201880035810207
```



```
Testing Metrics
Accuracy Score : 0.769653152740917
Precision Score : 0.4494949494949495
Recall Score : 0.5126273814798405
F1 Score : 0.4789898571724281
```

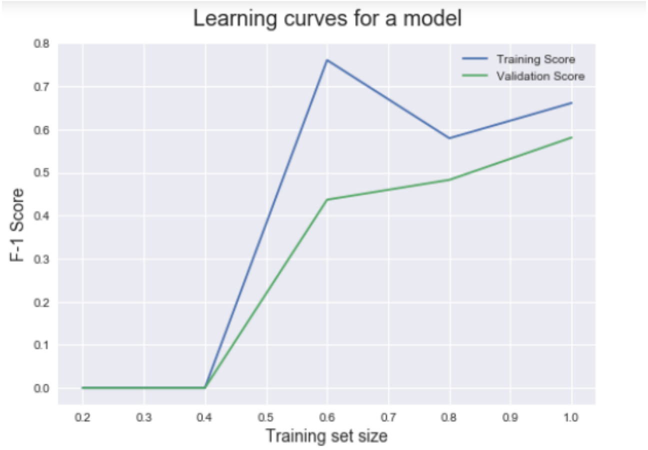
Feature Importance : Credit Score , Term, Current Loan Amount are the most important features.



Learning Curve:

As the data influx increases the train and test scores converge, which implies that the model is generalizing well.

The final Training and Validation accuracy metrics are 66% and 58% respectively.



Boosting:

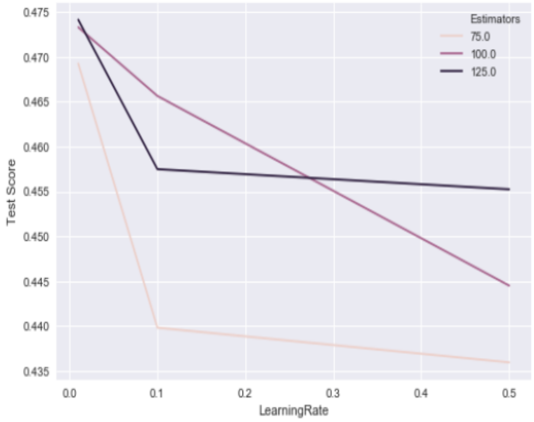
Parameters:

The parameters experimented with are the learning rate and estimators.

Estimators are the weak learns we need to boost and improve the accuracy of the required classification.

The best parameters are Estimators: 125

And Learning Rate: 0.01

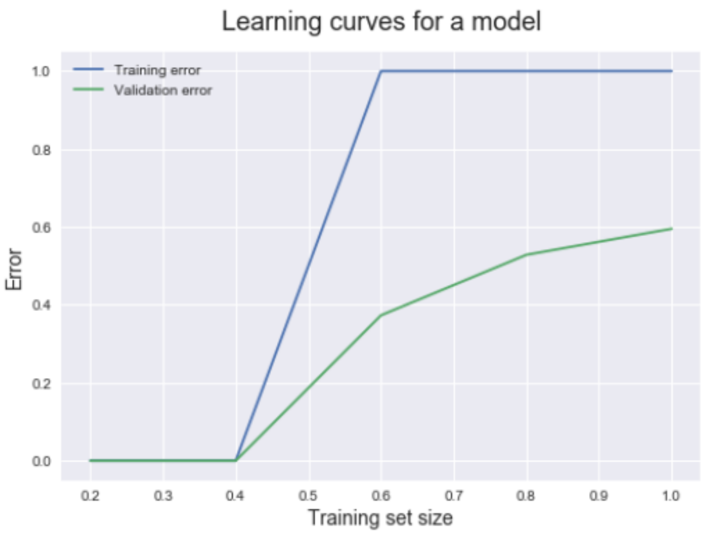


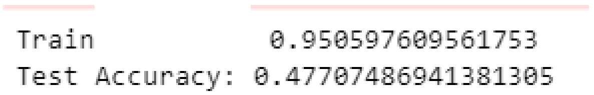
Learning Curve, Train and Test:

With increase in the influx of training data the test score is 59% and the validation increases but shows a very low accuracy score as discussed in all the above cases, the data is inconsistent along with very low target variable

Thus, there may a point of overfitting the data.

The below measures also show the same without cross validation train score is 95% and test is 47%





```
Train          0.950597609561753
Test Accuracy: 0.47707486941381305
```

Comparison of The Different Models:

SVM	Decision Tree	Ensemble Method(Boosting)
Does not change predict the accurate target classification but the accuracy is at 73%	The Decision tree is the fastest of all the algorithms when performed and does an easy job in classification and has an accuracy better than some kernels of SVM.	Does not change much from Decision tree thus boosting is pointless to this dataset.
F-1 Score:44%	F1-Score:47%	F-1 Score:47%

What can be done?
There may be a lot of changes like changing the continuous features to labels and over sample the data.