GROUP 04

PRESENTATION ON CHURN PREDCITION PROBLEM

MACHINE LEARNING TECHNIQUES

WHAT IS THE DATA ABOUT!

0	customer_id
1	vintage
2	age
3	gender
4	dependents
5	occupation
6	city
7	customer_nw_category
8	branch_code
9	days_since_last_transaction
10	current_balance

11	previous_month_end_balance
12	average_monthly_balance_prevQ
13	average_monthly_balance_prevQ2
14	current_month_credit
15	previous_month_credit
16	current_month_debit
17	previous_month_debit
18	current_month_balance
19	previous_month_balance
20	churn

Cleaned the data, dropped null values

PREPROCESSING OF THE DATA

Target variable

```
y = churn['churn']
```

Only two classes- Binary Classification

```
y.unique()
```

```
array([0, 1], dtype=int64)
```

Features

```
category_cols=['customer_id', 'vintage', 'age', 'gender', 'dependents', 'occupation',
       'city', 'customer_nw_category', 'branch_code',
       'days_since_last_transaction', 'current_balance',
       'previous_month_end_balance', 'average_monthly_balance_prevQ',
       'average_monthly_balance_prevQ2', 'current_month_credit',
       'previous_month_credit', 'current_month_debit', 'previous_month_debit',
       'current_month_balance', 'previous_month_balance']
```

Converted categorical features into numeric, added constant, splitted the data into train and test

BUILDING AND DIAGNOSING THE MODELS

Fitting the model using 1st feature set

```
logR_1=sm.Logit(y_train_1,X_train_1)

# Fit the model
logR_1=logR_1.fit()
```

```
Optimization terminated successfully.
         Current function value: 0.424048
         Iterations 8
```

From summary of 1st feature set

Model:	Logit	Pseudo R-squared:	0.153

Features with p<0.05

	Coef.	Std.Err.	z	P>|z|
current_balance	-0.0004	0.0001	-5.2830	0.0000
average_monthly_balance_prevQ	0.0002	0.0001	2.6857	0.0072

Fitting the model using 2nd feature set

```
logR_2= sm.Logit(y_train_2,X_train_2)
logR_2=logR_2.fit()
logR_2.summary2()
```

```
Optimization terminated successfully.
         Current function value: 0.479419
         Iterations 7
```

From summary of 2nd feature set

Model:	Logit	Pseudo R-squared:	0.042

	Coef.	Std.Err.	z	P>|z|
average_monthly_balance_prevQ	0.0002	0.0000	5.0051	0.0000
current_balance	-0.0005	0.0000	-10.0018	0.0000

CHECKED THE VIF VALUES AND FINALISED THE MODEL!

```
var_inf_factor(X_2)
```

	Feature	VIF_Value
0	average_monthly_balance_prevQ	4.989556
1	current_balance	4.989556

Both the VIFs >4.0

```
logR_3.params
```

```
average_monthly_balance_prevQ    0.000172
current_balance                 -0.000484
dtype: float64
```

The final LR model:

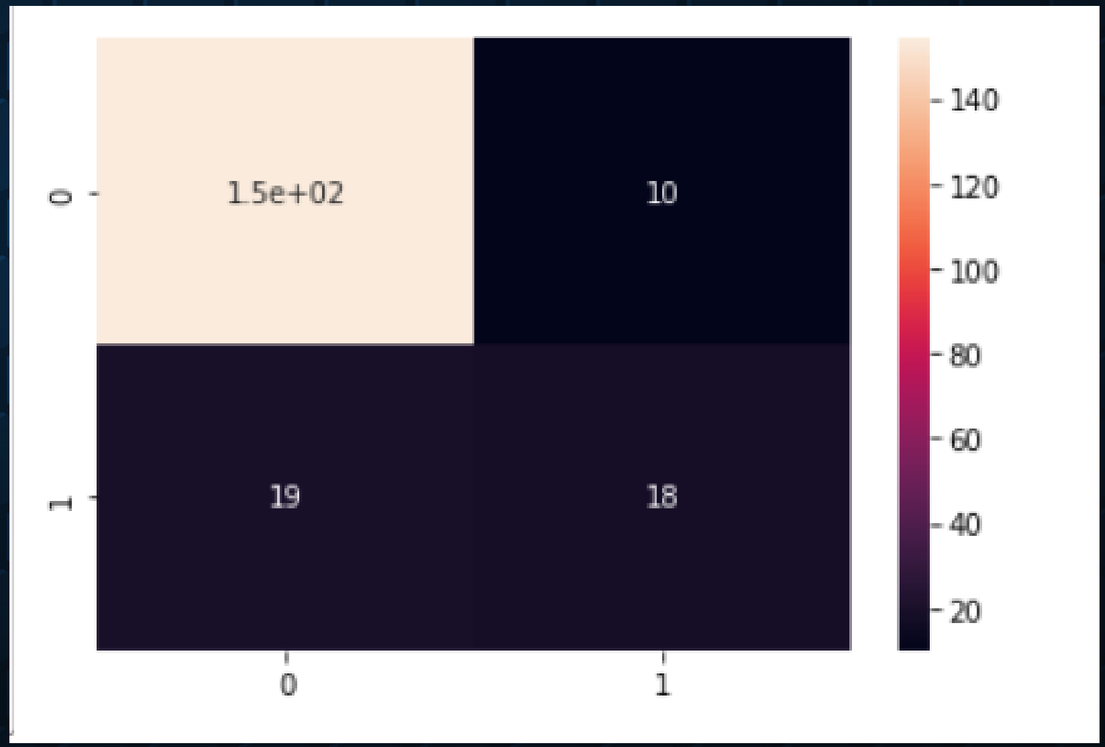
Z = average_monthly_balance_prevQ * 0.000172 + current_balance * -0.000484

PREDICTION USING THE MODEL

	Actual Class	Predicted Prob.
1099	0	0.016492
1123	0	0.127642
1264	0	0.001696
224	1	0.337550
1228	0	0.362362
...	...	...
247	0	0.239615
510	0	0.379636
1144	0	0.400206
1024	0	0.069019
243	0	0.004455

201 rows × 2 columns

Confusion matrix



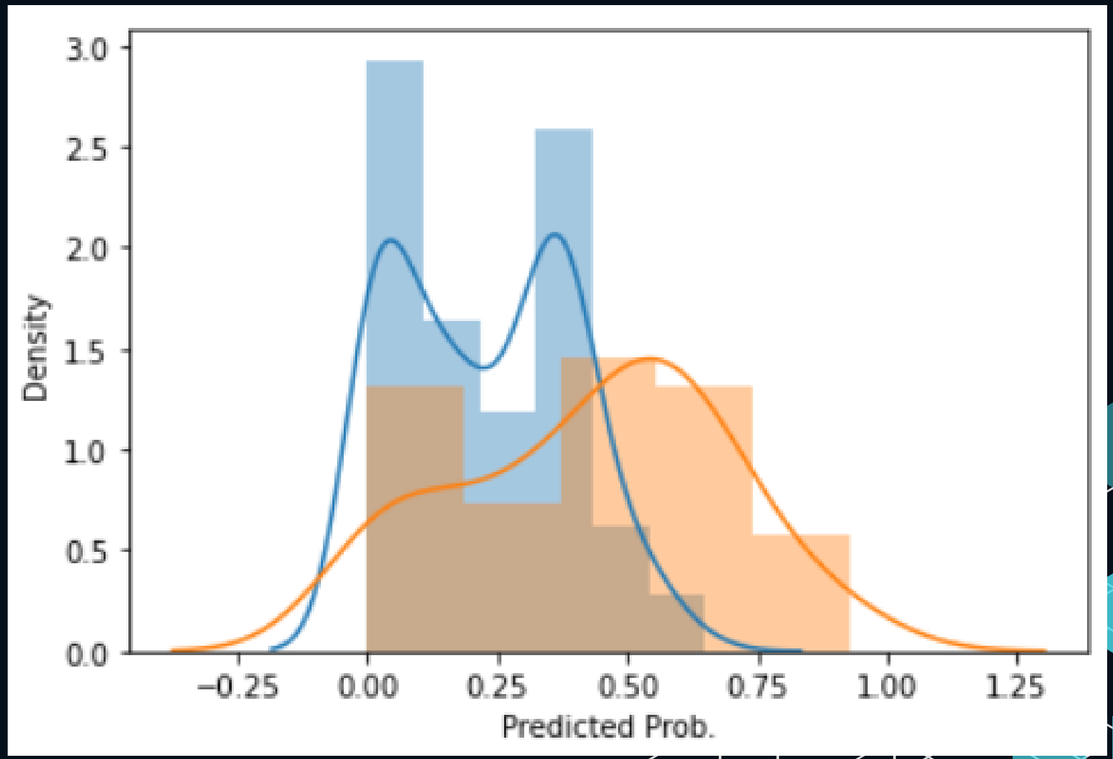
CLASSIFICATION REPORT & PLOTTING THE DISTRIBUTION

Classification report

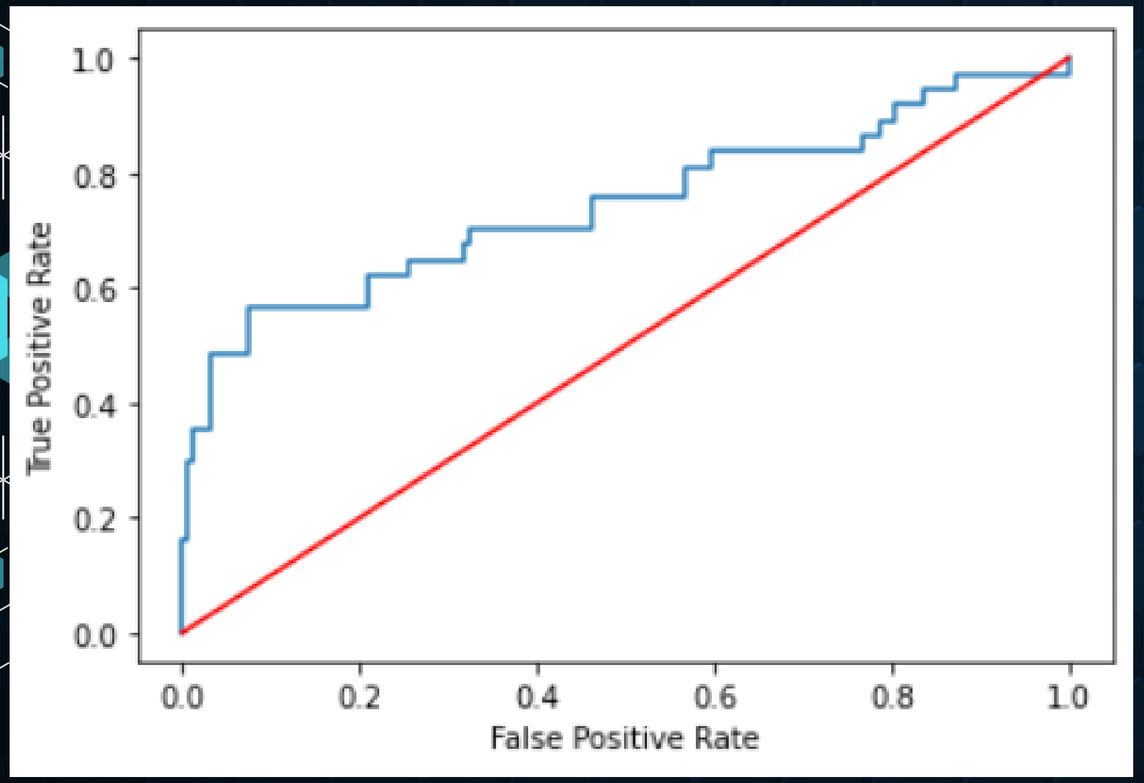
The classification report:

	precision	recall	f1-score
0	0.89	0.94	0.91
1	0.64	0.49	0.55
accuracy			0.86
macro avg	0.77	0.71	0.73
weighted avg	0.84	0.86	0.85

Distribution of pred prob corresponding class=0,1

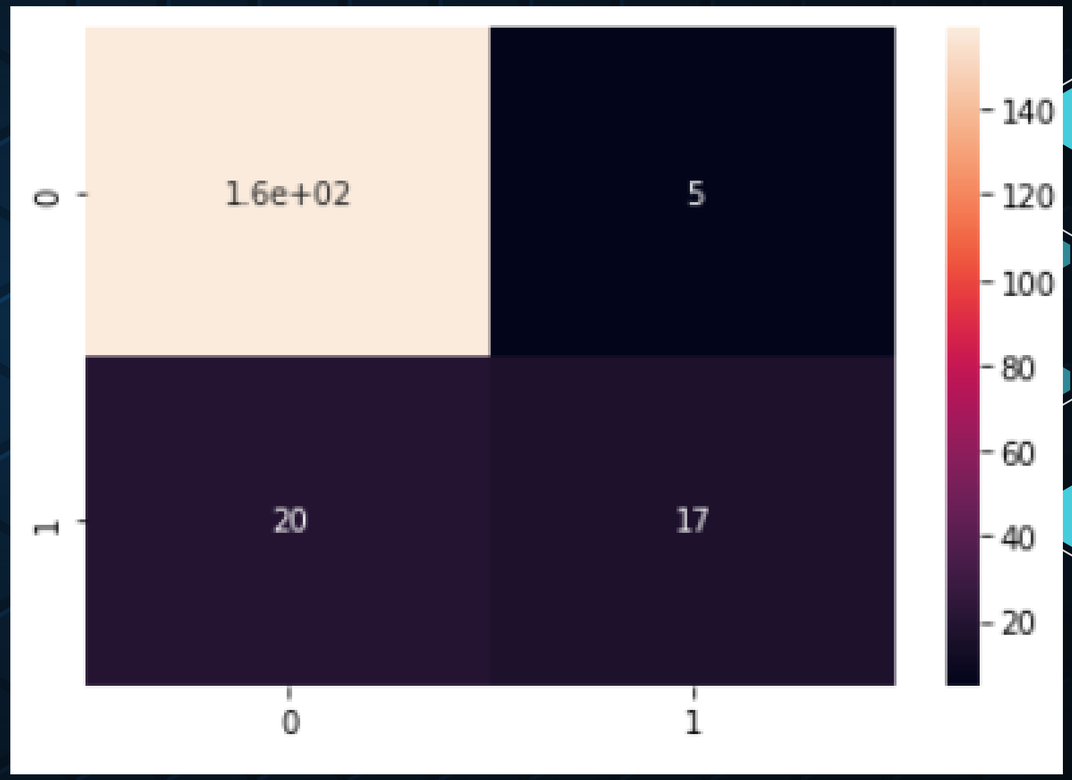


ROC CURVE



The ROC-AUC-Score: 0.7127554383651944

New confusion matrix



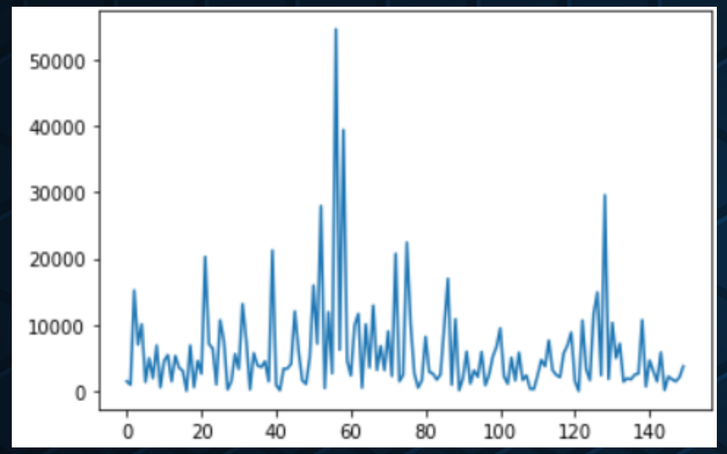
The new score: 0.7144858272907053

DECISION TREE CLASSIFIER

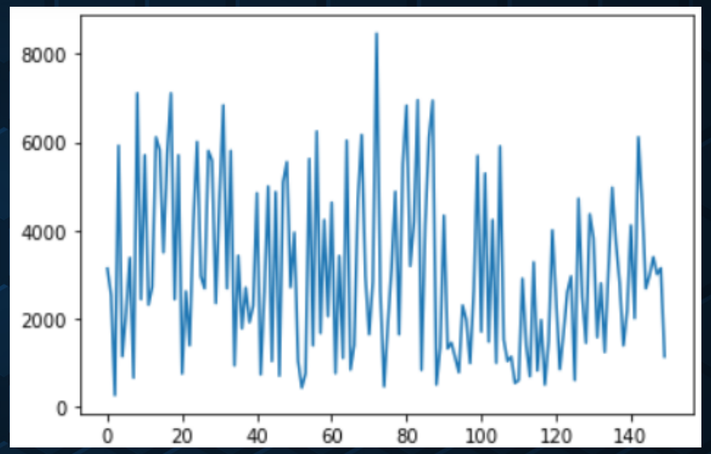
Considering following independent variables:
current_balance, days_since_last_transaction, vintage, current_month_credit

VISUALISATION OF DISTRIBUTION OF THESE 4 VARIABLES

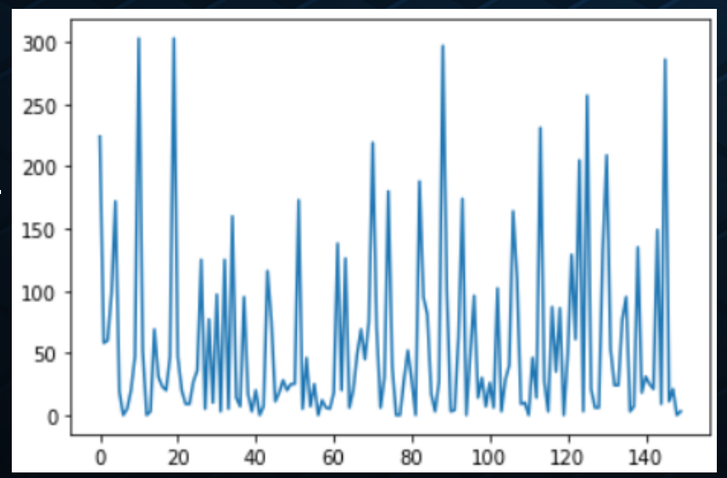
current_balance



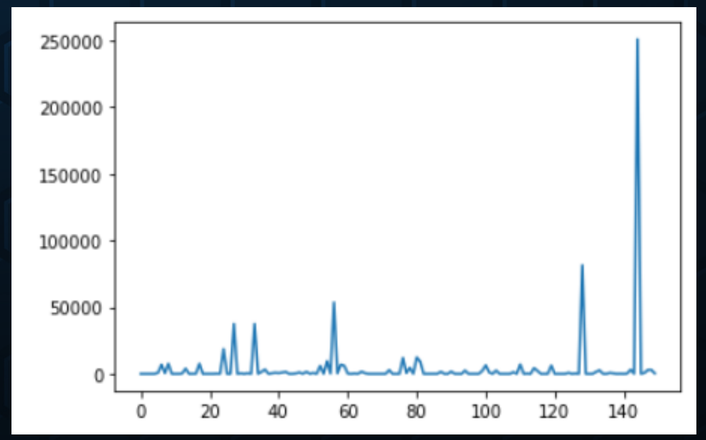
vintage



days_since_last_transaction

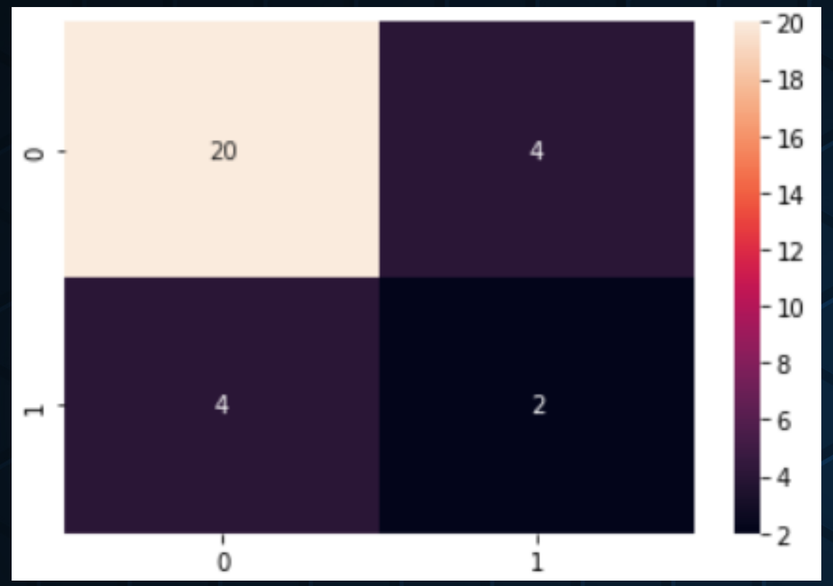


current_month_credit

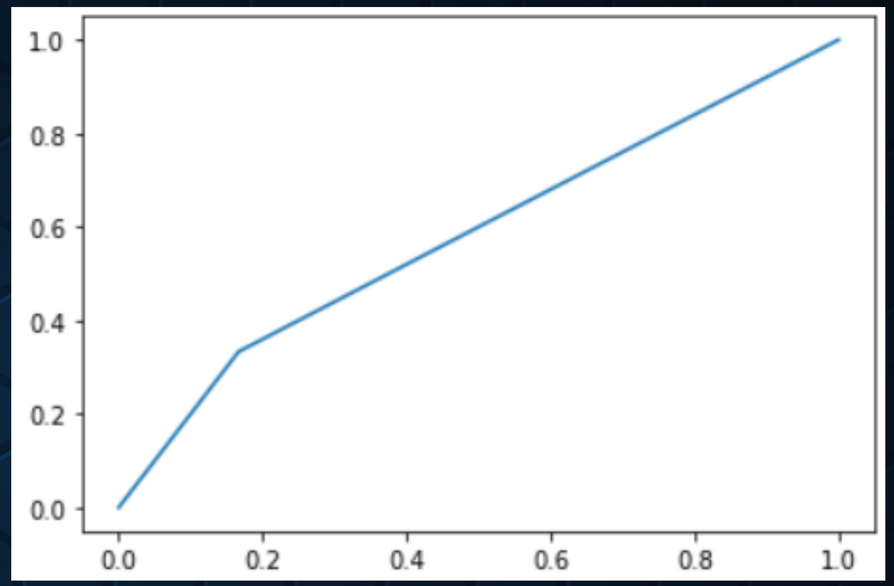


A GLANCE AT PERFORMANCE MEASURES

Confusion matrix



ROC Curve

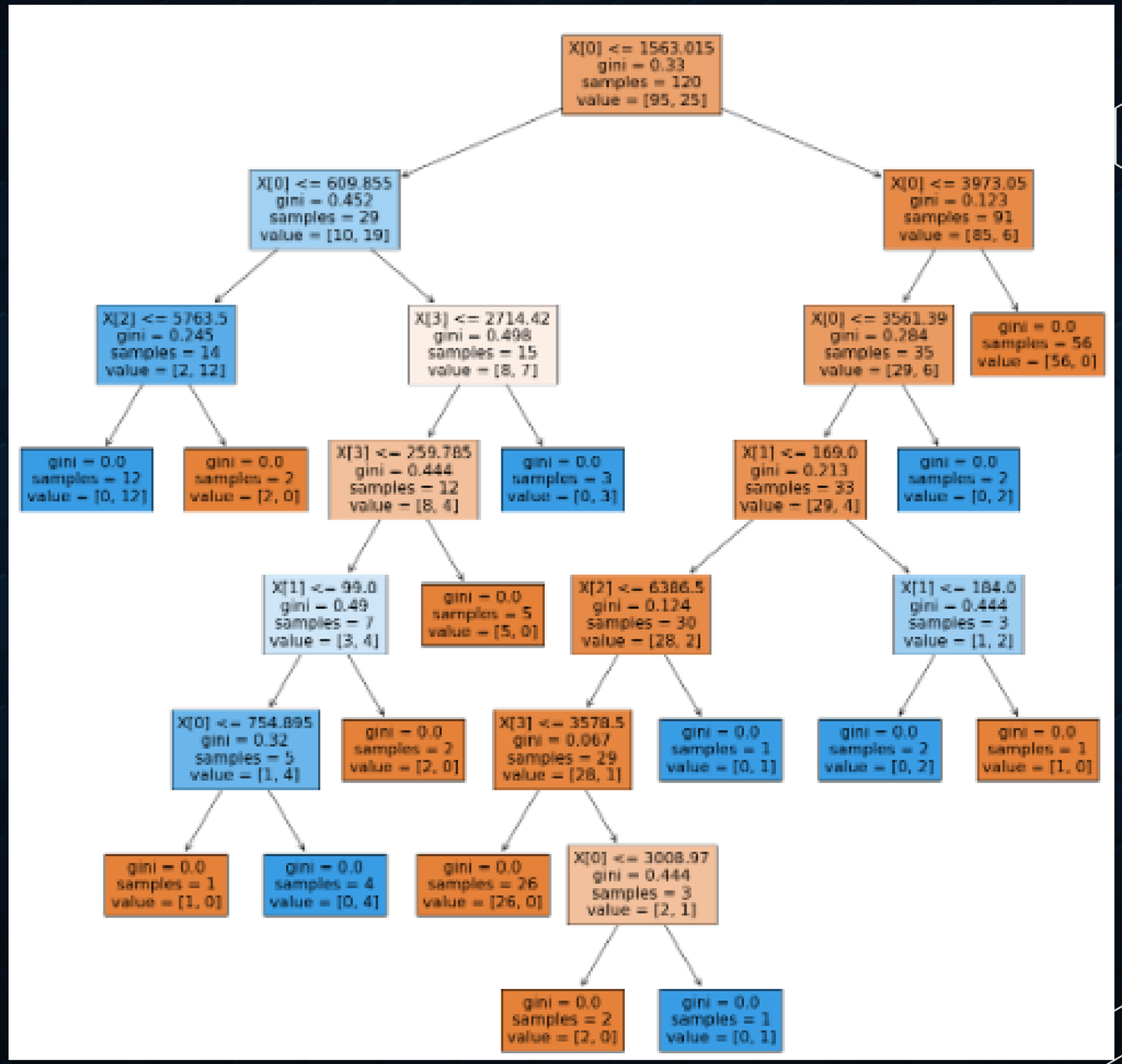


Classification report

The classification report:

	precision	recall	f1-score
0	0.83	0.83	0.83
1	0.33	0.33	0.33
accuracy			0.73
macro avg	0.58	0.58	0.58
weighted avg	0.73	0.73	0.73

VISUALISATION OF THE TREE



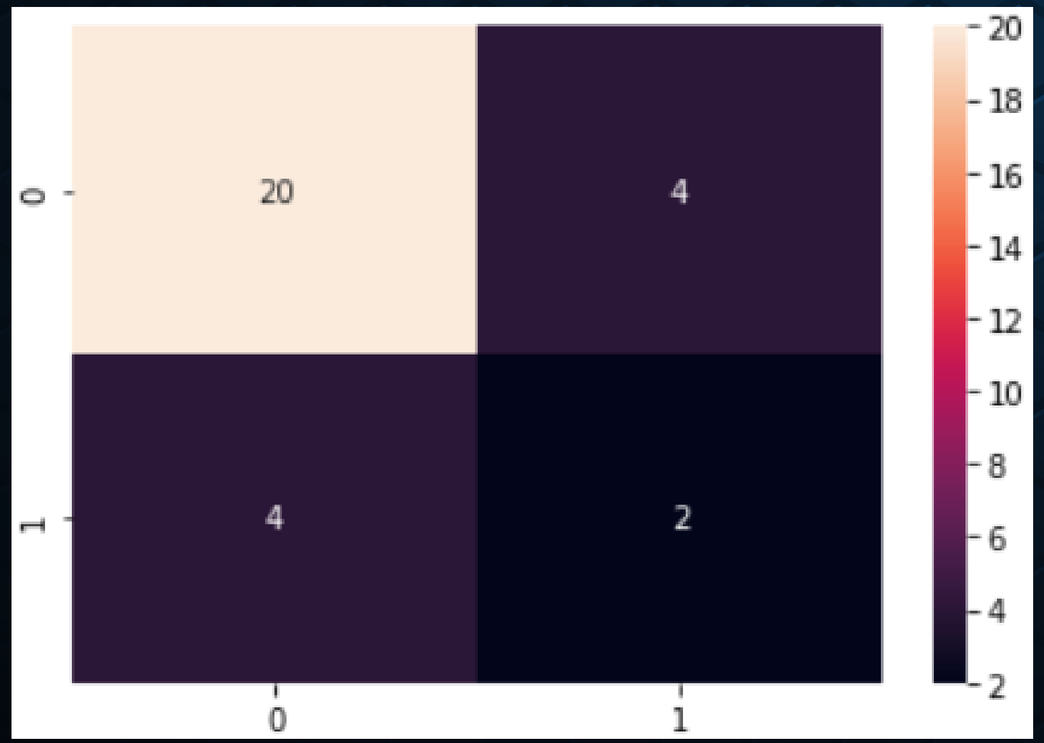
GINI INDEX AND OPTIMIZED DECION TREE

Calculating gini value:

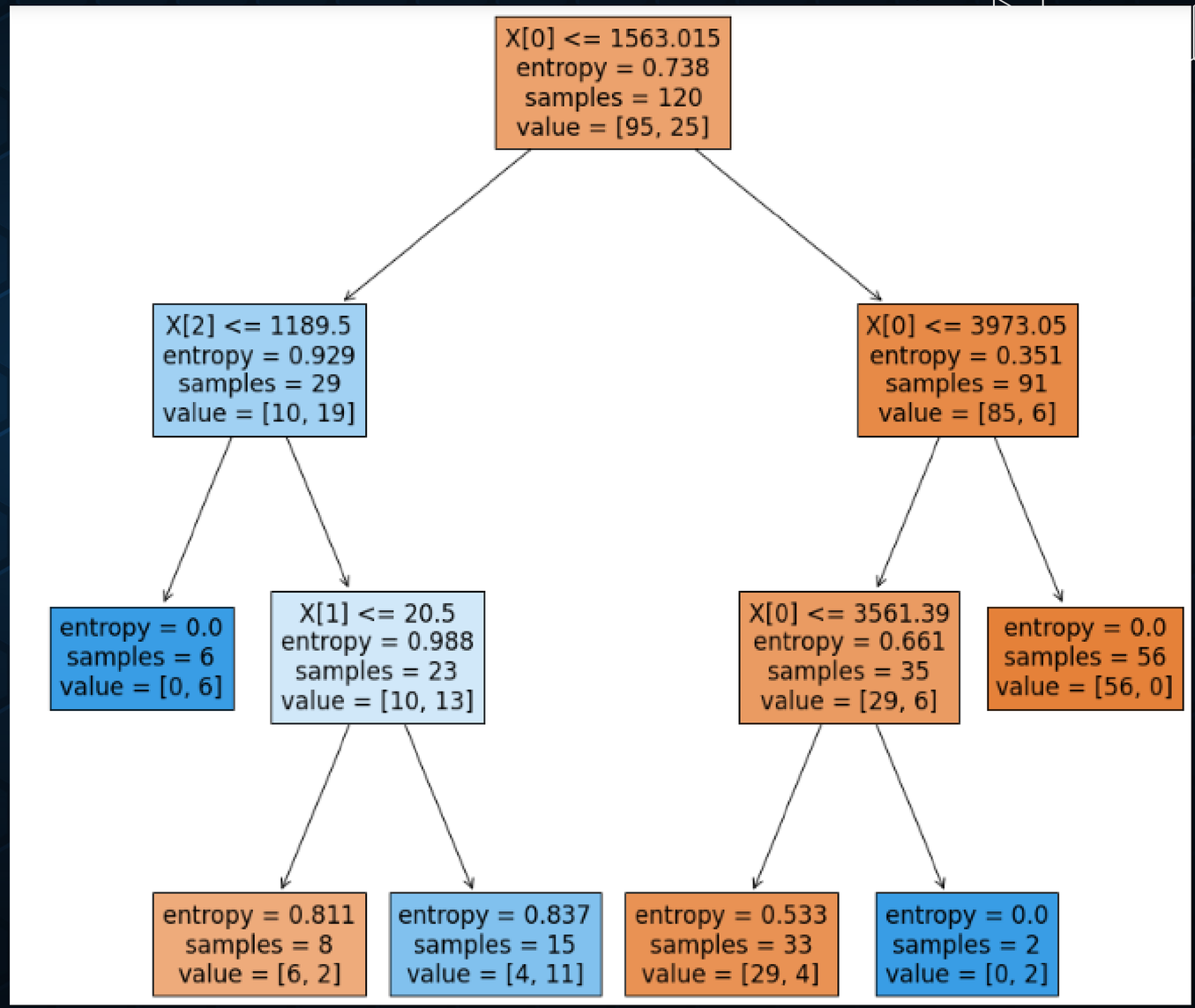
gini= 1- (95/120)**2 - (25/120)**2

```
Gini: 0.32986111111111116
```

Confusion matrix after modifying the model



Optimized Decision Tree



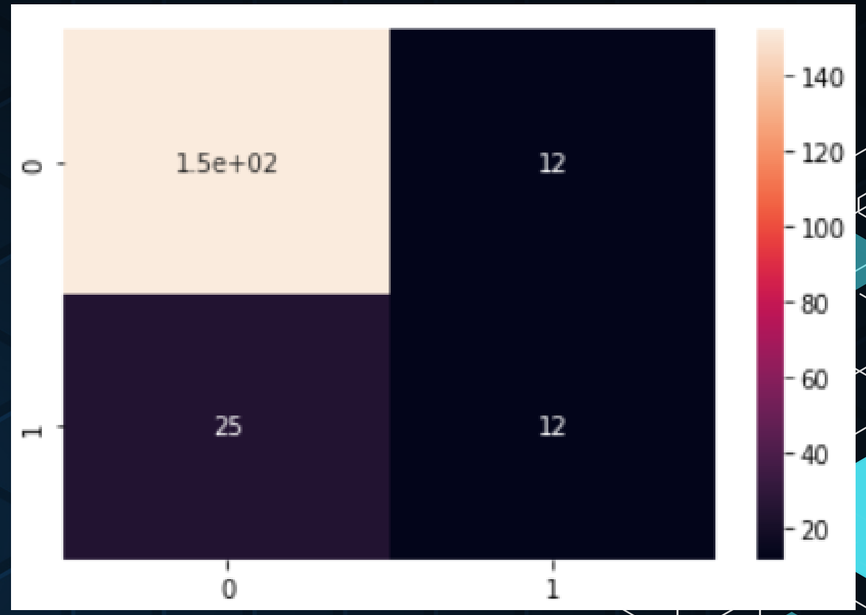
IMPLEMENTATION OF SVC

```
from sklearn.svm import SVC

svc_lin=SVC(kernel='linear',probability=True)
svc_lin=svc_lin.fit(X_train, y_train)
y_pred=svc_lin.predict(X_test)
y_pred
```

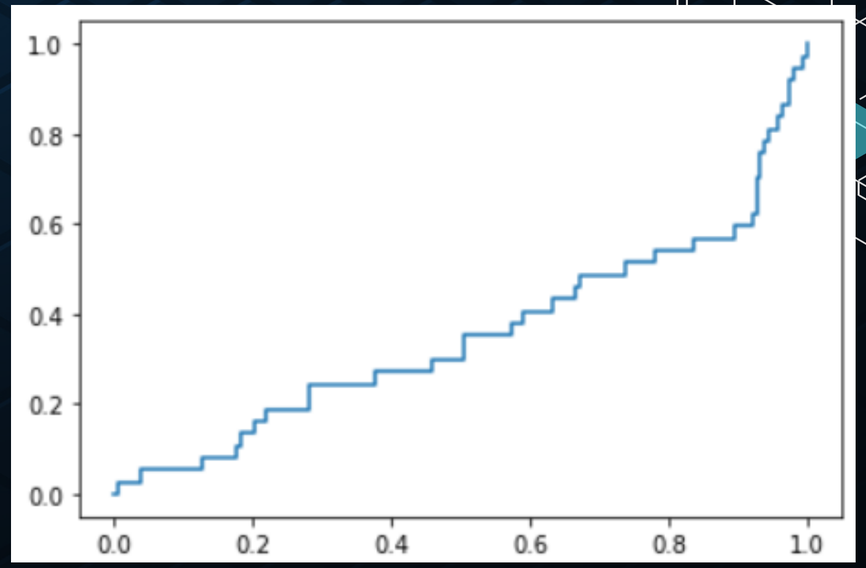
Checking the performance

```
The Report:
              precision    recall  f1-score

           0       0.86      0.93      0.89
           1       0.50      0.32      0.39

    accuracy                           0.82
   macro avg       0.68      0.63      0.64
weighted avg       0.79      0.82      0.80

The ROC-AUC-Score: 0.6255767963085036
```

Confusion matrix



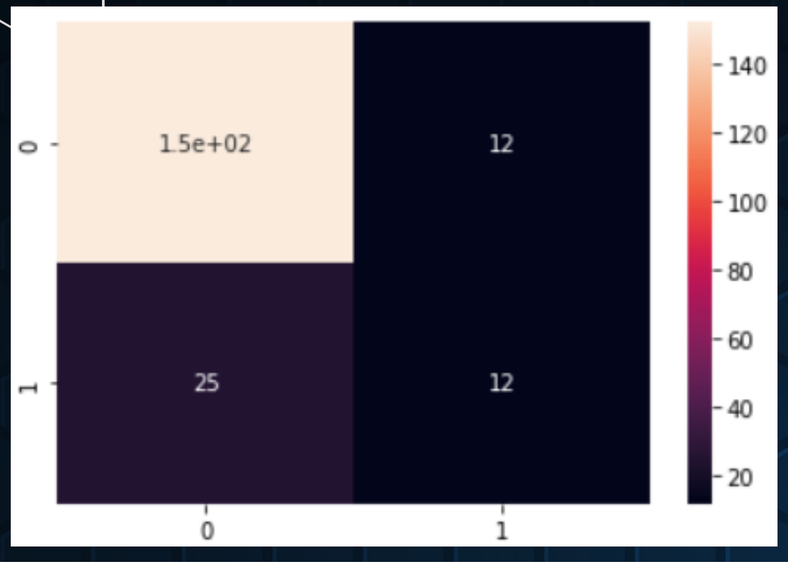
Plotted by taking:
FPR
(Sensitivity) =>
X- axis
TPR
(1-Specificity)
=> Y -axis



HYPER PARAMETER TUNING

LINEAR

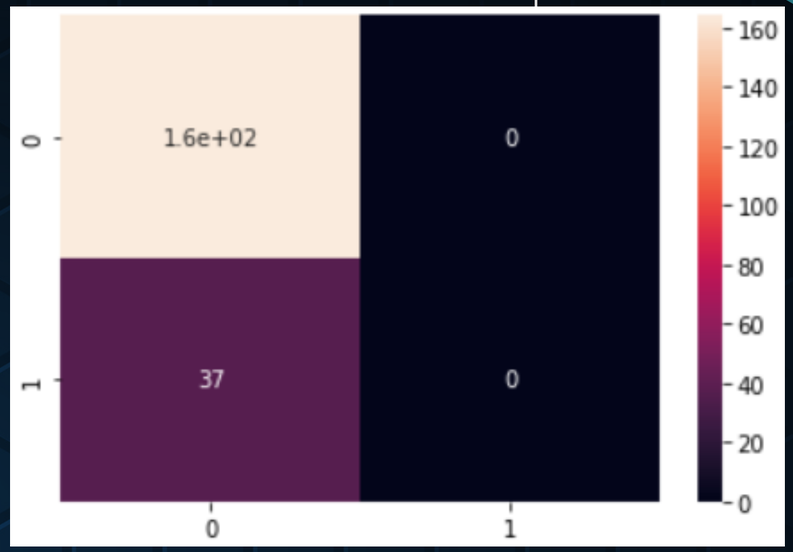
accuracy:
0.82



The ROC-AUC-Score: 0.6255767963085036

POLY

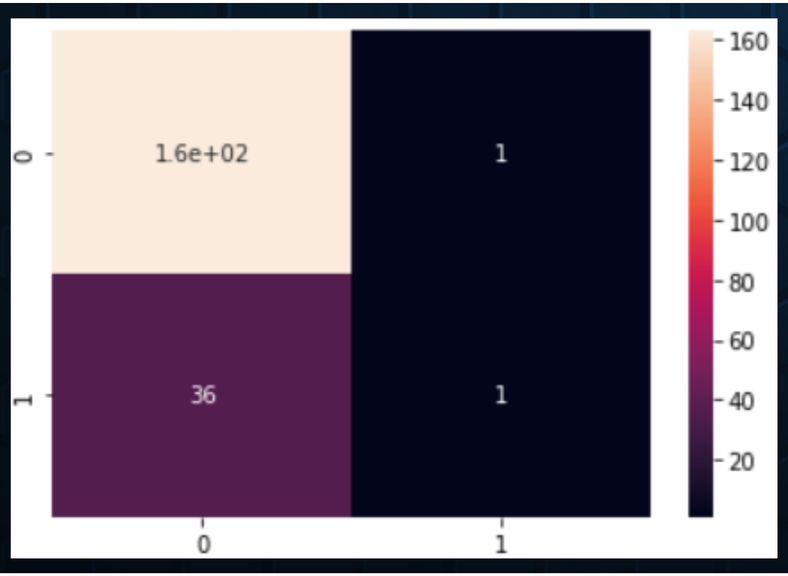
accuracy:
0.82



The ROC-AUC-Score: 0.5

RBF

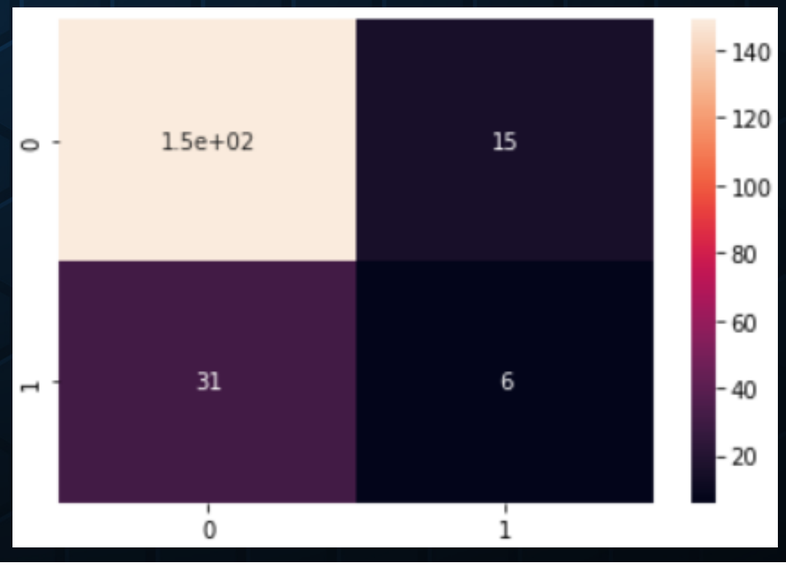
accuracy:
0.82



The ROC-AUC-Score: 0.5104647330257086

SIGMOID

accuracy:
0.77



The ROC-AUC-Score: 0.5353493737640079

TUNING REGULARISATION PARAMETER

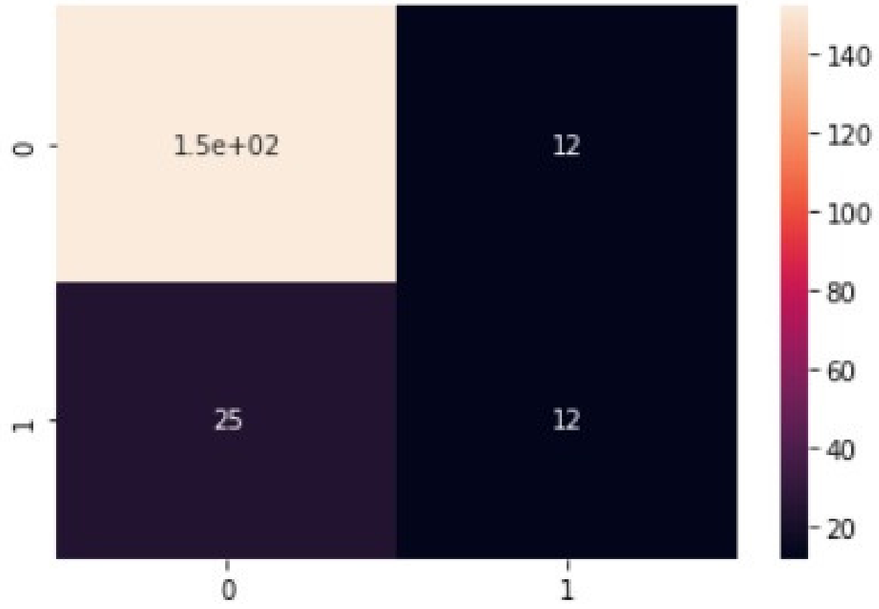
```
SVC_tuning_C(C_list)
```

```
C: 0.1 ===> Score: 0.5911338167435728
C: 1 ===> Score: 0.6255767963085036
C: 2 ===> Score: 0.6255767963085036
C: 3 ===> Score: 0.6255767963085036
C: 4 ===> Score: 0.6255767963085036
C: 5 ===> Score: 0.6255767963085036
C: 10 ===> Score: 0.6255767963085036
C: 15 ===> Score: 0.6120632827949902
C: 20 ===> Score: 0.6255767963085036
C: 25 ===> Score: 0.6255767963085036
C: 30 ===> Score: 0.6255767963085036
```

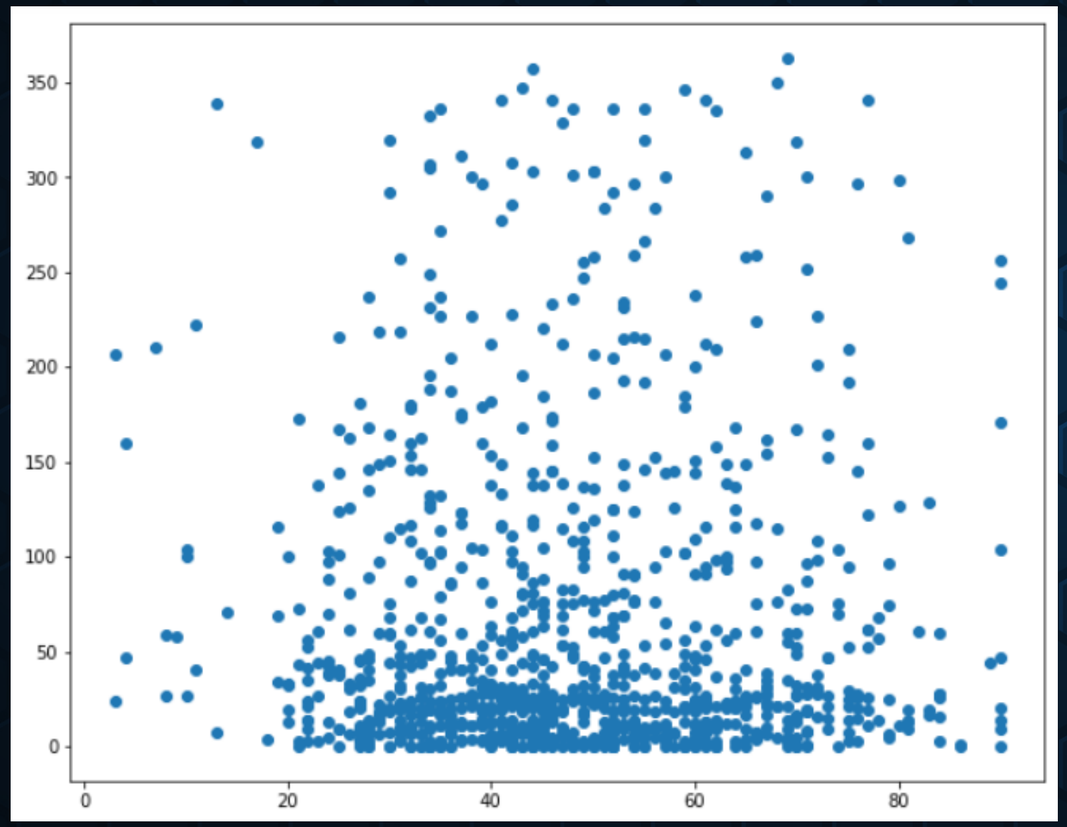
Afer tuning, the best value of C : 20

So, the best model is the one with kernel='linear' and C=20

```
The Confusion Matrix:
ROC-AUC-Score: 0.6255767963085036
 The report:                precision    recall  f1-score

           0       0.86      0.93      0.89       164
           1       0.50      0.32      0.39        37

    accuracy                           0.82       201
   macro avg       0.68      0.63      0.64       201
weighted avg       0.79      0.82      0.80       201
```



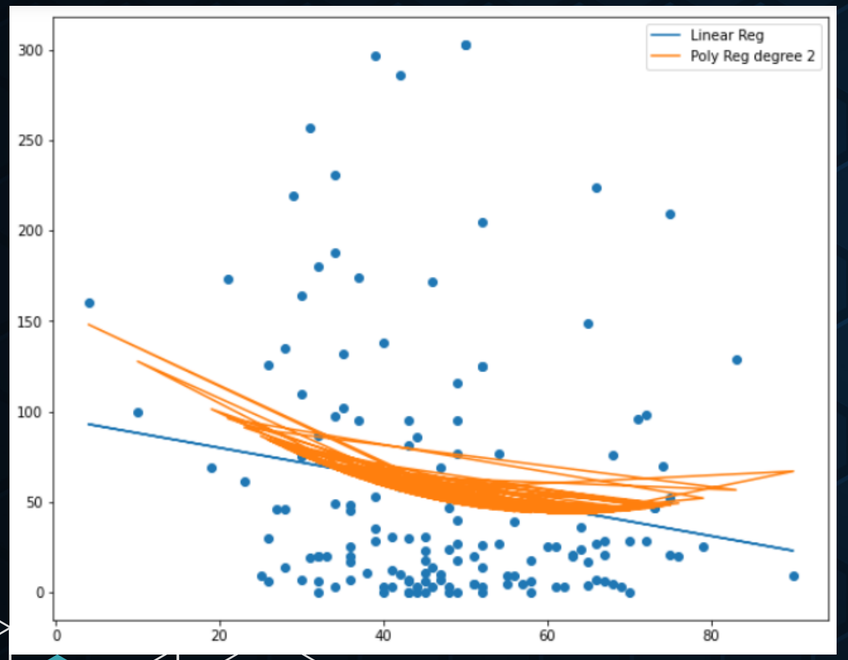
LINEAR REGRESSION



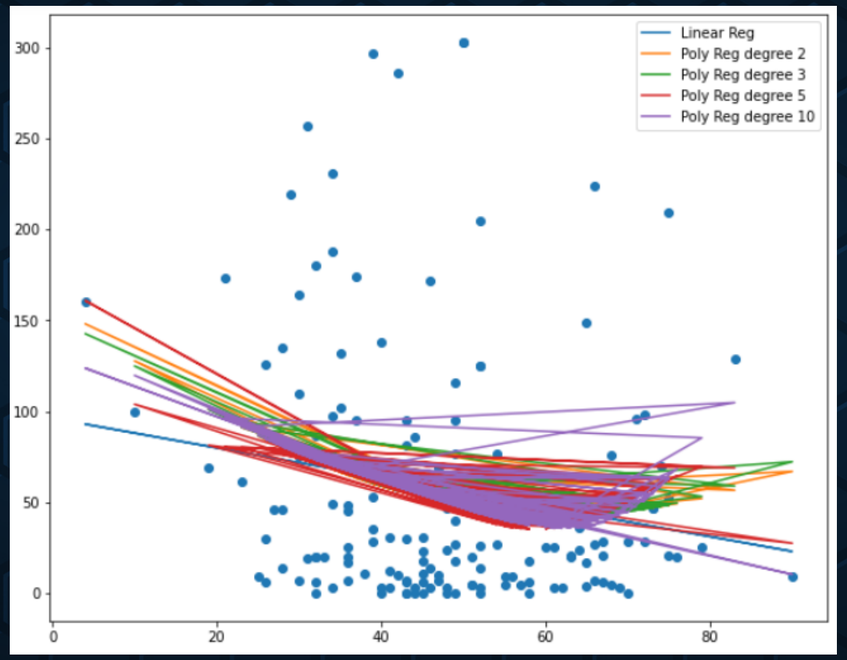
We could not arrive at any insights from this graph

POLYNOMIAL REGRESSION

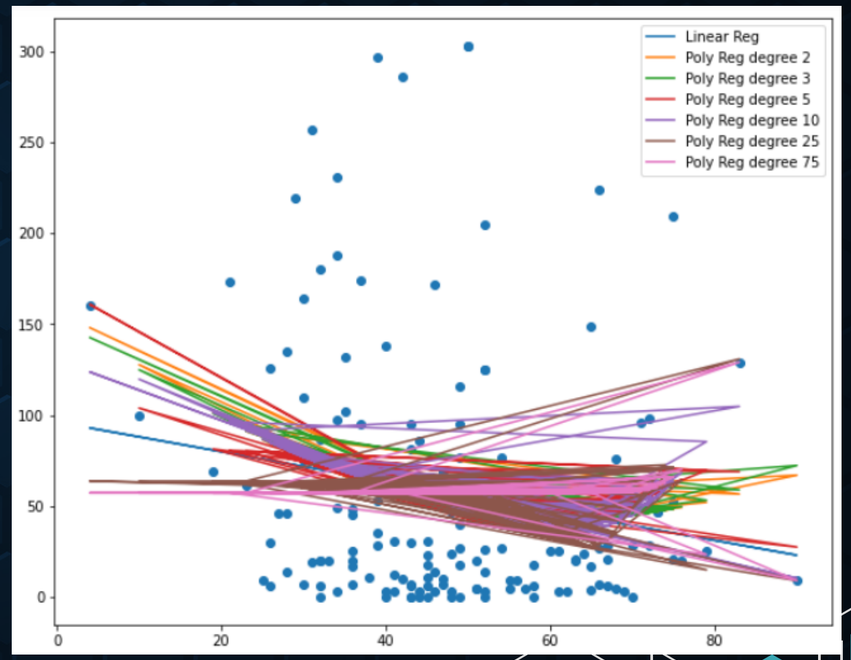
Degree 2



Degree 10



Degree 75



We could not arrive at any insights from these graphs

OVERFITTING AND RESOLVING IT

LASSO

```
R2 score: Lasso - Train 0.179673071629624
R2 score: Lasso - Test -0.02646467690051590 4
MSE: Lasso - Train 45537487.56024348
MSE: Lasso - Test 38800942.07038902
```

RIDGE

```
R2 score: ridge - Train 0.1796730740128922 3
R2 score: ridge - Test -0.0264647276321854 36
MSE: ridge - Train 45537487.427947074
MSE: ridge - Test 38800943.98807466
```

ELASTIC NET

```
R2 score: enet - Train 0.06602328236730137
R2 score: enet - Test -0.021578456717382455
MSE: enet - Train 51846345.26779231
MSE: enet - Test 38616240.19945722
```

Even after applying Lasso, Ridge and Elastic Net for regularisation, we were getting R^2 as low as 0.17. Hence, we can conclude that this model does not fit our data

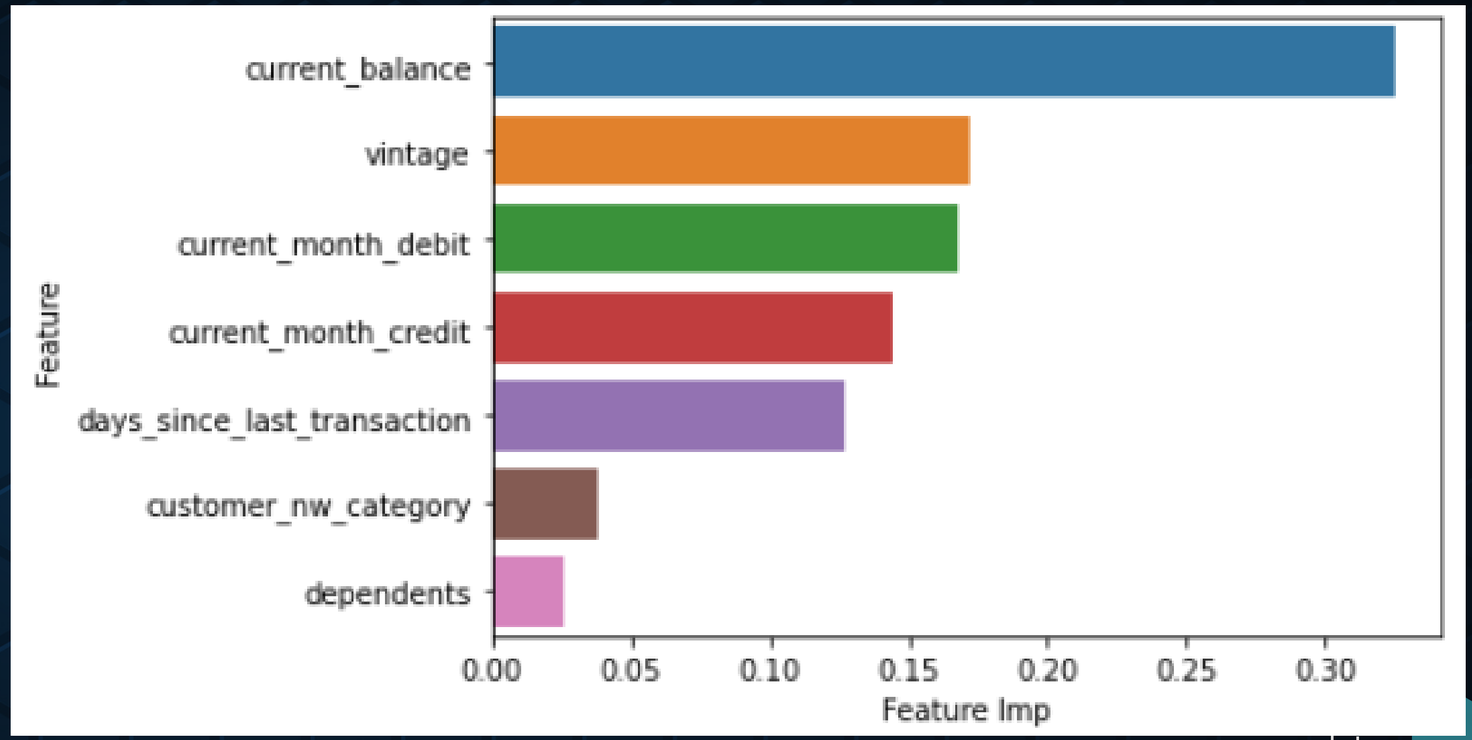
ENSEMBLE LEARNING

RANDOM FOREST CLASSIFIER

Accuracy: 0.83

HYPER PARAMETER TUNING USING GRIDSEARCHCV

Accuracy: 0.82



We could deduce that current balance is the most important feature, followed by vintage, followed by current month debit

BOOSTING

ADABOOST CLASSIFIER

Accuracy: 0.85

After improving the model using hyper parameter tuning, accuracy: 0.85

GRADIENT BOOSTING CLASSIFIER

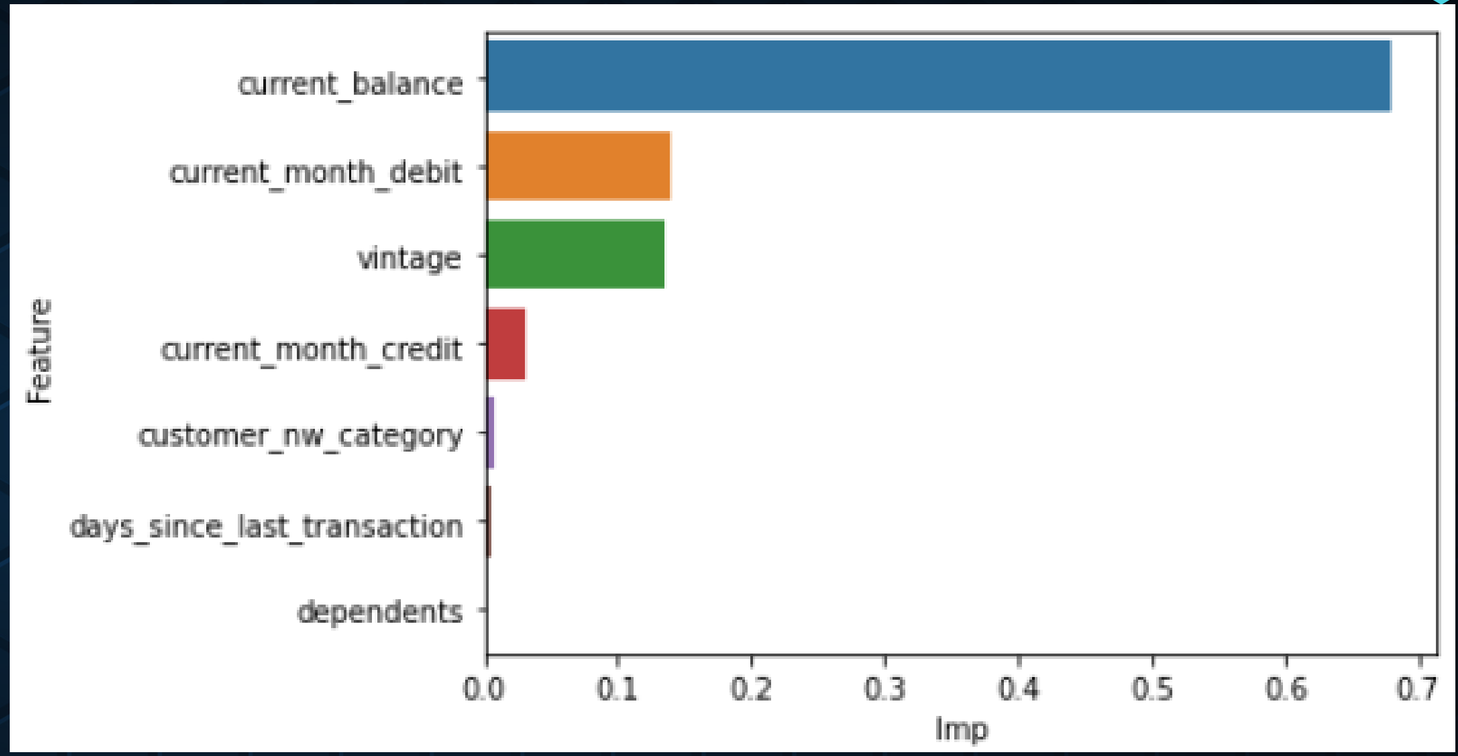
Accuracy: 0.84

After improving the model using hyper parameter tuning, accuracy: 0.85

XGBOOST CLASSIFIER

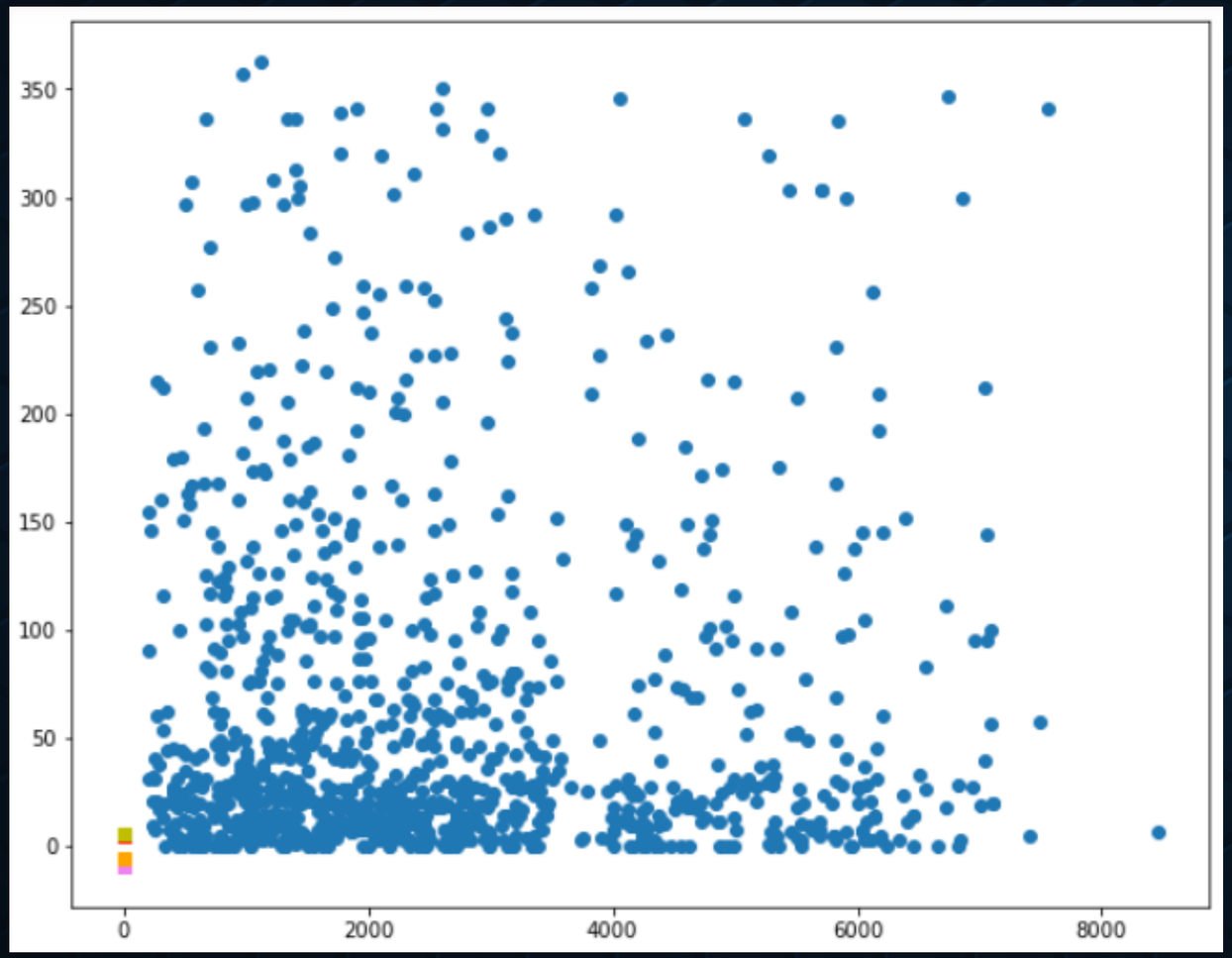
Accuracy: 0.82

After improving the model using hyper parameter tuning, accuracy: 0.85



We could deduce that current balance is the most important feature, followed by current month debit, followed by vintage

K-MEANS CLUSTERING

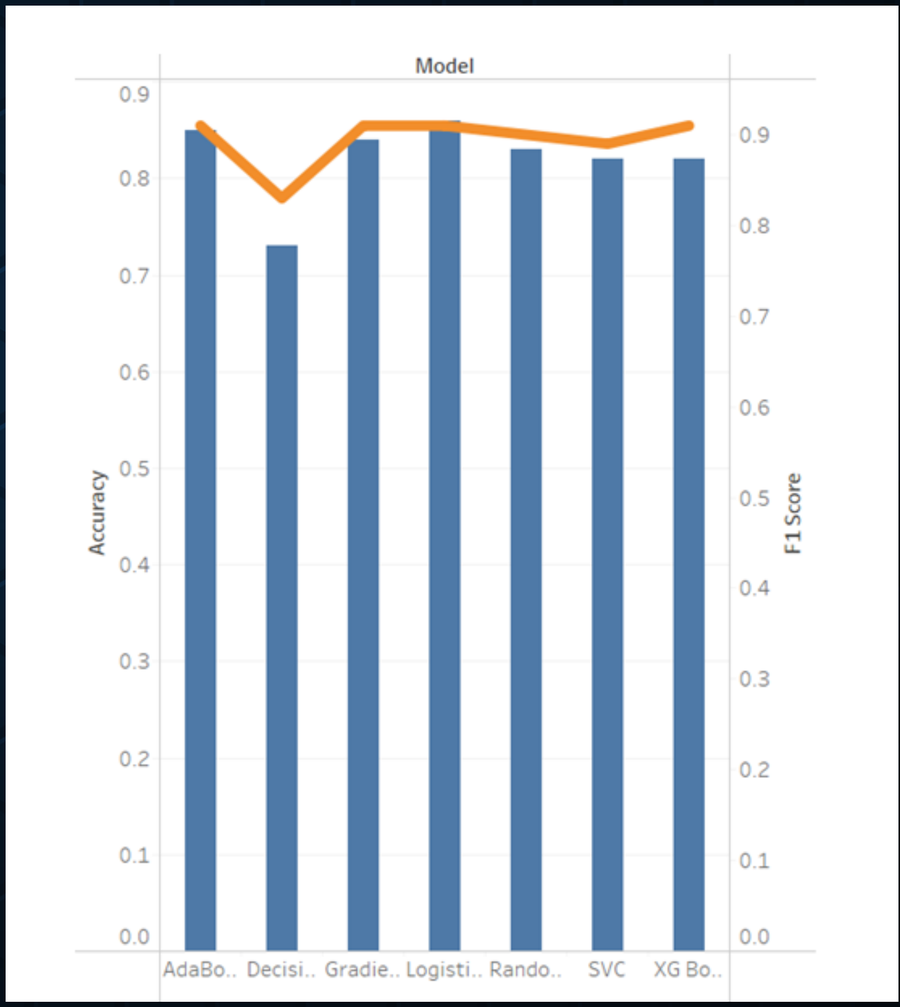


No cluster formed due to not being able to find any centroids. Hence, we can say that this model is does not fit the data set.

RESULTS

We run the following models on the dataset and then compare the results of the models. And the accuracy corresponding to each model is:

MODEL	ACCURACY	F1 SCORE
Logistic Regression	0.86	0.91
Decision Tree	0.73	0.83
SVC	0.82	0.89
Random Forest Classifier	0.83	0.90
AdaBoost Classifier	0.85	0.91
Gradient Boosting Classifier	0.84	0.91
XG Boost	0.82	0.91

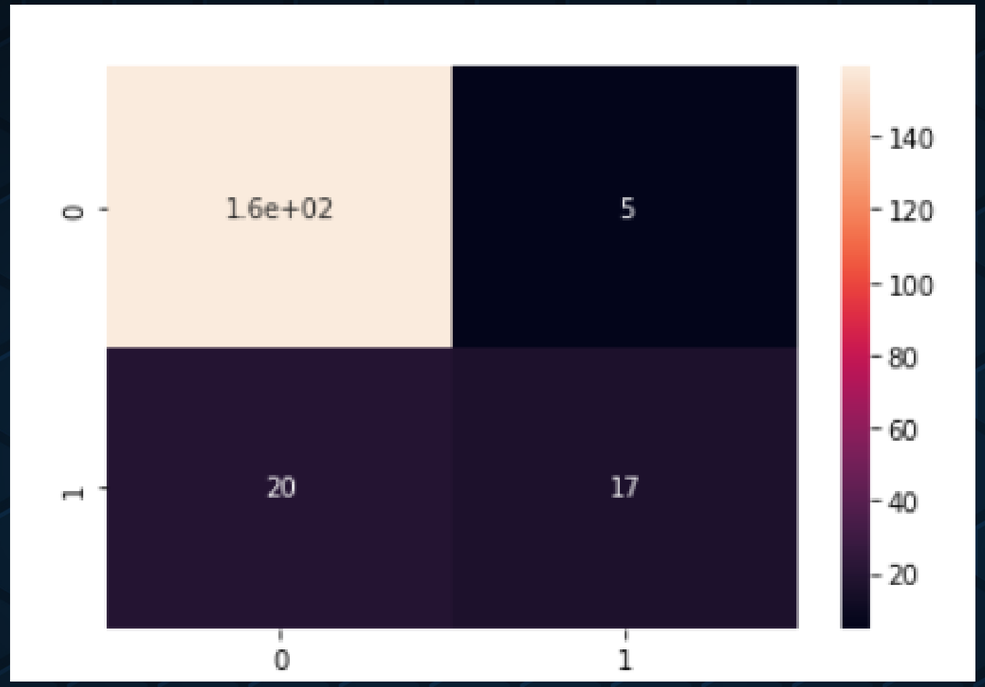


We used logit function and the variables that came out to be most contributing to the model are :

1) CURRENT BALANCE
2) DAYS SINCE LAST PREDICTION

$$Z = days\_since\_last\_transaction * 0.000172 + current\_balance * -0.000484$$

CONCLUSION



THE BEST CLASSIFIER FOR CHURN PREDICTION DATA SET IS USING LOGIT FUNCTION (LOGISTIC CLASSIFIER)

WE COULD CONCLUDE WITH AN ACCURACY OF 86% THAT 18% OF THE CUSTOMER WILL CHOOSE TO CHURN OUT OF THE BANK.

THANK YOU

UNDER THE KIND GUIDANCE OF
PROF. SIBY ABRAHAM SIR