Group Project-Mid Semester Report Submission

Credit Card Default Prediction System

EED363: Applied Machine Learning
Spring Semester, 2020

Prof. Madan Gopal
Course Instructor

Submitted By:

Aayushi Mer (1710110007)
Pranika Kakkar (1710110253)

Abstract

Credit risk plays an important role in the banking industry business. Credit card has been one of the most booming businesses in banking sectors but with its pros comes some cons also. With the growing numbers of credit card users the default rate is also increasing which is jeopardizing the banking sector. So to tackle this growing problem the world needs solutions and one of it is to try and predict in time which customer can result in a default. In this report we are applying various Machine Learning algorithms to predict and analyse the default rate and hence to see which all features are important to make a robust and efficient model.

Table of Contents

1.Introduction

The dataset which contains data of clients from a financial institute in Taiwan is provided by UCI Machine Learning Repository. We will study this data to predict the client's chances to default on his/her credit card.High credit card default rates can make a business in trouble even bankrupt. To default is to fail to make a payment on a debt by the due date. The health of the credit card industry is best measured not by the number of people with cards, but rather the number who pay their bills.

Serious delinquency rates are measured as the percentage of balances that are 90 or more days past the due date. The delinquency rate indicates the percentage of past-due loans within the borrowers' entire loan portfolio. In a developing country like ours this plays an important role as there are many entrepreneurs coming up and they need money to start their business which is funded by loans and more money is needed by them to set up their business so default rate also needs to be in check by the banks or else they can run into serious problems. Hence a need for a risk prediction model comes into existence which should be able to classify the probability of defaulters and non-defaulters. Credit card default prediction is one of the main predictions that banks are concerned with including credit scoring to better understand why customers are likely to default.

1.1 Problem Statement

To build a robust and accurate Machine Learning model which can help the banks in predicting Credit Card Default by classifying defaulters and non-defaulters.

1.2 Dataset

The dataset was taken from **UCI Machine Learning Repository.**

The dataset contains information on default payments, demographic factors, credit data, history of payment, and bill statements of credit card clients in Taiwan from April 2005 to September 2005.

ATTRIBUTES = 25

INSTANCES = 30,000

Documentation of the dataset as per the Repository :-

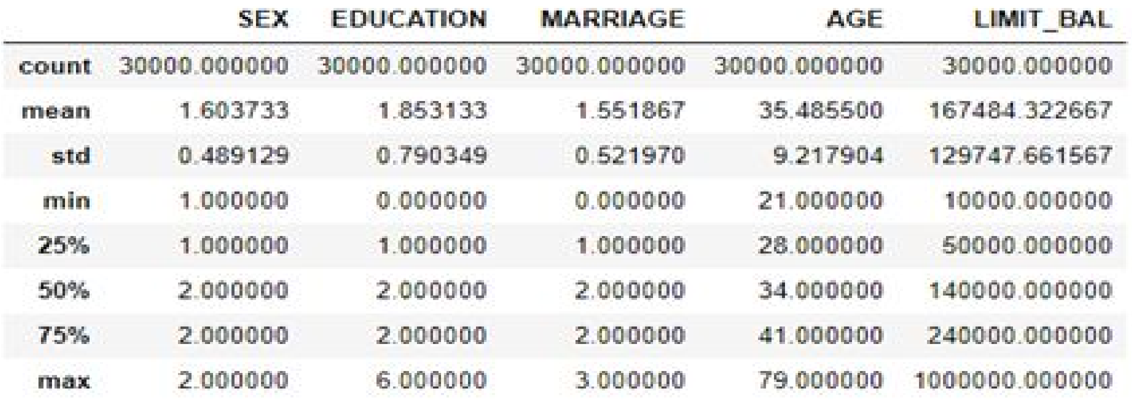
1. ID: ID of each client
2. LIMIT_BAL: Amount of given credit in NT dollars (includes individual and family/supplementary credit
3. SEX: Gender (1=male, 2=female)
4. EDUCATION: (1=graduate school, 2=university, 3=high school, 4=others, 5=unknown, 6=unknown)
5. MARRIAGE: Marital status (1=married, 2=single, 3=others)
6. AGE: Age in years
7. PAY_0: Repayment status in September, 2005 (-1=pay duly, 1=payment delay for one month, 2=payment delay for two months, ... 8=payment delay for eight months, 9=payment delay for nine months and above)
8. PAY_2: Repayment status in August, 2005 (scale same as above)
9. PAY_3: Repayment status in July, 2005 (scale same as above)
10. PAY_4: Repayment status in June, 2005 (scale same as above)
11. PAY_5: Repayment status in May, 2005 (scale same as above)
12. PAY_6: Repayment status in April, 2005 (scale same as above)
13. BILL_AMT1: Amount of bill statement in September, 2005 (NT dollar)
14. BILL_AMT2: Amount of bill statement in August, 2005 (NT dollar)
15. BILL_AMT3: Amount of bill statement in July, 2005 (NT dollar)

16. BILL_AMT4: Amount of bill statement in June, 2005 (NT dollar)
17. BILL_AMT5: Amount of bill statement in May, 2005 (NT dollar)
18. BILL_AMT6: Amount of bill statement in April, 2005 (NT dollar)
19. PAY_AMT1: Amount of previous payment in September, 2005 (NT dollar)
20. PAY_AMT2: Amount of previous payment in August, 2005 (NT dollar)
21. PAY_AMT3: Amount of previous payment in July, 2005 (NT dollar)
22. PAY_AMT4: Amount of previous payment in June, 2005 (NT dollar)
23. PAY_AMT5: Amount of previous payment in May, 2005 (NT dollar)
24. PAY_AMT6: Amount of previous payment in April, 2005 (NT dollar)
25. default.payment.next.month: Default payment (1=yes, 0=no)

In PAY_x if the x is negative ie. (-x) then it indicates paying duly for x months and positive x shows how many months the payment has been delayed.

Here in the dataset PAY_1 is not here so later we will convert PAY_0 to PAY_1 for our convenience. Default Payment next month has binary values 0 and 1 which will be used as predictors to classify.

2 . Data Analysis: To explore the data



	SEX	EDUCATION	MARRIAGE	AGE	LIMIT_BAL
count	30000.000000	30000.000000	30000.000000	30000.000000	30000.000000
mean	1.603733	1.853133	1.551867	35.485500	167484.322667
std	0.489129	0.790349	0.521970	9.217904	129747.661567
min	1.000000	0.000000	0.000000	21.000000	10000.000000
25%	1.000000	1.000000	1.000000	28.000000	50000.000000
50%	2.000000	2.000000	2.000000	34.000000	140000.000000
75%	2.000000	2.000000	2.000000	41.000000	240000.000000
max	2.000000	6.000000	3.000000	79.000000	1000000.000000

Table 2: Description function output

Inferences drawn from Table 1.1

1. From this table we can see that the median education is 1.853 which indicates that most clients are either from a graduate school and a university .
2. The median value of “MARRIAGE” suggests that most of them are married rather than others(which might contain widow, divorcee, etc).
3. The median age comes out to be 35.485 so most of the clients are around this age and the median Limit of credit card is around 167,484.32 but it has a large standard deviation which tells us that the limit is varied a lot among different clients.
4. The youngest of clients is 21 years and oldest is around 79 years and the maximum Limit of credit card is 10000000 whereas lowest is 10000 only.
5. The limit of credit card means the maximum amount the user can spend using the credit card and it depends on various factors. In total 30,000 different clients data is present in the dataset.

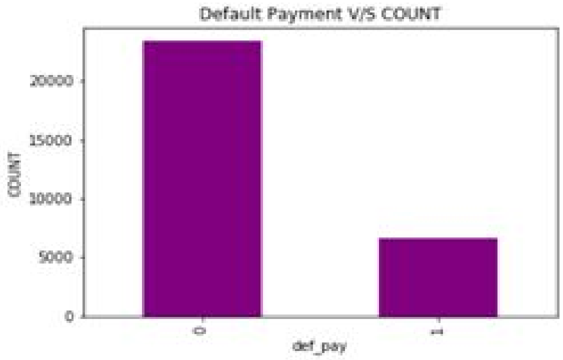
Changing column names

Changing PAY_0 to PAY_1 for own convenience as all other columns were starting from 1 and default.payment.next.month changed to def_pay so that it doesn't create problems later on when the column is used with different functions using the dot operator.

Checking for missing data

There is no missing data in the dataset. 'ID' column in no way is affecting our prediction so we drop that column.

2.1 Graphical Analysis of Data



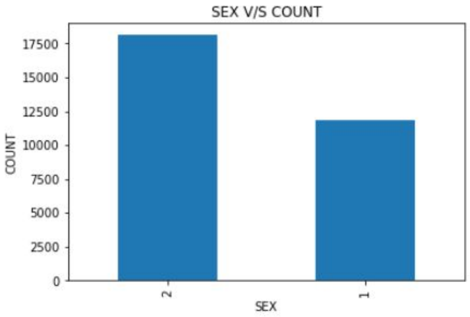


Figure 2.1: Default v/s Count

Figure 2.2: Sex v/s Count

- Def_pay v/s Count (figure 2.1) tells whether the person defaulted or not in reality, according to the Default Payment vs Count graph the non-defaulters are more (23364) than defaulters (6636).Hence, 22.12% are defaulters shown in figure 2.1.
- Sex vs Count (figure 2.2) shows that the number of female credit card holders are more than males shown in figure 2.2.

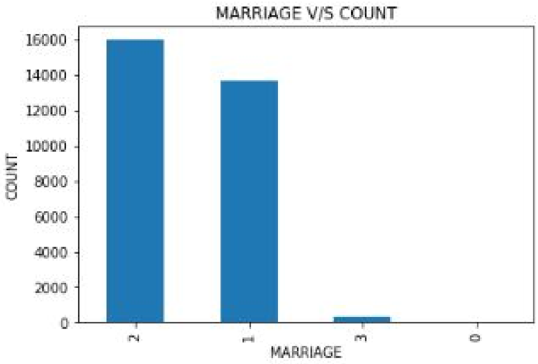


Figure 2.3: Marriage v/s Count

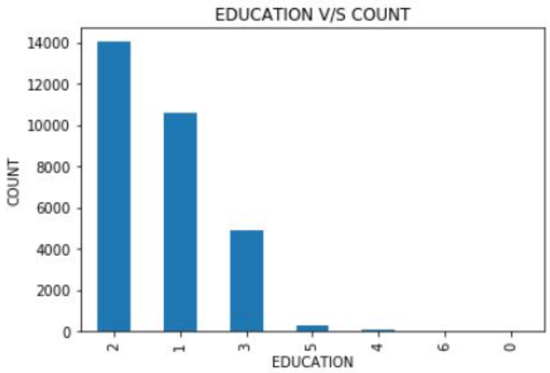


Figure 2.4: Education v/s Count

- Marriage vs Count (figure 2.3) shows that single people have more number of credit cards than married and others which includes divorcees, widows have the least number of credit cards accounting for only 323. The unlabelled value 0 carries very little weight and it can be merged with others.
- Education vs Count (figure 2.4) shows that people with education uptill university have the highest number of credit cards followed by graduate school and then high school. 5, 6, 0 which are unknown labels can be combined with others as they account for less number together when compared to 1, 2 and 3.

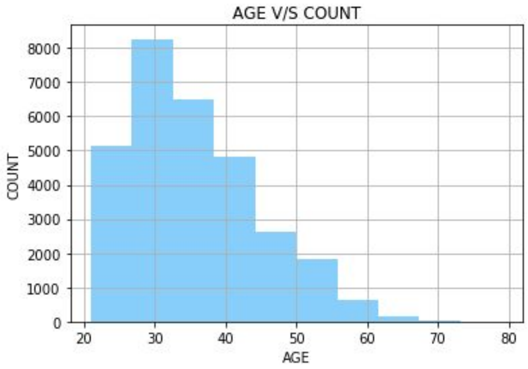


Figure 2.5: Age v/s Count

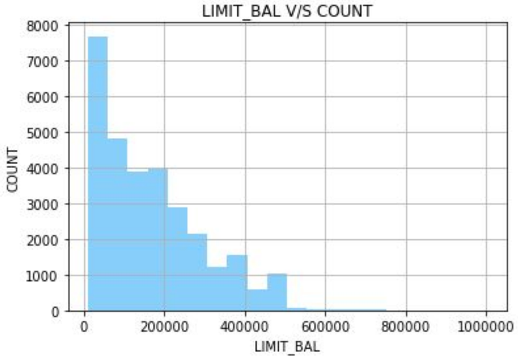


Figure 2.6: Limit_Bal v/s Count

- Age v/s Count (figure 2.5) shows that the majority of the card holders are in the range 21-45 after that the number of credit card holders is decreasing.

- Limit_Bal v/s Count (figure 2.6) shows that most of the people have a limit in the range 0-200000. People with large credit lines are quite less in number.

2.2 Data Cleaning

In *MARRIAGE* 0 can be categorized as 'other' which is 3 because the number is very small and 0 is undocumented.
In *EDUCATION* 0 is Undocumented 5 & 6 are both labeled unknown
CASE 1 - documenting 5 & 6 both as Other(Category - 4)
CASE 2 - documenting 5 & 6 both as Unknown(Category - 5)
Category 5, 6 and 0 together constitute a small number of data so they were combined with 4.
In *PAY_X* which is repayment status of each month we have negative undocumented values also like -2,-1, and 0 which according to some published reports mean that when x is negative ie. (-x) then it indicates paying duly for x months and positive x shows that how many months the payment has been delayed. So to remove the ambiguity combining the negative value together with 0 which means paying duly is the best option because they are clear of their dues.

2.3 Data Visualization

To understand the data in a better way plotting graphs of different attributes with each other is the best method to see which all attributes weigh more and are more influential in the final prediction of default.

Credit line can be accepted as a good estimator of whether a person can default or not. In many cases if a person is defaulting a lot then the bank tends to lower their credit line so that they cannot spend that much. The graph between def_pay vs LIMIT_BAL shows which credit line has the maximum number of defaulters. Default rate in general decreases as the credit line increases because people with higher credit lines are less likely to fault. A person’s credit line is increased by the bank if and only if they are capable of returning the money back in time to the bank otherwise the bank can go bankrupt.

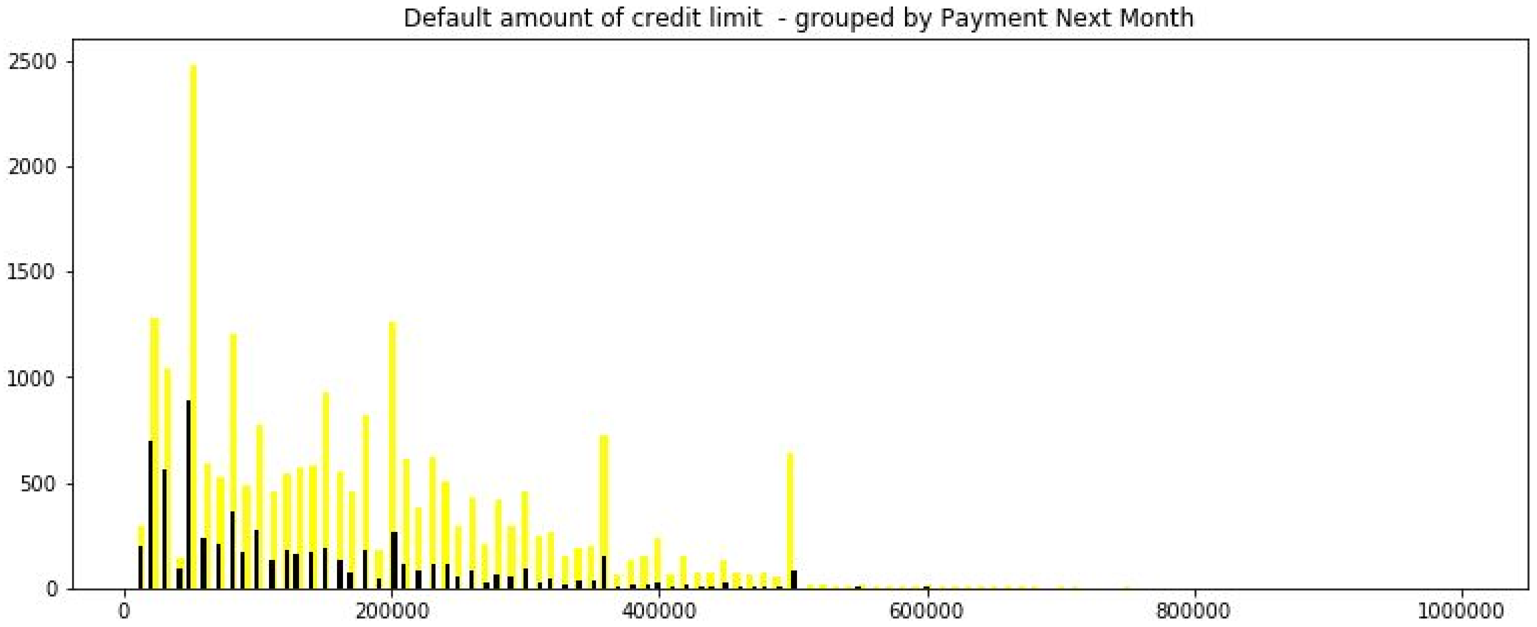
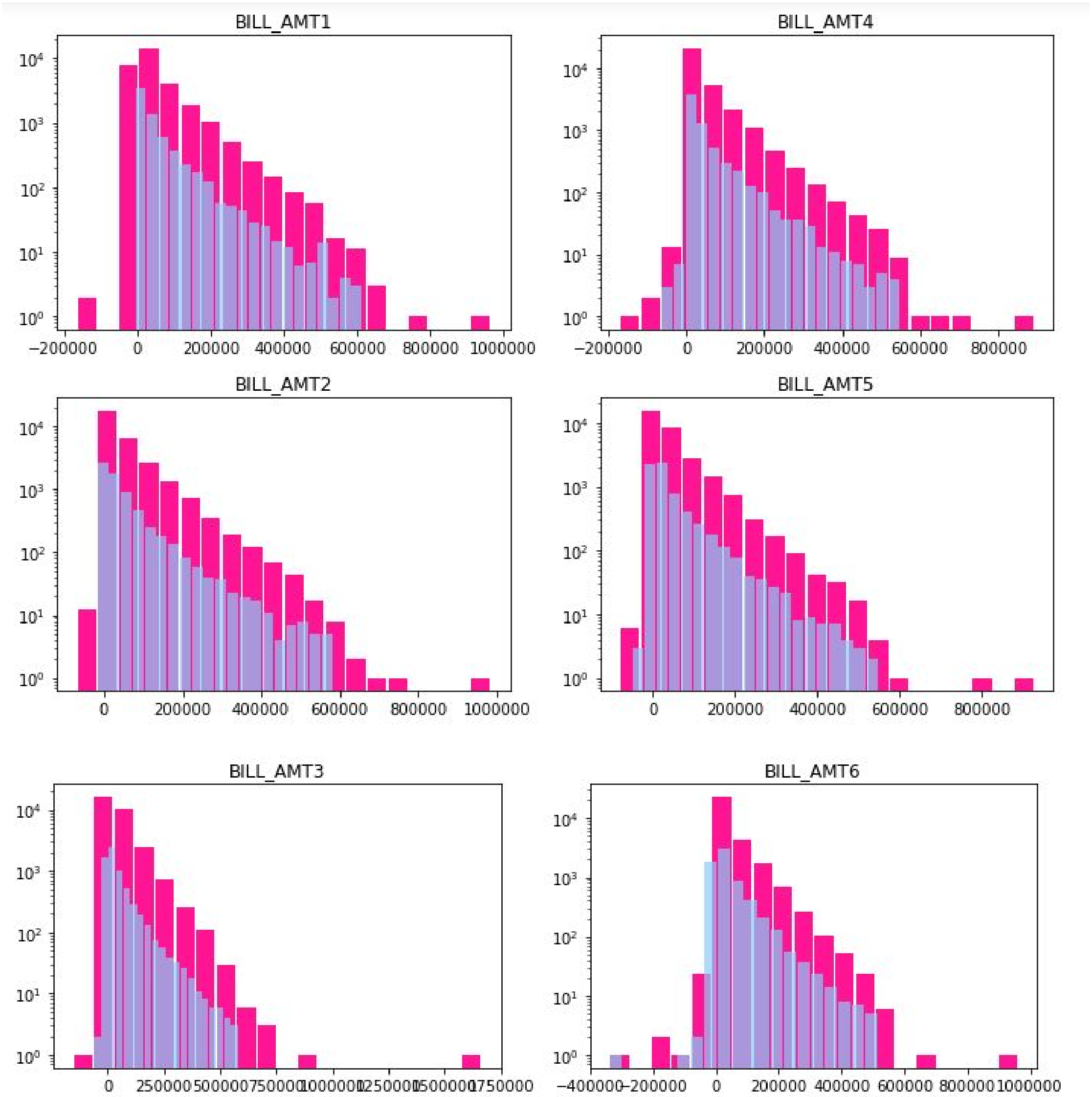


Figure 3.1: Default amount of credit limit

According to figure 3.1 the highest number of defaults are happening in the lower range of LIMIT_BAL and the maximum defaults are happening around 50,000. But as the credit line is increasing the number of defaults are decreasing with a large margin after 400,000 the number of defaulters is very less. In the figure yellow denotes Non-defaulters, black denotes defaulters.

The figure 3.2 shows a graph that has been plotted between ‘XBILL_AMTx’ and “def_pay”=1(Defaulters). Log scale is used for *y-axis* so as to get a better inference and visualization from the graph and nonposy=clip is used so that the negative values will not be masked rather they will be clipped to a very small positive value. ‘BILL_AMTx’ are the amount of bill statements from April to September 2005, through graphs it can be inferred that everyone is spending a lot with their credit cards and most of the defaults are happening when the people are in the spending range of around 0-200000.



●

Figure 3.2: Bill amount vs default pay(for all the months)

Figure 3.3 shows relation between PAY_AMTx and Defaulters, the previous amounts of payment the client has made to the bank. Most of the defaults are within the range of 0-100000.

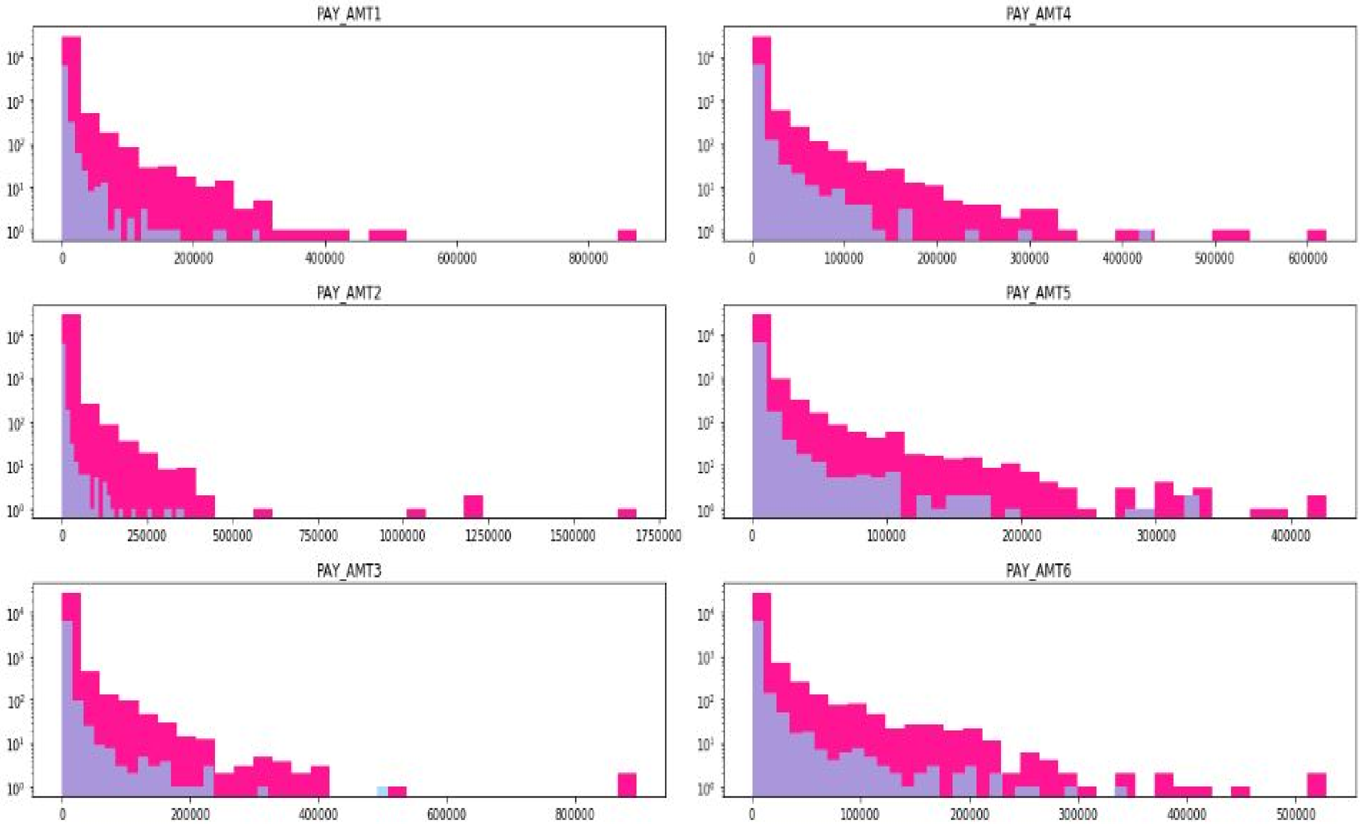


Figure 3.3: Pay Amount vs default pay (for all the months)

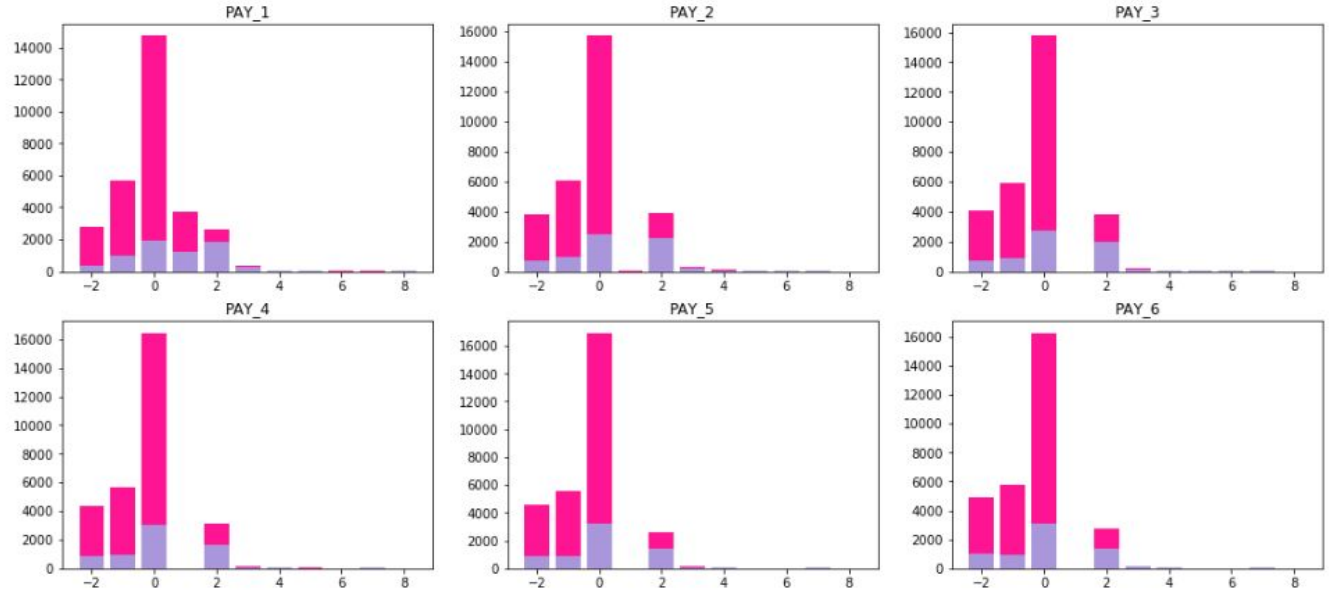
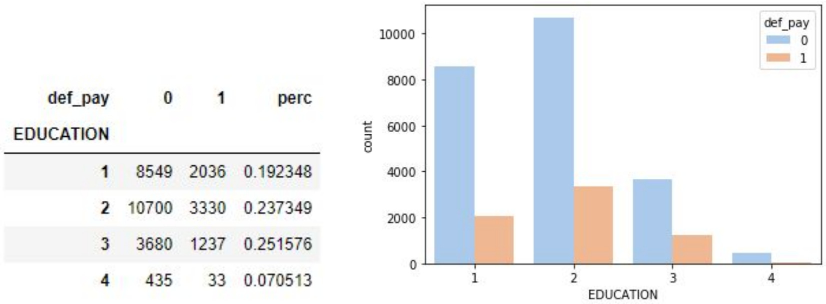


Figure 3.4: Pay_x vs default pay (for all the months)

Figure 3.4 represents the graphs plotted between PAY_x and Defaulters: repayment status of the clients in the respective months from April to September 2005. An increase of 2 (defaulting for 2 months) can be observed which implies that since two months the client has not paid the amount duly and there is a risk of defaulting involved if the client continues to do so.

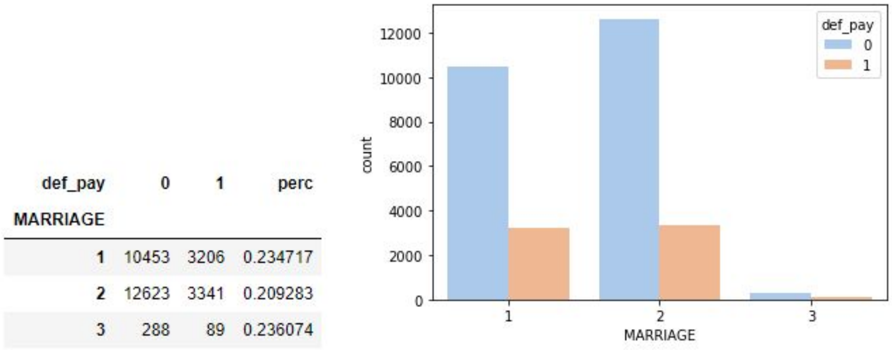
Correlation between different attributes

According to figure 3.5,In ‘EDUCATION’ the percentage is highest amongst the 3rd category which is clients upto high school education and then it is followed by clients till university education. The reason can be that they do not have a stable job or source of income as compared to the ones who are graduates and undergraduates. Perc is the percentage of default among the different attributes.



def_pay	0	1	perc
EDUCATION			
1	8549	2036	0.192348
2	10700	3330	0.237349
3	3680	1237	0.251576
4	435	33	0.070513

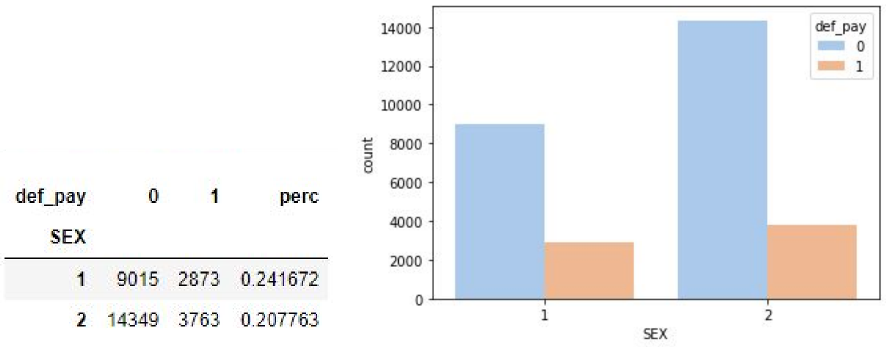
Figure 3.5: Correlation between Education & Default Pay



def_pay	0	1	perc
MARRIAGE			
1	10453	3206	0.234717
2	12623	3341	0.209283
3	288	89	0.236074

Figure 3.6: Correlation between Marriage & Default Pay

According to figure 3.6, In MARRIAGE married and others are slightly more likely to default than single clients. This can be because the married clients have a duty of supporting their families and they tend to spend more because they might also have children.



def_pay	0	1	perc
SEX			
1	9015	2873	0.241672
2	14349	3763	0.207763

Figure 3.7: Correlation between Marriage & Default Pay

According to figure 3.7, SEX: Females conduct more default than males but percetange shows that even though the number of males is less but they tend to default more when compared to females.

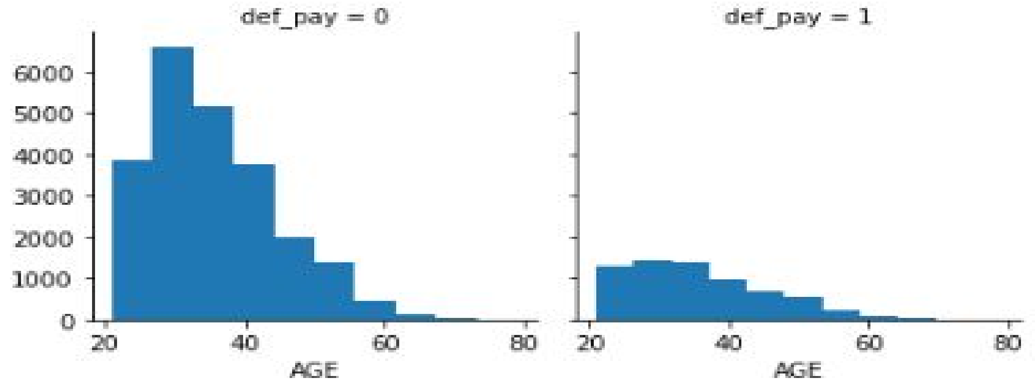


Figure 3.8: Face Grid of AGE with respect to default payment

According to figure 3.8, Range 30-40 has the maximum credit card holders and the default clients are also maximum in this range but with an increase in age the defaults happening are decreasing.

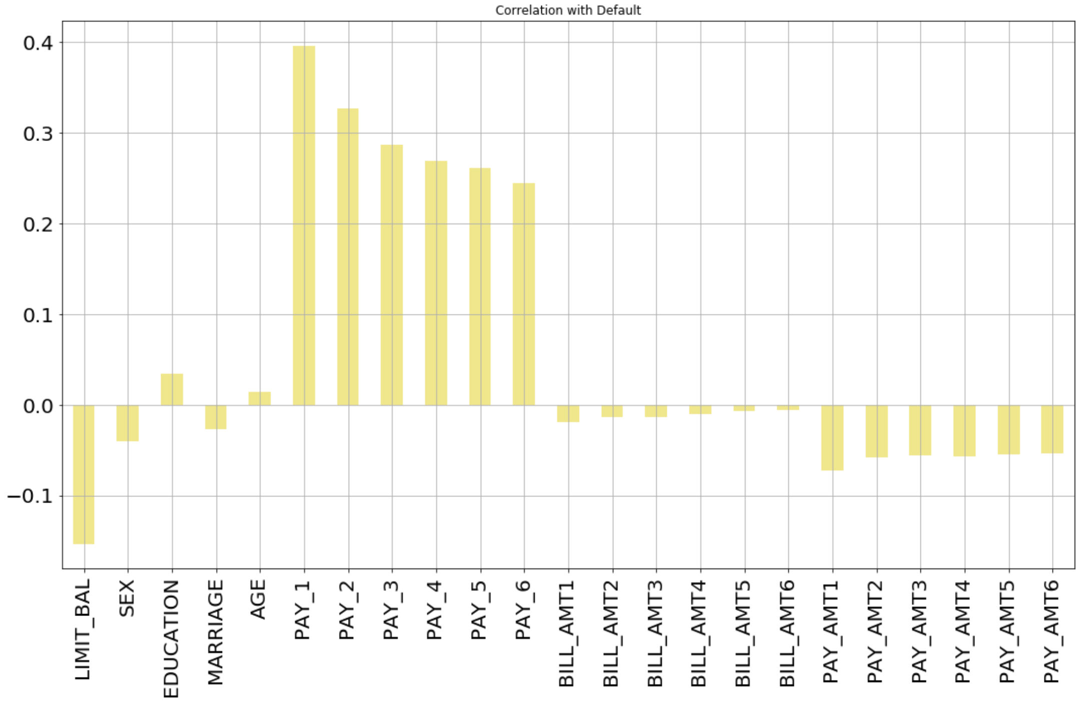
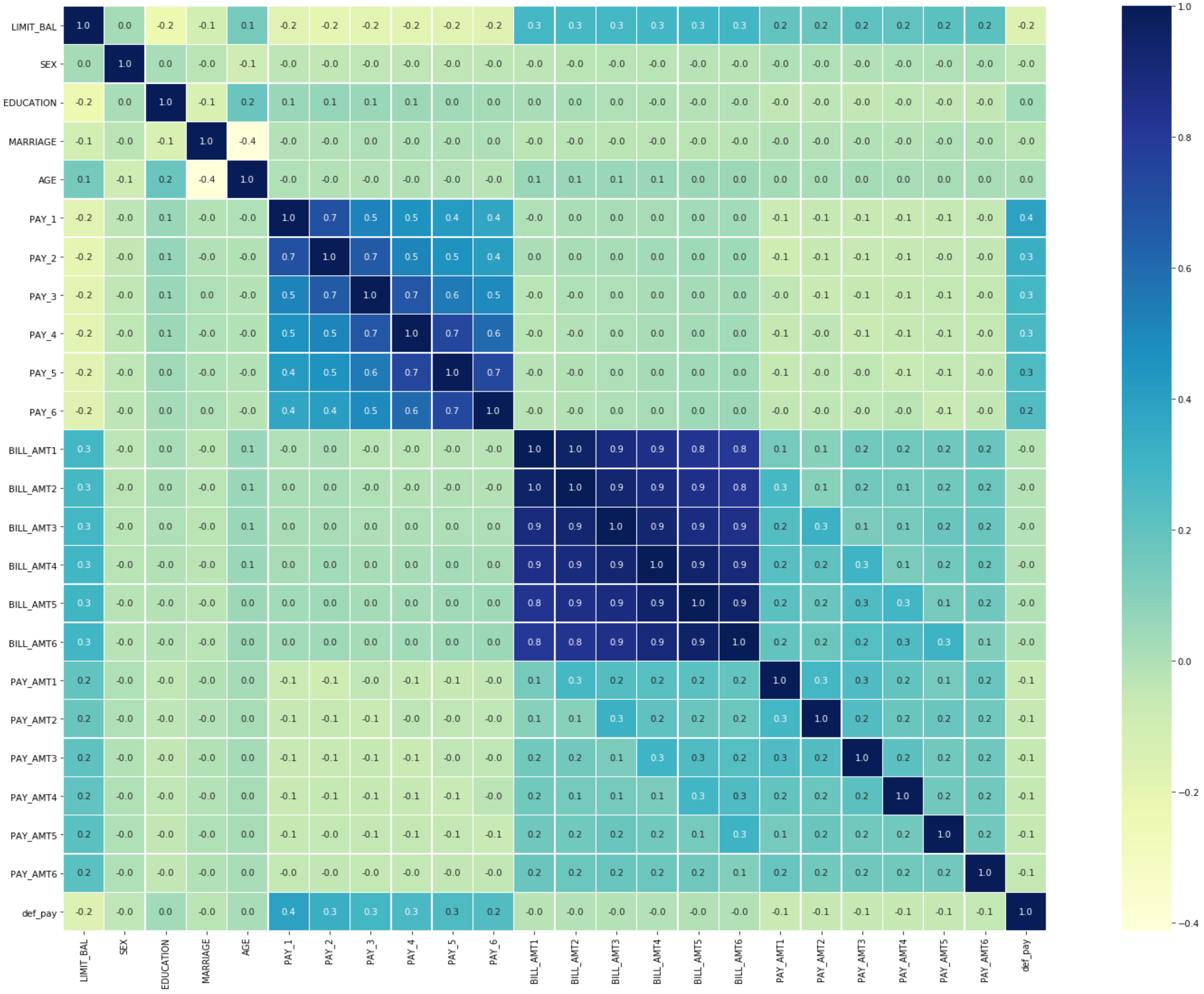


Figure 3.9: Correlation between all attributes & Default Pay

According to figure 3.9, ‘PAY_X’ terms seem to have a strong correlation with def_pay which is our predictor. Education along with age is also a somewhat good predictor. The highest negative correlation with default occurs with LIMIT_BAL, indicating that customers with lower limit balance are more likely to default. But in this data some of the uncorrelated attributes also play an important role.

In the below given Figure 3.10, the Correlation Matrix has been made in the form of a heat map which shows that PAY_1, PAY_2, PAY_3, PAY_4, PAY_ 5, PAY_6 are highly correlated to the class value. These attributes represent the history of the past payment of a credit card holder wherein PAY_1 to PAY_6 shows the payment status of each month. The only feature with a notable positive correlation with the dependent variable ‘Default’ is repayment status during the last month (September).



	LIMIT_BAL	SEX	EDUCATION	MARRIAGE	AGE	PAY_1	PAY_2	PAY_3	PAY_4	PAY_5	PAY_6	BILL_AMT1	BILL_AMT2	BILL_AMT3	BILL_AMT4	BILL_AMT5	BILL_AMT6	PAY_AMT1	PAY_AMT2	PAY_AMT3	PAY_AMT4	PAY_AMT5	PAY_AMT6	def_pay
LIMIT_BAL	1.0	0.0	-0.2	-0.1	0.1	-0.2	-0.2	-0.2	-0.2	-0.2	-0.2	0.3	0.3	0.3	0.3	0.3	0.3	0.2	0.2	0.2	0.2	0.2	0.2	-0.2
SEX	0.0	1.0	0.0	-0.0	-0.1	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0
EDUCATION	-0.2	0.0	1.0	-0.1	0.2	0.1	0.1	0.1	0.1	0.0	0.0	0.0	0.0	0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	0.0
MARRIAGE	-0.1	-0.0	-0.1	1.0	-0.4	-0.0	-0.0	0.0	-0.0	-0.0	0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0
AGE	0.1	-0.1	0.2	-0.4	1.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	0.1	0.1	0.1	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
PAY_1	-0.2	-0.0	0.1	-0.0	-0.0	1.0	0.7	0.5	0.5	0.4	0.4	-0.0	0.0	0.0	0.0	0.0	0.0	-0.1	-0.1	-0.1	-0.1	-0.1	-0.0	0.4
PAY_2	-0.2	-0.0	0.1	-0.0	-0.0	0.7	1.0	0.7	0.5	0.5	0.4	0.0	0.0	0.0	0.0	0.0	0.0	-0.1	-0.1	-0.1	-0.1	-0.0	-0.0	0.3
PAY_3	-0.2	-0.0	0.1	0.0	-0.0	0.5	0.7	1.0	0.7	0.6	0.5	-0.0	-0.0	0.0	0.0	0.0	0.0	-0.0	-0.1	-0.1	-0.1	-0.1	-0.0	0.3
PAY_4	-0.2	-0.0	0.1	-0.0	-0.0	0.5	0.5	0.7	1.0	0.7	0.6	-0.0	-0.0	0.0	0.0	0.0	0.0	-0.1	-0.0	-0.1	-0.1	-0.1	-0.0	0.3
PAY_5	-0.2	-0.0	0.0	-0.0	-0.0	0.4	0.5	0.6	0.7	1.0	0.7	-0.0	-0.0	0.0	0.0	0.0	0.0	-0.1	-0.0	-0.0	-0.1	-0.1	-0.0	0.3
PAY_6	-0.2	-0.0	0.0	0.0	-0.0	0.4	0.4	0.5	0.6	0.7	1.0	-0.0	-0.0	0.0	0.0	0.0	0.0	-0.0	-0.0	-0.0	-0.0	-0.1	-0.0	0.2
BILL_AMT1	0.3	-0.0	0.0	-0.0	0.1	-0.0	0.0	-0.0	-0.0	-0.0	-0.0	1.0	1.0	0.9	0.9	0.8	0.8	0.1	0.1	0.2	0.2	0.2	0.2	-0.0
BILL_AMT2	0.3	-0.0	0.0	-0.0	0.1	0.0	0.0	-0.0	-0.0	-0.0	-0.0	1.0	1.0	0.9	0.9	0.9	0.8	0.3	0.1	0.2	0.1	0.2	0.2	-0.0
BILL_AMT3	0.3	-0.0	0.0	-0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.9	1.0	0.9	0.9	0.9	0.2	0.3	0.1	0.1	0.2	0.2	-0.0
BILL_AMT4	0.3	-0.0	-0.0	-0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.9	0.9	1.0	0.9	0.9	0.2	0.2	0.3	0.1	0.2	0.2	-0.0
BILL_AMT5	0.3	-0.0	-0.0	-0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.9	0.9	0.9	1.0	0.9	0.2	0.2	0.3	0.3	0.1	0.2	-0.0
BILL_AMT6	0.3	-0.0	-0.0	-0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.8	0.9	0.9	0.9	1.0	0.2	0.2	0.2	0.3	0.3	0.1	-0.0
PAY_AMT1	0.2	-0.0	-0.0	-0.0	0.0	-0.1	-0.1	-0.0	-0.1	-0.1	-0.0	0.1	0.3	0.2	0.2	0.2	0.2	1.0	0.3	0.3	0.2	0.1	0.2	-0.1
PAY_AMT2	0.2	-0.0	-0.0	-0.0	0.0	-0.1	-0.1	-0.1	-0.0	-0.0	-0.0	0.1	0.1	0.3	0.2	0.2	0.2	0.3	1.0	0.2	0.2	0.2	0.2	-0.1
PAY_AMT3	0.2	-0.0	-0.0	-0.0	0.0	-0.1	-0.1	-0.1	-0.1	-0.0	-0.0	0.2	0.2	0.1	0.3	0.3	0.2	0.3	0.2	1.0	0.2	0.2	0.2	-0.1
PAY_AMT4	0.2	-0.0	-0.0	-0.0	0.0	-0.1	-0.1	-0.1	-0.1	-0.1	-0.0	0.2	0.1	0.1	0.1	0.3	0.3	0.2	0.2	0.2	1.0	0.2	0.2	-0.1
PAY_AMT5	0.2	-0.0	-0.0	-0.0	0.0	-0.1	-0.0	-0.1	-0.1	-0.1	-0.1	0.2	0.2	0.2	0.2	0.1	0.3	0.1	0.2	0.2	0.2	1.0	0.2	-0.1
PAY_AMT6	0.2	-0.0	-0.0	-0.0	0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	0.2	0.2	0.2	0.2	0.2	0.1	0.2	0.2	0.2	0.2	0.2	1.0	-0.1
def_pay	-0.2	-0.0	0.0	-0.0	0.0	0.4	0.3	0.3	0.3	0.3	0.2	-0.0	-0.0	-0.0	-0.0	-0.0	-0.0	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	1.0

Figure 3.10 : Heatmap of all attributes

3. Feature Engineering

Feature Engineering is the method of coming up with new features given the current dataset that could potentially be better predictors of credit card defaulters.

Categorical Variables

The numerical and categorical attributes are seperated . Then for the categorical data which includes the following columns:'SEX', 'EDUCATION', 'MARRIAGE',' PAY_1', 'PAY_2', 'PAY_3', 'PAY_4', 'PAY_5', 'PAY_6' dummy variables get created.

One Hot Encoding

This is a process by which all the categorical data is converted into numerical form which is easier to understand for ML Algorithms.After creating the dummy variables the number of columns is increased because it divides every categorical data.

Numerical Variables

The data which contains all the numerical attributes is feature scaled with the help of StandardScaler() by scikit which standardize features by removing the mean and scaling to unit variance. The columns included in numerical df are: ‘ID’, ‘LIMIT_BAL’, ‘AGE’, ‘BILL_AMT1’, ‘BILL_AMT2’, ‘BILL_AMT3’, ‘BILL_AMT4’, ‘BILL_AMT5’, ‘BILL_AMT6’, 'PAY_AMT1', 'PAY_AMT2', 'PAY_AMT3', 'PAY_AMT4', 'PAY_AMT5', 'PAY_AMT6'. Def_pay has not been included here because we don't need to scale it.

fit_transform is used to fit the data and then transform it.

Motivation for standardization

Standardization is essential for the chosen dataset because most of the numerical attributes are given in different units which acts as a major drawback. Hence, data standardization is a very effective method for preprocessing of data because if not standardized then the ones with higher weights will tend to learn faster and it will affect the model as some of the weights will not be weighted as much as they should have.

Concatenation of Data

The categorical and numerical data is concatenated to form the complete data which could be further used in algorithms.

3.1 Splitting the Data

The data was split into 70:30 where 70% was used for training the data and 30% was used as a testing set to verify the model prediction accuracy. Random_state is also present there which takes up random data from the data set for a particular set value like here we have set it as 0.

3.2 Error Metrics for Skewed Data

To see how good of a model is made metrics like Confusion Matrix, Recall, Precision, F1 score and accuracy is used.

Classification accuracy

The chosen data set is skewed as the number of defaults are very less in comparison to the non-defaulters. Hence,there exists a class imbalance.In such cases, classification accuracy is not the right measure to rate the model. Sometimes the false negatives can cost a lot so cost estimation plays an important role. This is known as the *accuracy paradox.*In order to resolve this issue other metrics like Precision,Recall and F-score are used.

Precision

It is defined as $Precision = \frac{TP}{TP + FP}$

Where, TP: Number of true positives and FP: Number of false positives.

The precision is intuitively the ability of the classifier to not label a sample as positive if it is negative.

Recall

$$Recall = \frac{TP}{TP + FN}$$

Where,TP : Number of true positives and FN: Number of false negatives.

The recall is intuitively the ability of the classifier to find all the positive samples. It is also called Sensitivity or the True Positive Rate indicating the number of how many people are doing default in reality.

F1 Score

$$F1\ Score = \frac{2 \times (\ Precision \times Recall)}{(\ Precision + Recall)}$$

.

F1 is usually more useful than accuracy, especially if you have an uneven class distribution.

4. Algorithms Implementation

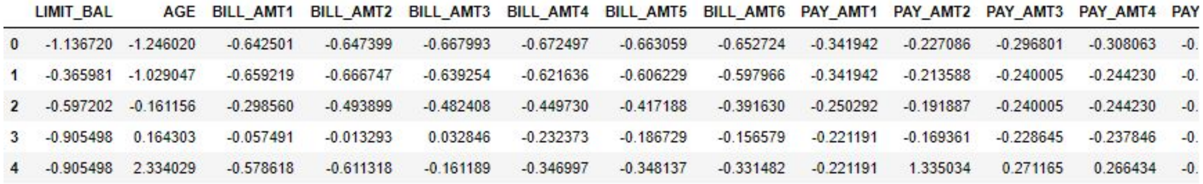
Implementation of three algorithms using two different approaches -

1. AGE and then standardizing it along with other Numerical attributes.
2. Made Age Bins and then one hot encoded it with other Categorical attributes.

The three algorithms which we have implemented yet are *Logistic Regression, Gaussian Naive Bayes and K-Nearest Neighbour.* We tried two approaches on all three of them to see how their Recall, Precision and F1 values were affected because our data is skewed and imbalanced so accuracy itself cannot always be seen as the best approach to find a good model.

First Approach:

Including AGE in our data set, here was included in the numerical_df which contains all the numerical attributes which will be standardised.



	LIMIT_BAL	AGE	BILL_AMT1	BILL_AMT2	BILL_AMT3	BILL_AMT4	BILL_AMT5	BILL_AMT6	PAY_AMT1	PAY_AMT2	PAY_AMT3	PAY_AMT4	PAY
0	-1.136720	-1.246020	-0.642501	-0.647399	-0.667993	-0.672497	-0.663059	-0.652724	-0.341942	-0.227086	-0.296801	-0.308063	-0.
1	-0.365981	-1.029047	-0.659219	-0.666747	-0.639254	-0.621636	-0.606229	-0.597966	-0.341942	-0.213588	-0.240005	-0.244230	-0.
2	-0.597202	-0.161156	-0.298560	-0.493899	-0.482408	-0.449730	-0.417188	-0.391630	-0.250292	-0.191887	-0.240005	-0.244230	-0.
3	-0.905498	0.164303	-0.057491	-0.013293	0.032846	-0.232373	-0.186729	-0.156579	-0.221191	-0.169361	-0.228645	-0.237846	-0.
4	-0.905498	2.334029	-0.578618	-0.611318	-0.161189	-0.346997	-0.348137	-0.331482	-0.221191	1.335034	0.271165	0.266434	-0.

Second Approach:

Formation of Age Bins by dividing the age into different categories, Age Bins are counted as categorical data now and hence they were one-hot encoded. The age bins were created with a gap of 10 yrs and Age Bin 2 ie. 30-40 has the highest amount. After this the data was again splitted into 70:30 ratio 30% for testing and 70% for training the model.

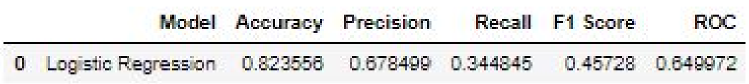
4.1 Algorithms Implemented

1. *Logistic Regression:*

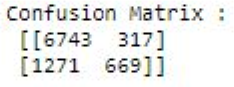
   Logistic Regression is used for classification problems like this one. This algorithm uses sigmoid function as the cost function. This analysis is conducted when the dependent variable is dichotomous (binary). In this a threshold point is set above which its corresponding output is classified as defaulter or vice versa.

   With AGE as a numerical Variable:

   We can see that even the accuracy is good as it stands at 82.35% but we can work more on the recall and precision trade off. Logistic regression has a fine enough positive prediction value.



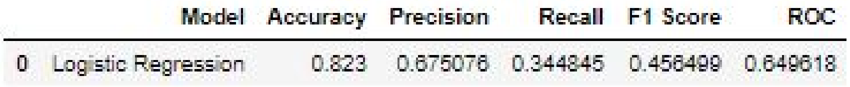
	Model	Accuracy	Precision	Recall	F1 Score	ROC
0	Logistic Regression	0.823556	0.678499	0.344845	0.45728	0.649972



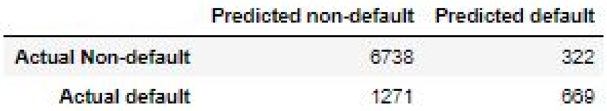
```
Confusion Matrix :
 [[6743  317]
 [1271  669]]
```

   confusion matrix. TN= 6743, FP= 317, FN= 1271, TP= 669.

   With Age Bins:



	Model	Accuracy	Precision	Recall	F1 Score	ROC
0	Logistic Regression	0.823	0.675076	0.344845	0.456499	0.649618



	Predicted non-default	Predicted default
Actual Non-default	6738	322
Actual default	1271	669

   We can see that after Age bin also, the change is very slight but our accuracy has decreased very slightly along with Recall and Precision value.

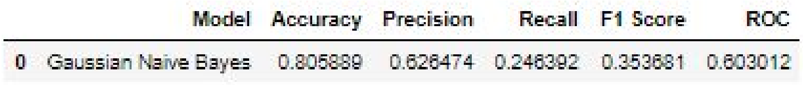
2. *Gaussian Naive Bayes:*

   It has a probabilistic approach and is based on Bayes theorem. The principle on which it is based is that every pair of features being classified is independent of each other. In Gaussian Naive Bayes, continuous values associated with each feature are assumed to be distributed according to a Gaussian distribution. A Gaussian distribution is also called Normal distribution. When plotted, it gives a bell shaped curve which is symmetric about the mean of the feature values.

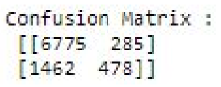
$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}$$

.

With AGE

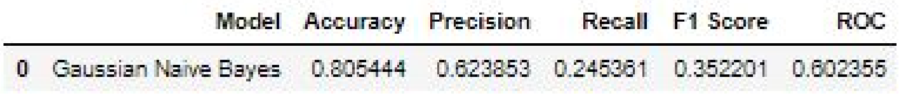


	Model	Accuracy	Precision	Recall	F1 Score	ROC
0	Gaussian Naive Bayes	0.805889	0.626474	0.246392	0.353681	0.603012

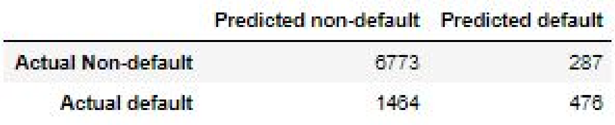


```
Confusion Matrix :
 [[6775  285]
 [1462  478]]
```

With Age Bins



	Model	Accuracy	Precision	Recall	F1 Score	ROC
0	Gaussian Naive Bayes	0.805444	0.623853	0.245361	0.352201	0.602355



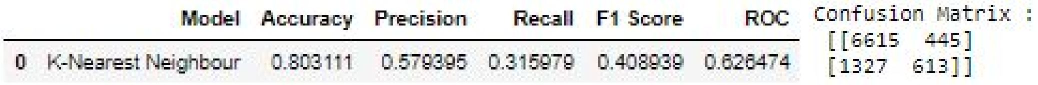
	Predicted non-default	Predicted default
Actual Non-default	6773	287
Actual default	1464	476

Here also we can see that the Accuracy ,Precision and ROC have decreased very slightly.

3. *K- Nearest Neighbour: with K=7 for both case*

KNN algorithms use data and classify new data points based on similarity measures. Classification in KNN is done by a majority vote to its neighbors. It uses euclidean distance as a way to measure distance and classify. It estimates how likely a data point is to be a member of one group or the other depending on what group the data points nearest to it are in.

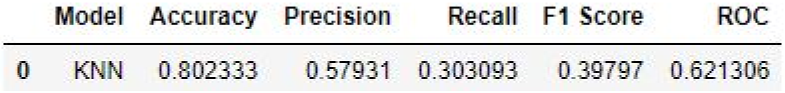
With AGE



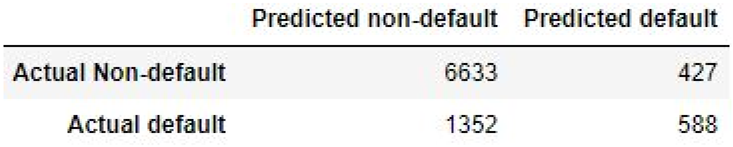
	Model	Accuracy	Precision	Recall	F1 Score	ROC
0	K-Nearest Neighbour	0.803111	0.579395	0.315979	0.408939	0.626474

```
Confusion Matrix :
 [[6615  445]
 [1327  613]]
```

With AgeBins



	Model	Accuracy	Precision	Recall	F1 Score	ROC
0	KNN	0.802333	0.57931	0.303093	0.39797	0.621306



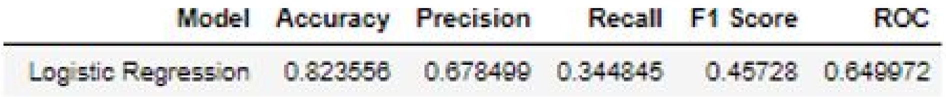
	Predicted non-default	Predicted default
Actual Non-default	6633	427
Actual default	1352	588

S.No	Parameter Name	Parameter Value	Reason/Significance
	K Neighbour	7	Error rate graph gave less error at this value.

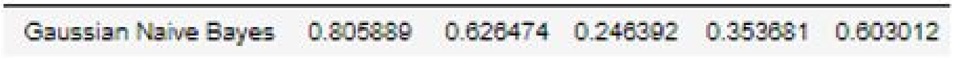
Table 4.1: Parameters for K-Nearest Neighbours

For all three Algorithms:

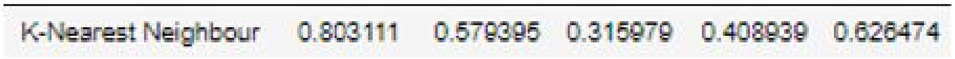
With AGE



Model	Accuracy	Precision	Recall	F1 Score	ROC
Logistic Regression	0.823556	0.678499	0.344845	0.45728	0.649972

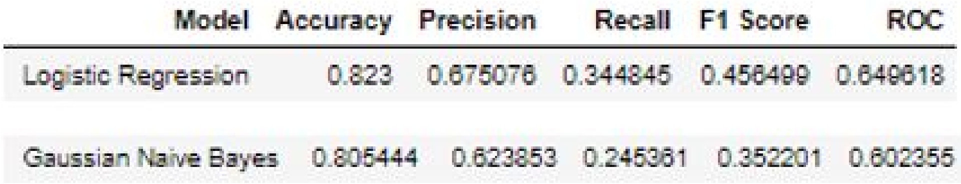


Gaussian Naive Bayes	0.805889	0.626474	0.246392	0.353681	0.603012



K-Nearest Neighbour	0.803111	0.579395	0.315979	0.408939	0.626474

With AGE BIN



Model	Accuracy	Precision	Recall	F1 Score	ROC
Logistic Regression	0.823	0.675076	0.344845	0.456499	0.649618
Gaussian Naive Bayes	0.805444	0.623853	0.245361	0.352201	0.602355

It was observed that by making Age bins not much difference was noticed in the accuracies, F1 score, precision or recall. They all decreased slightly when AGE was categorised and used as Age Bins. KNN was slower than Gaussian Naive Bayes and Logistic Regression, KNN is also called lazy learner because it does not generate a model of the data set beforehand and it scans the database each time a prediction is needed. Logistic Regression has the best recall rate amongst all the algos implemented yet and Gaussian Naive Bayes has the least. Accuracy was also the best in Logistic Regression but accuracy paradox doesn't let us rejoice it. Precision is also best in logistic regression and least in KNN.

4. *Support Vector Machine (Kernel=Linear)*

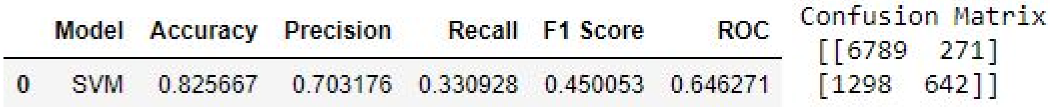
SVM is a supervised algorithm used for both classification and regression problems but mostly it is used for classification problems. The idea of SVM algorithm is simple as it creates a line or a hyperplane which separates the data into classes. It uses a technique called the kernel trick to transform the data and then based on these transformations it finds an optimal boundary between the possible outputs. Linear Kernel is used when the data is Linearly separable, that is, it can be separated using a single Line. It is one of the most common kernels to be used. It is mostly used when there are a large number of features in a particular Data Set like in Credit Card default data.

S.No	Parameter Name	Parameter Value	Reason/Significance
1.	kernel	linear	1. It is simple and easy to

0			train. 2. Gives faster prediction and scales well to very large datasets

Table 4.2: Parameters for Support Vector Machine

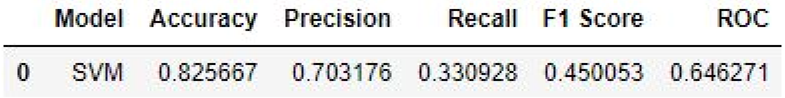
With AGE



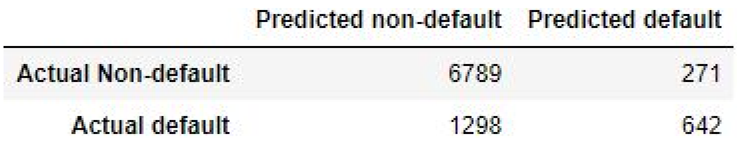
	Model	Accuracy	Precision	Recall	F1 Score	ROC
0	SVM	0.825667	0.703176	0.330928	0.450053	0.646271

```
Confusion Matrix
 [[6789  271]
 [1298  642]]
```

With AgeBins



	Model	Accuracy	Precision	Recall	F1 Score	ROC
0	SVM	0.825667	0.703176	0.330928	0.450053	0.646271



	Predicted non-default	Predicted default
Actual Non-default	6789	271
Actual default	1298	642

5. *Decision Trees*

Decision trees build classification or regression models in the form of a tree structure. It breaks down a data set into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. A decision tree is a structure that includes a root node, branches, and leaf nodes. Leaf node represents a classification or decision. Each internal node denotes a test on an attribute, each branch denotes the outcome of a test, and each leaf node holds a class label. It clearly lays out the problem so that all options can be challenged and allow us to analyze fully the possible consequences of a decision. Provide a framework to quantify the values of outcomes and the probabilities of achieving them.

In supervised algorithms, we use the Gini Impurity or Entropy to split nodes since data labels are known. Gini Impurity and Entropy are criteria to split a node in a decision tree.

They are the standard metrics to compute “impurity” or “information level” which then guides to split a node in the decision tree based on the information that exists at that node.

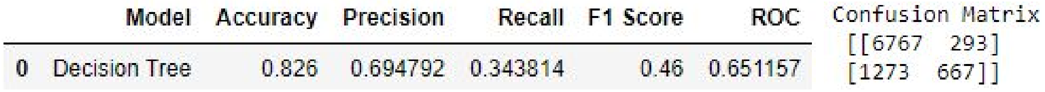
$$Gini = 1 - \sum_{j=1}^{c} p_j^2 \qquad Entropy = - \sum_{j=1}^{c} p_j \log p_j$$

Gini Impurity is calculated with less computation, i.e., no logarithmic function is involved. Hence we used **gini** as it takes less time to compute the results. Higher value of maximum depth causes overfitting, and a lower value causes underfitting. Splitter allows us to choose the split strategy. Gini measures the impurity of the node.

S.No	Parameter Name	Parameter Value	Reason/Significance
1.	max_depth	5	Best accuracy after this accuracy became constant
2.	criteria	gini	Takes lesser time than entropy criterion
3.	splitter	best	It chooses the best split

Table 4.3: Parameters for Decision Trees

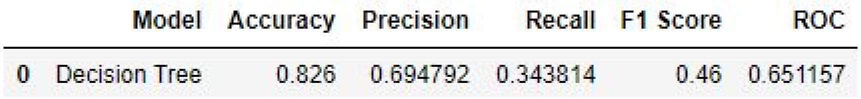
With AGE



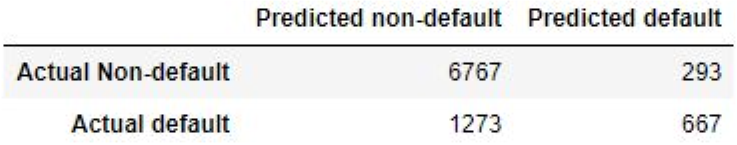
	Model	Accuracy	Precision	Recall	F1 Score	ROC
0	Decision Tree	0.826	0.694792	0.343814	0.46	0.651157

```
Confusion Matrix
 [[6767  293]
 [1273  667]]
```

With Age Bins



	Model	Accuracy	Precision	Recall	F1 Score	ROC
0	Decision Tree	0.826	0.694792	0.343814	0.46	0.651157



	Predicted non-default	Predicted default
Actual Non-default	6767	293
Actual default	1273	667

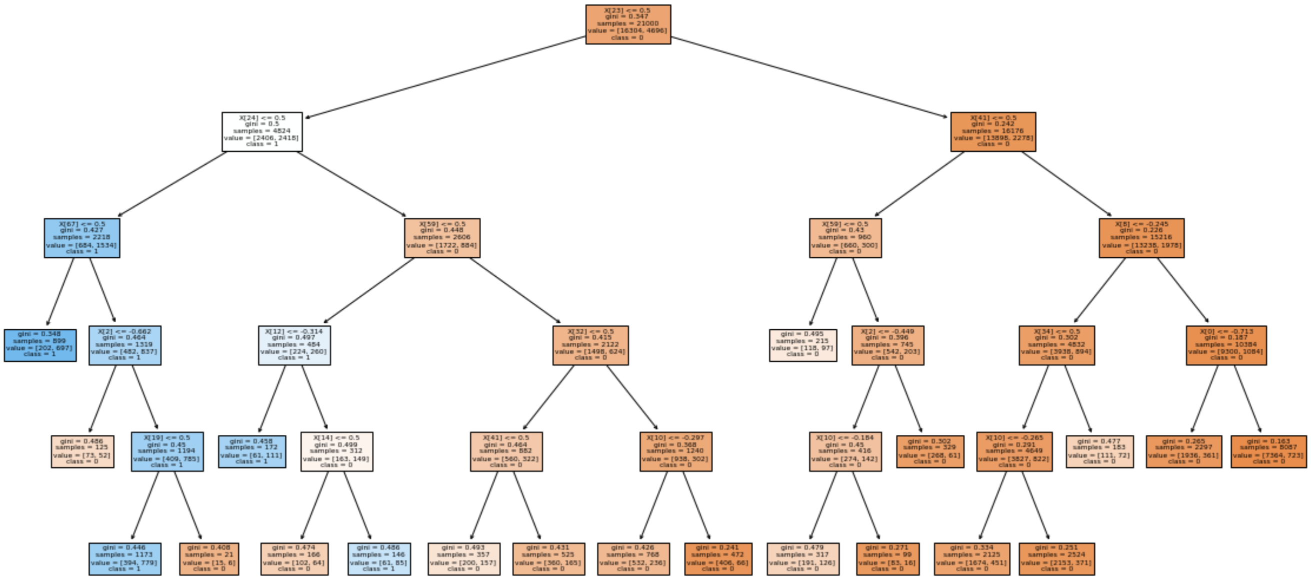

X[23] <= 0.5 gini = 0.347 samples = 21000 value = [16304, 4696] class = 0
X[24] <= 0.5 gini = 0.5 samples = 4824 value = [2406, 2418] class = 1
X[41] <= 0.5 gini = 0.242 samples = 16176 value = [13898, 2278] class = 0
X[67] <= 0.5 gini = 0.427 samples = 2218 value = [684, 1534] class = 1
X[59] <= 0.5 gini = 0.448 samples = 2606 value = [1722, 884] class = 0
X[59] <= 0.5 gini = 0.43 samples = 960 value = [660, 300] class = 0
X[8] <= -0.245 gini = 0.226 samples = 15216 value = [13238, 1978] class = 0
gini = 0.348 samples = 899 value = [202, 697] class = 1
X[2] <= -0.662 gini = 0.464 samples = 1319 value = [482, 837] class = 1
X[12] <= -0.314 gini = 0.497 samples = 484 value = [224, 260] class = 1
X[32] <= 0.5 gini = 0.415 samples = 2122 value = [1498, 624] class = 0
gini = 0.495 samples = 215 value = [118, 97] class = 0
X[2] <= -0.449 gini = 0.396 samples = 745 value = [542, 203] class = 0
X[34] <= 0.5 gini = 0.302 samples = 4832 value = [3938, 894] class = 0
X[0] <= -0.713 gini = 0.187 samples = 10384 value = [9300, 1084] class = 0
gini = 0.486 samples = 125 value = [73, 52] class = 0
X[19] <= 0.5 gini = 0.45 samples = 1194 value = [409, 785] class = 1
gini = 0.458 samples = 172 value = [61, 111] class = 1
X[14] <= 0.5 gini = 0.499 samples = 312 value = [163, 149] class = 0
X[41] <= 0.5 gini = 0.464 samples = 882 value = [560, 322] class = 0
X[10] <= -0.297 gini = 0.368 samples = 1240 value = [938, 302] class = 0
X[10] <= -0.184 gini = 0.45 samples = 416 value = [274, 142] class = 0
gini = 0.302 samples = 329 value = [268, 61] class = 0
X[10] <= -0.265 gini = 0.291 samples = 4649 value = [3827, 822] class = 0
gini = 0.477 samples = 183 value = [111, 72] class = 0
gini = 0.265 samples = 2297 value = [1936, 361] class = 0
gini = 0.163 samples = 8087 value = [7364, 723] class = 0
gini = 0.446 samples = 1173 value = [394, 779] class = 1
gini = 0.408 samples = 21 value = [15, 6] class = 0
gini = 0.474 samples = 166 value = [102, 64] class = 0
gini = 0.486 samples = 146 value = [61, 85] class = 1
gini = 0.493 samples = 357 value = [200, 157] class = 0
gini = 0.431 samples = 525 value = [360, 165] class = 0
gini = 0.426 samples = 768 value = [532, 236] class = 0
gini = 0.241 samples = 472 value = [406, 66] class = 0
gini = 0.479 samples = 317 value = [191, 126] class = 0
gini = 0.271 samples = 99 value = [83, 16] class = 0
gini = 0.334 samples = 2125 value = [1674, 451] class = 0
gini = 0.251 samples = 2524 value = [2153, 371] class = 0

Figure 4.1: Decision Tree graph for AGE

Figure 4.1 and figure 4.2 show us the split of the decisions the tree has made and each node has a specific class and the gini measures the impurity value of the node. Leaf nodes are the pure nodes when all of its records belong to the same class either 0 or 1. This helps in visualizing the tree and to interpret it.

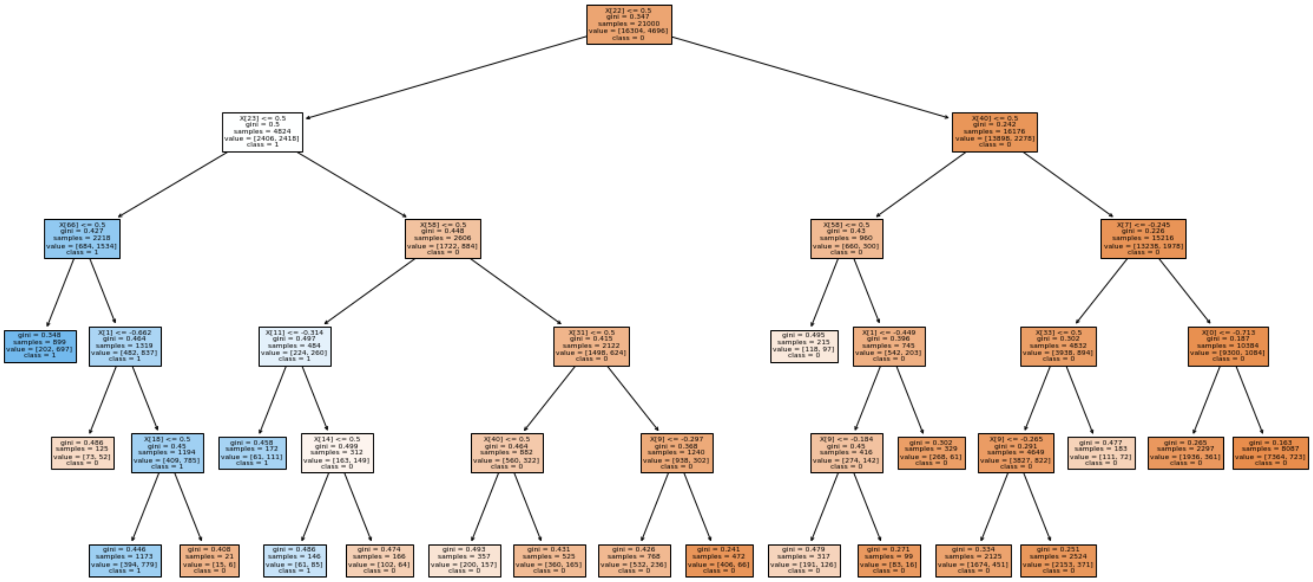

X[22] <= 0.5 gini = 0.347 samples = 21000 value = [16304, 4696] class = 0
X[23] <= 0.5 gini = 0.5 samples = 4824 value = [2406, 2418] class = 1
X[40] <= 0.5 gini = 0.242 samples = 16176 value = [13898, 2278] class = 0
X[66] <= 0.5 gini = 0.427 samples = 2218 value = [684, 1534] class = 1
X[58] <= 0.5 gini = 0.448 samples = 2606 value = [1722, 884] class = 0
X[58] <= 0.5 gini = 0.43 samples = 960 value = [660, 300] class = 0
X[7] <= -0.245 gini = 0.226 samples = 15216 value = [13238, 1978] class = 0
gini = 0.348 samples = 899 value = [202, 697] class = 1
X[1] <= -0.662 gini = 0.464 samples = 1319 value = [482, 837] class = 1
X[11] <= -0.314 gini = 0.497 samples = 484 value = [224, 260] class = 1
X[31] <= 0.5 gini = 0.415 samples = 2122 value = [1498, 624] class = 0
gini = 0.495 samples = 215 value = [118, 97] class = 0
X[1] <= -0.449 gini = 0.396 samples = 745 value = [542, 203] class = 0
X[33] <= 0.5 gini = 0.302 samples = 4832 value = [3938, 894] class = 0
X[0] <= -0.713 gini = 0.187 samples = 10384 value = [9300, 1084] class = 0
gini = 0.486 samples = 125 value = [73, 52] class = 0
X[18] <= 0.5 gini = 0.45 samples = 1194 value = [409, 785] class = 1
gini = 0.458 samples = 172 value = [61, 111] class = 1
X[14] <= 0.5 gini = 0.499 samples = 312 value = [163, 149] class = 0
X[40] <= 0.5 gini = 0.464 samples = 882 value = [560, 322] class = 0
X[9] <= -0.297 gini = 0.368 samples = 1240 value = [938, 302] class = 0
X[9] <= -0.184 gini = 0.45 samples = 416 value = [274, 142] class = 0
gini = 0.302 samples = 329 value = [268, 61] class = 0
X[9] <= -0.265 gini = 0.291 samples = 4649 value = [3827, 822] class = 0
gini = 0.477 samples = 183 value = [111, 72] class = 0
gini = 0.265 samples = 2297 value = [1936, 361] class = 0
gini = 0.163 samples = 8087 value = [7364, 723] class = 0
gini = 0.446 samples = 1173 value = [394, 779] class = 1
gini = 0.408 samples = 21 value = [15, 6] class = 0
gini = 0.486 samples = 146 value = [61, 85] class = 1
gini = 0.474 samples = 166 value = [102, 64] class = 0
gini = 0.493 samples = 357 value = [200, 157] class = 0
gini = 0.431 samples = 525 value = [360, 165] class = 0
gini = 0.426 samples = 768 value = [532, 236] class = 0
gini = 0.241 samples = 472 value = [406, 66] class = 0
gini = 0.479 samples = 317 value = [191, 126] class = 0
gini = 0.271 samples = 99 value = [83, 16] class = 0
gini = 0.334 samples = 2125 value = [1674, 451] class = 0
gini = 0.251 samples = 2524 value = [2153, 371] class = 0

Figure 4.2: Decision Tree graph for AgeBins

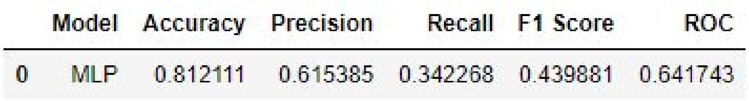
6. *Multilayer Perceptron (MLP)- Neural Networks*

A multilayer perceptron (MLP) is a class of feedforward artificial neural networks (ANN). It is often applied to supervised learning problems and is suitable for classification problem where inputs are assigned a class or label. They are composed of one or more layers of neurons. Data is fed to the input layer, there may be one or more hidden layers providing levels of abstraction, and predictions are made on the output layer, also called the visible layer.

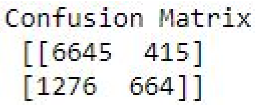
S.No	Parameter Name	Parameter Value	Reason/Significance
1.	hidden_layer_size	90	For best test accuracy that could be obtained was when 90 neurons were present in the ith layer.
2.	alpha	0.0001	The hidden_layer_size and alpha value were finely tuned in order to get the best accuracy
3.	learning_rate_init	0.1	Used to control the updating of weights with solver='adam' which is default, 0.1 was used to get best accuracy.

Table 4.4: Parameters for MLP

With AGE

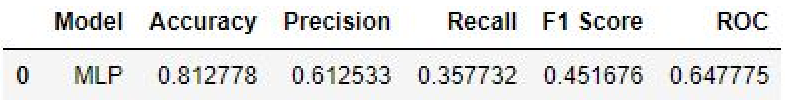


	Model	Accuracy	Precision	Recall	F1 Score	ROC
0	MLP	0.812111	0.615385	0.342268	0.439881	0.641743



```
Confusion Matrix
 [[6645  415]
 [1276  664]]
```

With Age Bins



	Model	Accuracy	Precision	Recall	F1 Score	ROC
0	MLP	0.812778	0.612533	0.357732	0.451676	0.647775

4.2 K-Fold Validation on Training data

K-Fold Cross Validation is where a given data set is split into a K number of sections/folds where each fold is used as a testing set at some point. This process is repeated until each fold of the 5 folds have been used as the testing set. For each such split, the **model** is fit to the training data, and predictive **accuracy** is assessed using the **validation** data. The results are then averaged over the splits.

Cross-validation is a resampling procedure used to evaluate machine learning models on a limited data sample. Here training and testing is done K times. It helps in assessing the effectiveness of the model, particularly in cases where there is **need** to mitigate overfitting and is also used in determining the hyper parameters of your model, in the sense that which parameters will result in the lowest test error.

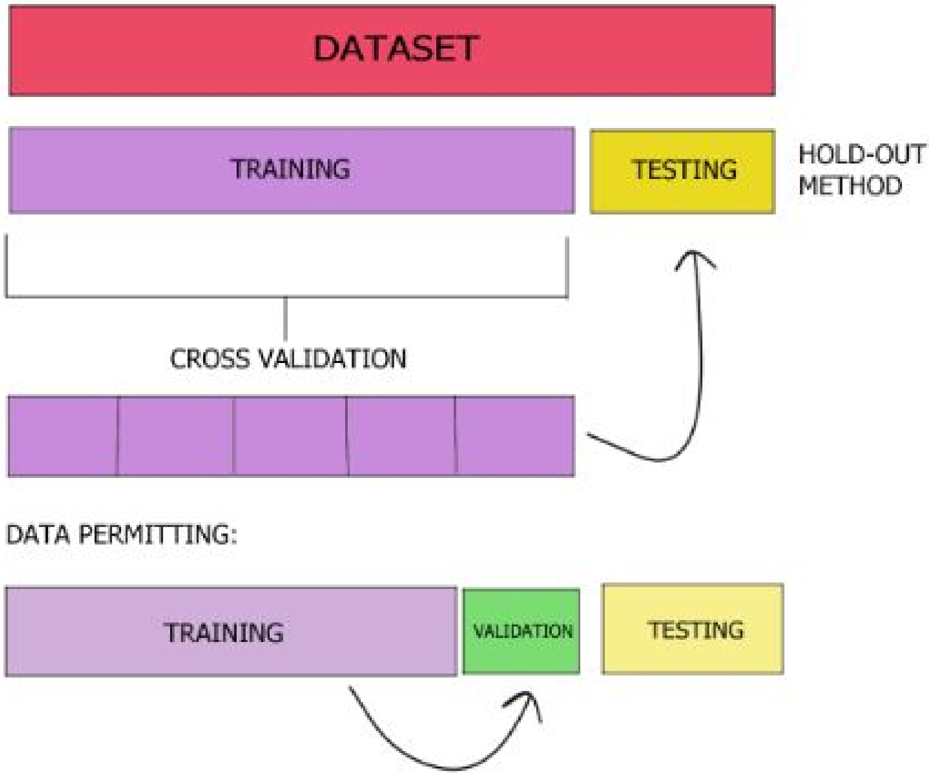


Table 4.2: Representation of cross-validation technique

S.No	Parameter Name	Parameter Value	Reason/Significance
3.	K	10	It is a standard value and some theoretical proofs back up 10-fold-cross-validation.

Table 4.5: Parameters for K fold

Through K-Fold cross-validation the mean value for f1 -According to Table 4.6, it can be inferred that Logistic regression has the highest mean f1-score followed by Decision Trees. The least mean f1-score was seen by Gaussian Naive Bayes for Credit Card default Model.

S.No	Algorithm	K fold Parameter 'K'	K fold f1 score
1.	Logistic Regression	10	0.464
2.	Gaussian Naive Bayes	10	0.210
3.	K-Nearest Neighbours	10	0.391
4.	SVM	10	0.449
5.	Decision Trees	10	0.460
6.	MLP	10	0.455

Table 4.6: Mean f1 score through K fold

In comparison with normal train test split methods it can be observed that with the scores obtained due to k-fold cross validation, logistic regression followed by decision trees perform best on our training dataset.

5. Receiver Operating Characteristic Curves (ROC-Curve)

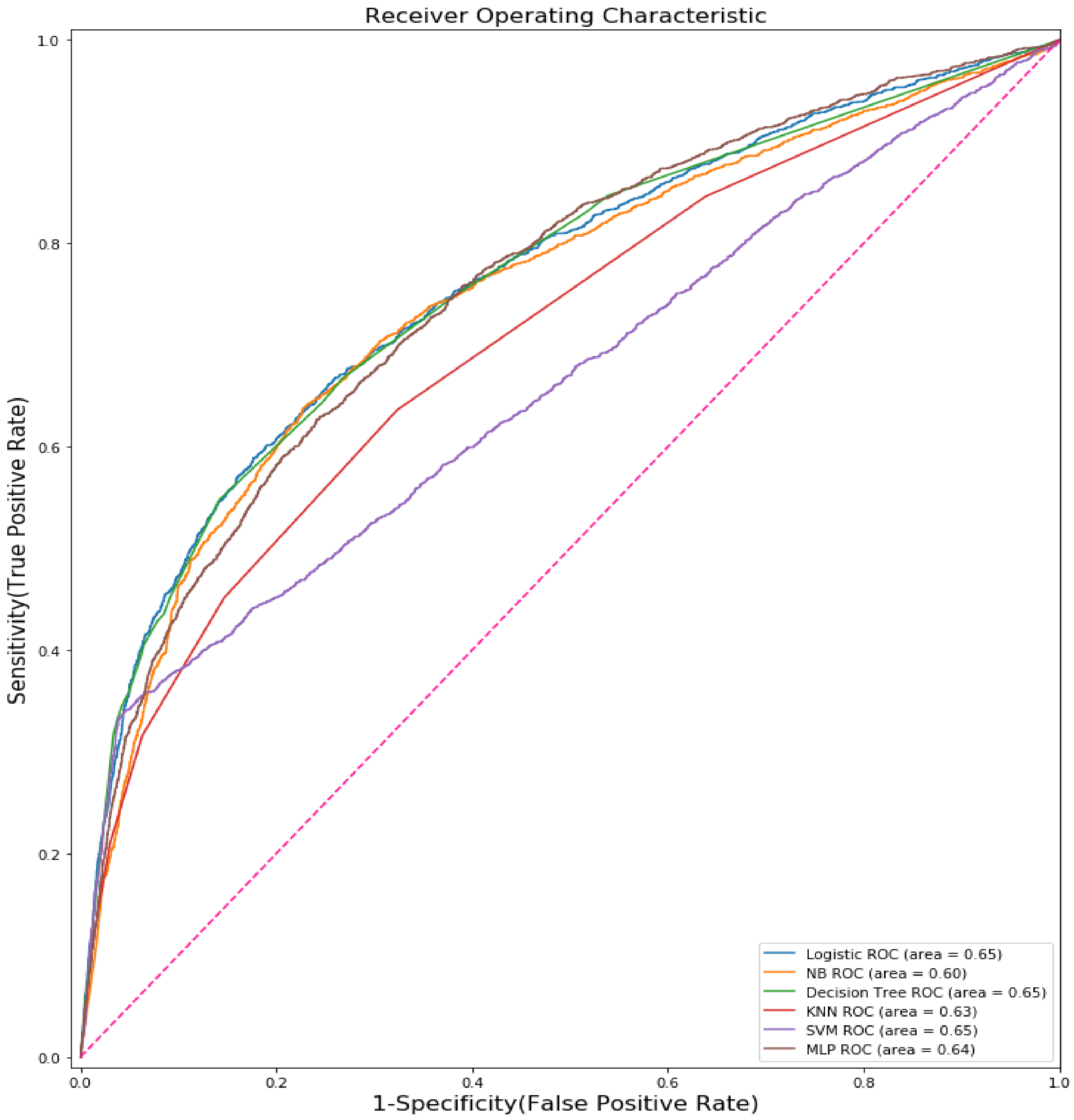


Figure 6.1: Receiver Operating Characteristics graph and their corresponding AUC for 0.5 threshold

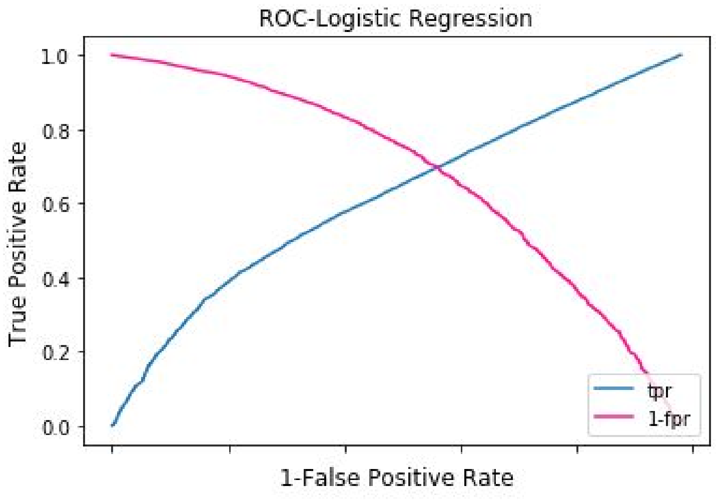
AUC - ROC curve is a performance measurement for classification problems at various thresholds settings. ROC is a probability curve and AUC represents degree or measure of separability. The area under the ROC curve helps to identify the best classifier for the model. Higher the area under curve better is the classification. Among all the six classifiers shown in figure 6.1 which are Logistic Regression, Decision Trees, and SVM have the highest area under the ROC curve which means they classify the data better than others. The ROC curve in figure 6.1 has been made at 0.5 threshold.

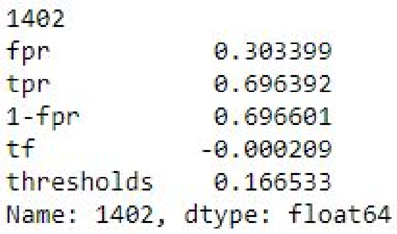
6.1 Finding Threshold for best Algorithms

The optimal cut off would be where TPR(True Positive Rate) is high and FPR(False Positive Rate) is low for Credit card default because to identify the defaulters is more important than classifying a wrong person as defaulter for the bank. If the client who is doing default is not identified then the bank can go bankrupt.

According to the graph shown in figure 6.1 we get three classifiers with the maximum area under the curve which is area=0.65. Further calculating the optimal threshold for the three classifiers

__1)Threshold for logistic regression__





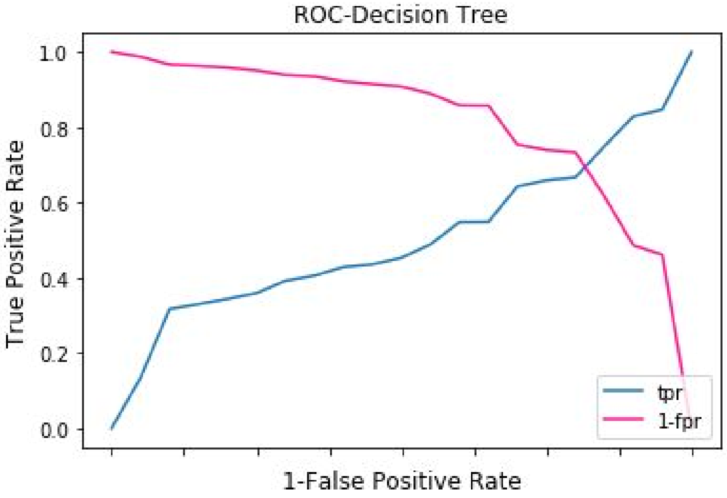
```
1402
fpr          0.303399
tpr          0.696392
1-fpr        0.696601
tf          -0.000209
thresholds   0.166533
Name: 1402, dtype: float64
```

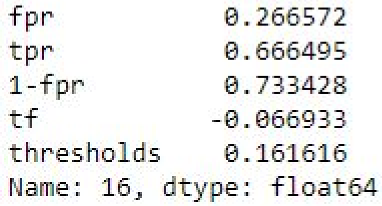
Figure 6.2: Receiver Operating Characteristics for logistic regression

The optimal cut off point for SVM is 0.168721, so anything above this can be labeled as 1 else 0. The TPR (Sensitivity) has a value of 0.696392 and FPR (Specificity) has a value of 0.303399 at threshold.

2) Threshold for Decision Tree

The optimal cut off point for SVM is 0.168721, so anything above this can be labeled as 1else 0. The TPR (Sensitivity) has a value of 0.666495 and FPR (Specificity) has a value of 0.266572 at threshold.



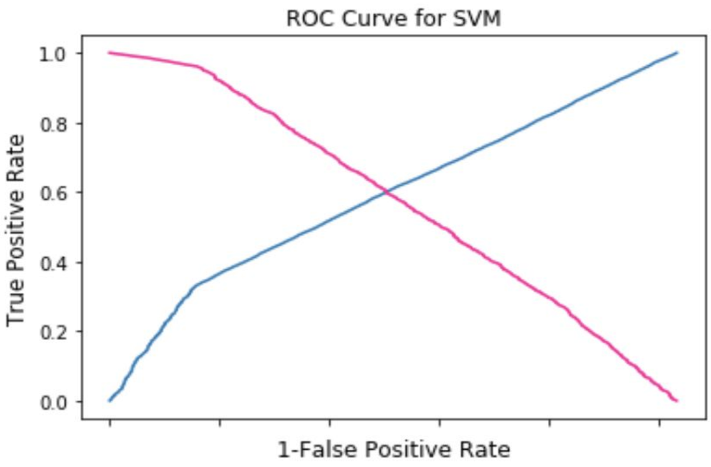


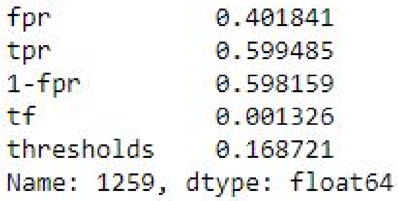
```
fpr            0.266572
tpr            0.666495
1-fpr          0.733428
tf            -0.066933
thresholds     0.161616
Name: 16, dtype: float64
```

Figure 6.3: Receiver Operating Characteristics for decision tree

3) Threshold for support vector machine

The optimal cut off point for SVM is 0.168721, so anything above this can be labeled as 1 else 0. The tpr(Sensitivity) has a value of 0.599485 and fpr(Specificity) has a value of 0.401841 at threshold.





```
fpr           0.401841
tpr           0.599485
1-fpr         0.598159
tf            0.001326
thresholds    0.168721
Name: 1259, dtype: float64
```

Figure 6.4: Receiver Operating Characteristics for Support Vector Machine

6. Solving the problem of Class Imbalance

Two class data are class-imbalanced if the primary class of interest is represented by just a few samples in the dataset, whereas the majority of the samples represent the other class. Credit Card Default Taiwan data is an imbalanced dataset which is quite visible as it has 23364 non-defaulters and 6636 defaulters out of the total 30000. It is skewed towards **non-defaulters** which can affect our prediction through models. Many machine learning algorithms are designed to maximize overall accuracy by default which in case of imbalanced data can result in a havoc. So to tackle this problem the given below methods were used :

1. Upsample (Oversampling)
2. Downsample (Undersampling)

All these methods are used but in the real world most of the data is imbalanced only its very rare to find a balanced data.

6.1 Upsampling

Majority and minority class was separated and then resampling of minority class was done with replacement. N_samples taken as 23364 as that's the majority class. With the help of upsampling more observations can be seen in the dataframe and the ratio of two classes becomes 1:1. Indexing problem was faced when upsampling was done so to solve it reindexing was used to index the data properly in a manner.

Disadvantages of upsampling

The real world data is mostly imbalanced and it's very rare to find balanced data so by oversampling we are **unnecessarily adding more weight to it.**

The problem is that the feature space is not known as to where the new points should be generated. This may lead to **overfitting** as the points generated may not be true representative of the actual minority class.

S.No	Algorithm	Training Accuracy	Testing Accuracy
1.	Logistic Regression	0.704	0.701
2.	Gaussian Naive Bayes	0.613	0.612
3.	K-Nearest Neighbours	0.805	0.728
4.	SVM	0.687	0.683
5.	Decision Trees	0.711	0.701
6.	MLP	0.753	0.723

Table 6.1: Training and Testing Accuracies for Downsampling

6.2 Downsampling

Down-sampling involves randomly removing observations from the majority class to prevent its signal from dominating the learning algorithm. Resampling of the majority class is done without replacement and by setting the number of samples to match that of the minority class which is 6636 out of the total 30000. The new Dataframe has fewer observations than the original and the ratio of the two classes is now 1:1 as the majority and minority have been made equal to 6636 by removing majority class data randomly.

Disadvantages of downsampling

Downsampling can lead to potential loss in relevant information which was there previously. If the minority class is rare or too small, one may end up downsampling the majority class severely and the classifier will be trained on very small data, which may lack any generalization capability and be practically useless for any purpose. But here our data is not too small so it's fine to apply downsampling.

S.No	Algorithm	Training Accuracy	Testing Accuracy
1.	Logistic Regression	0.7064	0.7059
2.	K-Nearest Neighbours	0.755	0.675
3.	Decision Trees	0.715	0.697
4.	MLP	0.713	0.704

Table 6.2: Training and Testing Accuracies for Downsampling

7. Inferences and Conclusions

The main aim of the project was to build a model which can help the banks in predicting Credit Card Default by classifying defaulters and non-defaulters correctly, it was done by analyzing the performance of different algorithms. First step was to investigate the data by using exploratory data analysis techniques including cleaning missing or invalid values and exploring the relationship between different features.

In total six algorithms were compared: Logistic Regression, Gaussian Naive Bayes, K-Nearest Neighbour, Support Vector Machine, Decision Trees and Multilayer Perceptron. Detecting credit card default is a risky problem because the banks want to catch the maximum amount of defaulters so that they can avoid landing themselves into any financial problems and they also want to minimize the wrong classification of defaulters as it affects their image in front of the clients.

Different performance metrics were seen as the data is imbalanced and skewed towards defaulters so accuracy cannot be chosen as the sole metric to evaluate the model.Therefore, recall and precision play an important part here as a good recall value signifies that more number of defaulters are classified correctly and high precision signifies that less false positives are being classified which is good for our case. ROC and F-score are the best metric to analyze our model. Two approaches were used with the imbalanced data namely AGE and AgeBins, by binning the age the defaulter can be classified within some distinctive features and the bank can use that information to select the clients and their credit limit.

Logistic Regression, SVM and Decision Trees showed the best result for classification among all the six algorithms because these three have the highest area under the ROCs. Through K-fold cross validation we saw that Logistic regression followed by a decision tree performs the best on our data in terms of f1-score. Most of the research papers also concluded that Logistic Regression and Tree algorithms showed the best result for credit card default. To remove the class imbalance upsampling and downsampling were performed on the data. In upsampling KNN followed by MLP showed the best accuracy and in downsampling best accuracy was shown by logistic regression followed by MLP.

With the growing number of credit card users, banks have been facing an escalating credit card default rate this model can help the banks to classify credit card default by making better decisions like which features are important when the bank needs to issue a credit card or what should be the credit limit for a particular person. Banks in such a way can make the most of the machine learning models which can contribute in boosting their performance and image in the industry.

8. Future Scope of the Project

In this we only used the data from UCI repository which had the data of a single bank only from 2005, so a bigger data set which consists of data from different countries and banks as well as more attributes will help in making the model more robust and better.

In this dataset cost-sensitive machine learning can be applied to solve the problem produced due to class imbalance and skewness of the data.

Generating Synthetic Samples is also a good method to sort out the imbalance of data, a technique similar to upsampling is Synthetic Minority Oversampling Technique. SMOTE is used for it and it uses a nearest neighbours algorithm to generate new and synthetic data.

9. Bibliography

1. https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients
2. https://www.kaggle.com/uciml/default-of-credit-card-clients-dataset#UCI_Credit_Card.csv
3. https://www.researchgate.net/publication/326171439_Default_Payment_Analysis_of_Credit_Card_Clients
4. https://www.researchgate.net/publication/328026972_Application_of_Machine_Learning_Algorithms_in_Credit_Card_Default_Payment_Prediction
5. https://nycdatascience.com/blog/student-works/credit-card-clients/
6. https://ieeexplore.ieee.org/document/8776802
7. https://www.researchgate.net/publication/328026972_Application_of_Machine_Learning_Algorithms_in_Credit_Card_Default_Payment_Prediction
8. https://escholarship.org/uc/item/9zg7157q
9. https://www.r-bloggers.com/logistic-regression-in-r-a-classification-technique-to-predict-credit-card-default/