Predictive Modelling for Default of Credit Card Users

Submitted By:

Divyesh Sethiya

Under Esteemed Guidance of

Mr. Ritesh Maury

TABLE OF CONTENTS

INTRODUCTION

Recently, the state vigorously promotes the economic construction of large-sized and medium-sized cities, they are now more focused on increasing people living standards and they are not limit to those only they are also emphasizing to make people spend more money. And we are enjoying the same and here comes the role of credit card, we as a human has a mentality to enjoy the service earlier and payment after options, however we sometimes overestimate our ability to repay the loans back to the banks in time, later it leads to either addition of interest or making default.

Now let‟s discuss the Banks or Credit card company part of it- With the huge number of banks selling credit card it also invites the increasing rate of making defaults. And with the huge number of users banks are finding effective way of identifying the customers who may make a default, so that the company can make their further financial decisions whether it is of increasing the limit of the person or giving credit to the new customers or using their funds elsewhere.

Credit card is one of the great revenue generators for the banking institutions and the credit card ownership is widely accepted. Some of the leaders in this market is Visa, MasterCard and Discover which acquires 70% of the market in US and till 2019 around $970 billion credit card debt has already aroused and this is increasing year by year with at least 3-4%. Credit card is not like the loan which we take for house or business which we care a lot, it is used for expenses which we do in day-to-day life and thus it became a necessary part of the day. With this advent use people sometimes forget to make the payment or they mistook their power of paying back and start making defaults and the banks starts charging interest on those defaults and sometime it becomes bad-debts which lets banks suffers a lot. Statistics shows around 3.8% of the customers make defaults in the year 2019.

To overcome this situation Bank are trying every possible resource to predict and forestall default behaviour. Earlier it wasn‟t possible as there were no such parameters to find and owing a credit card was such a hectic work. Earlier Loan officer where have all the authorities whether to give credit to whom and sometimes their predictions were personal and banks has to make sacrifice for that but now the machine learning can do all the worksstarting from their eligibility to finding their credit score and the financial history. And with the introduction of non-banking institutions, scrutiny is not done properly as their only priority is adding new customers; they care less about defaults these models can help them to make relevant forecasting about their payments and defaults.

This paper will address the primary contributing factors to default probability by means of the different classification models.

INDUSTRY REVIEW

Introduction to Domain:

Risk analytics always plays a major role in banking as most of their activities revolves around granting loans, credit card loan investments and mortgage. Identifying the customers who are defaulting such credit card bill payment is a matter of interest for Bank as it is one of their major booming services.

We all have witnessed some news in the past that banking industry facing the problems of not getting credit card payments on right time and also with the increasing rate ofcredit card users the problem is increasing to give the credit card to right customers. We will be using Supervised Machine Learning Models to find the right customers for the banks who is supposed to pay the bill at given time without fail. Credit default risk management is essential to financial institutions as it directly effects business. Now a days with the advance support of big data technology various platforms have brought opportunities to business butat the same time they are also facing risk of credit card defaults which directly relates to sustainable and healthy platform building. One of the major problems faced by this sector is ever changing behaviour of economy is the increasing rate of defaults the banking authorities are finding it more difficult to correctly access the credit request and overcome the risk of people non-defaulting the credit card.

The most critical questions in this risk analytics industry are as follows:

- How risky the Credit card user is?
- Considering the history of Credit card user should we extend his/her limit or should we lend him credit another time.

In the given problem, machine learning model predicts whether an individual should be given more credit, less credit or should we consider him to drop off the credit card by assessing given attributes and hence help the sector by easing their process.

LITERATURE REVIEW

Prediction of Default Probability of Credit-Card Bills

https://www.scirp.org/journal/paperinformation.aspx?paperid=97459

According to information economics, if the information related to a transaction is asymmetrical, the progenitors of "adverse selection" and "moral hazard" will arise, which are important factors for the formation of default risk. Research by American economists Stiglitz and Weiss showed that adverse selection and im-proper incentives always exist in the credit market. In terms of information, broadly speaking, banks and other lenders are in a position of relative weakness while borrowers are in a position of comparative strength. Banks often do not know the borrower"s repayment motivation, repayment ability and "project risk". It is hardly a secret that, to obtain a short-term advantage, certain borrowers distort or conceal negative information in their dealings with lenders; more positive "information" may simply be invented. Financial institutions are aware of their information disadvantage. As a result, they tend to charge higher interest rates, which crowd out credit worthy borrowers, while those who continue actively looking for loans are more likely to represent potential non- performing loan risks. In other words, good clients can be squeezed out of the credit marketby poor ones. On the other hand, when financial institutions do provide borrowers with funds, those borrowers may have a strong motivation to hide certain facts of operation and profitability in order to obtain economic benefit. If the borrower uses the loan for high-risk investment projects after the loan contract is signed, once that project fails, he/she transfersthe risk to the bank. The borrower may simply refuse to pay either principal or interest, thus effectively avoiding the debt obligation. As a result of this "moral hazard", the financial institution must absorb the loss of both interest and principal.

Benefits of Relationship Banking: Evidence From Consumer Credit Markets

https://www.sciencedirect.com/science/article/abs/pii/S0304393218300928?via%3Dihub

Using a unique dataset that contains various information about the credit card customers, we show that relationship accounts exhibit lower probabilities of default and attrition, and have higher utilization rates, than non-relationship accounts. Dynamic information about changes in the behaviour of a customer's other accounts at the same bank helps predict the behaviour of the credit card account over time. These results imply that relationship banking offers significant potential benefits to banks: information the lender has at its disposal can be used to mitigate credit risk on the credit card account.

Impact on Business

Information provided by RBI suggests a very interesting factor pertaining to the future of card business in various parts of country. India has 6.7 card payments per inhabitant, where as it is very number intensive for developed countries like Australia (249.3), Canada (247.9), Korea (260.8), France (143.4), UK (201.7), China (14.4), Russia (47), etc. The table shows gradual increase in card uses across the country where POS means is a hardware system for processing card payments at retail locations.

So, with increase of credit business, where it is business need to provide credit to more riskier customers and the whole business revolves around their timely payment of the credit provided. With advent of certain non-banking entities like Simpl, LazyPay etc., proper verification is not done before giving credits. There is an increase of defaults due to these marketing techniques of acquiring more customers.

Predicting these defaults beforehand can be proved very healthy for business by avoiding delivering credit to such customers or decreasing the amount of credit so that losses will be minimal in case default. Prediction is done by observing various pattern seen among previous customers who have either defaulted or delayed the repayment process.

Previous Review

Yeh and Lien applied a had published a research paper in 2009 and they had used 6 models to compare the accuracy of credit-card clients default probability and the models were: K-nearest Neighbor (KNN) classifiers, logistic regression (LR), discriminant analysis (DA), naive Bayesian (NB) classifiers, artificial neural networks (ANNs) and classification trees (CTs).

There was not so much difference between the scores of all the models performed. But among all artificial neural networks (ANNs) did the better job and gives more accurate results.

In 2016 another paper was published by Vendakesh and Jacob where they used Random Forest classifier and those give the best accuracy rate among all the models done earlier.

As the Credit card uses is increasing drastically the delinquency rate is also increasing hand in hand and the small Banks which are not able to predict future as they don‟t have the good mapping techniques, they suffer a lot as we can see from the image below. It is also observed that small banks in order to acquire more customer and increasing their consumer base they don‟t do proper verification and provide the loans for short-term gains. As sometime the customer couldn‟t able to do the repayment it converts to bad debts or long-term debts for the banks.

PROBLEM STATEMENT

To build a classification model to find whether a bank customer will default their credit card loan payment for the next month or not.

With advent of credit business, it has become necessary to recognize the future defaults from the certain patterns shown by previous defaulters. The dataset in this projectwill help us to analyse the inherent attributes only specific to persons defaulting it using machine learning algorithms.

Data Imported from UCI Machine Learning Repository.

Link: https://archive.ics.uci.edu/ml/machine-learning-databases/00350/

About Data

This 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.

This research aimed at the case of customers default payments in Taiwan and compares the predictive accuracy of probability of default among six data mining methods. From the perspective of risk management, the result of predictive accuracy of the estimated probability of default will be more valuable than the binary result of classification - credible or not credible clients. Because the real probability of default is unknown, this study presented the novel Sorting Smoothing Method to estimate the real probability of default. With the real probability of default as the response variable (Y), and the predictive probability of default as the independent variable (X), the simple linear regression result (Y = A + BX) shows that the forecasting model produced by artificial neural network has the highest coefficient of determination; its regression intercept (A) is close to zero, and regression coefficient (B) to one. Therefore, among the six data mining techniques, artificial neural network is the only one that can accurately estimate the real probability of default.

Steps To Be Taken While Solving the Problem:

1. Scrap the data from various resources.
2. Understand the data, the features present in data and rows also.
3. Clean the data. Do the EDA process on it.
    - Checking data types
    - Check the anomalies

- Check outliers
- Treating the data point wherever needed.

4. Visualize the data to understand better.
    - Plotting various plots to understand relationship between each other.
    - Plot various plots to find the pattern and nature.
5. Hypothesis testing to check Independency.
6. Modelling
    a. Fit the base model.
        i. Logistic Regression
        ii. Decision Tree
        iii. Naïve Bayes
        iv. KNN
    b. Draw Instances
    c. Hyper Parameter Tuning
7. Evaluation of the performance of model
8. Conclusions.

DATA DICTIONARY

Variable Identification:

Independent Variables: There are 24 independent variables listed below.

1. ID	2. LIMIT_BAL
3. SEX	4. EDUCATION
5. MARRIAGE	6. AGE
7. PAY_0	8. PAY_2
9. PAY_3	10. PAY_4
11. PAY_5	12. PAY_6
13. BILL_AMT1	14. BILL_AMT2
15. BILL_AMT3	16. BILL_AMT4
17. BILL_AMT5	18. BILL_AMT6
19. PAY_AMT1	20. PAY_AMT2
21. PAY_AMT3	22. PAY_AMT4
23. PAY_AMT5	24. PAY_AMT6

Target Variables:

1. default.payment.next.month

Variable Information/Data Description:

Variable	Data Type	Description
1. ID	int64	ID number of client
2. LIMIT_BAL	float64	Limit given to holder (NT dollars)
3. SEX	int64	Gender (1=Male, 2=Female)
4. EDUCATION	int64	Educational Qualification (0=Unknown, 1=Graduate School, 2=University, 3=High School, 4=Others, 5=Unknown, 6=Unknown)
5. MARRIAGE	int64	Marital Status (0=Unknown, 1=Married, 2=Single, 3=Others)
6. AGE	int64	Age in years
7. PAY_0	int64	History of Past Payment (Repayment status in September 2005) (-2= Unknown, -1= Pay Duly, 0 = Unknown, 1= One Month Delay, 2= Two Months Delay, 3=

		Three Months Delay, 4= Four Months Delay, 5= Five Months Delay, 6= Six Months Delay, 7= Seven Months Delay, 8= Delay of Eight or above Months)
8. PAY_2	int64	History of Past Payment (Repayment status in August 2005) (-2= Unknown, -1= Pay Duly, 0 = Unknown, 1= One Month Delay, 2= Two Months Delay, 3= Three Months Delay, 4= Four Months Delay, 5= Five Months Delay, 6= Six Months Delay, 7= Seven Months Delay, 8= Delay of Eight or above Months)
9. PAY_3	int64	History of Past Payment (Repayment status in July 2005) (-2= Unknown, -1= Pay Duly, 0 = Unknown, 1= One Month Delay, 2= Two Months Delay, 3= Three Months Delay, 4= Four Months Delay, 5= Five Months Delay, 6= Six Months Delay, 7= Seven Months Delay, 8= Delay of Eight or above Months)
10. PAY_4	int64	History of Past Payment (Repayment status in June 2005) (-2= Unknown, -1= Pay Duly, 0 = Unknown, 1= One Month Delay, 2= Two Months Delay, 3= Three Months Delay, 4= Four Months Delay, 5= Five Months Delay, 6= Six Months Delay, 7= Seven Months Delay, 8= Delay of Eight or above Months)
11. PAY_5	int64	History of Past Payment (Repayment status in May 2005) (-2= Unknown, -1= Pay Duly, 0 = Unknown, 1= One Month Delay, 2= Two Months Delay, 3= Three Months Delay, 4= Four Months Delay, 5= Five Months Delay, 6= Six Months Delay, 7= Seven Months Delay, 8= Delay of Eight or above Months)
12. PAY_6	int64	History of Past Payment (Repayment status in April 2005) (-2= Unknown, -1= Pay Duly, 0 = Unknown, 1= One Month Delay, 2= Two Months Delay, 3= Three Months Delay, 4= Four Months Delay, 5= Five Months Delay, 6= Six Months Delay, 7= Seven Months Delay, 8= Delay of

		Eight or above Months)
13. BILL_AMT1	float64	Amount of Bill Statement (September 2005) (in NT Dollars)
14. BILL_AMT2	float64	Amount of Bill Statement (August 2005) (in NT Dollars)
15. BILL_AMT3	float64	Amount of Bill Statement (July 2005) (in NT Dollars)
16. BILL_AMT4	float64	Amount of Bill Statement (June 2005) (in NT Dollars)
17. BILL_AMT5	float64	Amount of Bill Statement (May 2005) (in NT Dollars)
18. BILL_AMT6	float64	Amount of Bill Statement (April 2005) (in NT Dollars)
19. PAY_AMT1	float64	Amount of Previous Payment in September 2005 (in NT Dollars)
20. PAY_AMT2	float64	Amount of Previous Payment in August 2005 (in NT Dollars)
21. PAY_AMT3	float64	Amount of Previous Payment in July 2005 (in NT Dollars)
22. PAY_AMT4	float64	Amount of Previous Payment in June 2005 (in NT Dollars)
23. PAY_AMT5	float64	Amount of Previous Payment in May 2005 (in NT Dollars)
24. PAY_AMT6	float64	Amount of Previous Payment in April 2005 (in NT Dollars)
25. default.payment.next.month	int64	Default Payment (0=No, 1= Yes)

Variable Categorization:

A variable can be defined as something that is subject to change. There are two types of variables, namely:

- Numerical
- Categorical

Numerical data refers to the data that is in the form of numbers, and not in any language or descriptive form. There are about 30000 rows and 25 columns in the dataset in which there are about 15 numerical columns and 9 categorical columns.

With initial analysis it has been found that there is total 25 columns in our data, out of which “ID” column seems to be redundant. So, after removing the given column we are left with 24 columns. As already mentioned, we have a categorical target column i.e. “default_payment_next_month”.

Rest 23 columns are further segregated into numerical and categorical type after fixing their initial datatypes. Resultant columns are summarized below.

- **Numerical Variables:**

  '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'.

- **Categorical Variables:**

  'SEX', 'EDUCATION', 'MARRIAGE', 'PAY_0', 'PAY_2','PAY_3', 'PAY_4', 'PAY_5', 'PAY_6'.

PRE-PROCESSING DATA ANALYSIS

Data Pre-processing can refer to manipulation or dropping data before it is used in order to ensure or enhance performance. It is an important step in the Data Mining Process.

Data Understanding:

As our data is quite big in size, around 30000 data points, we need perform Exploratory Data Analysis. And gather various insights from the data to understand user behaviour. After performing EDA operations, we will modify the data according to the standard practices for model building. Check for correlation between the various features.

Data Preparation:

After we are done with the EDA, we will get to know about the necessary features for building our efficient model. We take the necessary steps according:

- Checking for null values and deleting the insignificant features from the dataset.
- Checking the data if it‟s normally distributed.
- Split the data into training and testing.
- Standardize the Training dataset.

Redundant Columns:

Some columns that have totally unique values / those which are not required for the analysis are considered redundant and are dropped.

Missing Value Analysis:

There are no missing values are found in features.

LIMIT_BAL	0
SEX	0
EDUCATION	0
MARRIAGE	0
AGE	0
PAY_0	0
PAY_2	0
PAY_3	0
PAY_4	0
PAY_5	0
PAY_6	0
BILL_AMT1	0
BILL_AMT2	0
BILL_AMT3	0
BILL_AMT4	0
BILL_AMT5	0
BILL_AMT6	0
PAY_AMT1	0
PAY_AMT2	0
PAY_AMT3	0
PAY_AMT4	0
PAY_AMT5	0
PAY_AMT6	0
default.payment.next.month	0

Unique datapoints in each categorical column are present in the data. We can see that below.

```
SEX [2 1]

EDUCATION [2 1 3 5 4 6 0]

MARRIAGE [1 2 3 0]

PAY_0 [ 2 -1  0 -2  1  3  4  8  7  5  6]

PAY_2 [ 2  0 -1 -2  3  5  7  4  1  6  8]

PAY_3 [-1  0  2 -2  3  4  6  7  1  5  8]

PAY_4 [-1  0 -2  2  3  4  5  7  6  1  8]

PAY_5 [-2  0 -1  2  3  5  4  7  8  6]

PAY_6 [-2  2  0 -1  3  6  4  7  8  5]
```

We already describe the unique data points in data dictionary for above. We will be doing some work on that while doing further analysis.

Data Preparation

	count	mean	std	min	25%	50%	75%	max
LIMIT_BAL	30000.0	167484.322667	129747.661567	10000.0	50000.00	140000.0	240000.00	1000000.0
SEX	30000.0	1.603733	0.489129	1.0	1.00	2.0	2.00	2.0
EDUCATION	30000.0	1.853133	0.790349	0.0	1.00	2.0	2.00	6.0
MARRIAGE	30000.0	1.551867	0.521970	0.0	1.00	2.0	2.00	3.0
AGE	30000.0	35.485500	9.217904	21.0	28.00	34.0	41.00	79.0
PAY_0	30000.0	-0.016700	1.123802	-2.0	-1.00	0.0	0.00	8.0
PAY_2	30000.0	-0.133767	1.197186	-2.0	-1.00	0.0	0.00	8.0
PAY_3	30000.0	-0.166200	1.196868	-2.0	-1.00	0.0	0.00	8.0
PAY_4	30000.0	-0.220667	1.169139	-2.0	-1.00	0.0	0.00	8.0
PAY_5	30000.0	-0.266200	1.133187	-2.0	-1.00	0.0	0.00	8.0
PAY_6	30000.0	-0.291100	1.149988	-2.0	-1.00	0.0	0.00	8.0
BILL_AMT1	30000.0	51223.330900	73635.860576	-165580.0	3558.75	22381.5	67091.00	964511.0
BILL_AMT2	30000.0	49179.075167	71173.768783	-69777.0	2984.75	21200.0	64006.25	983931.0
BILL_AMT3	30000.0	47013.154800	69349.387427	-157264.0	2666.25	20088.5	60164.75	1664089.0
BILL_AMT4	30000.0	43262.948967	64332.856134	-170000.0	2326.75	19052.0	54506.00	891586.0
BILL_AMT5	30000.0	40311.400967	60797.155770	-81334.0	1763.00	18104.5	50190.50	927171.0
BILL_AMT6	30000.0	38871.760400	59554.107537	-339603.0	1256.00	17071.0	49198.25	961664.0
PAY_AMT1	30000.0	5663.580500	16563.280354	0.0	1000.00	2100.0	5006.00	873552.0
PAY_AMT2	30000.0	5921.163500	23040.870402	0.0	833.00	2009.0	5000.00	1684259.0
PAY_AMT3	30000.0	5225.681500	17606.961470	0.0	390.00	1800.0	4505.00	896040.0
PAY_AMT4	30000.0	4826.076867	15666.159744	0.0	296.00	1500.0	4013.25	621000.0
PAY_AMT5	30000.0	4799.387633	15278.305679	0.0	252.50	1500.0	4031.50	426529.0
PAY_AMT6	30000.0	5215.502567	17777.465775	0.0	117.75	1500.0	4000.00	528666.0
default.payment.next.month	30000.0	0.221200	0.415062	0.0	0.00	0.0	0.00	1.0

1. Dropping the Irrelevant Column i.e. „ID“ column.

2. Finding description of data.
    a. LIMIT_BAL: Limit Bal is the amount offered to the customer that he/she can spend and it ranges from 10000 to 1000000.

    b. AGE:
        i. Min = 21
        ii. Max = 79

    c. BILL_AMT: Bill_Amt are the amount that the Card Holder had spent using credit card from April to September.
        i. Bill_Amt1 ranges from: -165580 to 964511
        ii. Bill_Amt2 ranges from: -69777 to 983931
        iii. Bill_Amt3 ranges from: -157264 to 1664089
        iv. Bill_Amt4 ranges from: -170000 to 891586
        v. Bill_Amt5 ranges from: -81334 to 927171
        vi. Bill_Amt6 ranges from: -339603 to 961664

    d. PAY_AMT: Pay_Amt are the amount that the Card Holder had paid to the Credit card Company for each month.
        i. Pay_Amt1 ranges from: 0 to 873552
        ii. Pay_Amt2 ranges from: 0 to 1684259
        iii. Pay_Amt3 ranges from: 0 to 896040
        iv. Pay_Amt4 ranges from: 0 to 621000
        v. Pay_Amt5 ranges from: 0 to 426529
        vi. Pay_Amt6 ranges from: 0 to 528666

3. Checking Null / Missing Values.
    a. No missing or null values present in dataset.

4. Changing the Feature Name and Value wherever needed. To make interpretation easy.
    a. „default.payment.next.month“ To „Default“
    b. „PAY_0“ To „PAY_1“
    c. In SEX Feature:
        i. 1 = Male, remained unchanged.
        ii. 2 = Female, changed to 0.

5. Clubbing the categories to each other where Information of category is not provided.
    a. EDUCATION
        i. „0, 5, 6“ clubbed to „4“.
        Here information for 0, 5, and 6 is not provided so club these values to 4 i.e. „others“ and these values counts are low. So decided to take this step.

    b. MARRIAGE

i. „0“ clubbed to „3“.

Here information for 0 is not provided so clubbing it to „others“.

Project Justification

1. **Project Statement:**

As the credit lending business of the banking industry has seen extreme competitions from numerous credit stratus. Credit loans are usually provided for many purposes some of which are personal use, educational purposes, medical purposes, travelling and business purposes. The banks need to make quick decision to grab the opportunity. This is where credit card company is failing to identify potential red flag in the customer“s portfolio.

2. **Complexity Involved:**

The challenge is to recognize Loan default applicants so that the bank does not grant loans to credit applicants who have bad credit history i.e. CIBIL score. Main challenges involved in loan default detection:

Enormous Data: The major problem is lot of data is processed every day and the model build must be fast enough so that Bank doesn“t give out loan to defaulter applicants.

Imbalanced Data: As sometimes the banks miss out to give loans to low credit score applicants who have never defaulted on their loan payments.

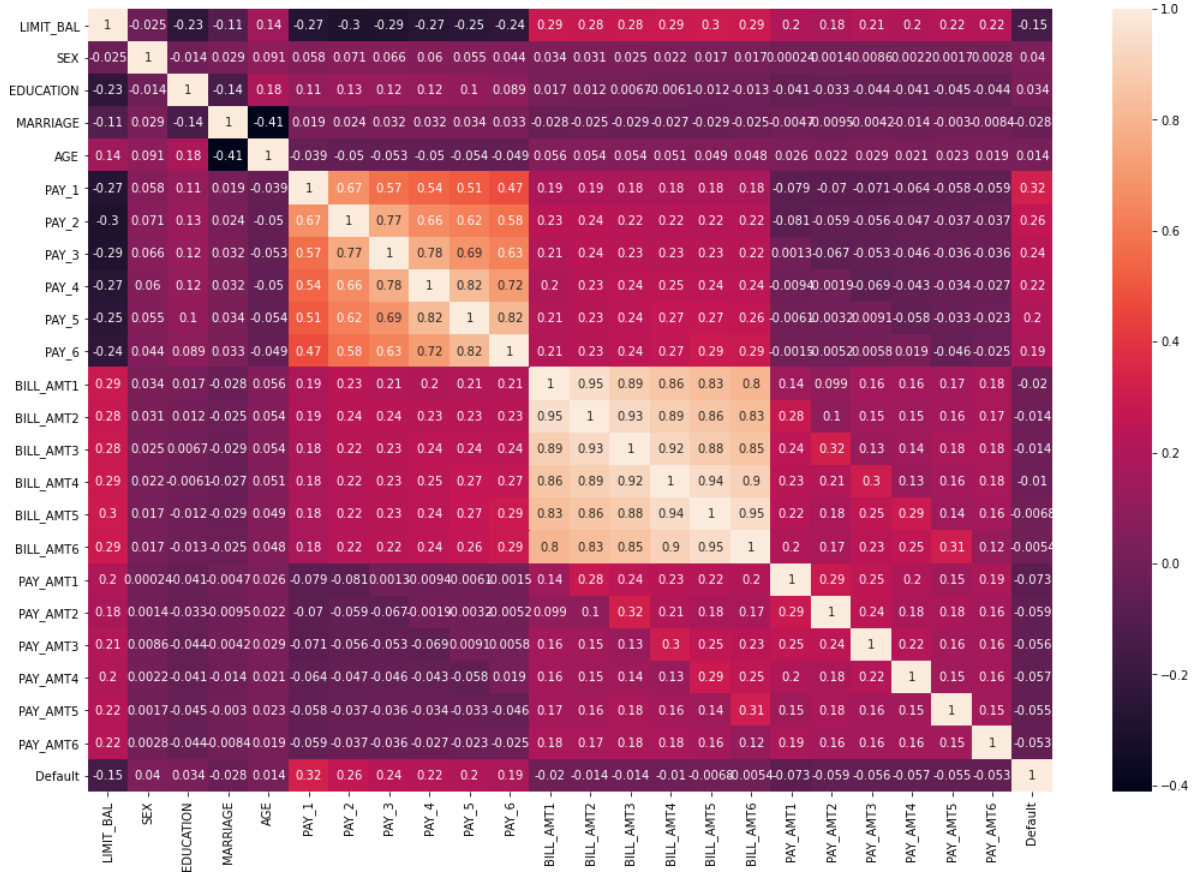
Data availability: As per government and banks policy and due to high competition, the banks don“t share the customer data.

3. **Project Outcome:**

Outcome of the project, banks or credit card providers will easily ably detect customers who will make default next month or in future by the given data. And also, these sectors will be able to utilize the funds elsewhere such as providing credit to the new customers rather than increasing the limits of existing risky customers. For this understanding, we are going to perform some machine learning classification models such as Logistic Regression, Decision Tree, KNN, etc.

EXPLORATORY DATA ANALYSIS

Plotting heat map to observe linear relationship with target variable, also presence of multicollinearity present among independent variables.



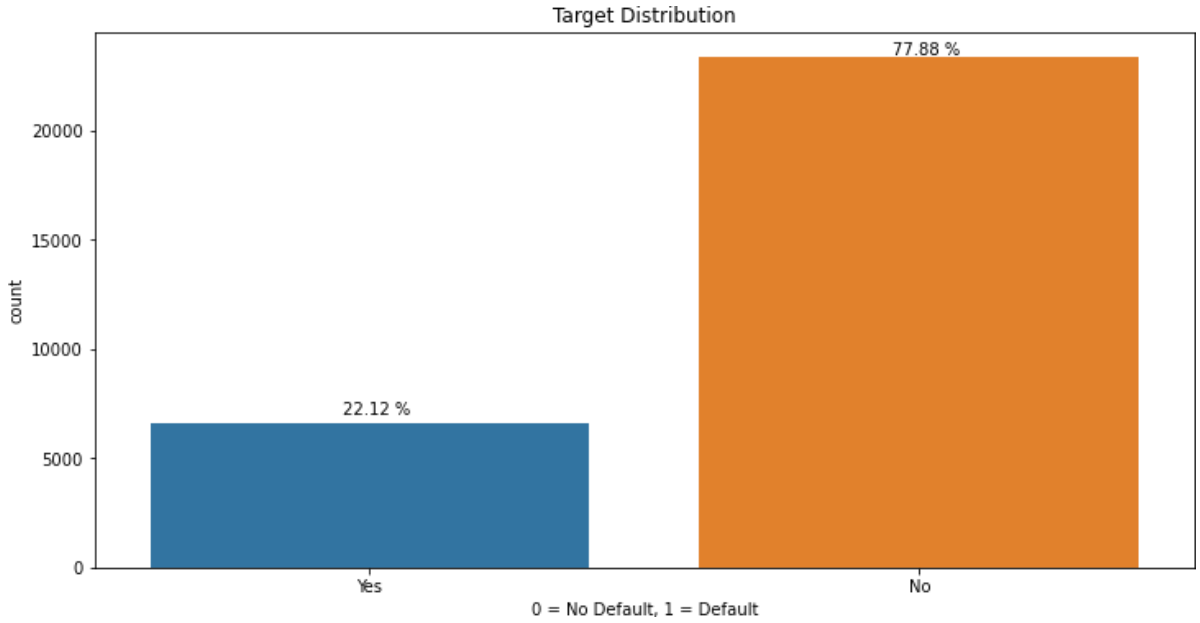
From this heatmap we can infer

- Pay_1 is the most correlated column with the Target Variable.
- All the PAY Columns (PAY_1, PAY_2, PAY_3, PAY_4, PAY_5, PAY_6) are correlated among each other.
- BILL_AMT Columns (BILL_AMT1, BILL_AMT2, BILL_AMT3, BILL_AMT4, BILL_AMT5, BILL_AMT6) are correlated among each other.

Relationship between Variables:

- **Univariate Analysis:**
  Univariate data visualization plots help us comprehend the enumerative properties as well as a descriptive summary of the particular data variable. These plots help in understanding the location/position of observations in the data variable, its distribution, and dispersion.

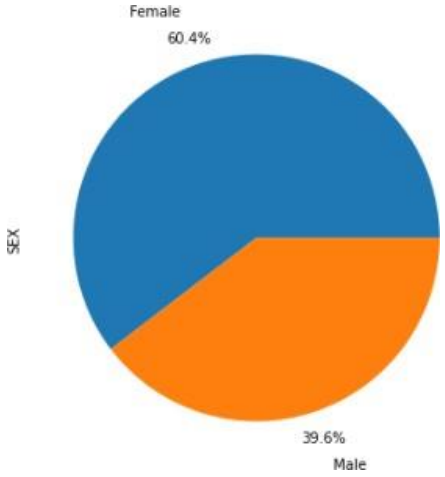
Target Distribution (Default):



Insights:

1. Around 6700 people are fall into „Yes“ category in Default.
2. Around 23000 people are fall into „No“ category in Default.

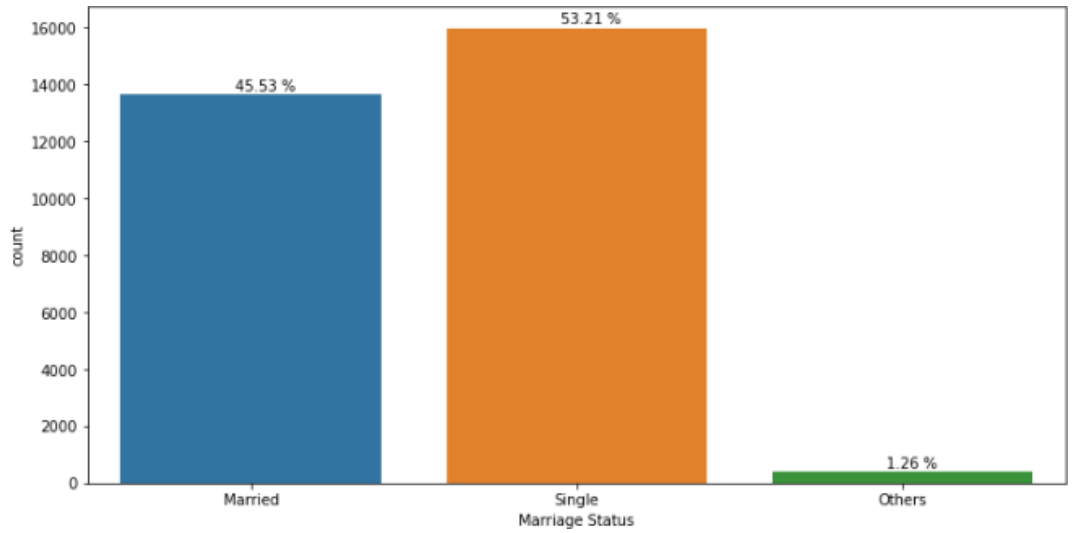
SEX Variable:



Insights:

1. 18112 around 60.4% are Female population.
2. 11888 around 39.6% are Male population.

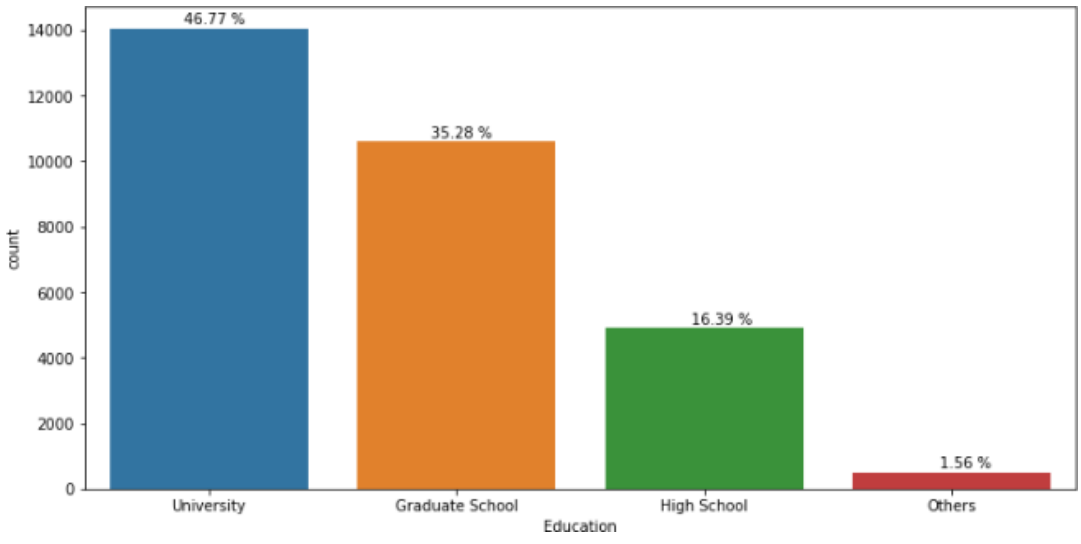
MARRIAGE Variable:



Insights:

1. 53.2% are Singles.
2. 45.5 % are Married.
3. 1.26 % are Others.

EDUCATION Variable:



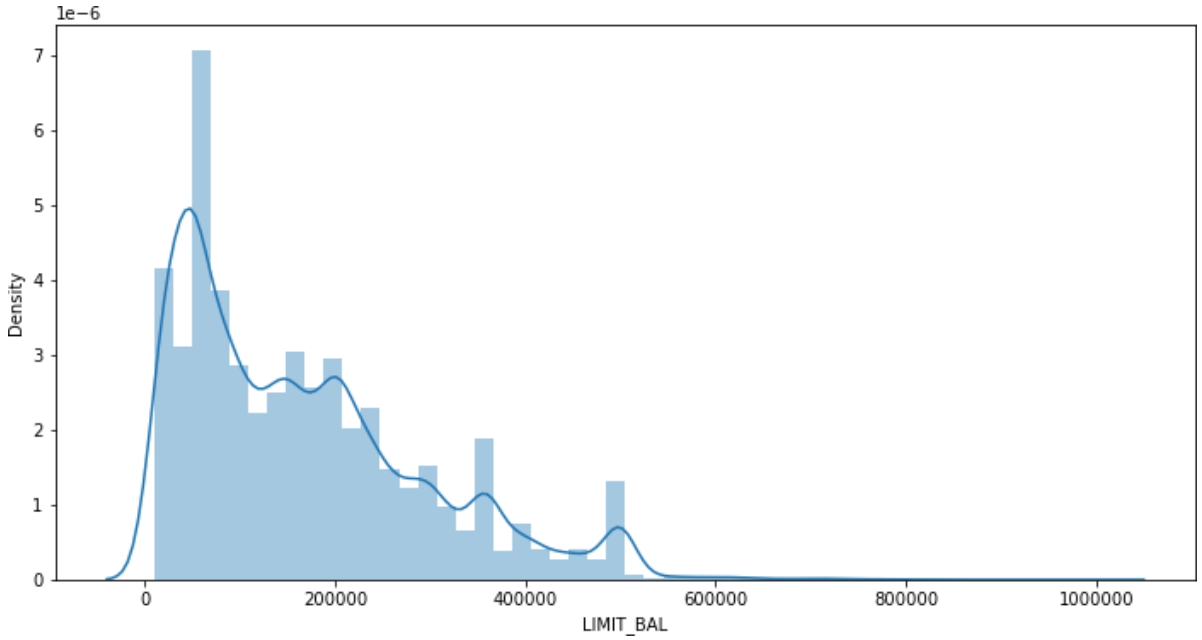
Insights:

In the Education Column there are 4 Values High School, Graduate School, University and Others.

1. University Contributes 46.77 %.

2. Graduate School contribute 35.28 %.
3. High School contributes 16.39 %.
4. Others contribute 1.56 %.

LIMIT_BAL Variable:

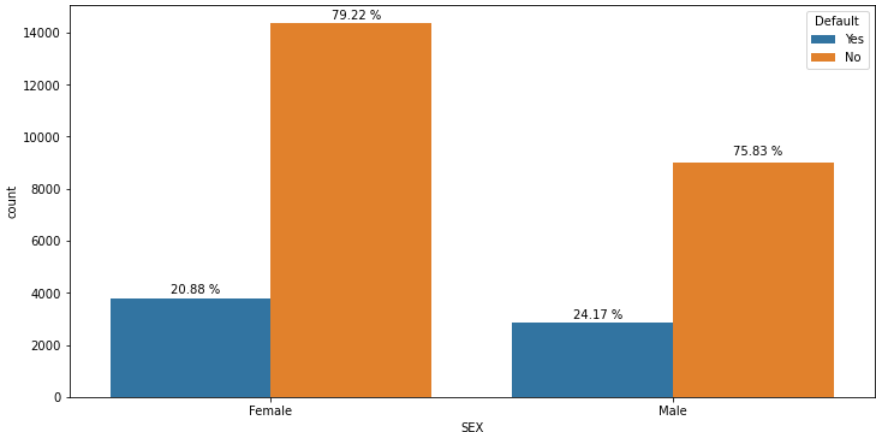


Insights:

Most of the customers have less than 200000 Credit Limit.

- **<u>Bivariate Analysis:</u>**
  Bivariate analysis is one of the statistical analyses where two variables are observed. One variable here is dependent while the other is independent. These variables are usually denoted by X and Y. So, here we analyse the changes that occurred between the two variables and to what extent.

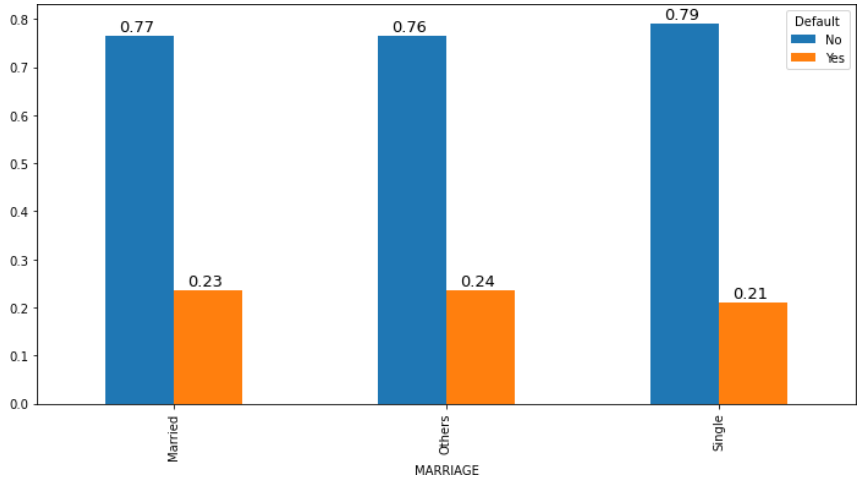
SEX vs Default Variables:



Insights:

1. There are more female than male in our dataset.
2. Men have slightly more chance of default.
3. Around 24% of male customers and around 20.8% of female customers are default.
4. From this we can say that Males are More Credit Card Defaulters.

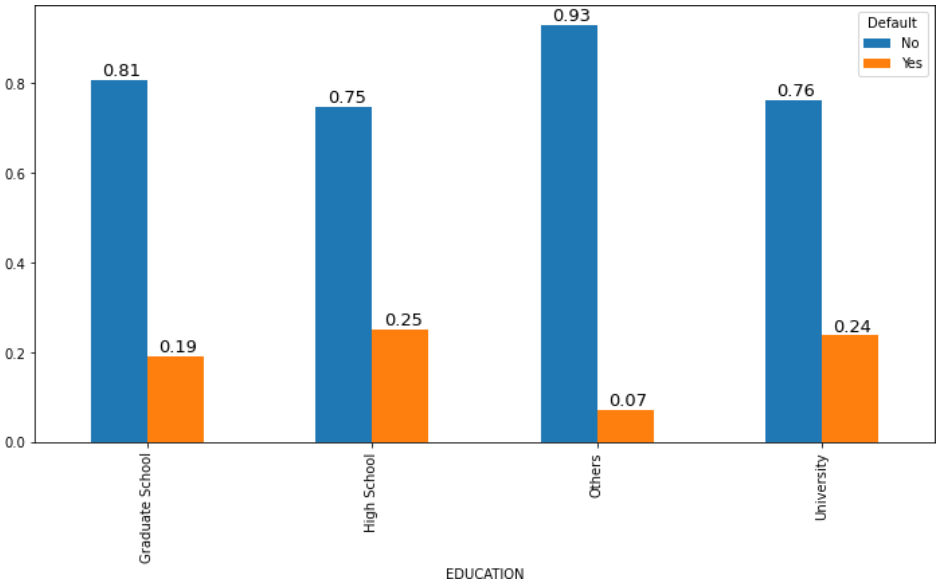
MARRAIGE vs Default Variables:



Insights:

1. In the Marriage Column 53.2% are Singles.
2. 45.5 % are Married.
3. 1.26 % are Others.
4. In Singles 0.79 % are Not Credit Card Defaulters and 0.21 are Credit Card Defaulters.
5. In Others 0.76 % are not Credit Card Defaulters while 0.24 % are credit card Defaulters.
6. In Marriage 0.77 are not Credit card Defaulters and 0.23 % are Credit Card Defaulters.
7. In Marriage Column as Married People contribute 45.53 % of Total Population and their Default rate is 0.23 so they are more defaulters.

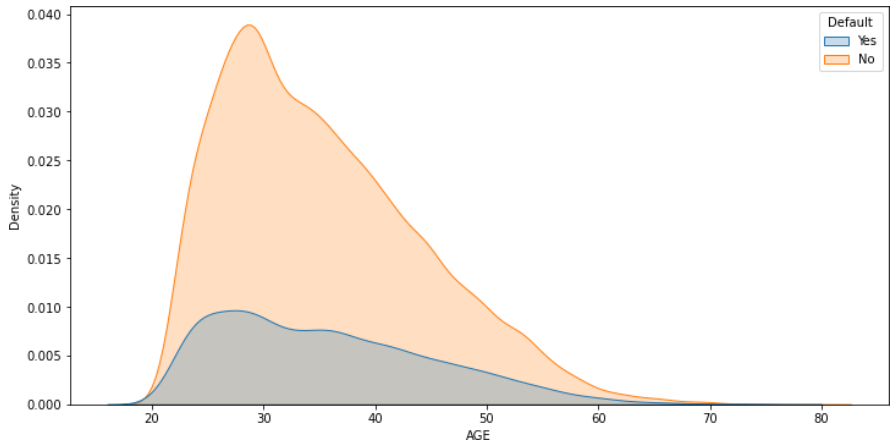
EDUCATION vs Default Variables:



Insights:

1. In the Education Column there are 4 Values High School, Graduate School, University and Others.
2. University Contributes 46.77 %, Graduate School contributes 35.28 %, High School Contributes 16.39 % and Others contributes 1.56 %
3. In Graduate0.81 % are not defaulters and 0.19 % are Defaulters. In High School 0.75 are no credit card defaulters and 0.25 are Defaulters. In University 0.76 % are Defaulters and 0.24 % are not Defaulters. In Others 0.93 % are not Defaulters and 0.07% are Defaulters.
4. As University Contribution is More in the Population 0.24 are Defaulters. Thus, we can say that they are most Defaulters. Afterwards High School with 0.25 % are maximum Defaulters.

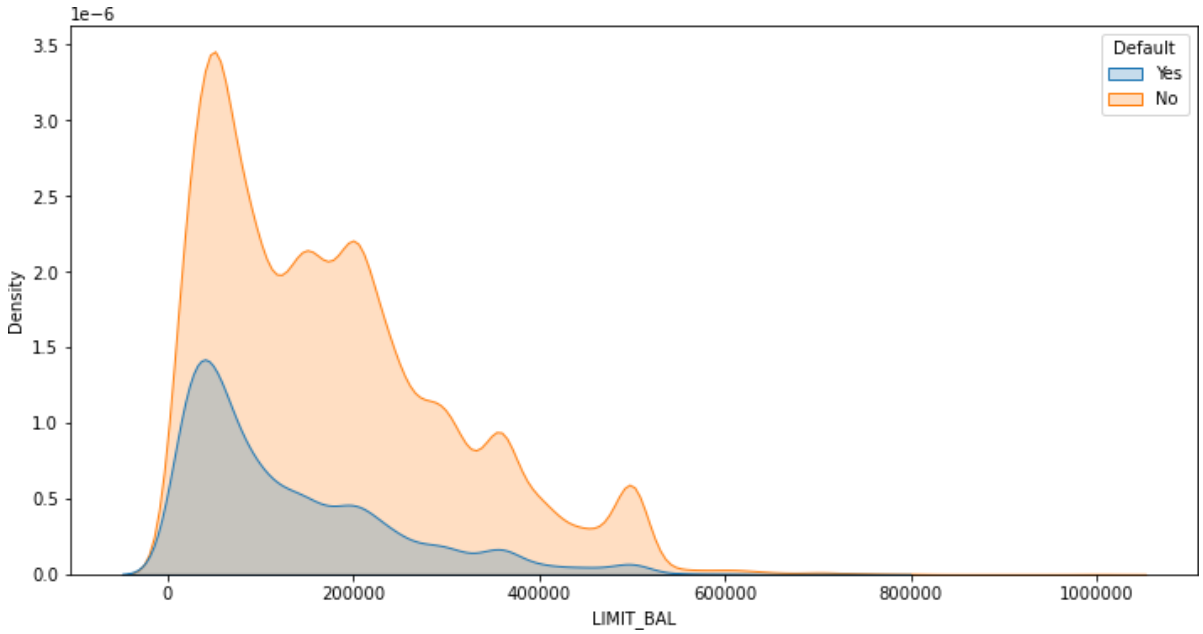
AGE vs Default Variables:



Insights:

1. Younger Age group are making more Default than Older Age Customer.

LIMIT_BAL vs Default Variables:

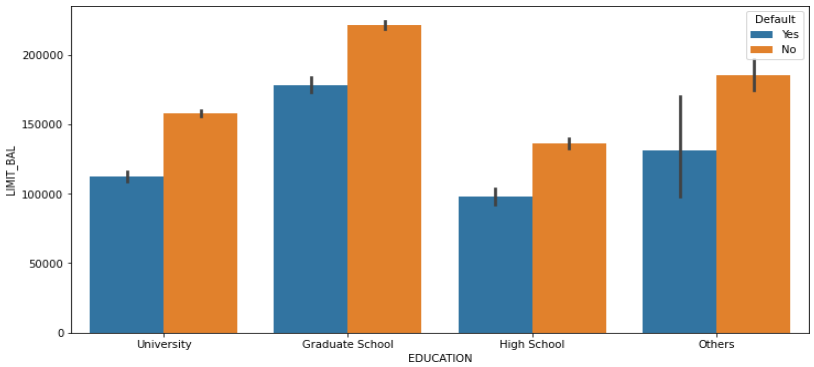


Insights:

1. Less LIMIT_BAL are making more defaults than High LIMIT_BAL.

- **<u>Multivariate Analysis:</u>**
  Multivariate analysis is based on the statistical principle of multivariate statistics, which involves observation and analysis of more than one statistical outcome variable at a time.

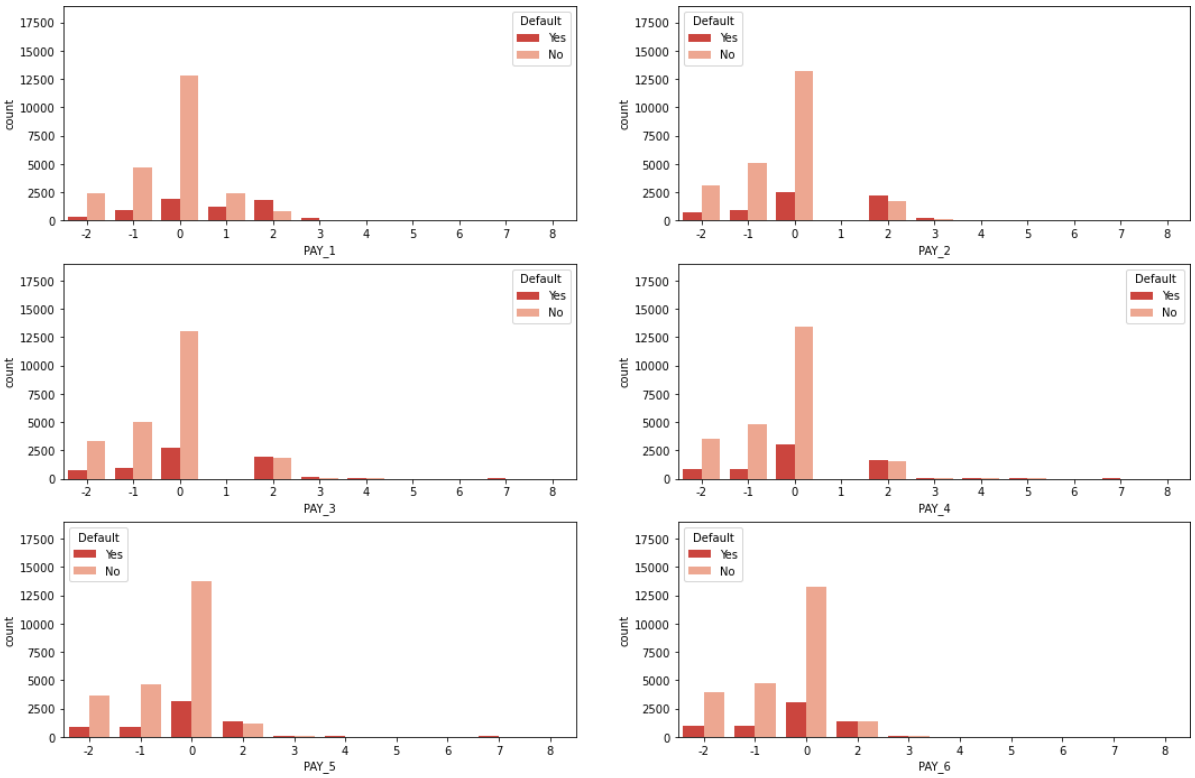
EDUCATION, LIMIT_BAL and Default Variables:



Insights:

1. Graduate students have more LIMIT_BAL and they are making more default than others.
2. We can see that average credit limit for low education level is minimum.

Repayment Features vs Default Variable:



Insights:

1. Most of the customers are paying the partial bill amount and duly paying.
2. When payment is delayed more than 2 months, the chances of default go higher than 50%.

STATISTICAL SIGNIFICANCE OF VARIABLES

Statistical significance is a determination that a relationship between two or more variables is caused by something other than chance. Statistical significance is used to provide evidence concerning the possibility of the null hypothesis, which hypothesizes that there is nothing more than random chance at work in the data.

- Statistical hypothesis testing is used to determine whether the result of a data set is statistically significant.
- Generally, a p-value of 5% or lower is considered statistically significant if p-value is less than the 0.05 we can say that variable is having the significance on the target variable where as a p-value that is greater than the significance level indicates that there is insufficient evidence in the sample to conclude that a non-zero correlation exists.
- To find the Statistical significance of variable we will perform Chi-square Test and Student t test.

Chi-Square Test for Independence:

A chi-square test for independence compares two variables in a contingency table to see if they are related. In a more general sense, it tests to see whether distributions of categorical variables from each other.

Student's t Test for Independence:

A t-test is a type of inferential statistic used to determine if there is a significant difference between the means of two groups, which may be related in certain features.

Hypothesis:

A statistical hypothesis is a hypothesis concerning the parameters or from of the probability distribution for a designated population or populations, or, more generally, of a probabilistic mechanism which is supposed to generate the observations basically It is a claim about population parameter.

- **Null Hypothesis:** Null hypothesis is a statistical theory that suggests there is no statistical significance exists between the populations. It is denoted by H0.
- **Alternative Hypothesis:** An Alternative hypothesis suggests there is a significant difference between the population parameters. It could be greater or smaller. Basically, it is the contrast of the Null Hypothesis. It is denoted by H1.
- Either of the hypothesis can be true.

Hypothesis testing based on 'LIMIT_BAL' and 'Default':

Basically, we stated Null and Alternative Hypothesis of „LIMIT_BAL" and „Default" from which „LIMIT_BAL" is independent variable Amount of given credit in NT dollars credit and „Default" is our Target Variable which is Dependent Variable.

Hypothesis testing performed here to check whether the „LIMIT_BAL“ for defaulter and non-defaulter are same or not.

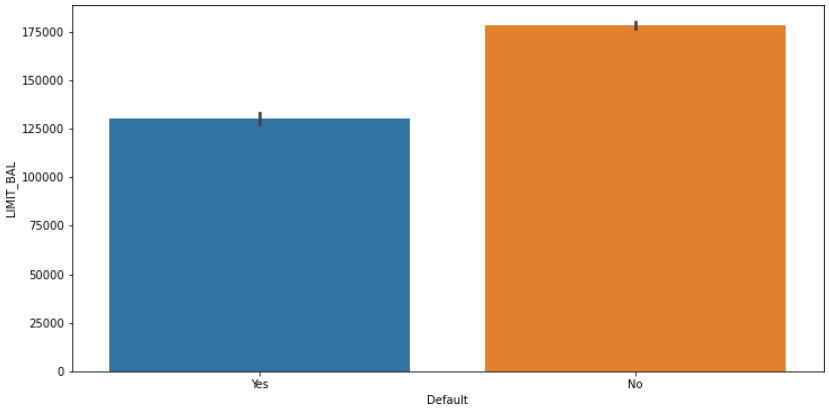
So,

Null Hypothesis:

H0: „LIMIT_BAL“ of defaulters is equal to „LIMIT_BAL“ of non-defaulter.

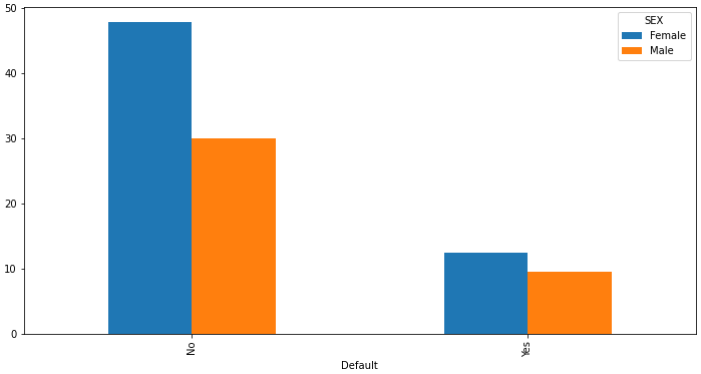
Alternative Hypothesis:

H1: „LIMIT_BAL“ of defaulters is not equal to „LIMIT_BAL“ of non-defaulter.



After performing T-Test for two samples, we decided to „Reject the Null Hypothesis i.e. “„LIMIT_BAL“ of defaulters is equal to „LIMIT_BAL“ of non-defaulter” as p-value (P-Value: 0.0) is less than significance level and concluded that customers with high credit limit are likelihood to be default than customers with less credit limit.

Hypothesis testing based on ‘SEX’ and ‘Default’:



We stated Null and Alternative Hypothesis of „SEX“ and „Default“ from which „SEX“ is independent shows gender of the user and „Default“ is our Target Variable which is Dependent Variable.

Hypothesis testing performed here to check whether „SEX“ and „Default“ is dependent or not.

So,

Null Hypothesis:

H0: „SEX“ and „Default“ is Independent

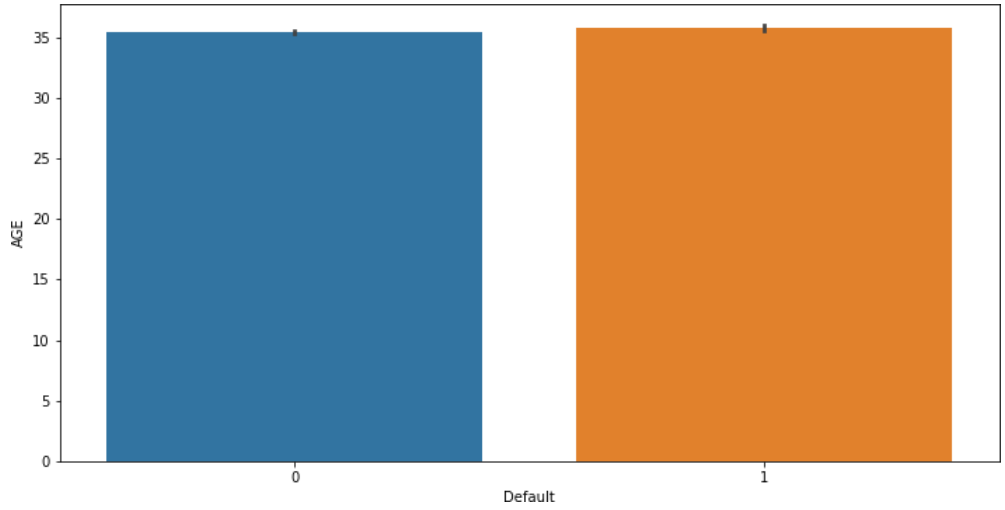
Alternative Hypothesis:

H1: „SEX“ and „Default“ is Dependent.

We performed the Chi-Square test on the above table to check independence of variables in a contingency table. And we get p-value as „4.944678999412026e-12“ which is less than significance value. So, we „Reject the null hypothesis“. And conclude that there is dependency in between SEX“ and „Default“ variables.

Hypothesis testing based on ‘AGE’ and ‘Default’:

(Here we took AGE mean to compare with „Default“ variable.)



We stated Null and Alternative Hypothesis of „AGE“ and „Default“ from which „AGE“ is independent shows age of the user which we took as mean age and „Default“ is our Target Variable which is Dependent Variable.

Hypothesis testing performed here to check whether „AGE“ and „Default“ is dependent or not.

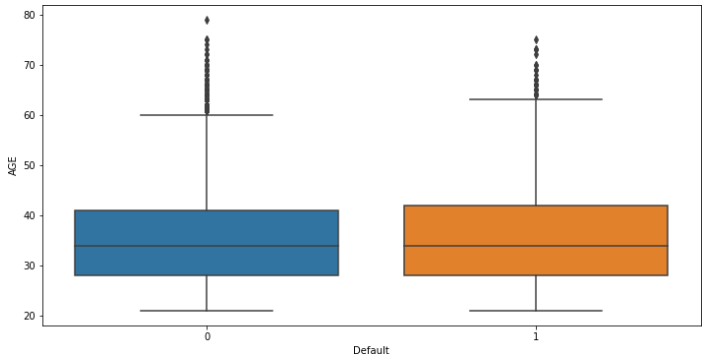
So,

Null Hypothesis:

H0: „AGE“ and „Default“ is Independent

Alternative Hypothesis:

H1: „AGE“ and „Default“ is Dependent.



We performed the T-test on the data to T-Test for two samples. And we get p-value as „0.01613684589016383“ which is less than significance value. So we „Reject the null hypothesis“. And conclude that „AGE“ and „Default“ both variables are Dependent.

Hypothesis testing based on ‘EDUCATION’ and ‘Default’:

We stated Null and Alternative Hypothesis of „EDUCATION“ and „Default“ from which „EDUCATION“ is independent shows highest education level provided by the user and „Default“ is our Target Variable which is Dependent Variable.

Hypothesis testing performed here to check whether „EDUCATION“ and „Default“ is dependent or not.

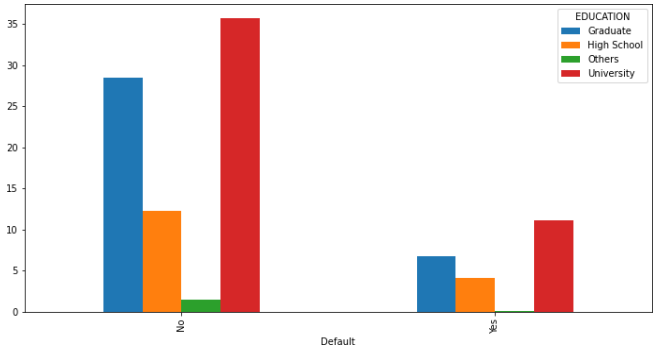
So,

Null Hypothesis:

H0: „EDUCATION“ and „Default“ is Independent

Alternative Hypothesis:

H1: „EDUCATION“ and „Default“ is Dependent.



We performed the Chi-Square test on the above table to check independence of variables in contingency table. And we get p-value less than significance value. So, we „Reject the null hypothesis.“ And can say „EDUCATION“ and „Default“ variables are dependent.

FEATURE ENGINEERING

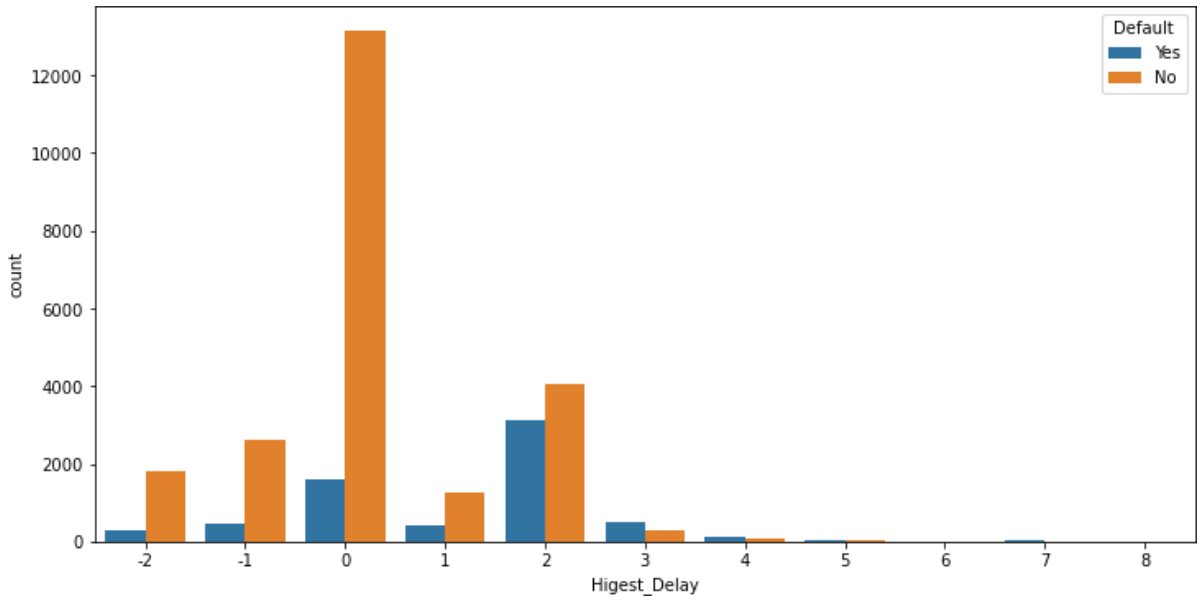
Feature engineering is the process of selecting, manipulating, and transforming raw data into features that can be used in supervised learning. In order to make machine learning work well on new tasks, it might be necessary to design and train better features.

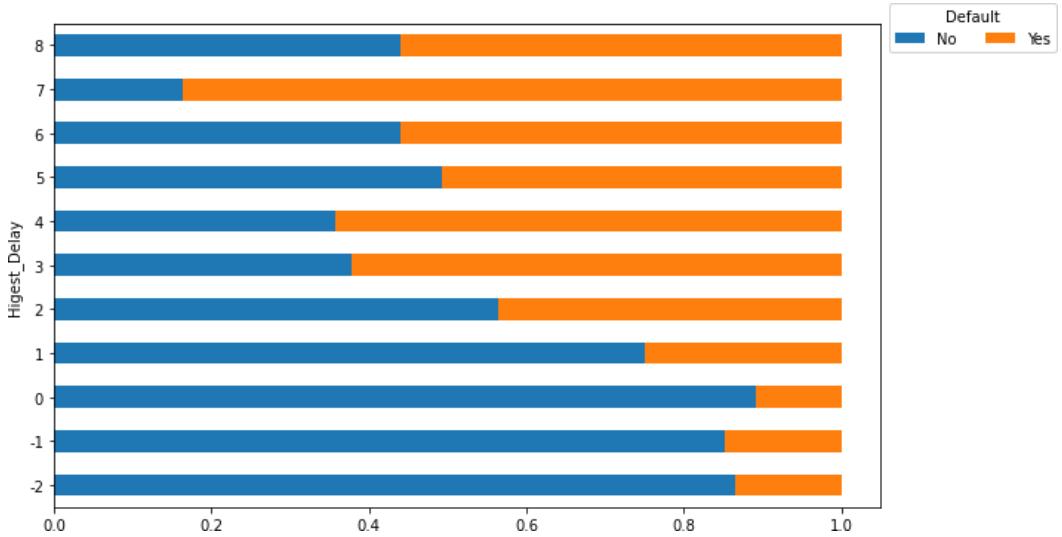
1. **Highest Delay**

Creating a new feature based on their payments, this feature will tell us that for maximum how many months he delayed the payment or highest payment delay.

Example:

If a customer delayed the payment of 8 months this column will show 8 (Means this customer delayed the payment for 8 months).



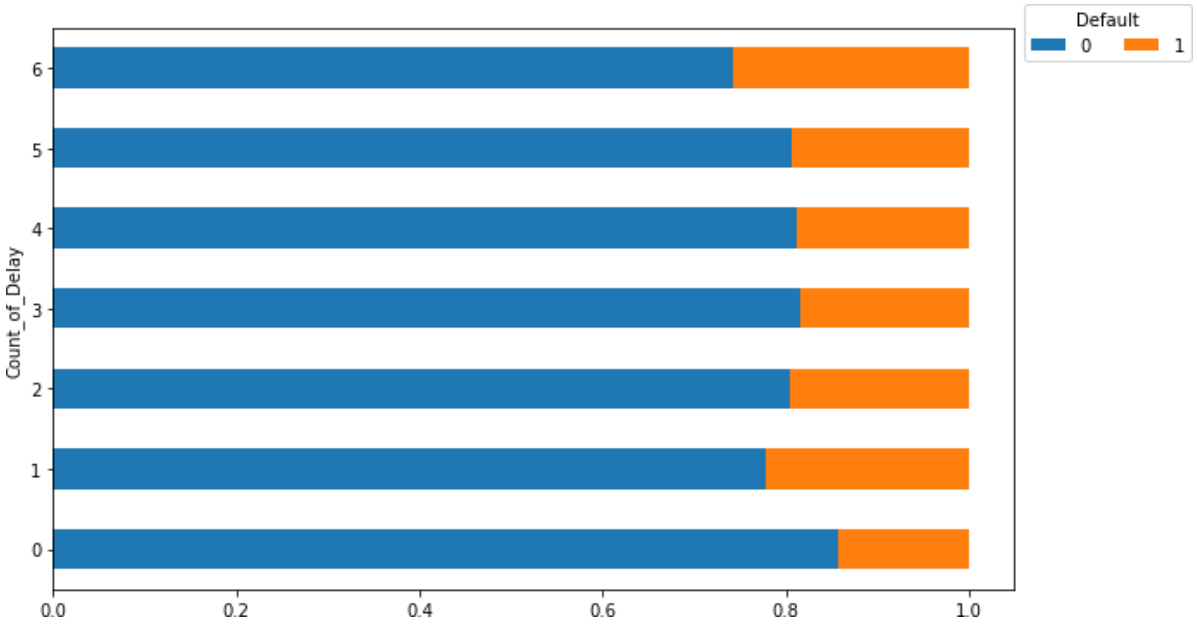


We can infer from above that most of the customer paying the bills partially and on time but the customer who delayed the payments for more than 1 month they are more likely to be defaulter.

Creating another feature which helps us to getting the information, how many times the customer did not pay the full amount.

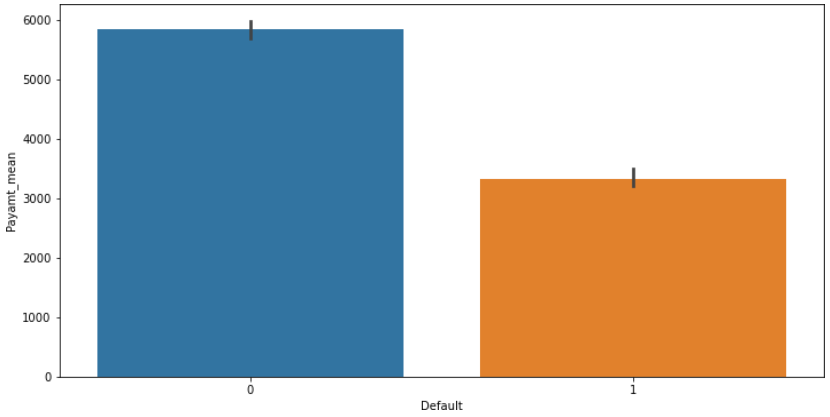
2. Count of Delay

Creating a new feature which will help us to get the information of how many times the customer did not pay the full amount.



From the above graph, we can conclude, customers who are making delay in making payments are likely to have default of their credit card.

3. Payment Mean



With this feature we can infer that the average payment paid by the default customer is less than the non-default customer.

Multicollinearity

a. Multicollinearity is a statistical concept where several independent variables in a model are correlated.
b. Two variables are considered to be perfectly collinear if their correlation coefficient is +/- 1.0.
c. Multicollinearity among independent variables will result in less reliable statistical inferences.
d. Multi-collinearity occurs when two or more independent variables (also known as predictors) are highly correlated with one another.
e. This means that an independent variable can be predicted from another independent variable.

VIF (Variance Inflation Factor)

Multicollinearity occurs when there are two or more independent variables in dataset, which have a high correlation among themselves. When some features are highly correlated, we might have difficulty in distinguishing between their individual effects on the dependent variable. Multicollinearity can be detected using various techniques, one such technique being the Variance Inflation Factor (VIF).

In VIF method, we pick each feature and regress it against all of the other features. For each regression, the factor is calculated as:

$$VIF = 1-(1/R^2)$$

	VIF_Factor	Features
0	inf	BILL_AMT4
1	inf	PAY_AMT4
2	inf	BILL_AMT5
3	inf	PAY_AMT2
4	inf	BILL_AMT3
5	inf	BILL_AMT2
6	inf	BILL_AMT1
7	inf	PAY_AMT3
8	inf	PAY_AMT5
9	inf	PAY_AMT1
10	inf	PAY_AMT6
11	inf	Payamt_mean
12	inf	billmean
13	inf	diff
14	inf	BILL_AMT6

15	13.442541	Count_of_Delay
16	12.420987	AGE
17	10.708620	bill_mean > Payment_mean
18	7.525389	EDUCATION
19	7.394274	MARRIAGE
20	5.046796	PAY_5
21	4.532702	PAY_4
22	4.345393	Higest_Delay
23	4.076982	LIMIT_BAL
24	3.975499	PAY_3
25	3.690588	PAY_6
26	3.384425	PAY_2
27	3.140213	PAY_1
28	1.695484	SEX

VIF factor of infinity is found in 14 columns which shows very high correlation. Columns having VIF factor of infinity are BILL_AMT4, PAY_AMT4, PAY_AMT5, PAY_AMT2, BILL_AMT3, BILL_AMT2, BILL_AMT1, PAY_AMT3, PAY_AMT5, PAY_AMT1, PAY_AMT6, PAYMNET_MEAN, BILL MEAN, DIFF, BILL_AMT6.

After performing various VIF methods one by one, we removed the mentioned variables: **'BILL_AMT4, BILL_AMT5, BILL_AMT3, BILL_AMT2, BILL_AMT6, Payamt_mean, billmean, diff, Count_of_Delay, AGE, bill_mean > Payment_mean, EDUCATION, MARRIAGE, PAY_5, PAY_3'.**

	VIF_Factor	Features
0	3.216943	Higest_Delay
1	2.930914	LIMIT_BAL
2	2.843044	PAY_4
3	2.722980	PAY_1
4	2.625009	PAY_2
5	2.391259	PAY_6
6	1.985859	BILL_AMT1
7	1.422138	SEX
8	1.324940	PAY_AMT1
9	1.281530	PAY_AMT3
10	1.247232	PAY_AMT2
11	1.236878	PAY_AMT4
12	1.217916	PAY_AMT5
13	1.209140	PAY_AMT6

Considering the above attributes, we achieved the VIF less than 5. And from here we will continue with these features for further processes.

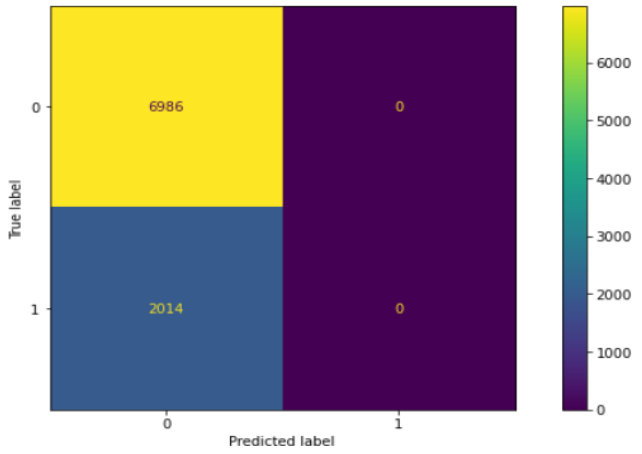
BASE MODELLING

We will build the models and will be including all the features which are in the dataset and other features also which are created with feature engineering.

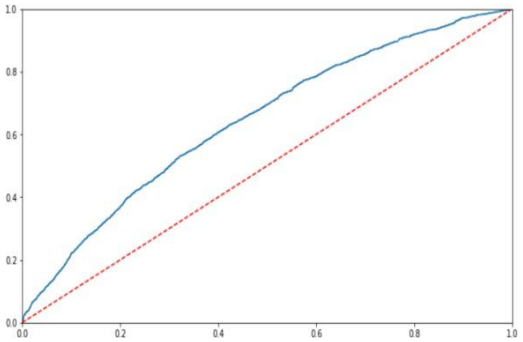
Logistic Regression Using Scikit-Learn:

Model Performance Evaluation:

```
Train_Accuracy_Score: 0.78
Test_Accuracy_Score: 0.776


roc_auc_score: 0.5


Classification_report:
              precision    recall  f1-score   support

           0       0.78      1.00      0.87      6986
           1       0.00      0.00      0.00      2014

    accuracy                           0.78      9000
   macro avg       0.39      0.50      0.44      9000
weighted avg       0.60      0.78      0.68      9000
```



Confusion Matrix



ROC AUC Curve

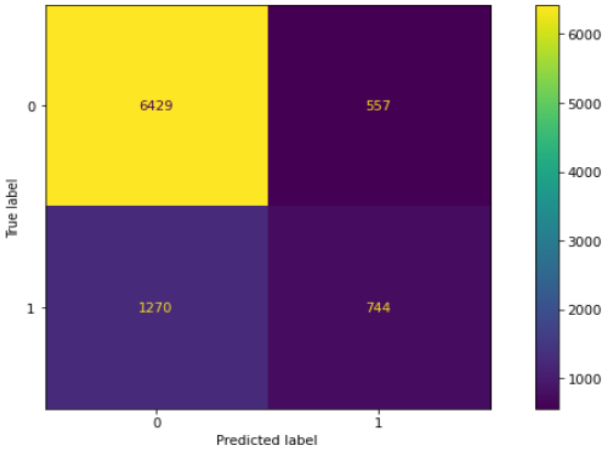
- As we can see the train and test accuracy of the model is 78% and 77.6% and roc auc score is 50%. With this model we did not find overfitting or underfitting as the both test and train score nearly same.
- If we see the confusion matrix and classification report we got 0 **False Positive** and 0 **True Negative** but huge False Negative which is not good. Because we want False Negative to be very low.
- Due to this False positive and False Negative we got specificity=1 and Sensitivity=0.

K-Nearest Neighbors Classification Model using Scikit-learn:

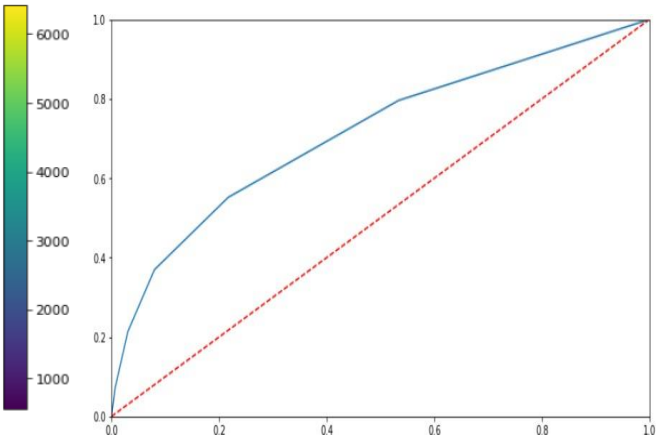
For this model we scaled the x_train and x_test with MinMaxScaler().

Model Performance Evaluation:

```
Train_Accuracy_Score: 0.844
Test_Accuracy_Score: 0.797


roc_auc_score: 0.645


Classification_report:
              precision    recall  f1-score   support

           0       0.84      0.92      0.88      6986
           1       0.57      0.37      0.45      2014

    accuracy                           0.80      9000
   macro avg       0.70      0.64      0.66      9000
weighted avg       0.78      0.80      0.78      9000
```



Confusion Matrix



ROC AUC Curve

- With K-Nearest Neighbors we got 84.1% train accuracy and 79.5% test accuracy by which we can say the model is little overfit.
- ROC AUC score for this model is 63.2% which is greater than logistic model.
- Sensitivity and specificity are 34% and 93% respectively which is again better than logistic model.
- F1 weightage average is 77% which is also greater than logistic model.

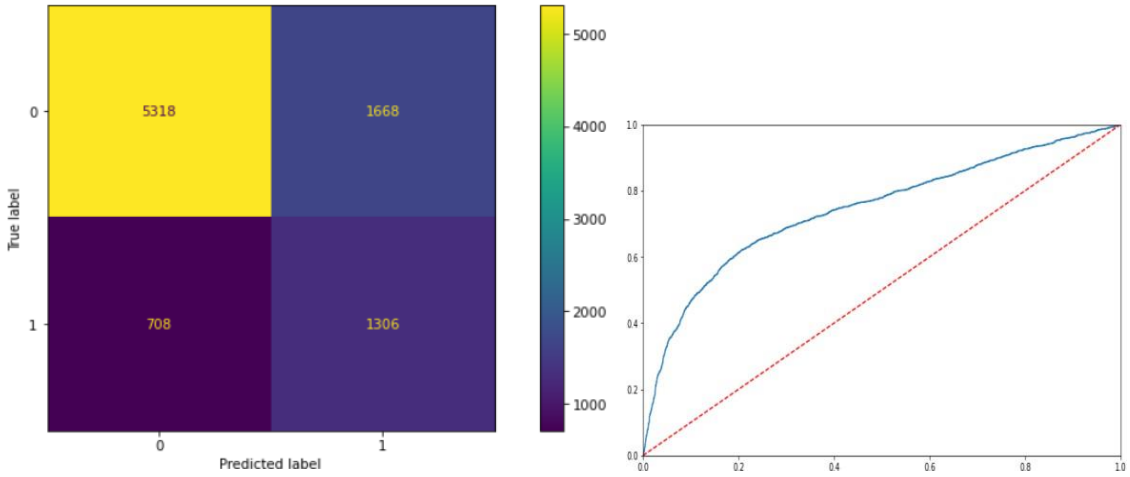
We have seen that K-Nearest model is better model than the logistic model in most of the aspect.

Gaussian Naïve Bayes (GNB) Model using Scikit-learn:

For this model we scaled the x_train and x_test with MinMaxScaler().

Model Performance Evaluation:

```
Train_Accuracy_Score: 0.734
Test_Accuracy_Score: 0.736


roc_auc_score: 0.705


Classification_report:
              precision    recall  f1-score   support

           0       0.88      0.76      0.82      6986
           1       0.44      0.65      0.52      2014

    accuracy                           0.74      9000
   macro avg       0.66      0.70      0.67      9000
weighted avg       0.78      0.74      0.75      9000
```



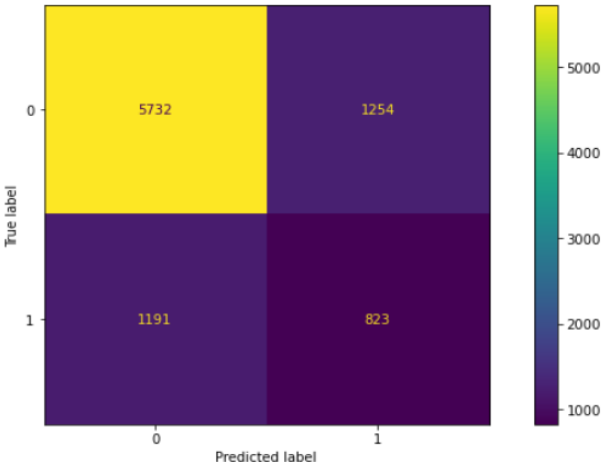
Confusion Matrix ROC AUC Curve

- With GNB model we got 71.2% train accuracy and 71.4% test accuracy.
- ROC AUC score is 69.5%.
- Sensitivity and specificity are 66% and 73% respectively.

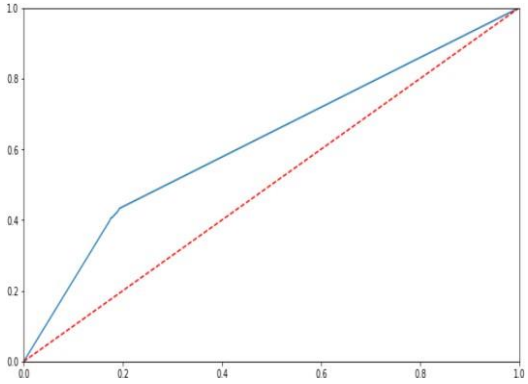
Decision Tree Classification Model using Scikit-learn:

Model Performance Evaluation:

```
Train_Accuracy_Score: 0.992
Test_Accuracy_Score: 0.728


roc_auc_score: 0.615


Classification_report:
              precision    recall  f1-score   support

           0       0.83      0.82      0.82      6986
           1       0.40      0.41      0.40      2014

    accuracy                           0.73      9000
   macro avg       0.61      0.61      0.61      9000
weighted avg       0.73      0.73      0.73      9000
```

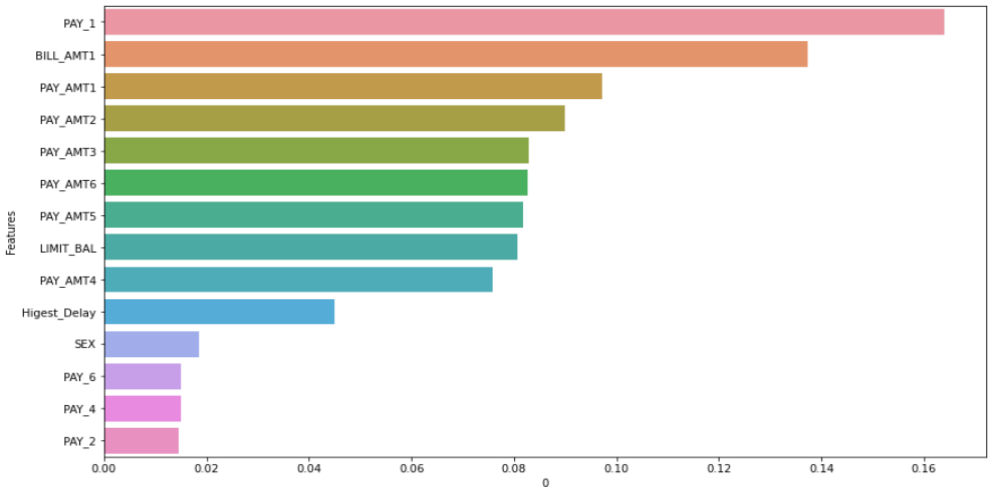


Confusion Matrix



ROC AUC Curve

Feature Importance:



- With Decision Tree model we got 99.9% train accuracy and 72.7% test accuracy and by this we can say that this model is overfitted. As we know that we allowed the tree to grow freely due to this it gets overfitted. We can prune the tree to reduce the overfitting.
- ROC AUC score is 61.7%.
- Sensitivity and specificity are 42% and 82% respectively.
- PAY_1 and Highest_Delay are most important features for predicting the target variable.

We will tune this model and can get rid of overfitting. As we got 42% sensitivity which is important to us and we got this value which is better than other models which are discussed above.

Base Model Comparison:

	Model_Name	Train_Accuracy	Test_Accuracy	ROC_Score	Specificity	Sensitivity	f1_weighted_avg
0	Logistic Regression	0.779905	0.776222	0.500000	1.000000	0.00000	0.6784
1	Decision Tree	0.991810	0.728333	0.614569	0.820498	0.40864	0.7298
2	Gaussian Naive Bayes	0.779571	0.776333	0.705000	0.760000	0.65000	0.7500
3	KNN	0.780000	0.776222	0.645000	0.920000	0.37000	0.7800

Inferences:

- We are getting best test accuracy in GNB model which is 77.633%.
- The highest f1 weighted average score which is 78% getting from KNN model.
- We are getting best Sensitivity score in GNB model which is 65%.
- The best Roc score is 70.5% which we are getting from GNB model.

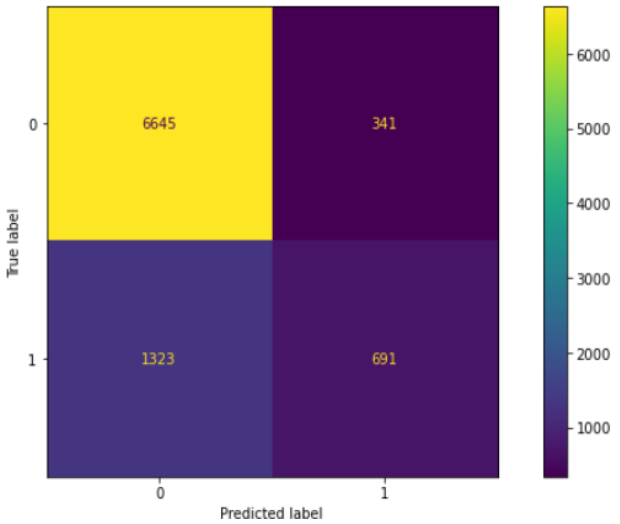
MODEL OPTIMIZATION

Hyper parameter tuning of Base Models which performed well

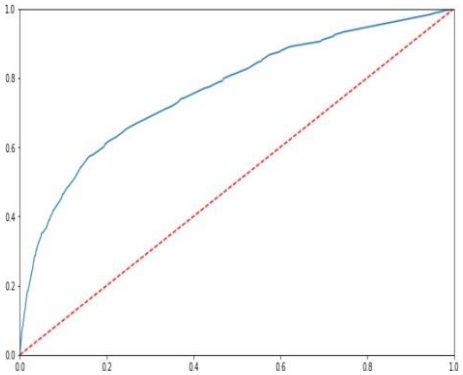
1. **Decision Tree**

Criterion = ['gini','entropy']
max_features = ['sqrt','log2',0.75,1]
min_samples_split = [5,10,20,30,50]
max_depth = [5,8,13]
min_samples_leaf = [5,10,15,30,40,50,60]

```
Train_Accuracy_Score: 0.824
Test_Accuracy_Score: 0.815


roc_auc_score: 0.647


Classification_report:
              precision    recall  f1-score   support

           0       0.83      0.95      0.89      6986
           1       0.67      0.34      0.45      2014

    accuracy                           0.82      9000
   macro avg       0.75      0.65      0.67      9000
weighted avg       0.80      0.82      0.79      9000
```



Confusion Matrix



ROC AUC Curve

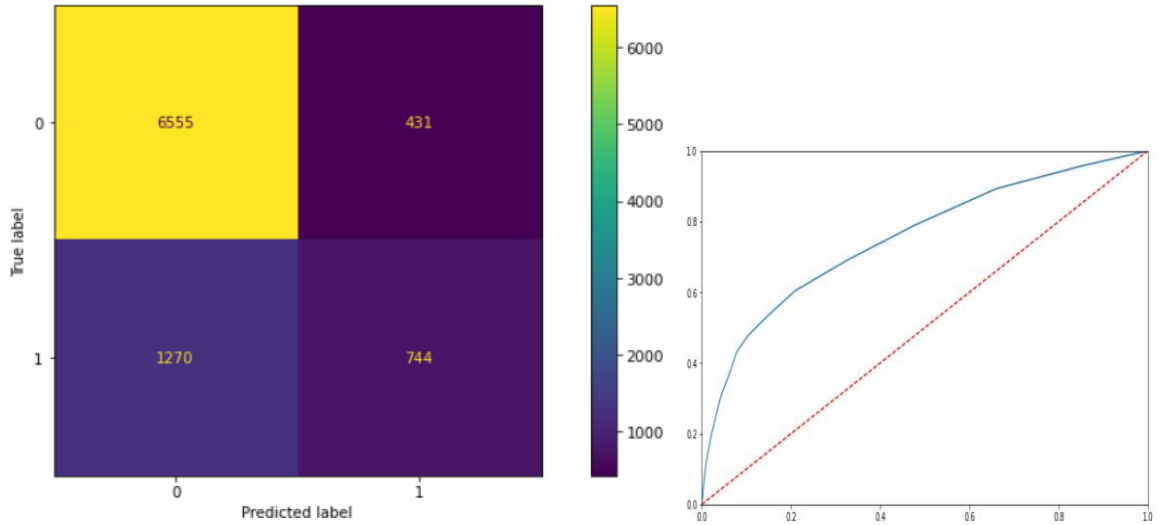
- With hyperparameter tuning of the parameters we got train accuracy 82.4% and test accuracy 81.5% and as you can see the overfitting problem is resolved.

- We got 95% specificity and 34% sensitivity. Sensitivity got decreased as compared to decision tree base model.
- Roc score got increased by approx. 3.3% which is now 64.7%.
- F1 weighted score got increased by approx. 6% and now we are getting 79%.

2. KNN

Best n_neighbors:17, p=1(Manhattan)

```
Train_Accuracy_Score: 0.826
Test_Accuracy_Score: 0.811


roc_auc_score: 0.654


Classification_report:
              precision    recall  f1-score   support

           0       0.84      0.94      0.89      6986
           1       0.63      0.37      0.47      2014

    accuracy                           0.81      9000
   macro avg       0.74      0.65      0.68      9000
weighted avg       0.79      0.81      0.79      9000
```



Confusion Matrix

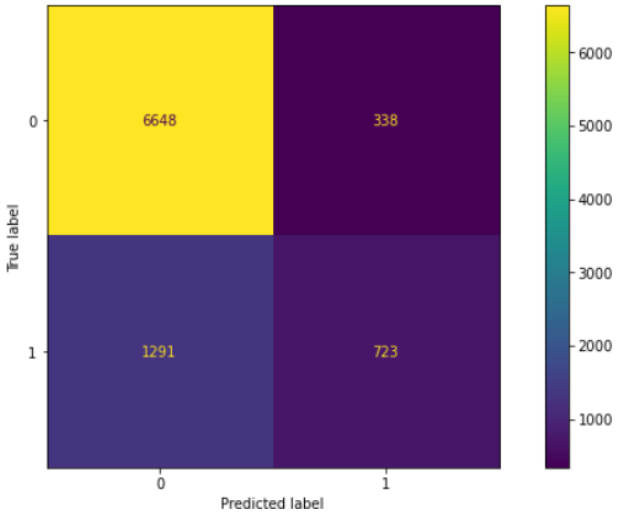
ROC AUC Curve

- With hyperparameter tuning of the parameters we got train accuracy 82.6% and test accuracy 81.1%.
- We got 94% specificity and 37% sensitivity. Sensitivity got decreased as compared to decision tree base model.
- Roc score got increased by little amount 0.9% which is now 65.4%.
- F1 weighted score got increased by 1% and now we are getting 79%.

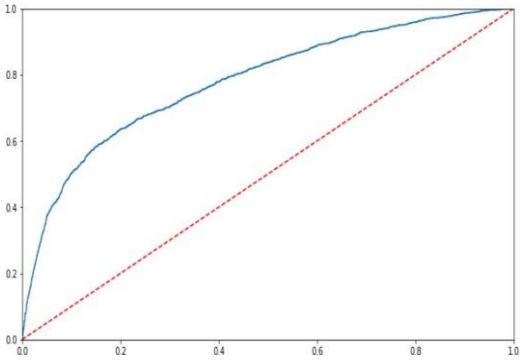
3. Gaussian Naïve Bayes

Model Performance Evaluation

```
Train_Accuracy_Score: 0.842
Test_Accuracy_Score: 0.819


roc_auc_score: 0.655


Classification_report:
              precision    recall  f1-score   support

           0       0.84      0.95      0.89      6986
           1       0.68      0.36      0.47      2014

    accuracy                           0.82      9000
   macro avg       0.76      0.66      0.68      9000
weighted avg       0.80      0.82      0.80      9000
```



Confusion Matrix



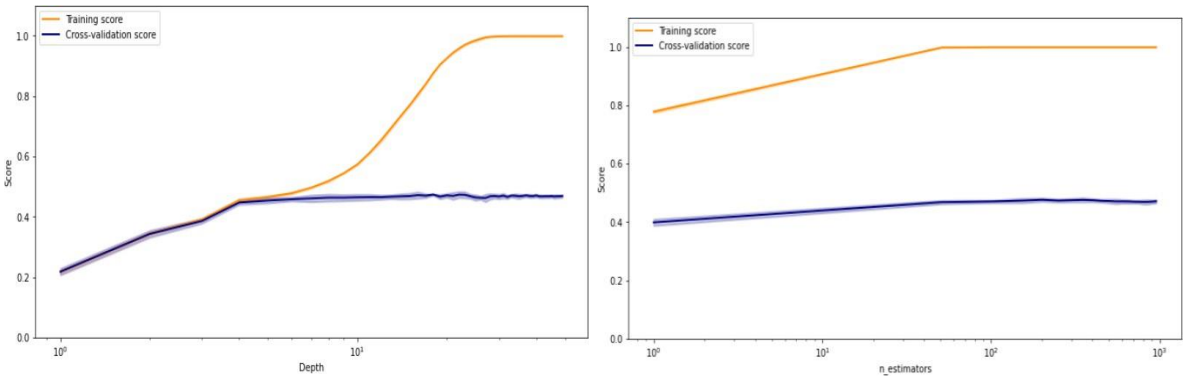
ROC AUC Curve

- With hyper parameter tuning of the parameters we got test accuracy 77.5% reduced by 0.1%.
- We got 83% specificity and 58% sensitivity. Sensitivity got decreased as compared to GNB base model.
- No change in Roc score.
- F1 weighted score got increased by 3% and now we are getting 78%.

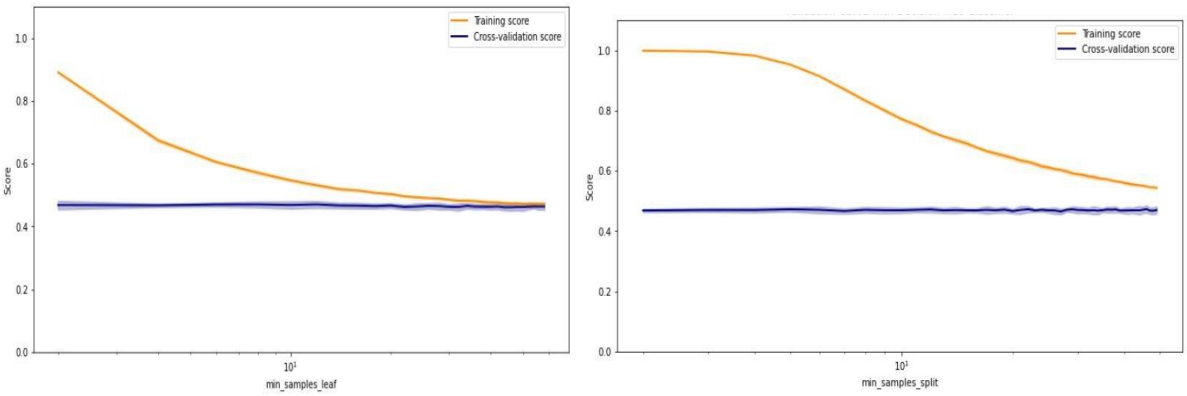
Boosting Algorithms:

1. Random Forest

Getting the optimum range of hyper parameters through validation curve:



We set tuned these hyper parameters and set the values and range according to the values got from above curves.



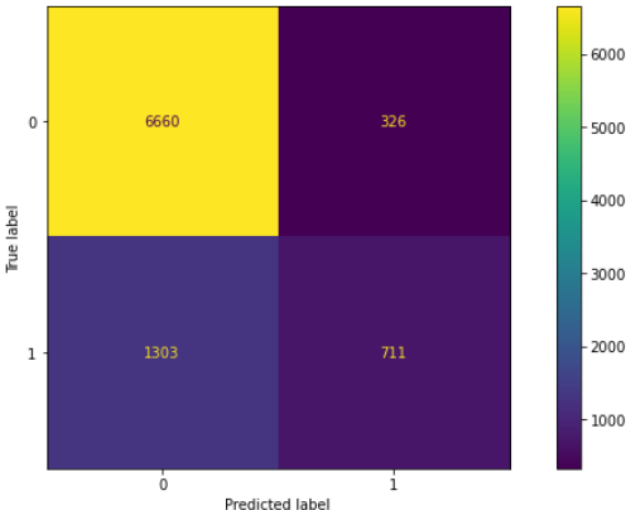
n_estimators=[50,500,1000,1500]
max_features=['sqrt','log2',0.75]
min_samples_split=[20,50]
max_depth=[5,13]
min_samples_leaf=[30,50]

And we got these as best parameters:

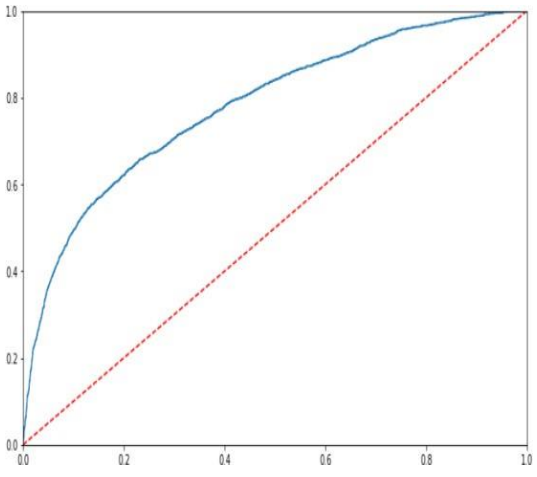
{'max_depth': 13,
 'max_features': 0.75,
 'min_samples_leaf': 30,
 'min_samples_split': 20,
 'n_estimators': 500}

Model Performance Evaluation:

```
Train_Accuracy_Score: 0.83
Test_Accuracy_Score: 0.819


roc_auc_score: 0.653


Classification_report:
              precision    recall  f1-score   support

           0       0.84      0.95      0.89      6986
           1       0.69      0.35      0.47      2014

    accuracy                           0.82      9000
   macro avg       0.76      0.65      0.68      9000
weighted avg       0.80      0.82      0.80      9000
```



Confusion Matrix

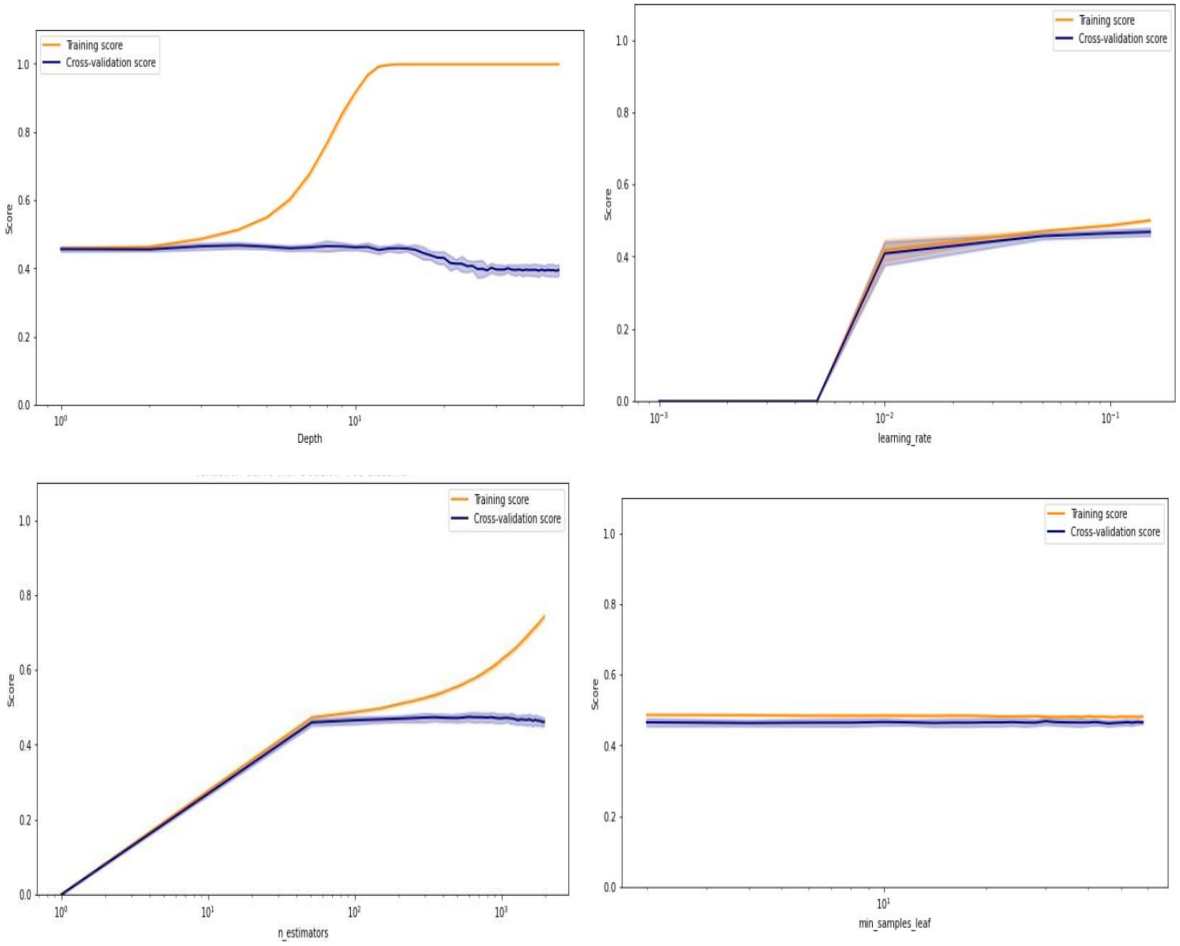


ROC AUC Curve

- We have got 81.9% test accuracy, 65.3% roc auc score.
- Spesificity 95% and sensitivity 35%.

2. Gradient Boost

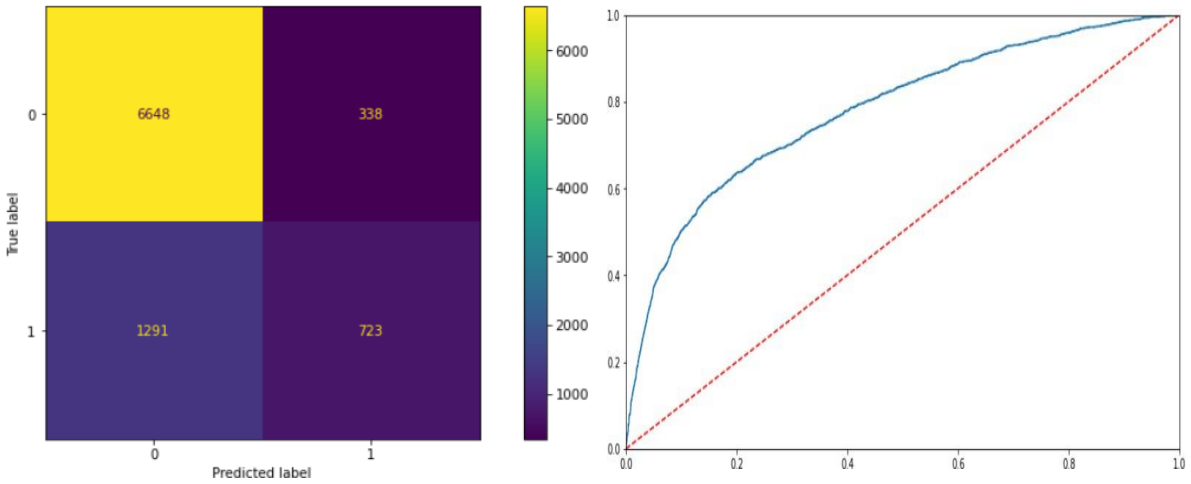
Getting the optimum range of hyper parameters through validation curve:



With the help of above curves, we can set the hyperparameters and find the best with GridSearchCV.

Model Performance Evaluation:

```
Train_Accuracy_Score: 0.842
Test_Accuracy_Score: 0.819


roc_auc_score: 0.655


Classification_report:
              precision    recall  f1-score   support

           0       0.84      0.95      0.89      6986
           1       0.68      0.36      0.47      2014

    accuracy                           0.82      9000
   macro avg       0.76      0.66      0.68      9000
weighted avg       0.80      0.82      0.80      9000
```



Confusion Matrix ROC AUC Curve

- We have got 81.9% test accuracy, 65.5% roc auc score.
- Spesificity 95% and sensitivity 36%.

3. XGBoost

tuning_parameters = {'learning_rate': [0.01,0.1, 0.2],'max_depth': range (3,8),

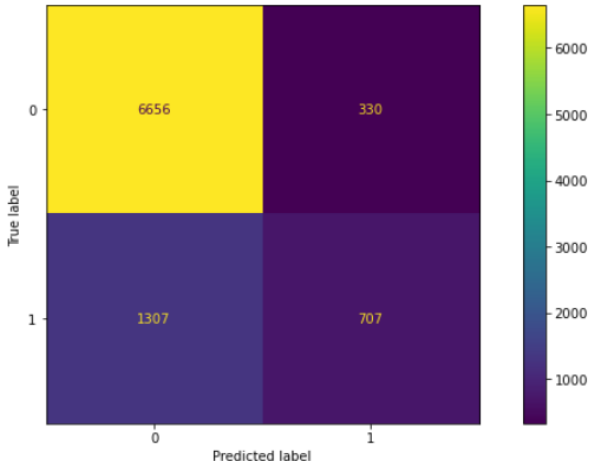
'gamma': [0, 1, 3],'n_estimator': [50,500,1500]}

With GridSearchCV we get the best parameters and these are:

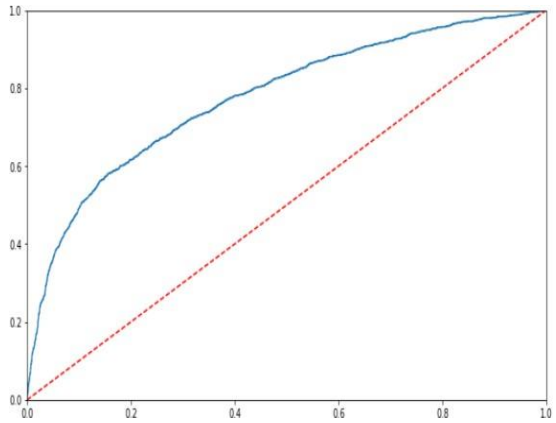
Best parameters for XGBoost: {'gamma': 1, 'learning_rate': 0.2, 'max_depth': 4, 'n_estimator': 50}

Model Performing Evaluation:

```
Train_Accuracy_Score: 0.833
Test_Accuracy_Score: 0.818


roc_auc_score: 0.652


Classification_report:
              precision    recall  f1-score   support

           0       0.84      0.95      0.89      6986
           1       0.68      0.35      0.46      2014

    accuracy                           0.82      9000
   macro avg       0.76      0.65      0.68      9000
weighted avg       0.80      0.82      0.79      9000
```



Confusion Matrix



ROC AUC Curve

- We have got 81.8% test accuracy, 65.2% roc auc score.
- Spesificity 95% and sensitivity 35%.

Model Comparison:

	Model_Name	Train_Accuracy	Test_Accuracy	ROC_Score	Specificity	Sensitivity	f1_weighted_avg
0	Logistic Regression	0.779905	0.776222	0.500000	1.000000	0.000000	0.6784
1	Decision Tree	0.991810	0.728333	0.614569	0.820498	0.408640	0.7298
2	Gaussian Naive Bayes	0.779571	0.776333	0.705000	0.760000	0.650000	0.7500
3	KNN	0.780000	0.776222	0.645000	0.920000	0.370000	0.7800
4	Decision Tree with tuning	0.824476	0.815111	0.647143	0.951188	0.343098	0.7914
5	KNN with tuning	0.797714	0.798778	0.649344	0.940000	0.370000	0.7900
6	Gaussian Naive Bayes with Tuning	0.778952	0.775667	0.704000	0.830000	0.580000	0.7800
7	Random Forest	0.991762	0.809667	0.651057	0.938162	0.363952	0.7897
8	AdaBoostClassifier	0.821048	0.816222	0.642028	0.957343	0.326713	0.7900
9	GradientBoostingClassifier	0.826619	0.818444	0.650350	0.954624	0.346077	0.7945
10	XGBClassifier	0.880571	0.813111	0.649212	0.945892	0.352532	0.7910
11	RandomForestClassifier with Tuning	0.829905	0.819000	0.653182	0.953335	0.353029	0.7959
12	GradientBoostingClassifier with Tuning	0.842476	0.819000	0.655302	0.951618	0.358987	0.7967
13	XGBClassifier with Tuning	0.833000	0.818111	0.651903	0.952763	0.351043	0.7949

- Getting maximum f1 weighted score in gradient boost model 79.67% and also the test accuracy which is 81.90%.
- Gaussian Naïve Bayes (GNB) without tuning (index no: 2) giving the best Roc score 70.50% and best sensitivity 65%.
- After tuning GNB model we increased the f1 weighted average score by 3% but sensitivity got decreased by 7%.

CONCLUSION

In this project we deployed 13 algorithms. We started with the base model and took the inferences from them after evaluating important measures we went for their parameter tuning to increase their performance.

We also treated outliers with power transformer method but it does not affect the model performances so we decided not to perform any transformation.

In this project we want to achieve best sensitivity which is to predict defaulters correctly because they might cause the loss to the business.

As we can see we are getting best f1 weighted score in Gradient boost model 79.67% and also the test accuracy 81.9%. But as discussed above our major focus is on to detect class 1 (defaulter) correctly which is 58% in Gaussian Naïve Bayes (with tuning). As we are getting best Sensitivity 65 % the ability to predict class 1 or can say predict True Positive correctly and have low False Positive in GNB base model and 70.50% Roc score.

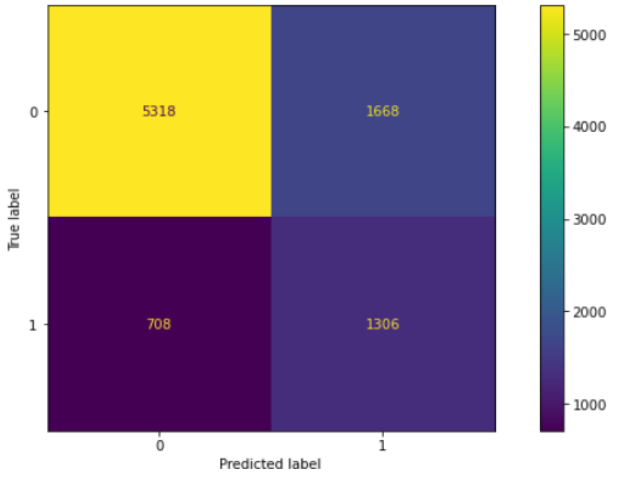
We will conclude that the Gaussian Naïve Bayes (index no 2 in comparison table) model is best suited for our problem.

Naïve Bayes

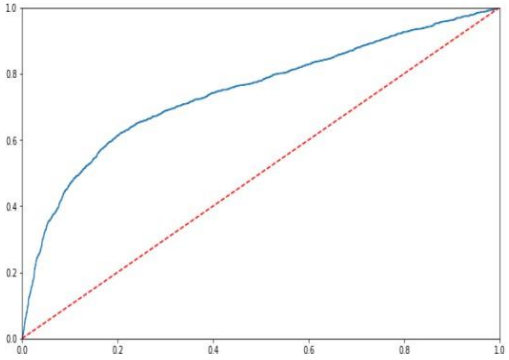
Important Features we consider when we did final model building on Naïve Bayes Classification technique. The feature importance is as follows:

```
Feature ranking:
1. Higest_Delay (0.087089)
2. PAY_1 (0.058533)
3. PAY_2 (0.053378)
4. PAY_4 (0.046222)
5. PAY_6 (0.041556)
6. BILL_AMT1 (0.000822)
7. SEX (-0.000578)
8. LIMIT_BAL (-0.001844)
9. PAY_AMT6 (-0.002311)
10. PAY_AMT4 (-0.002756)
11. PAY_AMT2 (-0.003222)
12. PAY_AMT5 (-0.003378)
13. PAY_AMT3 (-0.003400)
14. PAY_AMT1 (-0.004333)
```

```
Train_Accuracy_Score: 0.734
Test_Accuracy_Score: 0.736


roc_auc_score: 0.705


Classification_report:
              precision    recall  f1-score   support

           0       0.88      0.76      0.82      6986
           1       0.44      0.65      0.52      2014

    accuracy                           0.74      9000
   macro avg       0.66      0.70      0.67      9000
weighted avg       0.78      0.74      0.75      9000
```



Confusion Matrix



ROC AUC Curve

Limitations

- As we selected GNB model as our final prediction model which have assumption that all input features are independent of one another rarely holds true in real world but when the assumption of independence holds, a Naive Bayes classifier performs better compared to other models as we have seen in our project.
- The Gradient boost and Xgboost can be further tuned to get better result.
- As we reduced multicollinearity in our dataset and data needs to be scaled before deploying this model.
- Model can be improved by tuning the parameters in right way so we can increase its performance although we tried but, in our case, sensitivity got reduced.

What we have learned

- How to proceed and what things to do first like understand the dataset properly so that we can understand the problem and what are the features and their nature.
- Correct way to perform analysis on variables and which things to be keep in mind when describing the variables and inferences.
- Outlier treatment not necessary all the time because it may have pattern in it and may affect our prediction. Although after treating the outliers it did not affect the models so we went with outliers.
- Scaling affects the model‟s performance for which the scaling is required but for other models it did not affect the model performance.
- Feature engineering helped us to understand the patterns and predicting the target variable in better way in tree-based models.
- Not all the time Boosting techniques help to increase the performance, it does in our case but not able to get better than base naïve bayes model.

Acknowledgment

We would like to give a big thanks to our mentor Mr. Ankush Bansal for his guidance throughout the entire duration of the project. He helped us to understand the data, techniques and the idea how industry look on the data and methods.