A

PROJECT REPORT ON

Customer Profiling For Loan Prediction in Indian Banking System

Submitted in partial fulfilment of the requirements for the post graduate program

in

Data Science & Engineering

by

SABIN HASHMI (SSID: KVE8XT49HD)
ANKIT KUMAR (SSID: 0CN6AERV71)
OWAIS ALI MOHAMMED (SSID: G321NLUUEW)
SAKSHI RANWA (SSID: YLYEGRJL51)

Under the guidance of
DR DIPANJAN GOSWAMI
PROFESSOR BUSINESS INTELLIGENCE, IIT DELHI

GREAT LEARNING
Learning for life
HSR LAYOUT :: BANGALORE

ABSTRACT

With the enhancement in the banking sector lots of people are applying for bank loans but the bank has its limited assets which it has to grant to limited people only, so finding out to whom the loan can be granted which will be a safer option for the bank is a typical process. The bank employees are not able to analyze or predict whether the customer can payback the amount or not (good customer or bad customer) for the given interest rate. The aim of this paper is to increase approval rate without altering/changing the NPA/default rate. The pre-processing technique used includes comprehensive data cleaning, data reduction, and transformation. An exploratory data analysis technique is used to deal with this problem and exploratory data analysis help us in understanding the data more visually. The results are shown in graphs that helps the bankers to understand the client's behavior. In order to predict the customers who are loan defaulters the machine learning based supervised binary classification model is implemented such as logistic regression which gave better precision and accuracy. Along with that AUC ROC curves was also checked which gave satisfactory result.

ACKNOWLEDGEMENT

Any endeavor in a specific field requires the guidance and support of many people for successful completion. The sense of achievement on completing anything remains incomplete if the people who were instrumental in its execution are not properly acknowledged. We would like to take this opportunity to verbalize our deepest sense of indebtedness to our project mentor, Dr. DR DIPANJAN GOSWAMI, who was a constant pillar of support and continually provided us with valuable insights to improve upon our project and make it a success. Further, we would like to thank our parents for encouraging us and providing us a platform wherein we got an opportunity to design our own project.

Date: 16/01/2020

Place: Bangalore

greatlearning
Learning for Life

GREAT LAKES
INSTITUTE OF MANAGEMENT
Global Mindset - Indian Roots

CERTIFICATE OF COMPLETION

I hereby certify that the project titled **Customer Profiling For Loan Prediction in Indian Banking System** for case resolution was undertaken and completed under my supervision by Dr. Dipanjan Goswami of Post Graduate Program in Data Science and Engineering (PGP – DSE).

Name of the Mentor: DR DIPANJAN GOSWAMI
Signature of the Mentor:
Date:16/01/2020
Place: Bangalore

DECLARATION

We GROUP-9 hereby declare that the Project report entitled “**CONSUMER PROFILING FOR LOAN PREDICTION IN INDIAN BANKING SYSTEM"** submitted to the **Great Learning, Bangalore** in partial fulfilment of the requirements for the award of the PGP-DSE is a record of original dissertation work done by the group, under the guidance and supervision of **DR**. **DIPANJAN GOSWAMI, IIT Delhi,** and it has not formed the basis for the award of any Degree/Diploma/Associateship/ Fellowship or other similar title to any candidate of any University.

The dataset provided by the mentor is used only for sole purpose of completing the Capstone project and is not used or will not be used for any beneficiary of the students in the group or any candidate of any Institute or Organization.

PLACE: Bengaluru
DATE: 16/01/2020

GROUP MEMBERS:
1. Ankit Kumar
2. Owais Ali Mohammed
3. Sabin Hashmi
4. Sakshi Ranwa

CONTENTS

1. INTRODUCTION

1.1 OVERVIEW OF THE DATASET
1.2 PROBLEM STATEMENT
1.3 PROBLEM SOLVING METHODOLOGY USED
1.4 IMPORTANCE OF THE ANALYSIS IN THE REAL WORLD

2. EXPLORATORY DATA ANALYSIS

2.1 DEPENDENCY OF DIFFERENT CRITERION FOR DECISION MAKING

3. DATA CLEANING

3.1 MISSING VALUES
3.2 GENERAL ASSUMPTIONS
3.3 MISSING VALUE IMPUTATION

4. STATISTICS ANALYSIS

4.1 CHI-SQUARE TEST FOR CATEGORICAL COLUMNS
4.2 T-TEST UNPAIRED

5. MODEL BUILDING

5.1 DEFINE X AND Y VARIABLE
5.2 HIGH CORRELATED COLUMNS
5.3 UPSAMPLING OF TARGET VARIABLE
5.4 ALGORITHMS USED
5.5 MODEL COMPARISON

CONCLUSION

1. INTRODUCTION

1.1 Overview of the Dataset

The dataset contains the detailed record of 95488 entries who have applied for the home loan from their respective banks along with other relevant information of the customers that is being recorded in 91 attributes such as Location_Name, Company_Category, Residence_Type, Negative_Area, Bank_Name,Residence_City,Is_Online_Lead,cust_loan_type,Net_Take_Home,Loan_Applied_Amount,Loan_Applied_tenure,Sanction_Amount,Sanction_Tenure,Disbursement_Amount,System_Approved_Amount, Month-1 Day-5, Month-1 Day-10, Month-1 Day-15, Month-1 Day-20, Month-1 Day 25, Month-1 Average Monthly Balance, Month-1_End of Month Balance, Month-2 Day-5, Month-2 Day-10, Month-2 Day-15, Month-2 Day-20, Month-2 Day-25, Month-2 End of Month Balance, Month-2 Average Monthly Balance, Month-3 Day-5, Month-3 Day-10, Month-3 Day-15, Month-3 Day-20, Month-3 Day-25, Month-3 Average Monthly Balance, Month-3 End of Month Balance, ACCOUNT_NUM, BANK_NAME, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89.

FEATURE NAME	FEATURE TYPE	FEATURE DESCRIPTION
Location_Name	NOMINAL	It contains the name of the states where the customer resides.(14 Types-Delhi, Bangalore, Mumbai,Chennai,Hyderabad,Pune,Ahmedabad, Calcutta,Jaipur,Chandigarh,Gurgaon,Cochin,Baroda,THANE)
Residence_City	NOMINAL	It contains the name of city of customer
Residence_Type	NOMINAL	The type of residence in which the customer lives. (9 Types-Rented with Family, Parental, Self Owned, Rented with Friends, Rented - Bachelor Staying alone, Company Provided - Staying with family, Company Provided-Staying Alone, Paying Guest, Hostel/Guest House/Hotel)
Negative_Area	NOMINAL	Means bank has a list of areas categorized as the negative or 'red' zones. Banks reject your loan application if you reside in or the property you want to purchase is located in the negative zone. (2 class- Y,N)
Bank_Name	NOMINAL	It contains 64 different varieties of bank in which the customer have their account.
BANK_NAME	NOMINAL	The entire 64 varieties of bank is being classified into three different types as 0,1,2,3
ACCOUNT_NUM	NOMINAL	Types of account number given to each customer. It’s of four different type as 0,1,2,3,4

Reference_Number	NOMINAL	It represents that whenever a loan is given to a particular customer the bank generates a reference code for them.
APP_REFCODE	NOMINAL	It is same as the reference code being given by the bank to customer
Company_Category	NOMINAL	The bank have a list of companies for the customers who are working in company to label them as A,B,C,D,E according the company category.
Net_take_home	NUMERIC	Net take home salary is the income that the employee actually takes home once tax and other such deductions are carried over with.
Loan_Applied_Amount	NUMERIC	It is the total amount for which he want loan.
Loan_Applied_tenure	NOMINAL	It is the total no. of years that the customer is willing to pay back the entire loan.
Sanction_Amount	NUMERIC	It is the amount mentioned in the sanction letter
Sanction_Tenure	NOMINAL	It is the tenure that is fixed by the bank for paying the loan and it changes from bank to bank. (Types-0,1,2,3,4,5)
Disbursement_Date_x	DATETIME	It is date when the amount was sanctioned to the customer.
Disbursement_Date_y	DATETIME	It is date when the amount was sanctioned to the customer.
Disbursement_Amount	NUMERIC	Once the loan is sanctioned by the bank the amount is being disbursed in one or more installments depending upon the terms and condition mentioned in the sanctioned letter is disbursement amount.
System_Approved_Amount	NUMERIC	It is the loan amount that is being approved by the system for a particular customer.
Is_Online_Lead	NOMINAL	Whether the customer is online lead or not Type-YES, No
cust_loan_type	NOMINAL	Tells about the loan status for each customer whether it’s accepted (1) or rejected (0).
Month-1 Day-5	NUMERIC	Balance of customers up to 5th day for first month
Month-1 Day-10	NUMERIC	Balance of customers up to 10th day for first month
Month-1 Day-15	NUMERIC	Balance of customers up to15th day for first month
Month-1 Day-20	NUMERIC	Balance of customers up to20th day for first month
Month-1 Day-25	NUMERIC	Balance of customers up to25th day for first month
Month-1 Average Monthly Balance	NUMERIC	Average balance of the customers for first month

Month-1 End of Month Balance	NUMERIC	Balance of customers at the end of first month
Month-2 Day-5	NUMERIC	Balance of customers up to 5th day for second month
Month-2 Day-10	NUMERIC	Balance of customers up to 10th day for second month
Month-2 Day-15	NUMERIC	Balance of customers upto15th day for second month
Month-2 Day-20	NUMERIC	Balance of customers up to 20th day for second month
Month-2 Day-25	NUMERIC	Balance of customers up to25th day for second month
Month-2 End of Month Balance	NUMERIC	Balance of customers at the end of Second month
Month-2 Average Monthly Balance	NUMERIC	Average balance of the customers for second month
Month-3 Day-5	NUMERIC	Balance of customers up to 5th day for third month
Month-3 Day-10	NUMERIC	Balance of customers up to 10th day for third month
Month-3 Day-15	NUMERIC	Balance of customers up to15th day for third month
Month-3 Day-20	NUMERIC	Balance of customers up to 20th day for third month
Month-3 Day-25	NUMERIC	Balance of customers up to 25th day for third month
Month-3 Average Monthly Balance	NUMERIC	Average balance of the customers for third month
Month-3 End of Month Balance	NUMERIC	Balance of customers at the end of third month
40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,78,79,80,81,82,83,84,85,86,87,88,89	NOMINAL	Arbitrary column supporting the analysis

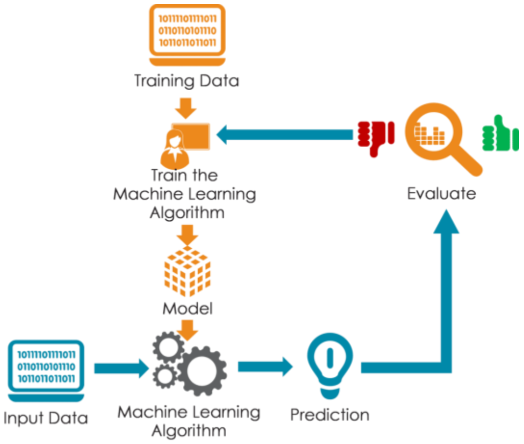
This document states the business problem of kotak and gives detailed understanding of the data points shared by kotak so far. There are 63 categorical columns, 26 numeric column and 2 date time column. Missing values is in huge numbers.

1.2 PROBLEM STATEMENT

The dataset contains data regarding details of 95488 entries who have applied for home loan. Kotak bank has high reject rate for the loan applications, the NPA is around 1%.The objective is to increase approval rate without altering/changing the NPA/default rate.

1.3 PROBLEM SOLVING METHODOLOGY USED

The methodology used to design the project was CRISP-DM. CRISP-DM stands for cross industry process for data mining. CRISP-DM is a comprehensive data mining methodology and process model that provides anyone—from novices to data mining experts—with a complete blueprint for conducting a data mining project. CRISP-DM breaks down the life cycle of a data mining project into six phases. It has 6 phases like Business understanding, Data understanding, Data preparation, Modeling, Evaluation and Deployment.


Training Data
Train the Machine Learning Algorithm
Evaluate
Model
Input Data
Machine Learning Algorithm
Prediction

1) Business understanding: The first phase of CRISP-DM process is to understanding the data in business perspective.

Business understanding

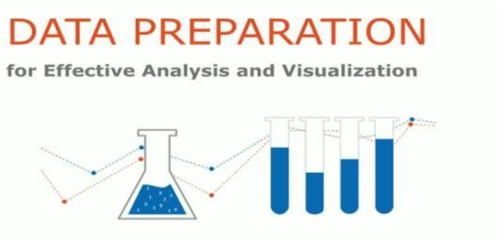
- *What is the problem that you are trying to solve?*

The objective of this stage is to uncover significant factors that could influence the outcomes of the project. Disregarding this step can mean that a great deal of effort is put into producing the right answers to the wrong questions. In this project the domain is related to details related to credit card details of customers. Location_Name, Residence_City, Residence_Type, Negative_Area, Bank_Name, Reference_Number, Company_Category, Is_Online_Lead, cust_loan_type ,BANK_NAME, ACCOUNT_NUM, Net_Take_Home, Loan_Applied_Amount, Loan_Applied_tenure, Sanction_Amount, Sanction_Tenure, Disbursement_Amount, System_Approved_Amount. These are the data collected from the bank so far about the customers and the details about them related to banks. The objective is to increase approval rate without altering/changing the NPA/default rate. To achieve it, we would first understand and treat the data. Then we will apply the base model which is logistic regression.

2) Data Understanding: The second phase of CRISP-DM process is for understanding the data.

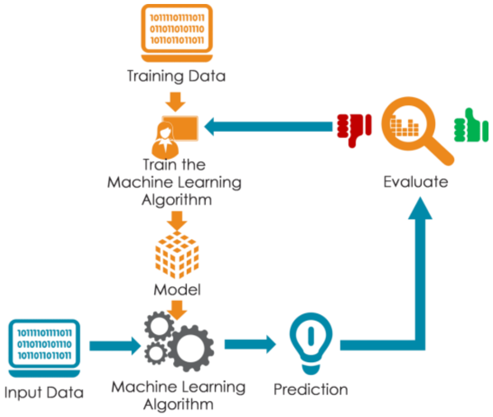
This process requires one to acquire the data listed in the project resources. This initial collection includes data loading, if this is necessary for data understanding. Cust_loan_type
Is our target variable which is stating that whether the loan applied by the customer is accepted or rejected. The predictor variables are Location_Name, Residence_City, Residence_Type, Negative_Area, Bank_Name, Reference_Number, Company_Category, Is_Online_Lead, BANK_NAME, ACCOUNT_NUM, Net_Take_Home, Loan_Applied_Amount, Loan_Applied_tenure, Sanction_Amount, Sanction_Tenure, Disbursement_Amount, System_Approved_Amoun. To perform the analysis we did Exploratory Data Analysis to better understand the predictor variables. It was checked if there were any outliers. We checked if the data is normal or not and what are the patterns seen when predictor variables are plotted against the target variable First, univariate analysis was performed to check how each variable is distributed, if the distribution is normal or not. Normal distribution is important because of the Central limit theorem. In simple terms, if you have many independent variables that may be generated by all kinds of distributions, the aggregate of those variables will tend toward a normal distribution.

3) Data Preparation: The goal of data preparation is the same as other data hygiene processes: to ensure that data is consistent and of high quality.



Inconsistent, low quality data can contribute to incorrect or misleading business intelligence. It can create errors and make analytics and data mining slow and unreliable. This is the process of collecting, cleaning, and consolidating data into one file or data table, primarily for use in analysis. Data preparation involves the rationalization and validation of data to make sure data is formatted consistently and that the data will be understood once removed from its source. It can involve changing the formats of dates, for example, or deleting duplicate fields. For data preparation, number of outliers were checked and observed to have more number of outliers in some features and z-score was performed in the numerical columns for transforming the data. Then the data was scaled using standard scalar.

4) Modelling: *Data modeling* is the process of producing a descriptive diagram of relationships between various types of information that are to be stored in a database.

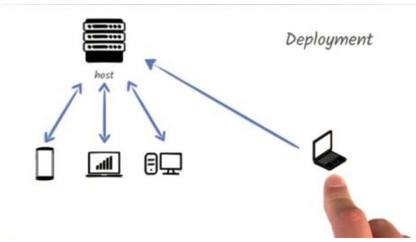

Training Data
Train the Machine Learning Algorithm
Evaluate
Model
Input Data
Machine Learning Algorithm
Prediction

Although one may have already selected a tool during the business understanding phase, at this stage one maybe be selecting some specific modelling technique.If multiple techniques can also be applied. After treating the data, we performed only logistic regression.

5) Evaluation: During this step it will be assessed if the model meets your business objective and to how much degree and seeking to determine if there is some business reason why this model is deficient.

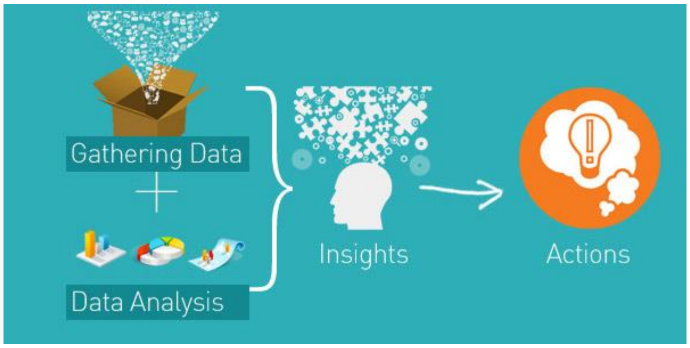
Another option is to test the model(s) on test applications in the real application, if time and budget constraints permit. The evaluation phase also involves assessing any other data mining results one generated. Data mining results involve models that are necessarily related to the original business objectives and all other findings that are not necessarily related to the original business objectives, but might also unveil additional challenges, information, or hints for future directions. To evaluate the model the data was separated into train and test. Then, the data was passed to various machine learning algorithms like logistic regression which is our base model. Accuracy score and confusion metrics were checked and are used to evaluate the data. K fold cross validation can also be performed to evaluate the data.

6) Deployment: In the deployment stage one may take their evaluation results and determine e a strategy for their deployment. It makes sense to consider the ways and means of deployment during the business understanding phase as well, because deployment is absolute crucial to the success of the project. This is where predictive analytics really helps to improve the operational side of the business. For the project, deployment steps have not been performed. The scope of the procedure we followed was till model evaluation.



1.4 IMPORTANCE OF THE ANALYSIS IN THE REAL WORLD

Conducting the analysis to produce the best results for the decisions to be made is an important part of the process, as is appropriately presenting the results. A fundamental aspect of analytics is decision support – in other words, providing material to support the human decision-making process.

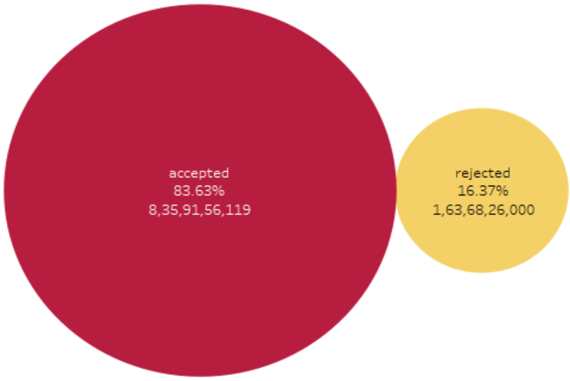


There are cases like in bank sectors, in recent years the number of peoples are increased who are applying for the loans for various reasons. The bank employees are not able to analyze or predict whether the customer can payback the amount or not for the given interest rate. Let's say, if someone needs to find the nature of the client who are applying for the personal loan. An exploratory data analysis technique is used to deal with this problem. The result which they will be getting by doing the analysis will show what type of loans will be preferred by majority of the clients. The results which one can show by plotting graphs (by doing analysis) that helps the bankers to understand the client's behavior.

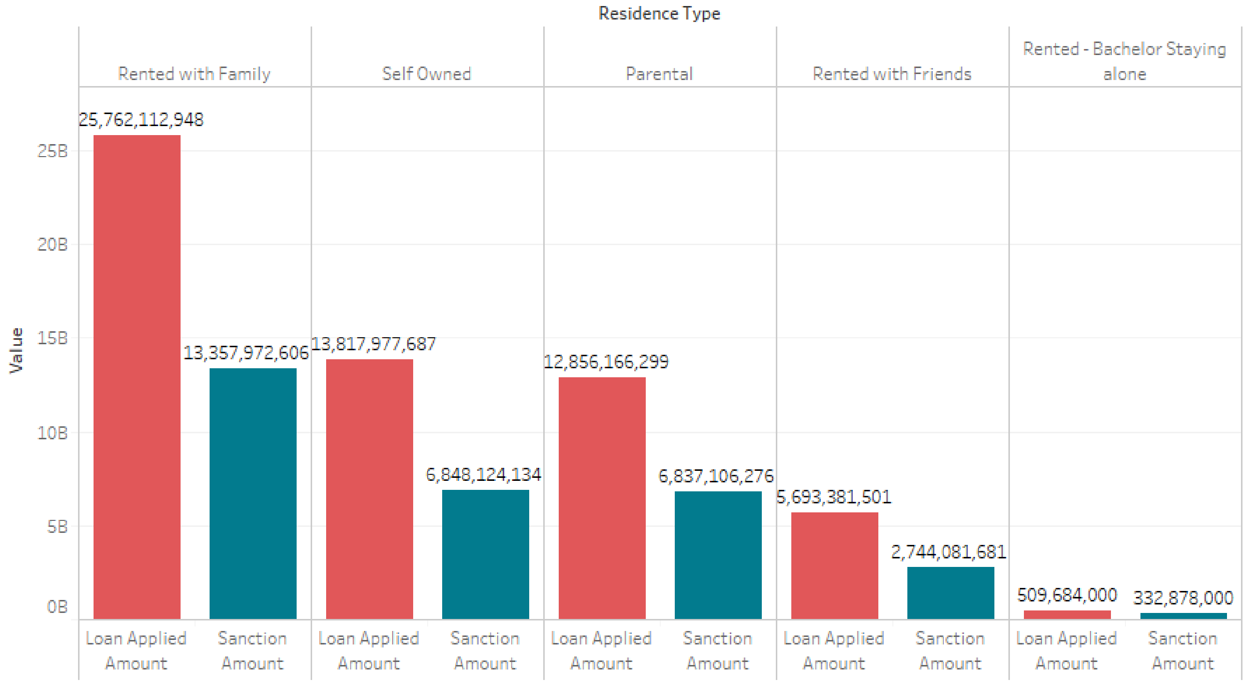
2. EXPLORATORY DATA ANALYSIS

2.1 Dependency of different criterion for decision making:

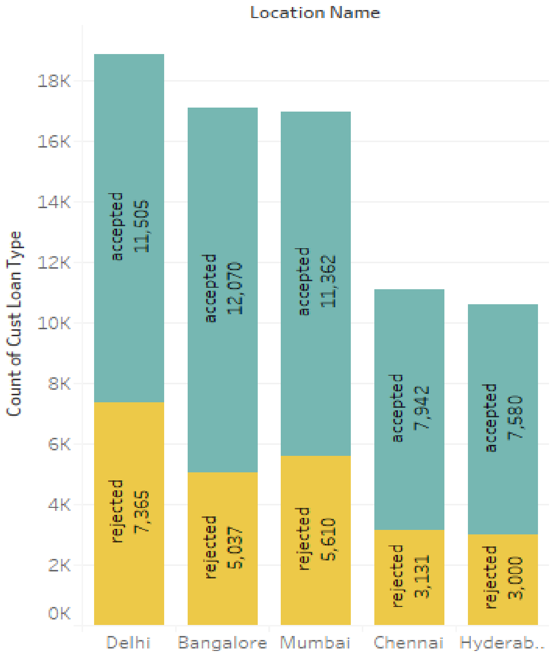
- **Acceptance and Rejection Rate:-** The following graph shows the ratio from accepted loan application and the corresponding sum amount to the rejected amount across all the regions.



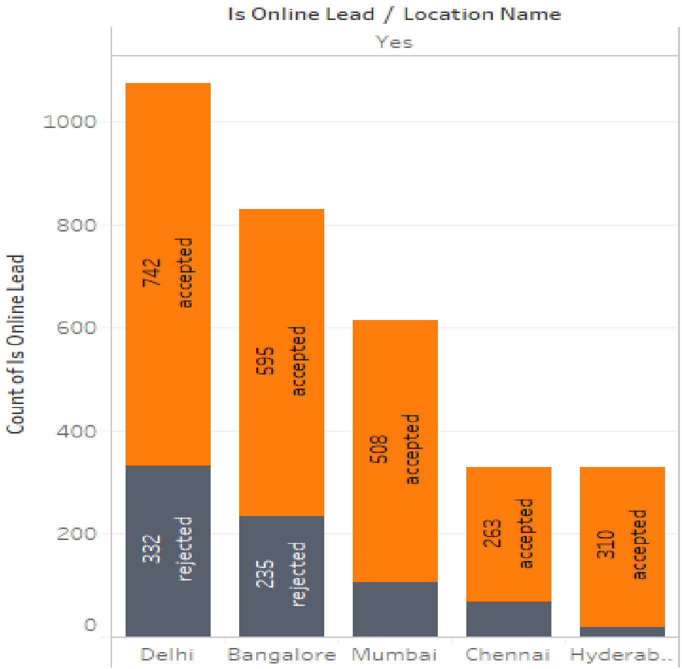
- **Criterion - Residence Type :-** The following graph indicates that the different ratio proportion of acceptance to rejection with respect to the residence type the customer is living in



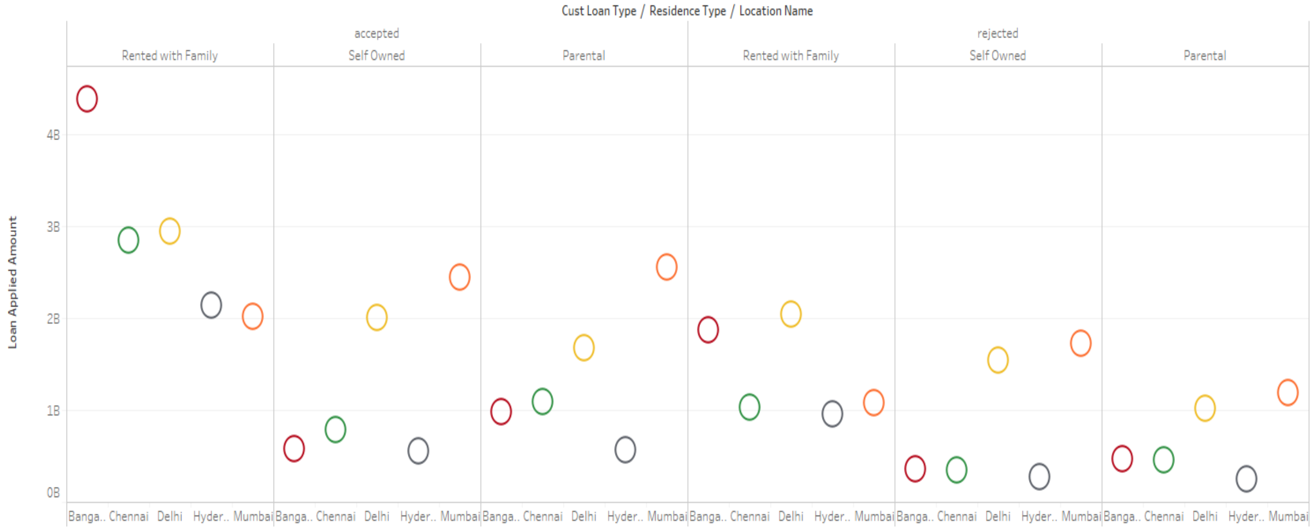
- **Criterion-Location:-** The following graph shows the number of application difference in case of acceptance and rejection with respect to the Hqs where the customer lodged loan application



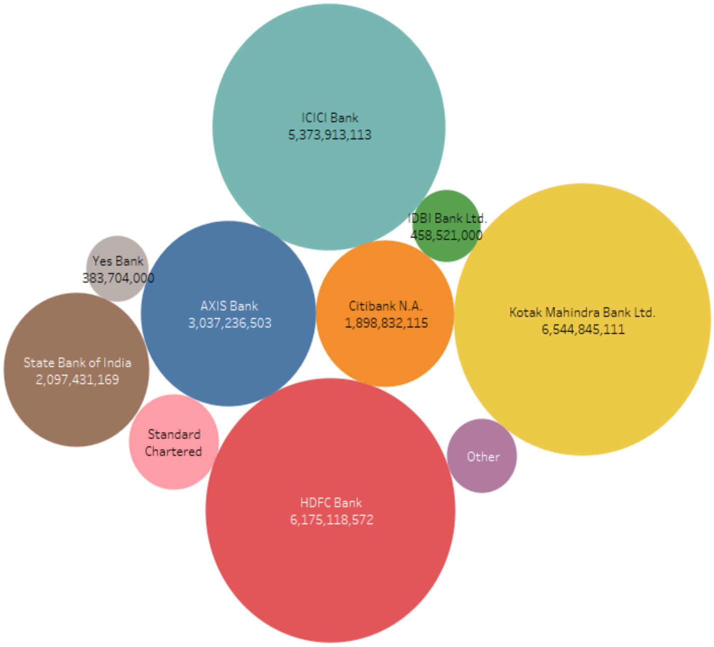
- **Trend of Online Applications:-** The following graph indicates the trend of online-loan applications in different locations and the response of them in case of acceptance and rejection.



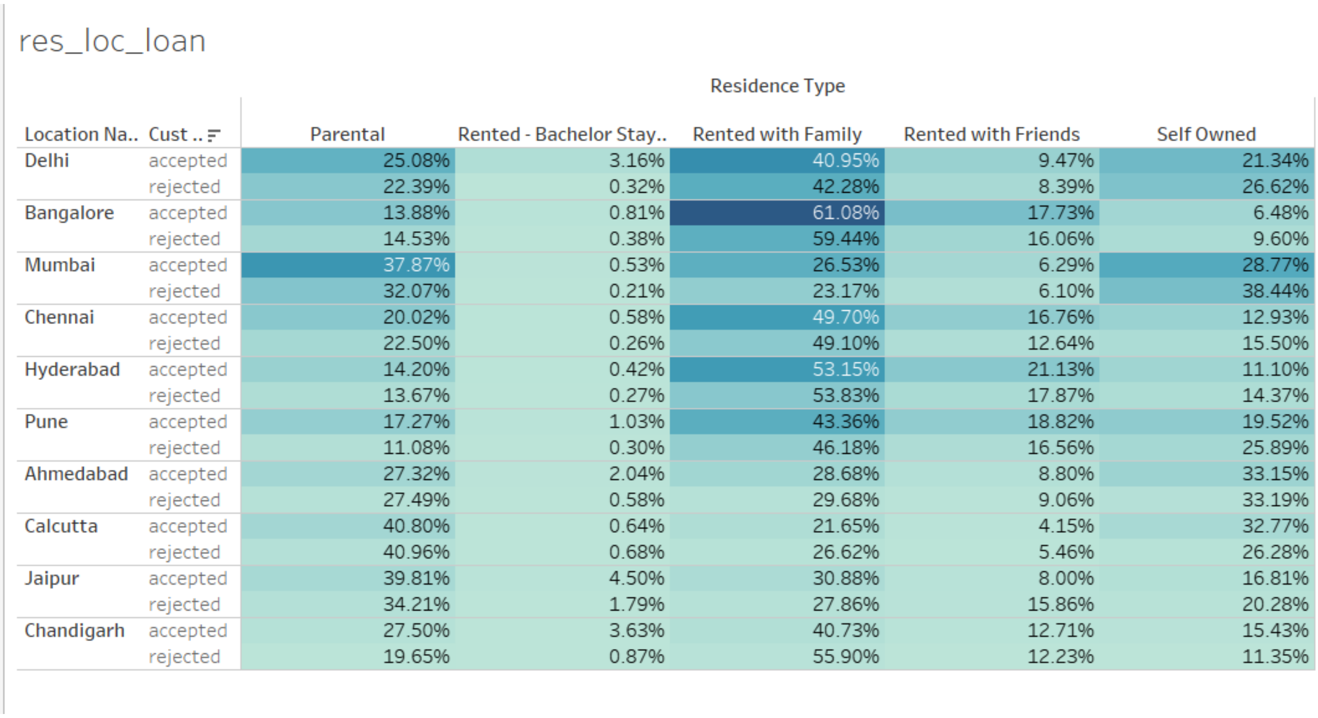
- **Loan amount with respect to residence type:-** The following graph includes the amount of loan applied by the customers at different locations and corresponding residence type.



- **Top Banks:-** The following graph indicates the top banks and the corresponding total sanction amount.



- **LOAN STATUS W.R.T LOACTION AND RESIDENT TYPE**



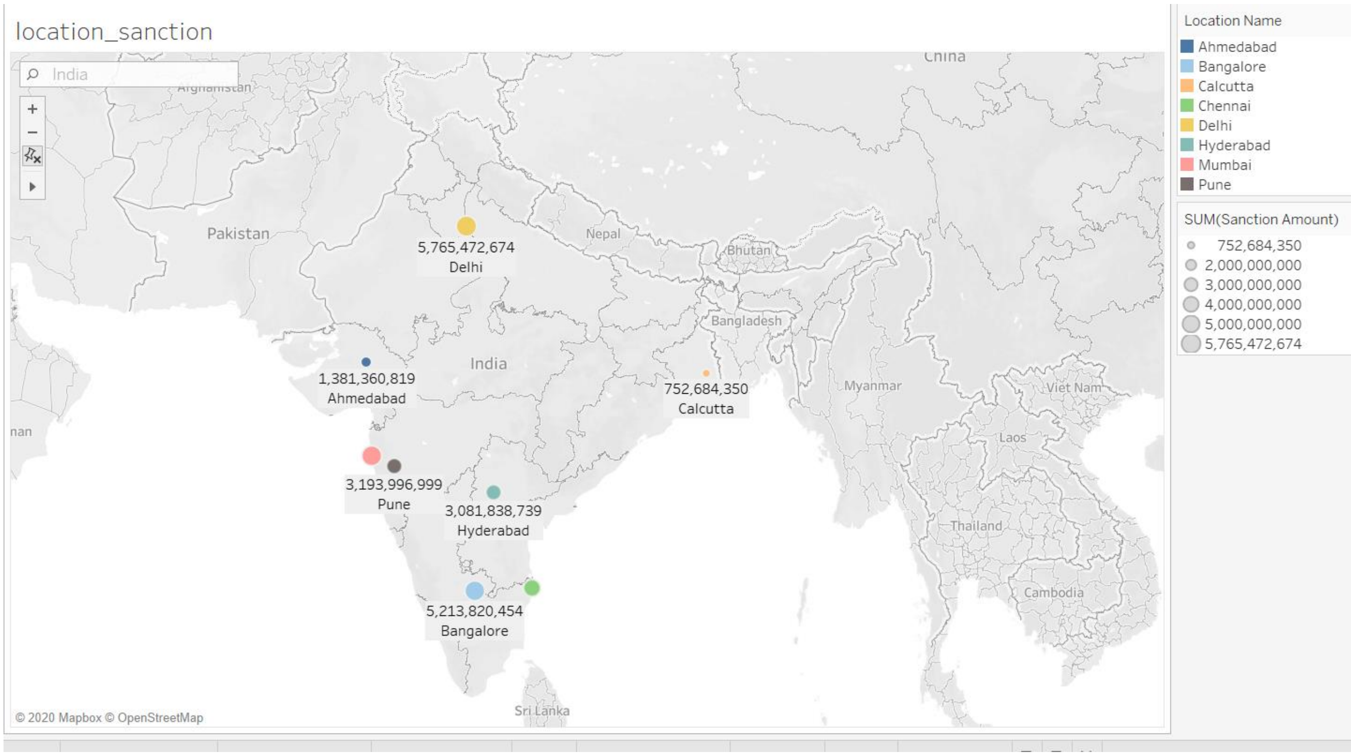
res_loc_loan

		Residence Type				
Location Na..	Cust ..	Parental	Rented - Bachelor Stay..	Rented with Family	Rented with Friends	Self Owned
Delhi	accepted	25.08%	3.16%	40.95%	9.47%	21.34%
	rejected	22.39%	0.32%	42.28%	8.39%	26.62%
Bangalore	accepted	13.88%	0.81%	61.08%	17.73%	6.48%
	rejected	14.53%	0.38%	59.44%	16.06%	9.60%
Mumbai	accepted	37.87%	0.53%	26.53%	6.29%	28.77%
	rejected	32.07%	0.21%	23.17%	6.10%	38.44%
Chennai	accepted	20.02%	0.58%	49.70%	16.76%	12.93%
	rejected	22.50%	0.26%	49.10%	12.64%	15.50%
Hyderabad	accepted	14.20%	0.42%	53.15%	21.13%	11.10%
	rejected	13.67%	0.27%	53.83%	17.87%	14.37%
Pune	accepted	17.27%	1.03%	43.36%	18.82%	19.52%
	rejected	11.08%	0.30%	46.18%	16.56%	25.89%
Ahmedabad	accepted	27.32%	2.04%	28.68%	8.80%	33.15%
	rejected	27.49%	0.58%	29.68%	9.06%	33.19%
Calcutta	accepted	40.80%	0.64%	21.65%	4.15%	32.77%
	rejected	40.96%	0.68%	26.62%	5.46%	26.28%
Jaipur	accepted	39.81%	4.50%	30.88%	8.00%	16.81%
	rejected	34.21%	1.79%	27.86%	15.86%	20.28%
Chandigarh	accepted	27.50%	3.63%	40.73%	12.71%	15.43%
	rejected	19.65%	0.87%	55.90%	12.23%	11.35%

Inference from the above graph are:

Highest percent of acceptance rate can be seen in the Rented with Family residence in Bangalore at 61.08% and also the rejection rate is highest in Bangalore at 59.44%. Followed by Hyderabad. This shows that rented families from these 2 cities have high and active loan applications compared to other residence types in different cities.

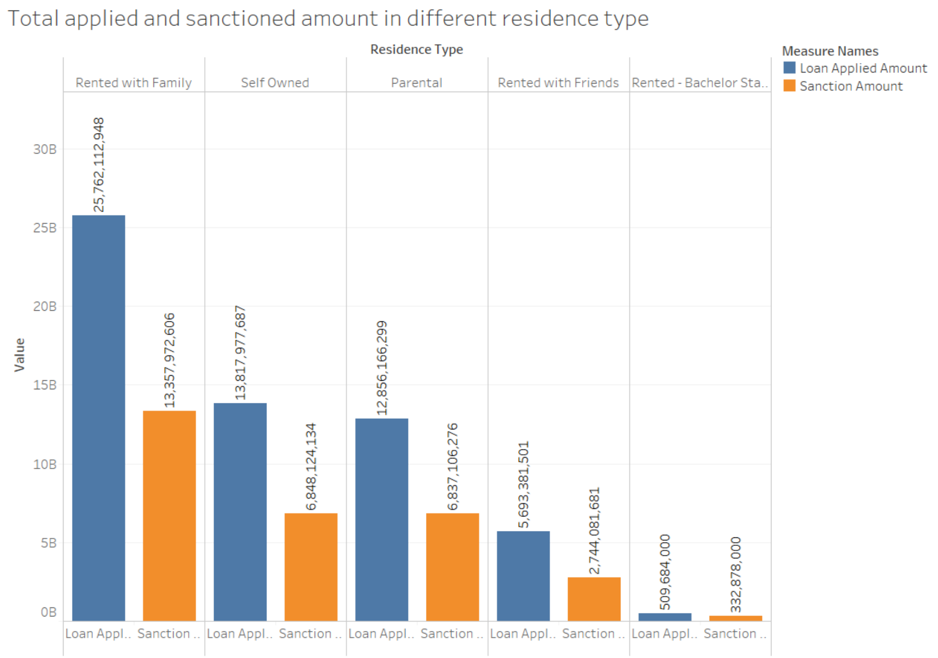
- **TOTAL SANCTION AMOUNT IN TOP FIVE LOCATIONS**



Above plot shows top 5 locations with highest sanction amount in the given time period

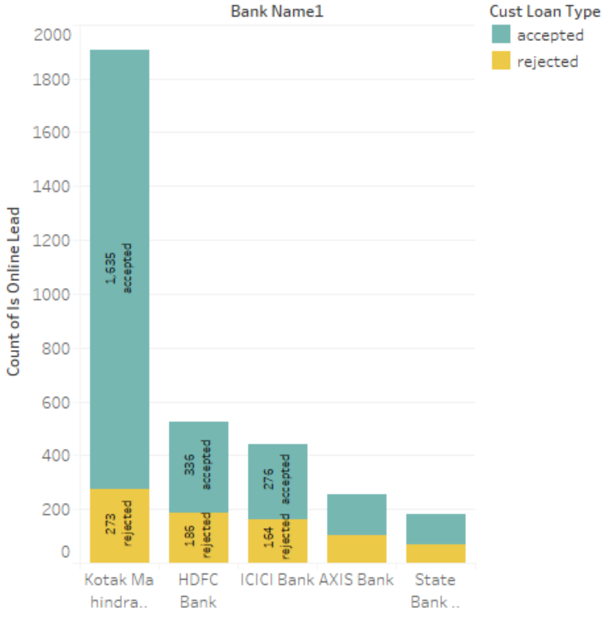
1. Delhi, Capital of India has highest sanction amount for Kotak customers.
2. Followed by Bangalore

- **TOTAL APPLIED SANCTIONED AMOUNT IN DIFFERENT RESIDENCE TYPE**

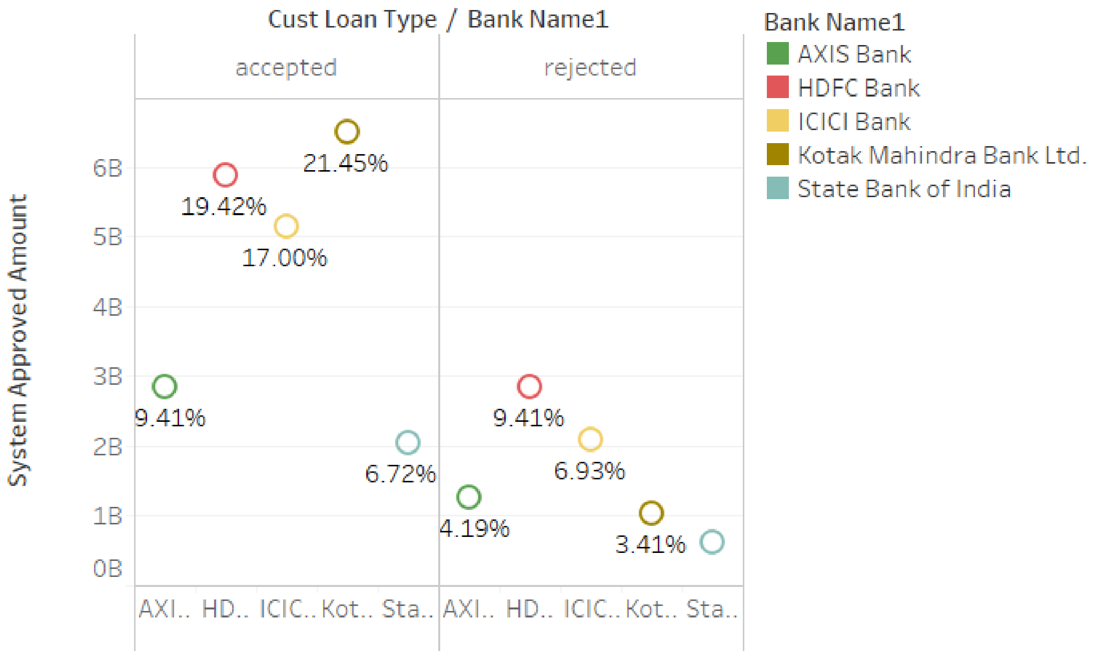


Loan Applied Amount and Sanction Amount for each Residence Type. Color shows details about Loan Applied Amount and Sanction Amount. The marks are labeled by Loan Applied Amount and Sanction Amount. The view is filtered on Residence Type, which has multiple members selected.

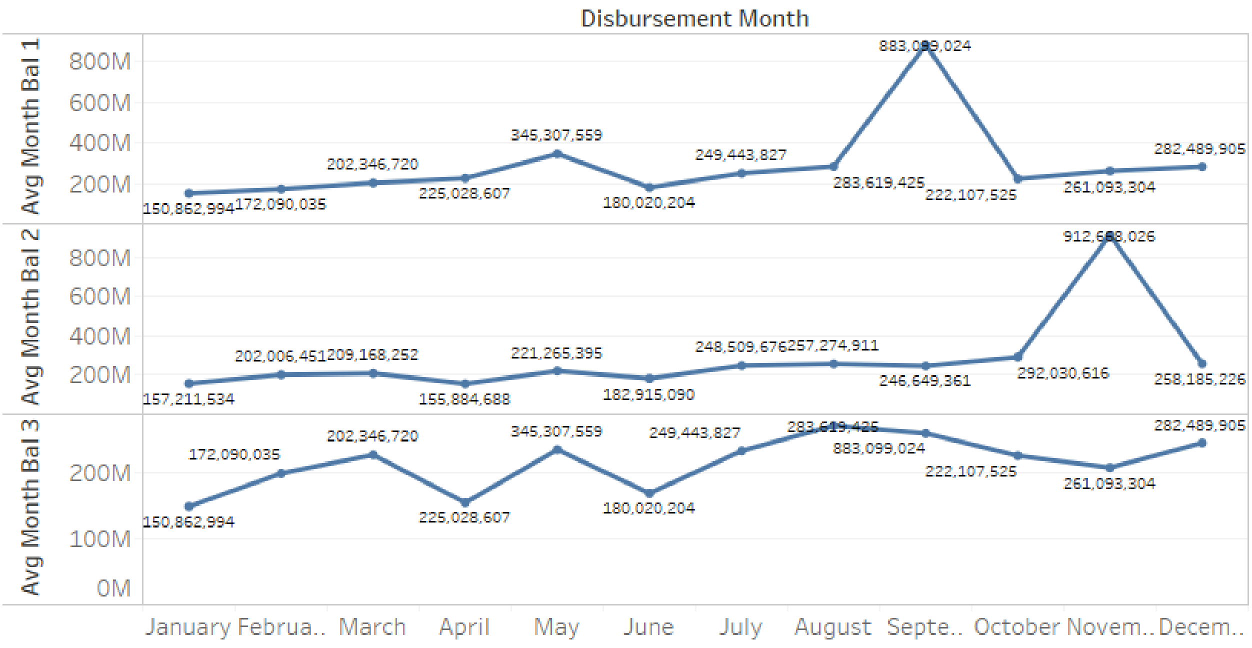
- **TOP 5 BANKS VS IS ONLINE LEAD APPLICATION STATUS**

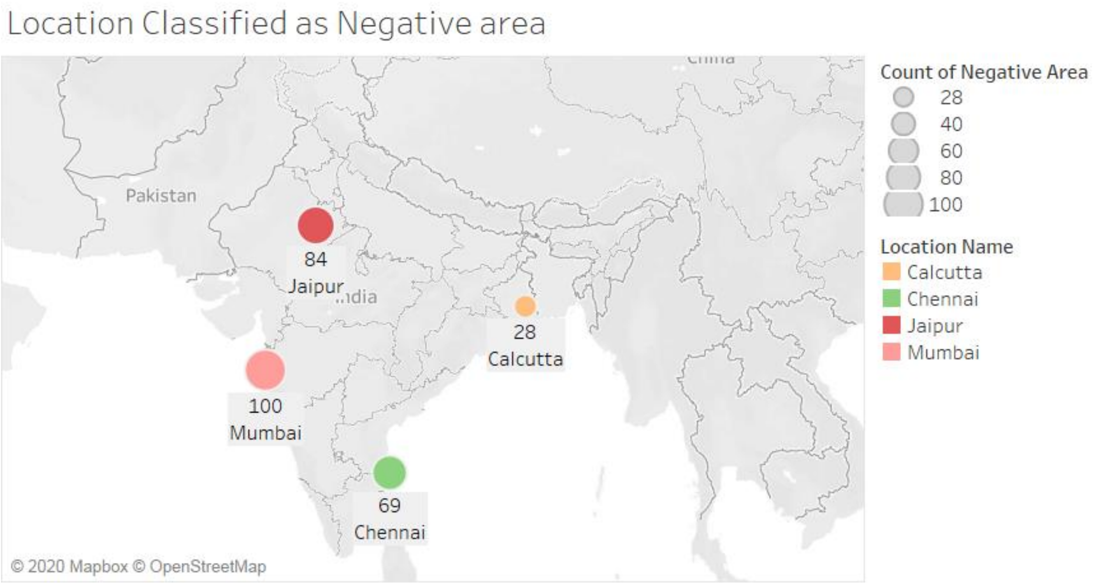


System_Approved_Amount in top 5 locations whether accepted or rejected.

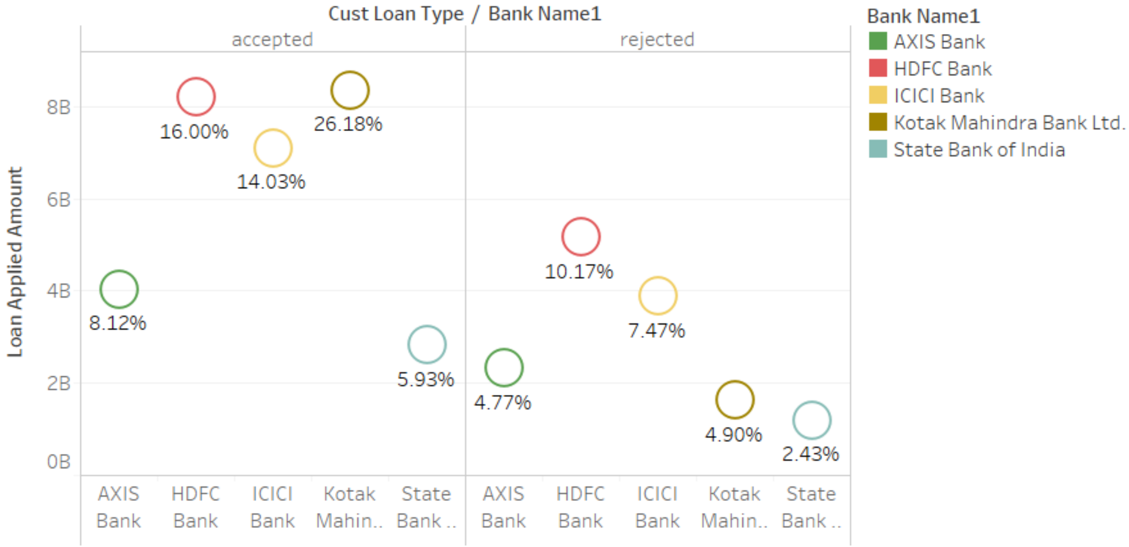


Avg Mothly Balance with respect to Disbursement Amount

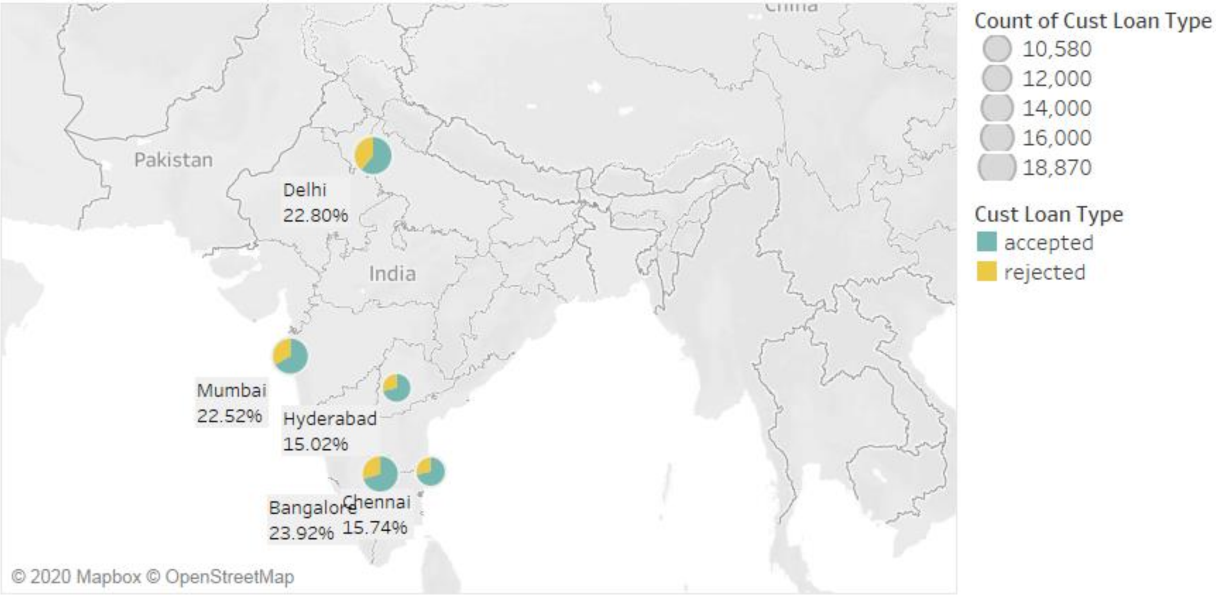




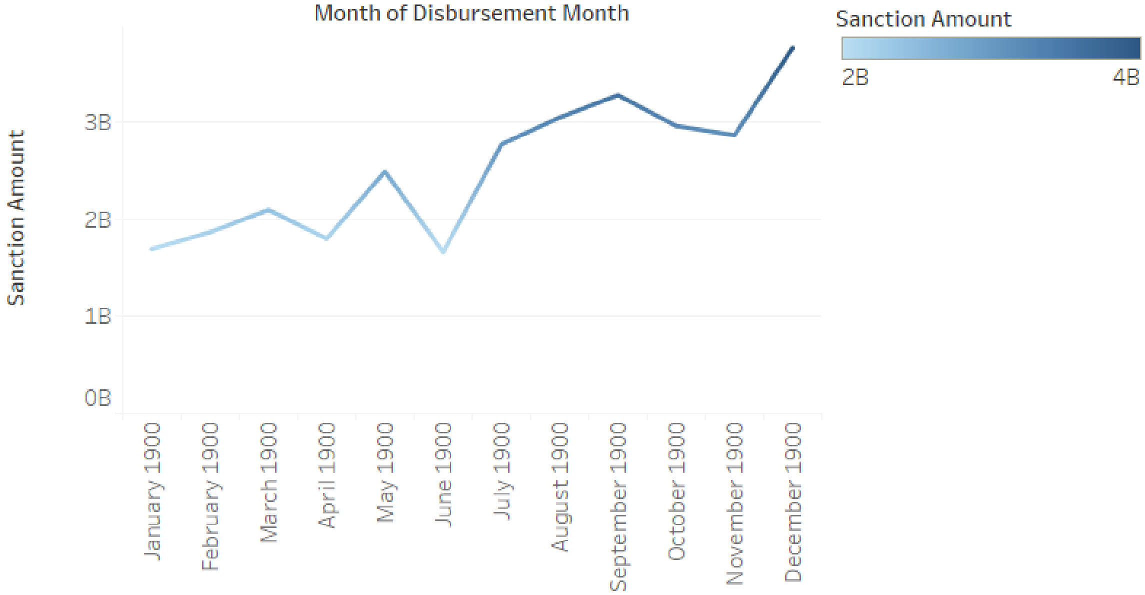
Top 5 banks with loan applications with respect to status



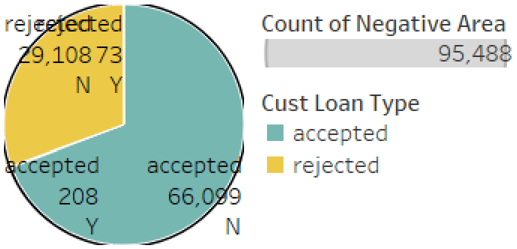
Status of top 5 location



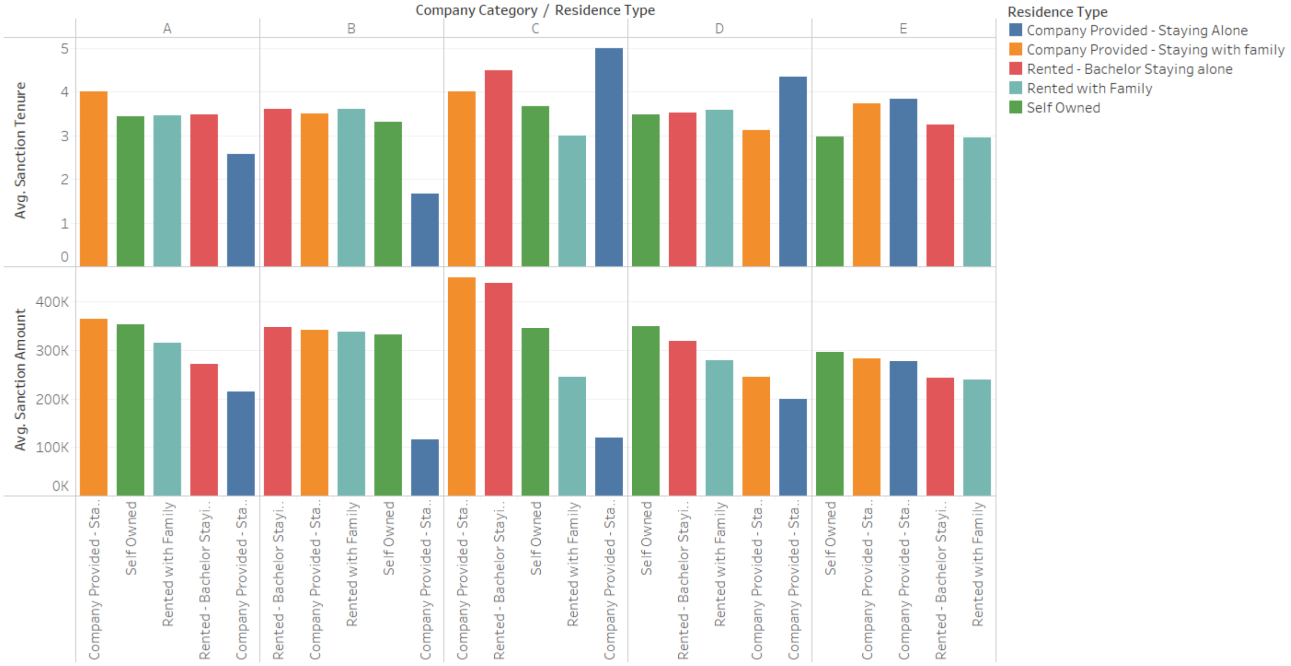
Sanction Amount with respect to Amount Disbursement.



Loan accepted or rejected in Negative area



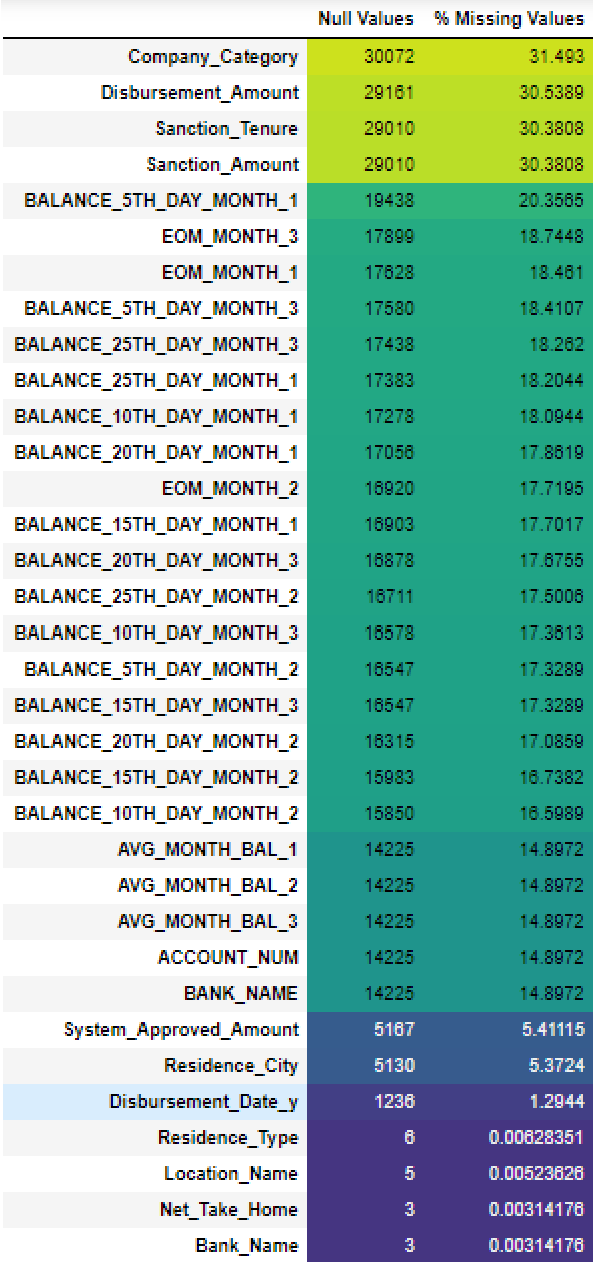
Avg sanction amount and avg sention tenure of Kotak with respect to different residence type



Average of Sanction Tenure and average of Sanction Amount for each Residence Type broken down by Company Category. Color shows details about Residence Type. The data is filtered on Bank Name1, which keeps Kotak Mahindra Bank Ltd.. The view is filtered on Residence Type, which keeps Company Provided - Staying Alone, Company Provided - Staying with family, Rented - Bachelor Staying alone, Rented with Family and Self Owned.

3. DATA CLEANING

3.1 Missing values :



	Null Values	% Missing Values
Company_Category	30072	31.493
Disbursement_Amount	29161	30.5389
Sanction_Tenure	29010	30.3808
Sanction_Amount	29010	30.3808
BALANCE_5TH_DAY_MONTH_1	19438	20.3565
EOM_MONTH_3	17899	18.7448
EOM_MONTH_1	17628	18.461
BALANCE_5TH_DAY_MONTH_3	17580	18.4107
BALANCE_25TH_DAY_MONTH_3	17438	18.262
BALANCE_25TH_DAY_MONTH_1	17383	18.2044
BALANCE_10TH_DAY_MONTH_1	17278	18.0944
BALANCE_20TH_DAY_MONTH_1	17056	17.8619
EOM_MONTH_2	16920	17.7195
BALANCE_15TH_DAY_MONTH_1	16903	17.7017
BALANCE_20TH_DAY_MONTH_3	16878	17.6755
BALANCE_25TH_DAY_MONTH_2	16711	17.5006
BALANCE_10TH_DAY_MONTH_3	16578	17.3613
BALANCE_5TH_DAY_MONTH_2	16547	17.3289
BALANCE_15TH_DAY_MONTH_3	16547	17.3289
BALANCE_20TH_DAY_MONTH_2	16315	17.0859
BALANCE_15TH_DAY_MONTH_2	15983	16.7382
BALANCE_10TH_DAY_MONTH_2	15850	16.5989
AVG_MONTH_BAL_1	14225	14.8972
AVG_MONTH_BAL_2	14225	14.8972
AVG_MONTH_BAL_3	14225	14.8972
ACCOUNT_NUM	14225	14.8972
BANK_NAME	14225	14.8972
System_Approved_Amount	5167	5.41115
Residence_City	5130	5.3724
Disbursement_Date_y	1236	1.2944
Residence_Type	6	0.00628351
Location_Name	5	0.00523626
Net_Take_Home	3	0.00314176
Bank_Name	3	0.00314176

As discussed in the previous sections, the data provided consists of a high amount of signal to noise ratio. Thus an in-depth understanding of each attribute in the data is very valid and specific. The data has a structure of 92 information attributes as discussed. There were a few logical assumptions been taken into account for taking each step of cleaning the data. Unlike different sensor data analysis having missing values or very strange anomalies in the banking, sector data is very rare. Thus the first assumption that has been taken upfront was even if there are any such anomalies or data missing, there has to be a reason for that not to be mentioned or mentioned in the information frame for a reason. That leads the team to proceed further for taking one attribute at a time and understand the nature of the data points in the attribute.

3.1 General Assumptions :

As a first step, we have analyzed the number of missing values and later figured out that the null values are too many along with there were none-registered entered in most of the columns represented by '-' symbol. These missing entries were converted into a null-value format for easy handling of the data.
Secondly, the 'Location_Name' and 'Residence_City' columns had a few data points and a few of them were meant to be the same as the others but only a portion of the data was registered to the database. For example, data points like 'Chenna' were the clear representation of the data point Chennai but due to missing a few letters the data points were taken as two different entries.
As a next step implementing string operations on different columns in order to obtain only numerical values in the data has done.
Treatment of Time Stamp was the next step we have taken in order to make the time stamp format into a numerical format so that the model algorithm can take it as an input. A few other attributes had the same small issues with half-registered entries in the case of data points.

Detailed assumptions:

Column 'Disbursement_Date_x' and 'Disbursement_Date_y' has the same values written in two different formats and 'y' is providing more values whereas 'x' contains a number of missing values, thus we removed 'Disbursement_Date_x' from the data completely from further analysis.
'Bank_Name' column is a categorical column with the representation of different bank names associated with Bank-Loan, but here the number of missing values are very high, thus
we came to a conclusion that it is missing because those banks are not coming within the preset bank list and thus considered them as a different entity.

The target variable: cust_loan_type' has four different classifications, even though the analysis objective is a binary classification problem, thus once we looked closer there are data points representing Accepted, Approved, Rejected and Unknown. Within the limits of our understanding, we came to the conclusion that Approved and Accepted are the same data points along with Unknown data points. The respective columns for all the 'Unknown' data points in the sanction amount category we can see that there is a sanction amount associated with the 'Unknown' data point. We assumed that if there is a sanction amount associated with these data points they represent the loan accepted categories only.

ATTRIBUTE NAME	CLEANING IDEA
Location_Name	Is filled with mode of the location name column.

Residence_City	data['Residence_City']=data['Residence_City'].fillna('Missing')
	data1=data[data['Residence_City']=='Missing']
	data2=data[data['Residence_City']!='Missing']
	data1['Residence_City']=data1['Location_Name']
	data=pd.concat([data1,data2])

Residence_Type	Is filled with mode of the residence type column
Bank_Name	The missing values in this column is filled ‘others’.(64 banks)
BANK_NAME	Is filled as 'No_bank' because that all corresponding rows doesnot have any sanction,disbursement amount in that.
Company_Category	The missing values in this column is filled ‘others’.
Loan_Applied_Amount	NO MISSING VALUES
Loan_Applied_tenure	NO MISSING VALUES
Sanction_Amount Sanction_Tenure Disbursement_Amount System_Approved_Amount	data1=data[data['cust_loan_type']=='accepted'] data2=data[data['cust_loan_type']=='rejected'] data2['Sanction_Amount']=0 data2['Sanction_Tenure']=0 data2['Disbursement_Amount']=0 data2['System_Approved_Amount']=0 data=pd.concat([data1,data2])
Disbursement_Date_y	Is filled with the mode of accepted disbursement date.
Net_Take_Home	No MISSING VALUES
Negative_Area	NO MISSING VALUES
Is_Online_Lead	NO MISSING VALUES
cust_loan_type	NO MISSING VALUES
Balance_5th_day_month_1 Balance_10th_day_month_1 Balance_15th_day_month_1 Balance_20th_day_month_1 Balance_25th_day_month_1 Eom_month_1 Balance_5th_day_month_2 Balance_10th_day_month_2 Balance_15th_day_month_2 Balance_20th_day_month_2 Balance_25th_day_month_2 Eom_month_2 Balance_5th_day_month_3 Balance_10th_day_month_3 Balance_15th_day_month_3 Balance_20th_day_month_3 Balance_25th_day_month_3 Eom_month_3	*def monthly_balance_imputation(column_to_impute,avg_monthly):* *list_monthly=[]* *for i,v in enumerate(column_to_impute.isnull()):* *if v==True:* *v=avg_monthly[i]* *else:* *v=column_to_impute[i]* *list_monthly.append(v)* *return list_monthly* *data['Month-1 Day-5']=monthly_balance_imputation(data['Month-1 Day-5'],data['Month-1 Average Monthly Balance'])* *data['Month-1 Day-10']=monthly_balance_imputation(data['Month-1 Day-10'],data['Month-1 Average Monthly Balance'])* *data['Month-1 Day-15']=monthly_balance_imputation(data['Month-1 Day-15'],data['Month-1 Average Monthly Balance'])* *data['Month-1 Day-20']=monthly_balance_imputation(data['Month-1 Day-20'],data['Month-1 Average Monthly Balance'])* *data['Month-1 Day-25']=monthly_balance_imputation(data['Month-1 Day-25'],data['Month-1 Average Monthly Balance'])*

	data['Month-1 End of Month Balance']=monthly_balance_imputation(data['Month-1 End of Month Balance'],data['Month-1 Average Monthly Balance'])

ACCOUNT_NUM,
Avg_month_bal_1,
Avg_month_bal_2,
Avg_month_bal_3
	*data1=data[data['BANK_NAME']=='0.0']
	data2=data[data['BANK_NAME']!='0.0']
	data1['Month-1 Average Monthly Balance']=0
	data1['Month-2 Average Monthly Balance']=0
	data1['Month-3 Average Monthly Balance']=0
	data1['ACCOUNT_NUM']=0 #Othercolumns will be automatically fixed.
	data=pd.concat([data1,data2])
	
	#First_Run
	def avg_balance_imputation(avg_column,day_5,day_10,day_15,day_20,day_25,month_end_bal):
	    list_avg=[]
	    for i,v in enumerate(avg_column.isnull()):
	        if v==True:
	v=np.mean([day_5[i],day_10[i],day_15[i],day_20[i],day_25[i],month_end_bal[i]])
	        else:
	            v=avg_column[i]
	        list_avg.append(v)
	    return list_avg
	
	data['Month-1 Average Monthly Balance']=avg_balance_imputation(data['Month-1 Average Monthly Balance'],
	data['Month-1 Day-5'],data['Month-1 Day-10'],
	data['Month-1 Day-15'],data['Month-1 Day-20'],
	data['Month-1 Day-25'],data['Month-1 End of Month Balance'])*

Monthly-Balance Assumptions :

We have data of three different months distributed among a major portion of the data structure. Month-1 represents the first 30 days after receiving the loan and the day number associated with the day count within each 30-days span time. Renaming them into something which is more reliable to understand was the second phase of this section of data.

3.2 Missing Value Imputation :

As mentioned above, there were missing values present in the data, along with that now the newly transformed null-values from '-' also present in the data. As explained before, the data provided cannot have a long list of missing values, even if they were represented with some notation we underlined the condition that it will be due to some solid reasons. The imputation of the data was done as follow:

Column Name	Method of Imputation	Logical Reason
Location_Name, Residence_Type And Disbursement Timestamp	Mode Imputation	The number of missing data points were less. New imputed values are aligning with other existing data points.
Company_Category	Creating a separate data point	The number of missing data was high. Since the actual data kind is not known to treat as a separate data point.
Net_Take_Home	Imputed with zero	All corresponding loan status was rejected, there are no values to be added here since the loan status is ‘Reject’
Average Monthly Balance	Using manual function	Used the function which takes up different balance data points, calculates average and re-plug into the respective column.
Day-wise balance in one month	Using Manual Function	In this, we assumed that if the data provided does not have a specific sequence of balance data, in the column we can take the data from average_column.

Conditional Imputation:

Here in the case of a few attributes, there were huge number of missing values associated, but when we take a closer analysis we can see that the attributes 'Sanction_Amount', 'Sanction_Tenure', 'Disbursement_Amount', 'System_Approved_Amount’ they have missing values but the corresponding cust_loan_type was ‘Rejected’ for all of them, ideally, for a rejected loan, there will not be any data attached about sanctioned amount, tenure, etc. Thus since the respective loan status is rejected we can impute these null values by the value zero. Along with that the un-registered ‘BANK_NAME’ column when there is no bank associated with the profile, the average balance of the month is assumed to be zero, once we imply the function for other day values it will be automatically get implemented all across. From 40 to the final attribute, they are dummified variables of a category in a sparse matrix format, since 40th column does not have any missing values and the range is between 0-1, the manual calculation on row-wise imputation has to be done on the data on the basic assumption we have take, if the value is missing in any column, then it will get imputed with the value of 40th column.

4. STATISTICS ANALYSIS

Statistical testing was done to look at the evidence for a particular hypothesis being true. This helped us to accomplish decisions. Hypothesis will be performed to statistically validate the impact of respective independent variables on the outcome. Since 7 variables are discrete, we will perform Chi-Square test to test feature significance.

4.1 CHI-SQUARE TEST

This test is performed between all 7 categorical independent variables and target column (cust_loan_type).

Hypothesis (Alpha=0.01)
Ho: *The feature is not significant predictor of Target.*
Ha: *The feature is significant predictor, i.e., it has high association with Target.*

CATEGORICAL COLUMNS	P-VALUE	RESULT
Location_Name	0.0	Has association
Company_Category	0.0	Has association
Residence_Type	0.0	Has association
Negative_Area	0.5940251890440905	Has no association (Insignificant)
Bank_Name	0.0	Has association
Residence_City	0.0	Has association
Is_Online_Lead	4.1905020497503496e-13	Has association
Loan_Applied_tenure	4.852138427273385e-56	Has association
Sanction_Tenure	0.0	Has association
Disbursement_Year	3.382129212553857e-165	Has association
Disbursement_Month	0.0	Has association
Disbursement_Day	0.0	Has association

4.2 T-TEST UNPAIRED

T-test unpaired is applied for the rest 31 continuous variables with respect to target variables. Since All the continuous variables are right skewed. So, the assumptions (Normality test and variance test) that we consider before performing t test is violated and can be checked by Shapiro test. So, mannwhitneyu non-parametric need to be performed

Hypothesis (Alpha=0.01)
Ho: *The feature is not significant predictor of Target i.e: mean of respective levels are same.*
Ha: *The feature is significant predictor, i.e: means of respective levels are not equal.*

Result: THE INSIGNIFICANT COLUMNS ARE: ['Month-2 Day-20', 'Month-2 Day-25', 'Month-1 Day-25', 'Month-3 Day-25']

The above mentioned columns are insignificant because the balance of End of the Month for that particular month can be similar to that of the mentioned insignificant columns.

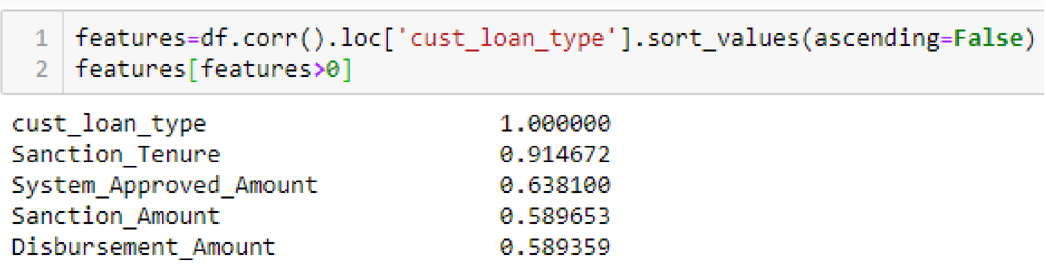
5. MODEL BUILDING

5.1 DEFINE X AND Y VARIABLE

Now, before applying any model, we have prepared the data and segregate the features and the label of the dataset. Variable X contains all the independent variables that are necessary to make prediction. Variable y has cust_loan_type as target variable.

5.2 HIGH CORRELATED COLUMNS :-

The column like sanction_tenure, system_approved_amount, sanction_amount and Disbursement_amount are highly correlated with the customer loan type and it will badly effect our model building. So, it need to be removed for the purpose of model building.

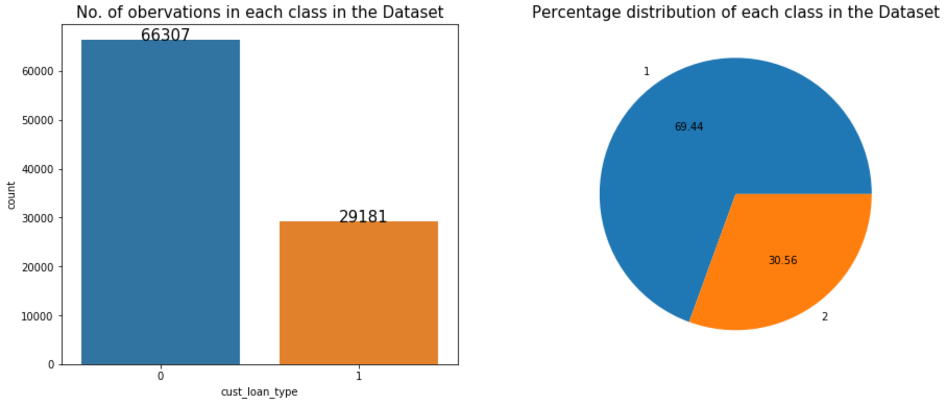


```
features=df.corr().loc['cust_loan_type'].sort_values(ascending=False)
features[features>0]
```

```
cust_loan_type                  1.000000
Sanction_Tenure                 0.914672
System_Approved_Amount          0.638100
Sanction_Amount                 0.589653
Disbursement_Amount             0.589359
```

5.3 UPSAMPLING OF TARGET VARIABLE :-

CUST_LOAN_TYPE :- Customer loan type status is divided into two categories as “Accepted” and “Rejected”, with respective proportions of 69.44% and 30.56%. And the minority class is only 30% as compared to majority so up sampling can be performed on the cust_loan_type column.



SMOTE :-

SMOTE stands for Synthetic Minority Oversampling Technique. This is a statistical technique for increasing the number of cases of the lower value in the dataset in a balanced way. This is done to avoid building a bias model where the train data would mostly or max contain the highest of the values. The module works by generating new values from existing minority cases that act as supply input. Implementation of SMOTE does not affect the majority cases

```
smote = SMOTE ( )
X_train_sm , y_train_sm = smote.fit_sample ( X_train , y_train.ravel ( ) )
```

ONE HOT ENCODING :- One hot encoding is a process by which categorical variables are converted into a form that could be provided to ML algorithms to do a better job in prediction. In our model building, the dataset contains seven categorical columns.

CATEGORICAL COLUMNS	LEVELS
Location_Name	5
Company_Category	15
Residence_Type	9
Negative_Area	2
Bank_Name	64
Residence_City	67
Is_Online_Lead	2

NORMALIZATION :- Transform features by scaling each feature to a given range. This estimator scales and translates each feature individually such that it is in the given range on the training set, e.g. between zero and one.

TRUNCATEDSVD :- Truncated Singular Value Decomposition (**SVD**) is a matrix factorization technique that is applied on the sparse matrix. Typically, **SVD** is used under the hood to find the principle components of a matrix.

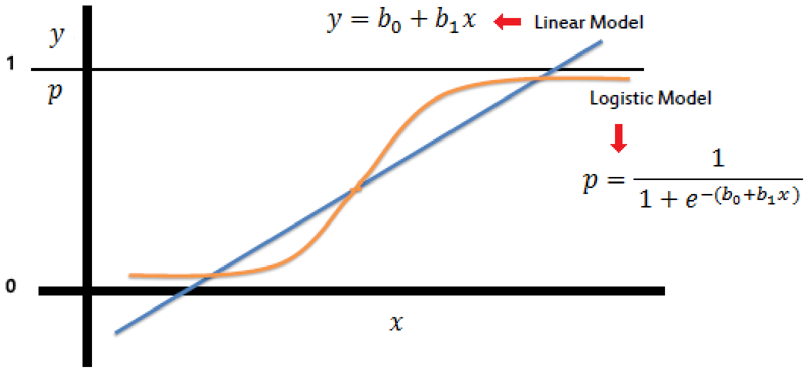
5.4 ALGORITHMS USED :-

Since our problem is a classification problem, we will be using the following algorithms in modelling:-

LogisticRegression

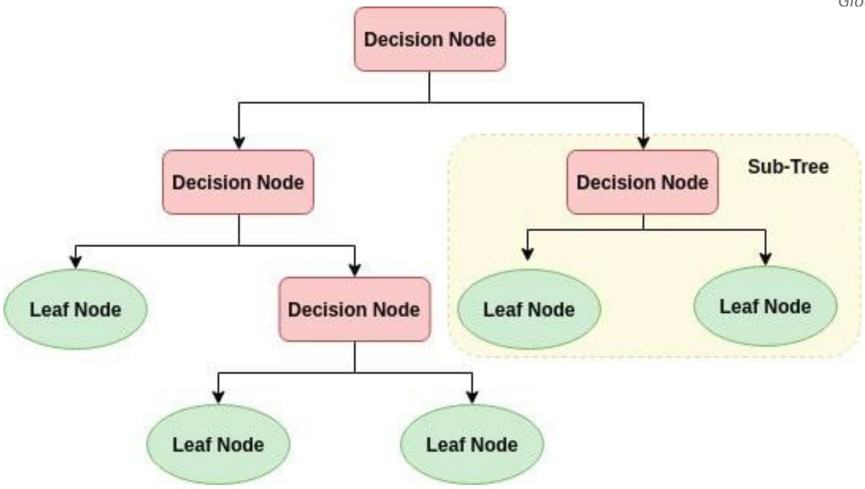
Logistic regression predicts the probability of an outcome that can only have two values (i.e. a dichotomy). The prediction is based on the use of one or several predictors (numerical and categorical). A linear regression is not appropriate for predicting the value of a binary variable for

two reasons: A linear regression will predict values outside the acceptable range (e.g. predicting probabilities outside the range 0 to 1). Since the dichotomous experiments can only have one of two possible values for each experiment, the residuals will not be normally distributed about the predicted line. On the other hand, a logistic regression produces a logistic curve, which is limited to values between 0 and 1. Logistic regression is similar to a linear regression, but the curve is constructed using the natural logarithm of the “odds” of the target variable, rather than the probability. Moreover, the predictors do not have to be normally distributed or have equal variance in each group.



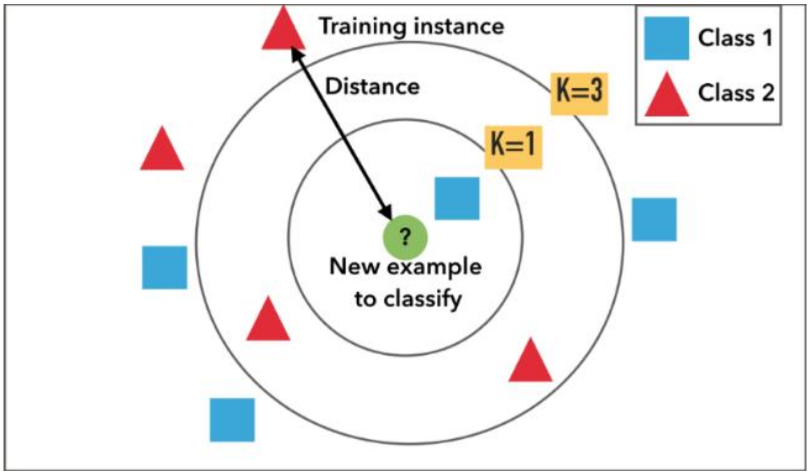
DecisionTree

A Decision tree (CART) is a schematic, tree-shaped diagram used to determine a course of action or show a statistical probability. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. A decision node has two or more branches. Leaf node represents a classification or decision. The topmost decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data.



KNeighborsClassifier

KNN is a simple yet powerful classification algorithm. It requires no training for making predictions, which is typically one of the most difficult parts of a machine learning algorithm. The KNN algorithm has been widely used to find document similarity and pattern recognition. The intuition behind the KNN algorithm is one of the simplest of all the supervised machine learning algorithms. It simply calculates the distance of a new data point to all other training data points. The distance can be of any type e.g. Euclidean or Manhattan etc. It then selects the K-nearest data points, where K can be any integer. Finally, it assigns the data point to the class to which the majority of the K data points belong.

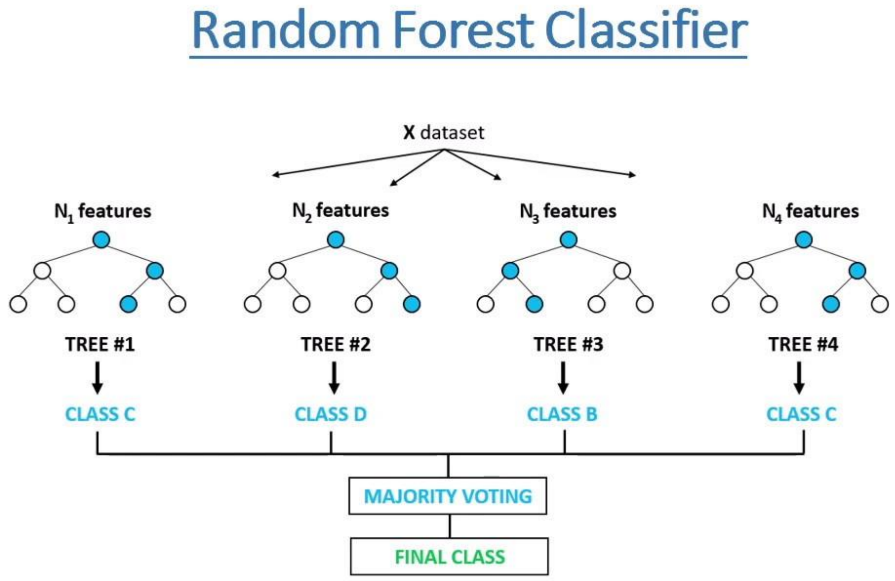


ExtraTreeClassifier

This class implements a meta estimator that fits a number of randomized decision trees (a.k.a. extra-trees) on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

RandomForestClassifier

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. To say it in simple words: Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction. One big advantage of random forest is, that it can be used for both classification and regression problems. Random Forest has nearly the same hyper parameters as a decision tree or a bagging classifier. Fortunately, we don't have to combine a decision tree with a bagging classifier and can just easily use the classifier-class of Random Forest. Like I already said, with Random Forest, you can also deal with Regression tasks by using the Random Forest regressor. Random Forest adds additional randomness to the model, while growing the trees. Instead of searching for the most important feature while splitting a node, it searches for the best feature among a random subset of features. This results in a wide diversity that generally results in a better model.



GaussianNB

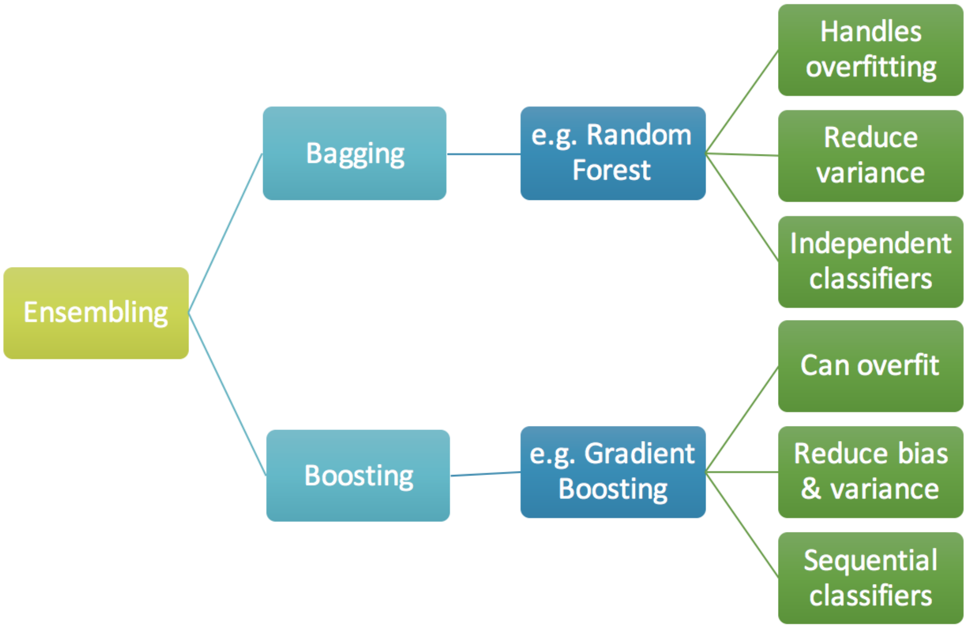
A Gaussian Naive Bayes algorithm is a special type of NB algorithm. It's specifically used when the features have continuous values. It's also assumed that all the features are following a gaussian distribution i.e, normal distribution.

GradientBoostingClassifier

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models

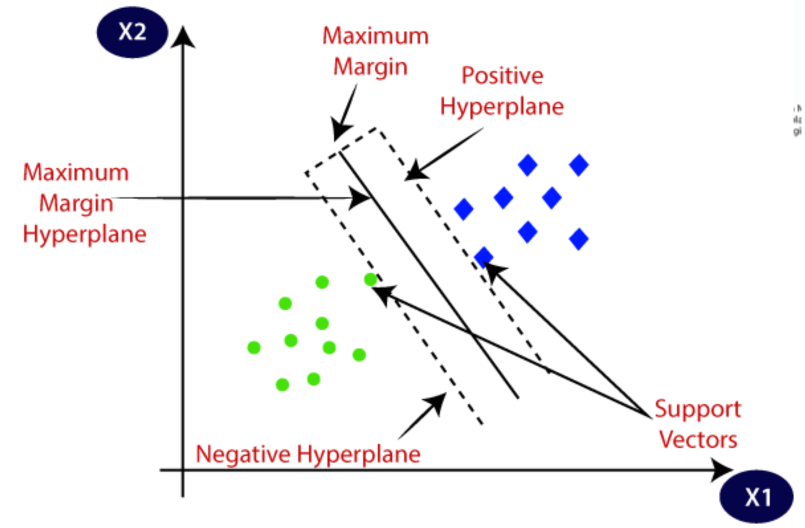
AdaBoostClassifier

An AdaBoost classifier is a meta-estimator that begins by fitting a classifier on the original dataset and then fits additional copies of the classifier on the same dataset but where the weights of incorrectly classified instances are adjusted such that subsequent classifiers focus more on difficult cases.



Support_vector_classifier

“Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. ... Support Vector Machine is a frontier which best segregates the two classes (hyper-plane/ line).



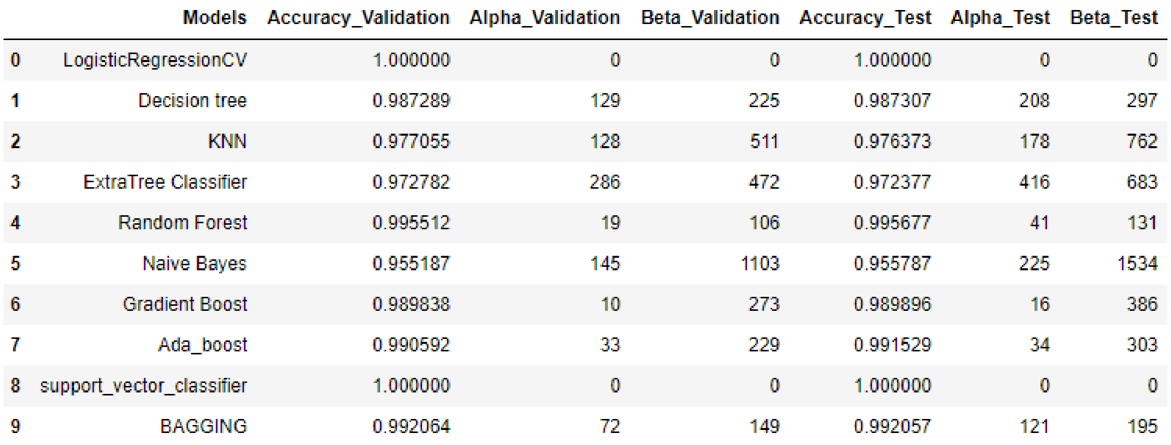
BaggingClassifier

A Bagging classifier is an ensemble meta-estimator that fits base classifiers each on random subsets of the original dataset and then aggregate their individual predictions (either by voting or by averaging) to form a final prediction. Such a meta-estimator can typically be used as a way to reduce the variance of a black-box estimator (e.g., a decision tree), by introducing randomization into its construction procedure and then making an ensemble out of it.
Each base classifier is trained in parallel with a training set which is generated by randomly drawing, with replacement, N examples(or data) from the original training dataset – *where N is the size of the original training set*. Training set for each of the base classifiers is independent of each other. Many of the original data may be repeated in the resulting training set while others may be left out.

Bagging reduces overfitting (variance) by averaging or voting, however, this leads to an increase in bias, which is compensated by the reduction in variance though.

5.5 MODEL COMPARISON:-

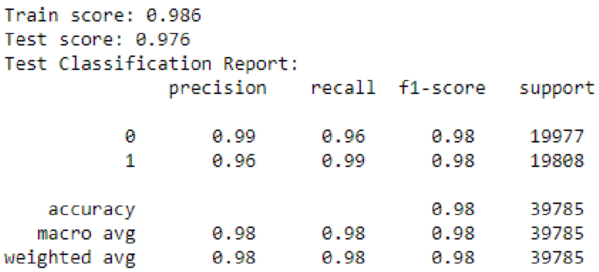
After applying all the classification models KNN gives better and satisfactory results as compared to all other classification models.



	Models	Accuracy_Validation	Alpha_Validation	Beta_Validation	Accuracy_Test	Alpha_Test	Beta_Test
0	LogisticRegressionCV	1.000000	0	0	1.000000	0	0
1	Decision tree	0.987289	129	225	0.987307	208	297
2	KNN	0.977055	128	511	0.976373	178	762
3	ExtraTree Classifier	0.972782	286	472	0.972377	416	683
4	Random Forest	0.995512	19	106	0.995677	41	131
5	Naive Bayes	0.955187	145	1103	0.955787	225	1534
6	Gradient Boost	0.989838	10	273	0.989896	16	386
7	Ada_boost	0.990592	33	229	0.991529	34	303
8	support_vector_classifier	1.000000	0	0	1.000000	0	0
9	BAGGING	0.992064	72	149	0.992057	121	195

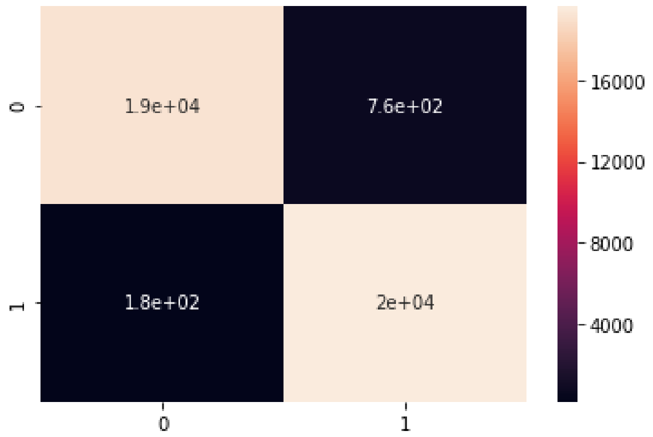
KNN

Classification report

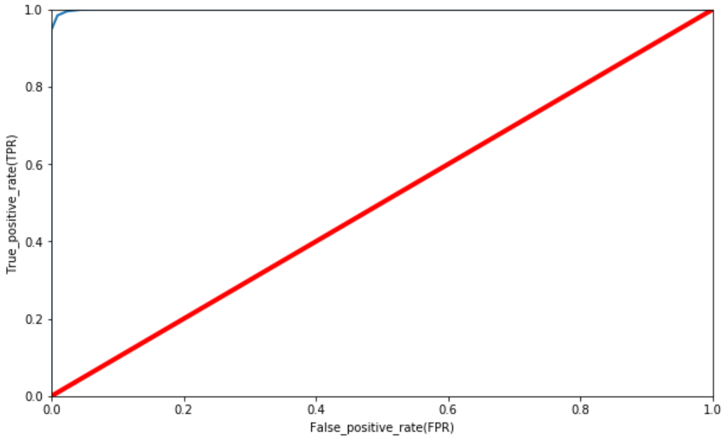


```
Train score: 0.986
Test score: 0.976
Test Classification Report:
              precision    recall  f1-score   support

           0       0.99      0.96      0.98     19977
           1       0.96      0.99      0.98     19808

    accuracy                           0.98     39785
   macro avg       0.98      0.98      0.98     39785
weighted avg       0.98      0.98      0.98     39785
```

CONFUSION MATRIX



ROC-AUC CURVE



```
ROC_AUC_SCORES 0.999468130876327
```

CONCLUSION

With SMOTE the model is giving best result (roc_auc_score = 0.99) with K nearest neighbour. Because the data is highly imbalanced so base models without any resampling techniques are not giving good result. Even with undersampling the model is not showing satisfactory result. One of the key reasons why SMOTE is giving better is that Over-sampling simply replicates the observations from the minority class, though this method does lead to no loss in information unlike Under-sampling.

The disadvantage of using this method is that the process of replication of observations in original data set, ends up adding multiple observations of several types, thus leading to over fitting. Although, the training accuracy of such data set will be high, but the accuracy on unseen data will be worse. Here's where the advantages of SMOTE come into play. Even though SMOTE is an oversampling technique, it creates synthetic observations instead of reusing existing observations and hence the chances of over fitting get lowered down.