PYTHON PROJECT

POST GRADUATION IN DATA ANALYTICS

PROJECT TITLE:

CREDIT RISK ANALYSIS (LOAN REPAYMENT)

Submitted and Presented By:

Arsh Modak

Saurav Jha

Saima Dhiyan

Ankita Talekar

ABSTRACT

A loan is money, property or other material goods given to another party in exchange for future repayment of the loan value amount, along with interest or other finance charges. A loan may be for a specific, one-time amount or can be available as an open-ended line of credit up to a specified limit or ceiling amount. Loans can come from individuals, corporations, financial institutions and governments.

Credit Analysis is the method by which one calculates the creditworthiness of a business or organization. In other words, it is the evaluation of the ability of a company to honour its financial obligations. The audited financial statements of a large company might be analysed when it issues or has issued bonds.

There could be times where the client or person receiving a loan may not be able to pay it, which could then lead to a loss to the lender. Institutions give loans to numerous people and if they are not able to pay it back the institution may face losses or even bankruptcy.

To prevent such a thing for happening, we use our technical knowledge of Machine Learning Algorithms to predict if the borrower of the loan will be able to pay back or no by utilizing prior knowledge of the client and the amount, they want to borrow along with other parameters such as background, financial status, etc.

INDEX

TABLE OF FIGURES

1. Introduction

Whenever an individual/corporation applies for a loan from a bank or financial institution, their credit history undergoes a rigorous check to ensure that whether they are capable enough to pay off the loan also referred to as credit-worthiness.

The issuers have a set of models and rules in place which take information regarding their current financial standing, previous credit history and some other variables as input and output a metric which gives a measure of the risk that the issuer will potentially take on issuing the loan. The measure is generally in the form of a probability and is the risk that the person will default on their loan (called the probability of default) in the future.

Based on the amount of risk that the issuer is willing to take (plus some other factors) they decide on a cut off of that score and use it to take a decision regarding whether to pass the loan or not. This is a way of managing credit risk. The whole process collectively is referred to as underwriting.

Variables	Types	Variables	Types
loan_amnt	float64	tot_coll_amt	float64
funded_amnt	float64	tot_cur_bal	float64
funded_amnt_inv	float64	total_rev_hi_lim	float64
int_rate	float64	acc_now_delinq	float64
installment	float64	member_id	int64
annual_inc	float64	default_ind	int64
dti	float64	term	object
delinq_2yrs	float64	grade	object
inq_last_6mths	float64	sub_grade	object
open_acc	float64	emp_title	object
pub_rec	float64	emp_length	object
revol_bal	float64	home_ownership	object
revol_util	float64	verification_status	object
total_acc	float64	issue_d	object
out_prncp	float64	pymnt_plan	object
out_prncp_inv	float64	purpose	object
total_pymnt	float64	title	object
total_pymnt_inv	float64	zip_code	object
total_rec_prncp	float64	addr_state	object
total_rec_int	float64	earliest_cr_line	object
total_rec_late_fee	float64	initial_list_status	object
recoveries	float64	last_pymnt_d	object
collection_recovery_fee	float64	next_pymnt_d	object
last_pymnt_amnt	float64	last_credit_pull_d	object
collections_12_mths_ex_med	float64	application_type	object
policy_code	float64		

The above-mentioned table shows the various variables that take part in the analysis. They are described as the following:

a. Float or Int (Numerical Variables)
b. Object (Categorical Variables)

There are 72 variables (columns) in total and the dataset has 855969 observations (rows).

2. Hardware and Software Details

Hardware:

- 12GB RAM (Min 8GB RAM required for smooth operations)
- 1TB HDD (As low as 128GB will suffice too)
- Intel(R) Core™ i5-7200 CPU @ 2.50GHz 2.71GHz

Software:

- OS: Windows 10
- Spyder (IDE for Python)
- Tableau

Language:

- Python 3

3. DATASET MANIPULATION

3.1 Summarization of the Dataset

	member_i	loan_amnt	funded_an	funded_an	int_rate	installmen	annual_inc	dti	delinq_2yr	inq_last_6	open_acc
count	855969	855969	855969	855969	855969	855969	855969	855969	855969	855969	855969
mean	34762690	14745.57	14732.38	14700.06	13.19232	436.2381	75071.19	18.12216	0.311621	0.680915	11.54245
std	23994177	8425.34	8419.472	8425.805	4.368365	243.7269	64264.47	17.42363	0.857189	0.964033	5.308094
min	70699	500	500	0	5.32	15.69	0	0	0	0	0
25%	10792732	8000	8000	8000	9.99	260.55	45000	11.88	0	0	8
50%	36975319	13000	13000	13000	12.99	382.55	65000	17.61	0	0	11
75%	58035586	20000	20000	20000	15.99	571.56	90000	23.9	0	1	14
max	73519693	35000	35000	35000	28.99	1445.46	9500000	9999	39	8	90

pub_rec	revol_bal	revol_util	total_acc	out_prncp	out_prncp	total_pym	total_pym	total_rec_	total_rec_	total_rec_
855969	855969	855523	855969	855969	855969	855969	855969	855969	855969	855969
0.194537	16910.53	55.0194	25.26927	8284.83	8281.449	7653.296	7622.221	5850.841	1755.046	0.31953
0.581585	22223.74	23.81158	11.81884	8461.947	8458.496	7909.384	7885.156	6676.411	2081.693	3.609399
0	0	0	2	0	0	0	0	0	0	0
0	6469	37.6	17	0	0	1969.69	1960.12	1239.95	451.27	0
0	11903	55.9	24	6290.25	6287.65	4976.16	4948.25	3286.89	1076.91	0
0	20857	73.5	32	13528.8	13522.51	10744.8	10697.33	8000	2233.98	0
86	2904836	892.3	169	49372.86	49372.86	57777.58	57777.58	35000.03	24205.62	358.68

recoveries	collection	last_pymi	collection	policy_co	acc_now_	tot_coll_a	tot_cur_b	total_rev_	default_ind
855969	855969	855969	855913	855969	855969	788656	788656	788656	855969
47.0895	4.95123	2225.99	0.01423	1	0.00494	225.413	139766	32163.6	0.05429
413.136	62.4786	4864.97	0.13371	0	0.07733	10489.4	153939	37699.6	0.22658
0	0	0	0	1	0	0	0	0	0
0	0	285.42	0	1	0	0	29870	14000	0
0	0	468.82	0	1	0	0	81008.5	23800	0
0	0	849.16	0	1	0	0	208703	39900	0
33520.3	7002.19	36475.6	20	1	14	9152545	8000078	9999999	1

- We can see that there is a total of 855969 candidate who have applied for loan where we have 72 variables justifying this data out of which 32 are missing values.
- Loan amount is one of the most important and unique variables in the entire picture where its standard mean comes to 14745.57133. Minimum loan amount is 500 and Maximum loan amount is 35000.
- Interest Rate equally plays a very important role. It is the proportion of a loan that is charged as interest to the borrower, typically expressed as an annual percentage of the loan outstanding. Standard mean comes to 13.19231961. Minimum loan amount is 5.32 and Maximum loan amount is 28.99.
- Installment here in the project talks about the sum of money due as one of several equal payments for something, spread over an agreed period of time where its standard mean comes to 436.2380718. Minimum loan amount is 15.69 and Maximum loan amount is 1445.46.
- Determining whether your income is sufficient or not will decide whether to grant the loan amount or not. The process here is not really easy. Standard mean comes to 75071.18596. Minimum Annual income is 500 and Annual income amount is 9500000.
- In finance the term **recovery** refers to collection of <u>amount due</u>. The normally recovery depends on the purpose, time and condition, business running process etc.
- Normally loan amount will be recovered on installment basis. Standard mean comes to 47.08949939. Minimum Annual income is 0 and Annual income amount is 33520.27.
- Collection recovery fee is also to be considered as the recollection of loan amount. Standard mean comes to 4.951227157. Minimum Annual income is 0 and Annual income amount is 7002.19.
- Total current balance-- Standard mean comes to 139766.2475. Minimum Annual income is 0 and Annual income amount is 8000078.
- Total collected amount refers to the final state of collected loan amount repaid back to bank or loan issuer. Standard mean comes to 225.4128822. the range her is between 0 to 9152545 where 0 is the minimum value and 9152545 is the highest.
- Values of default index comes in the range of 0 to 1.

3.2 Finding Missing Values

Since this dataset is not a sensitive dataset, we decided to keep the threshold as 50%

This means, if any variable has missing values above 50% of the total values, we will simply drop the column and not consider it in our modelling phase.

To get this information, we created a data frame which consisted of the following:

a. Variable Name
b. Number of Missing Values
c. Percentage of Values Missing

Out of the 72 variables, we dropped 20 variables since they contained missing values over 50%.

Variable Name	No. of Missing Values	Percentage Missing
dti_joint	855529	99.9
annual_inc_joint	855527	99.9
verification_status_joint	855527	99.9
il_util	844360	98.6
mths_since_rcnt_il	843035	98.5
inq_last_12m	842681	98.4
open_il_24m	842681	98.4
open_il_12m	842681	98.4
open_il_6m	842681	98.4
open_acc_6m	842681	98.4
open_rv_12m	842681	98.4
open_rv_24m	842681	98.4
total_bal_il	842681	98.4
max_bal_bc	842681	98.4
all_util	842681	98.4
inq_fi	842681	98.4
total_cu_tl	842681	98.4
desc	734157	85.8
mths_since_last_record	724785	84.7
mths_since_last_major_derog	642830	75.1
mths_since_last_delinq	439812	51.4
next_pymnt_d	252971	29.6
total_rev_hi_lim	67313	7.9
tot_cur_bal	67313	7.9
tot_coll_amt	67313	7.9
emp_title	49443	5.8
emp_length	43061	5
last_pymnt_d	8862	1
revol_util	446	0.1
collections_12_mths_ex_med	56	0
last_credit_pull_d	50	0
title	33	0

4. DATA VISUALIZATION

Fig 1.1

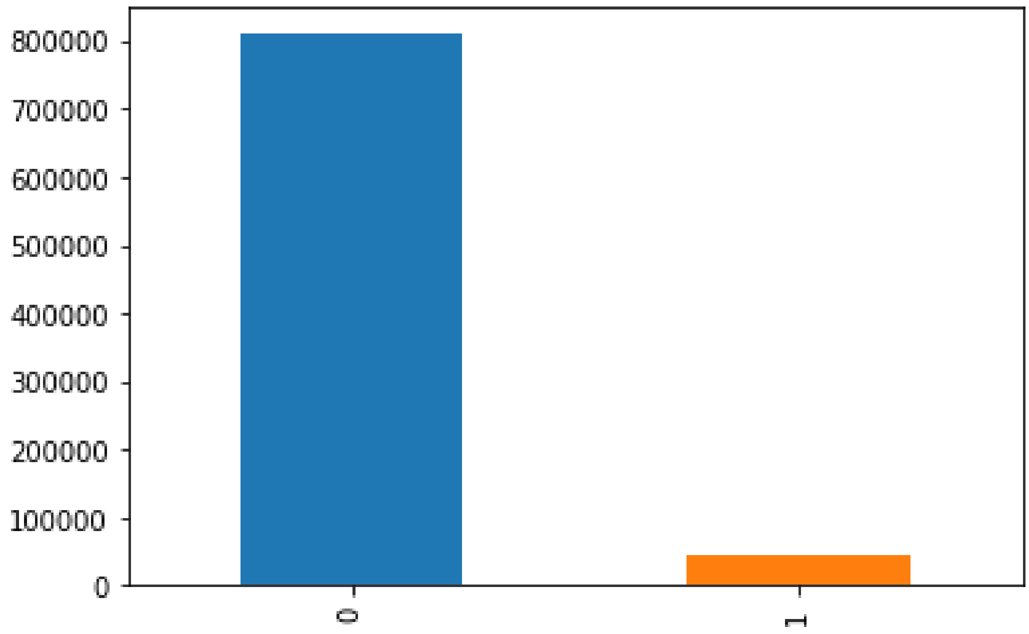


Fig 1.2

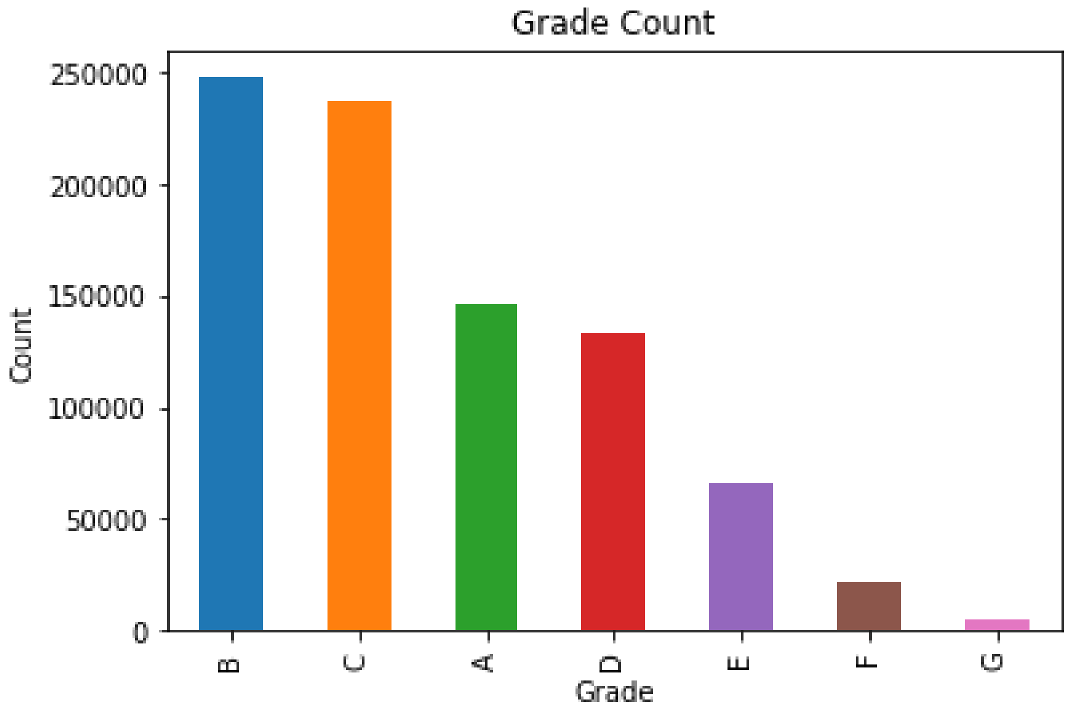


Fig 1.3

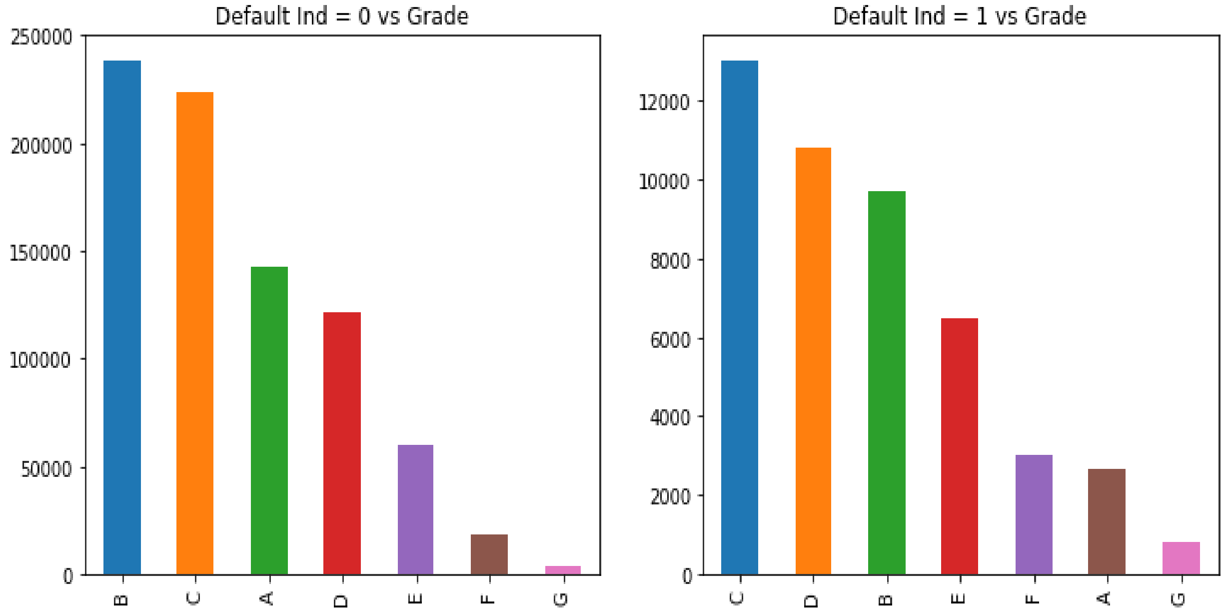
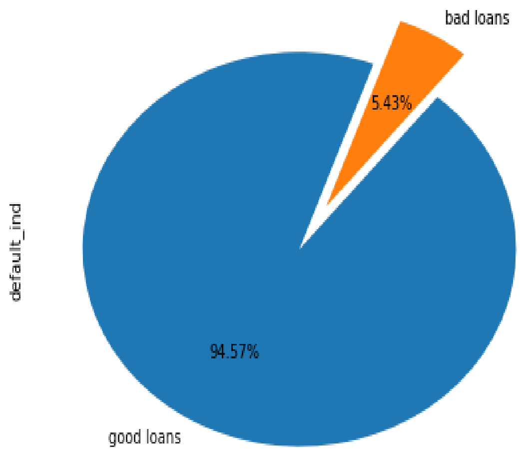


Fig 2.1(Left) and Fig 2.2(Right)



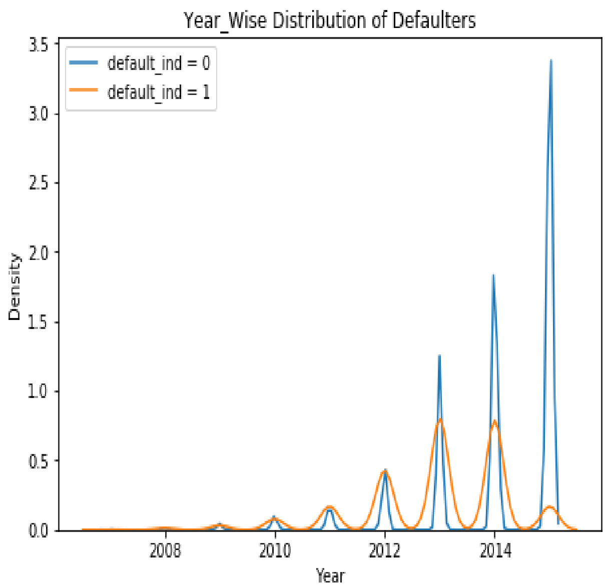


Fig 3.1

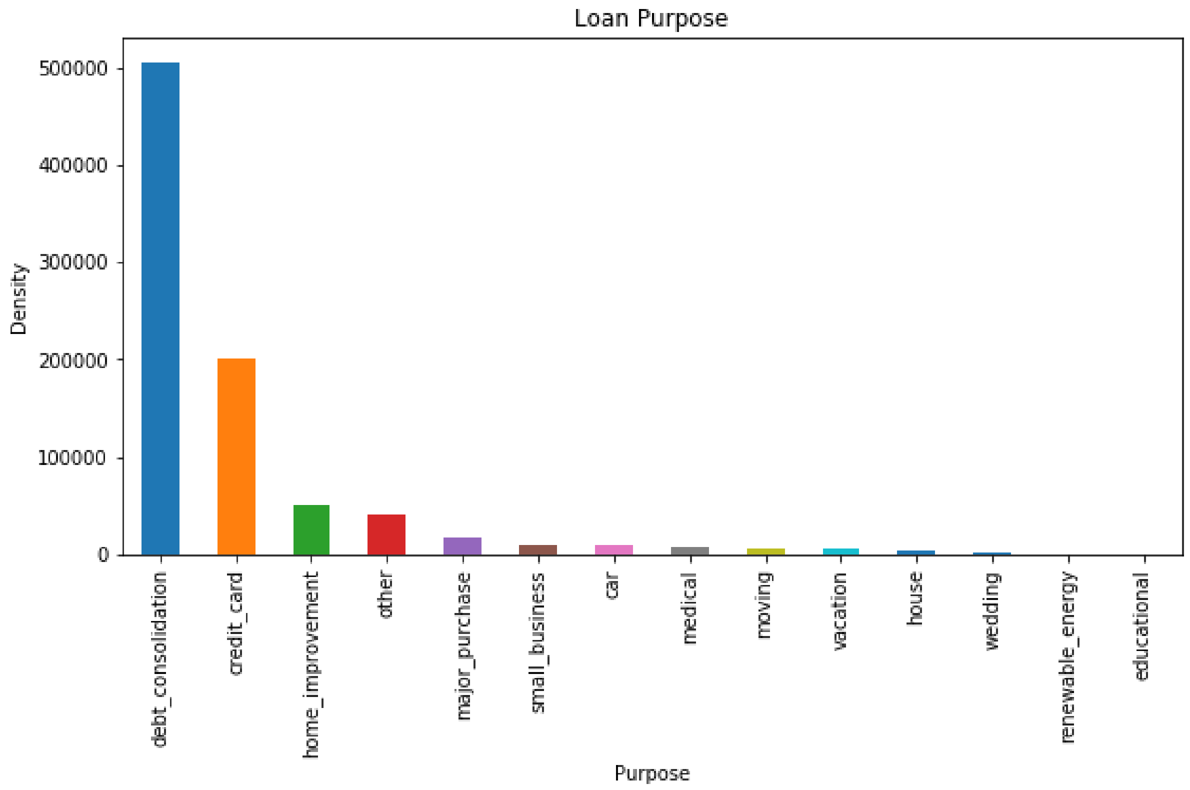

Loan Purpose
Density
500000
400000
300000
200000
100000
0
debt_consolidation
credit_card
home_improvement
other
major_purchase
small_business
car
medical
moving
vacation
house
wedding
renewable_energy
educational
Purpose

Fig 3.2

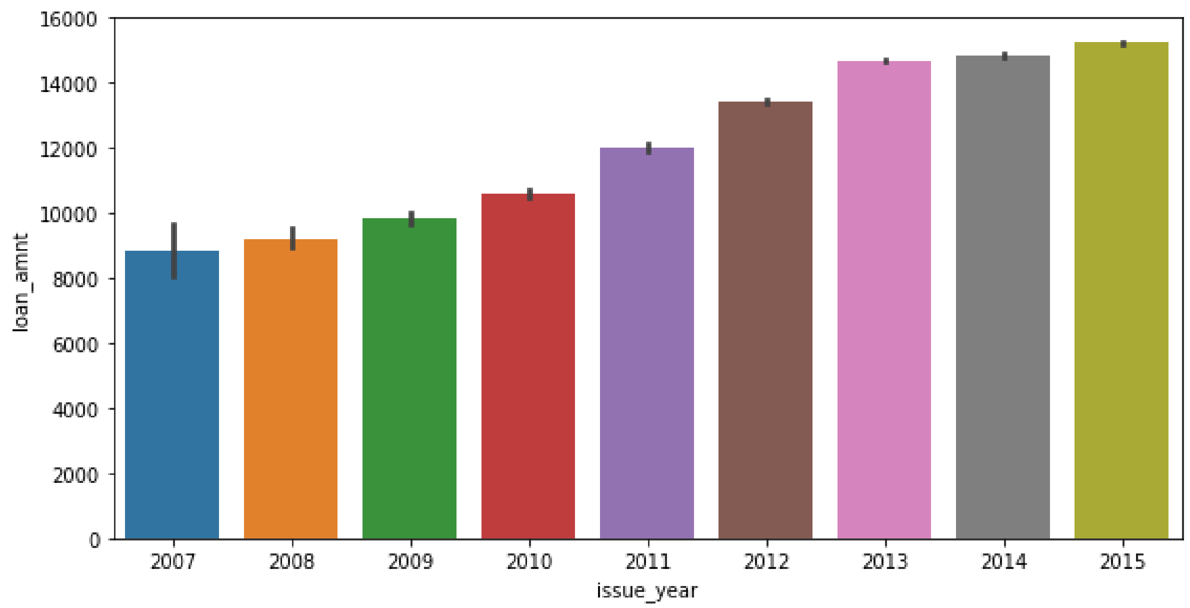

16000
14000
12000
10000
8000
6000
4000
2000
0
loan_amnt
2007
2008
2009
2010
2011
2012
2013
2014
2015
issue_year

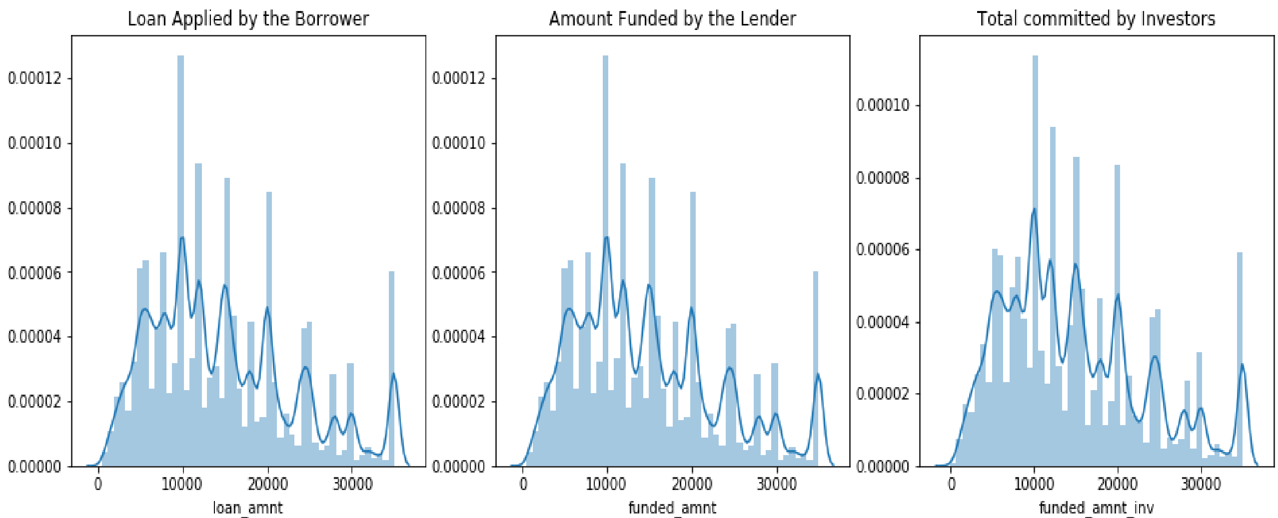
Fig 4

Purpose VS Annual Income

Purpose	
car	583,963,805
credit_card	15,039,253,894
debt_consolidation	37,367,983,264
educational	17,442,697
home_improvement	4,508,672,615
house	284,702,720
major_purchase	1,275,405,967
medical	593,144,136
moving	353,049,502
other	2,856,320,866
renewable_energy	40,492,626
small_business	875,637,909
vacation	303,988,873
wedding	158,549,104

Sum of Annual Inc broken down by Purpose. Color shows sum of Annual Inc. The marks are labeled by sum of Annual Inc.

Fig 5.1(Left), 5.2(Middle) and 5.3(Right)



5. STEPS PERFORMED

Step 1: Importing the libraries

- **Pandas** - Used for data manipulation and analysis.
- **NumPy**-Adds support to large multi-dimensional arrays and matrices along with large collection mathematical functions.
- **Matplotlib**- Plotting library for python.
- **Seaborn-** It is a data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.
- **Sci-Kit Learn-** It is a machine learning library that features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means .

Step 2: Importing the dataset.

- Dataset is imported by using 'read.csv' keyword and specifying delimiter as '\t' and index column as '0'.

Step 3: Checking shape of the dataset and null values.

- To check the shape of the dataset: loan_credit.shape.
- To find number of null values in each column: loan_credit.isnull().sum()

Step 4: Finding Threshold (as 50%) to drop columns with missing values.

- This is done to Set the threshold for the dropping of columns containing null values above 50% of the total observations
- Syntax: half_count= round(len(loan_credit1)/2, 0)

Step 5: Dropping columns with Null Values above threshold

- Dropping the Variables that contain Null Values above the set threshold value
- Syntax: loan_credit1 = loan_credit1.dropna(thresh = half_count, axis = 1)

Step 6: Getting Summary

- Summary of the data set is printed by using 'describe' keyword

Step 7: Making a list of columns to drop and then permanently drop them.

Step 8: Convert "employee_length" to standard numerical values (Integers)

- Replacing the Values in "emp_length" from categorical to numerical. This method is also known as Manual Label Encoding
- Syntax: loan_credit1['emp_length'] = loan_credit1['emp_length'].replace({'2 years': 2, '1 year': 1, '4 years': 4, '8 years': 8,'10+ years': 10, '9 years': 9.0, '< 1 year': 0, '6 years': 6, '7 years': 7, '3 years': 3, '5 years': 5})

Step 9: Importing missing values with mean and mode

- For Continuous Variables we use the mean to impute missing values
- For Categorical Variables we use the mode to impute missing values

Step 10: Manual encoding for Term, Verification status, Home Ownership.

- This is done to convert string lengths to numeric values.

Step 11: Converting 'issue_d' to datetime format for the next step.

Step 12: Splitting Dataset into Train and Test data.

- We split the data into training and testing datasets
- The parameter we used to split is:
  - For training dataset: <= May 2015
  - For testing dataset: > May 2015

Step 13:

- Drop "issue_d" from the dataset
- Perform Label Encoding on the categorical variables

Step 14: Creating X(Independent Variables) and Y(Dependent Variable) Arrays

- Creating X and Y array where X contains all variables except dependent variable.
- Y array will have only the dependent variable.

Step 15: Standardisation using "Standard Scaler"

- We are using standard scalar to give us a range between -3 and 3.

Step 16: Logistic Regression

- We used Logistic Regression for training the model with the training dataset
- Further we used this model to predict outcomes of the test dataset

Step 17: Metrics

- We used the following Metrics to score out model:
  a. Confusion Matrix
  b. Accuracy
  c. Classification Report

Step 18: Changing threshold of CFM

- After calculating the parameters in the aforementioned step, we found the Type I error and the Type II Error.
- We tried to reduce both the errors to a minimum by changing the threshold of the confusion matrix to its optimum value which in this case was 0.62.
- By default the value is 0.5

Step 19: Sensitivity, Specificity and ROC Curve

- Next, we found the following values:
  a. TPR (True Positive Rate) aka Sensitivity
  b. FPR (False Positive Rate) aka Specificity
  c. AUC (Area Under Curve)
- Using the TPR and FPR we plotted the ROC Curve

5.1: OTHER MODELS

Other models that we used for training and testing are as follows:

1. Naïve Bayes
2. K-Fold Cross Validation
3. Random Forest
4. K-Means
5. Extra Trees Classifier
6. AdaBoost
7. Gradient Boosting

6.RESULTS

The following pages will contain the results of the various models that were used in this project:

1. Logistic Regression

```
==== LOGISTIC REGRESSION ====
Confusion Matrix:

[[256421    259]
 [    70    241]]


Classification Report:

             precision    recall  f1-score   support

        0.0       1.00      1.00      1.00    256680
        1.0       0.48      0.77      0.59       311

avg / total       1.00      1.00      1.00    256991


Accuracy of the Model:  0.9987197995260534
```

2. Logistic Regression with Adjusted Threshold of Confusion Matrix

```
ADJUSTED THRESHOLD: 0.62 FOR LOGISTIC REGRESSION

Confusion Matrix:

[[256501    179]
 [    70    241]]


Classification Report:

             precision    recall  f1-score   support

        0.0       1.00      1.00      1.00    256680
        1.0       0.57      0.77      0.66       311

avg / total       1.00      1.00      1.00    256991


Accuracy of the Model:  0.9990310944741255
```

3. Naïve Bayes

```
==== NAIVE BAYES ====
Confusion Matrix:

[[256629     51]
 [   306      5]]


Classification Report:

             precision    recall  f1-score   support

        0.0       1.00      1.00      1.00    256680
        1.0       0.09      0.02      0.03       311

avg / total       1.00      1.00      1.00    256991


Accuracy of the Model:  0.9986108462942282
```

4. K-Fold Cross Validation

The mean result of 10 folds: 0.99659

The max result of the best fold: 0.9981

5. Extra Trees Classifier

```
==== EXTRA TREES CLASSIFIER ====
Confusion Matrix:

[[151638 105042]
 [     8    303]]


Classification Report:

             precision    recall  f1-score   support

        0.0       1.00      0.59      0.74    256680
        1.0       0.00      0.97      0.01       311

avg / total       1.00      0.59      0.74    256991


Accuracy of the Model:  0.5912308213128086
```

6. Random Forest

```
Confusion Matrix:

[[ 96617 160063]
 [     1    310]]


Classification Report:

             precision    recall  f1-score   support

        0.0       1.00      0.38      0.55    256680
        1.0       0.00      1.00      0.00       311

avg / total       1.00      0.38      0.55    256991


Accuracy of the Model:  0.37716106789731935
```

7. **AdaBoost**

```
Confusion Matrix:

[[ 96072 160608]
 [      4    307]]


Classification Report:

             precision    recall  f1-score   support

        0.0       1.00      0.37      0.54    256680
        1.0       0.00      0.99      0.00       311

avg / total       1.00      0.38      0.54    256991


Accuracy of the Model:  0.3750286975030254
```

8. **Gradient Boosting**

```
Confusion Matrix:

[[120269 136411]
 [      3    308]]


Classification Report:

             precision    recall  f1-score   support

        0.0       1.00      0.47      0.64    256680
        1.0       0.00      0.99      0.00       311

avg / total       1.00      0.47      0.64    256991


Accuracy of the Model:  0.4691876369211373
```

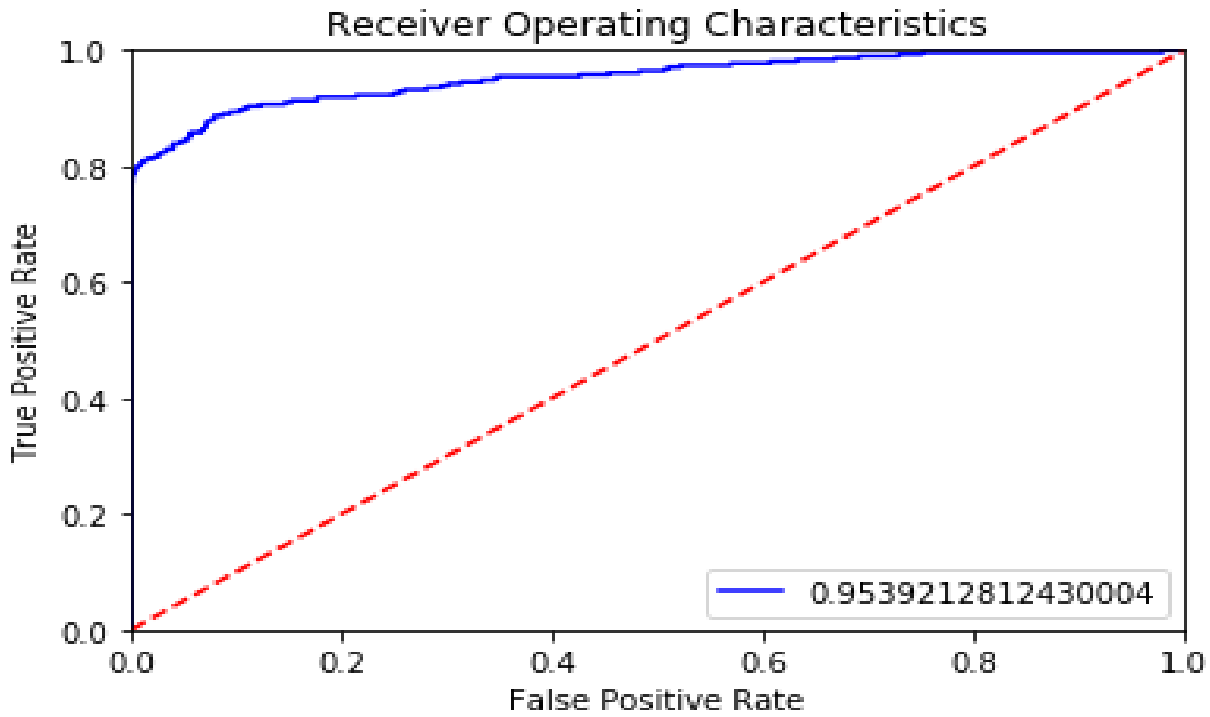
9. Decision Tree (With Entropy)

```
Confusion Matrix:

[[109172 147508]
 [     5    306]]


Classification Report:

             precision    recall  f1-score   support

        0.0       1.00      0.43      0.60    256680
        1.0       0.00      0.98      0.00       311

avg / total       1.00      0.43      0.60    256991


Accuracy of the Model:  0.4259993540629827
```

10.Decision Tree (With Gini Index)

```
Confusion Matrix:

[[100444 156236]
 [     4    307]]


Classification Report:

             precision    recall  f1-score   support

        0.0       1.00      0.39      0.56    256680
        1.0       0.00      0.99      0.00       311

avg / total       1.00      0.39      0.56    256991


Accuracy of the Model:  0.3920409664151663
```

ROC Curve for Logistic Regression with Adjusted Threshold :



Area Under Curve: 0.954

7.CONCLUSION

After comparing the results of the various models performed, we came to a conclusion that Logistic Regression with a modified threshold of 0.62 for the confusion matrix gives us the best model for the prediction of defaulters.

This conclusion is also supported by the aforementioned ROC Curve which depicts the curve elbow near to 1.0 thus increasing rh AUC value.

We found the Type I and Type II error this model to be the least out of all the models we used to predict the outcome. They are 179 and 70 respectively with the total observation count of 256991.

It is important for the lenders to make sure they are not conned and do not face any loss. To make sure the borrowers will return the money with interest, the lenders must peruse the background, financial status, credit history etc to reduce the risk of facing a loss.

This model can predict if the lenders can safely loan money to the borrower using various parameters with an accuracy of 99.9%

8.REFERENCES

1. http://budgeting.thenest.com/mean-loan-goes-underwriting-23201.html

2. http://www.investopedia.com (a great source to find meanings of BFSI terminology and jargon)

3.www.w3school.com

4. www.wikipedia.org

5. www.stackoverflow.com