Loan Status Prediction for Lending Club

Post Graduate Program in Data Science Engineering

Location: **Pune** Batch: **Nov 2019**

Submitted by:

Aakanksha Patil

Abhishek Deshmukh

Sanket Kumar

Ravi Mahato

Mentored by

Mr. Muppidi Srikar

Table of Contents

1. Abstract

Lending Club is an American peer-to-peer lending company, a marketplace for personal loans that matches borrowers who are seeking a loan with investors looking to lend money and make a return.

Each borrower fills out a comprehensive application, providing their past financial history, the reason for the loan, and more.

Approved loans are listed on the Lending Club website, where the lenders/investors browse the details of the borrowers and decide which borrower they shall fund. If the loan is fully paid off on time, the investors make profits.

But at times, loans are not completely paid off and the borrowers default on the loan. This problem shall be addressed by doing this project. The model thus developed shall help to determine if the applicant falls in the category of Non Defaulter or Defaulter by analyzing the details from the comprehensive application form and their past financial history and the details about loan demanded. This shall help the business to predict which applicant can default over the time and thus, rejecting the loan application and saving the business from loss.

2. Introduction

2.1Domain and Feature Review

Lending Club is an American peer-to-peer lending company, a marketplace for personal loans that matches borrowers who are seeking a loan with investors looking to lend money and make a return. Being into the business of lending, this company works very similar to a bank. Hence, it can be considered into banking like domain.

2.2Dataset Information.

The dataset available is fetched from the Lending Club website. It has the details of the approved loans from year 2007 to 2018. There are 22,60,668 instances and 145 attributes / features. This dataset contains information on current loans, completed loans and default loans.

2.3Problem Statement

From business point of view, it can be very helpful if one has an idea which applicant shall Default. This will help business in sanctioning the loan of right applicant and thus, enjoy profits and also help to maintain a healthy customer base. According to the problem statement, we had to build a short-term and specific model for a year. In finance sectors, business every year is not always same, it depends on market conditions, which again varies with different parameters. For example, a good customer can also default in cases when his/her employment is at stake or interest rate is higher. The data with us had 13lakhs entries from 10 years, which was good for a generalized model. But since our objective was to analyze the factors impacting Loan Status in a fiscal year, we picked the recent data which was 2018 – the first half (Jan to June) as train and July as test data. Our model is not a generalized model but a business oriented, realistic and specific model - meaning needs to be calibrated and updated with time and business. Keeping in mind this business objective, the Machine Learning model developed shall help to determine if the applicant falls in the category of Non Defaulter or Defaulter by analyzing the details from the comprehensive application form and their past financial history and the details about loan demanded. This shall help the business to predict which applicant can default over the time and thus, rejecting the loan application and saving the business from loss.

3. Data Preparation

3.1 Target Variable

The main goal is to predict who will default on the loan, a column which reflects this information is treated as the target variable. The 'loan status' column is one such column which speaks about different status of the sanctioned loan. Hence, this column shall serve as target variable.

The sub categories of this column are:

a. Fully paid: Loan has been fully paid off.
b. Charged Off: Loan for which there is no longer a reasonable expectation of further payments.
c. Does not meet the credit policy. Status: Fully Paid: While the loan was paid off, the loan application today would no longer meet the credit policy and wouldn't be approved on to the marketplace.
d. Does not meet the credit policy. Status: Charged Off: While the loan was charged off, the loan application today would no longer meet the credit policy and wouldn't be approved on to the marketplace.
e. Current: Loan is up to date on current payments
f. In Grace Period: The loan is past due but still in the grace period of 15 days.
g. Late (31-120 days): Loan hasn't been paid in 31 to 120 days (late on the current payment).
h. Late (16-30 days): Loan hasn't been paid in 16 to 30 days (late on the current payment).
i. Default: Loan is defaulted on and no payment has been made for more than 121 days.

Since the ML model should learn from the past loans, from above sub categories only Charged Off and Fully Paid describe the final outcome of the loan. The other categories describe the ongoing loans (Current, Grace Period, Late, etc.) and thus, there is no point in including those categories. Thus, it can be considered as a binary

classification problem. Thus, charged off corresponds to Defaulters and Fully Paid corresponds to Non Defaulters.

The available dataset thus, is filtered with respect to the target variables. The dataset now contains 13 lakh rows and 145 columns. Moving ahead, keeping in mind the available computational limitations and According to the problem statement, we had to build a short-term and specific model for a year. In finance sectors, business every year is not always same, it depends on market conditions, which again varies with different parameters. For example, a good customer can also default in cases when his/her employment is at stake or interest rate is higher. Our model is not a generalized model but a business oriented, realistic and specific model - meaning needs to be calibrated and updated with time and business. The data with us had 13lakhs entries from 10 years, which was good for a generalized model. But since our objective was to analyze the factors impacting Loan Status in a fiscal year, we picked the recent data which was 2018 – the first half, Jan to May as train and June as test data.

Thus, the dataset size drops down to 35648 row, 144 attributes and 1 target. The train data set has 31345 rows, whereas test dataset has 4303 rows.

3.2 Redundant Feature Elimination

The dataset captures all the features right from the point of loan application, loan sanction and completion of payment. The ML model needs to predict using the details in the loan application and previous financial history and not using any details pertaining to the features captured after the loan is sanctioned. The dataset includes such 37 features which are dropped. The total data size thus is now 35648 rows and 107 features.

Further, features having only one unique value and features conveying similar information are dropped. Also, column like zip code which do not convey any information are dropped. Thus, further the total data size is 35648 rows and 101 features.

Feature	Explanation
acc_now_delinq	The number of accounts on which the borrower is now delinquent.
acc_open_past_24mths	Number of trades opened in past 24 months.
addr_state	The state provided by the borrower in the loan application
all_util	Balance to credit limit on all trades
annual_inc	The self-reported annual income provided by the borrower during registration.
annual_inc_joint	The combined self-reported annual income provided by the co-borrowers during registration
application_type	Indicates whether the loan is an individual application or a joint application with two co-borrowers
avg_cur_bal	Average current balance of all accounts
bc_open_to_buy	Total open to buy on revolving bankcards.
bc_util	Ratio of total current balance to high credit/credit limit for all bankcard accounts.
chargeoff_within_12_mths	Number of charge-offs within 12 months
collections_12_mths_ex_med	Number of collections in 12 months excluding medical collections
delinq_2yrs	The number of 30+ days past-due incidences of delinquency in the borrower's credit file for the past 2 years
delinq_amnt	The past-due amount owed for the accounts on which the borrower is now delinquent.
dti	A ratio calculated using the borrower's total monthly debt payments on the total debt obligations, excluding mortgage and the requested LC loan, divided by the borrower's self-reported monthly income.
dti_joint	A ratio calculated using the co-borrowers' total monthly payments on the total debt obligations, excluding mortgages and the requested LC loan, divided by the co-borrowers' combined self-reported monthly income
earliest_cr_line	The month the borrower's earliest reported credit line was opened
emp_length	Employment length in years. Possible values are between 0 and 10 where 0 means less than one year and 10 means ten or more years.
grade	LC assigned loan grade
home_ownership	The home ownership status provided by the borrower during registration or obtained from the credit report. Our values are: RENT, OWN, MORTGAGE, OTHER

il_util	Ratio of total current balance to high credit/credit limit on all install acct
initial_list_status	The initial listing status of the loan. Possible values are – W, F
inq_fi	Number of personal finance inquiries
inq_last_12m	Number of credit inquiries in past 12 months
inq_last_6mths	The number of inquiries in past 6 months (excluding auto and mortgage inquiries)
installment	The monthly payment owed by the borrower if the loan originates.
int_rate	Interest Rate on the loan
issue_d	The month which the loan was funded
loan_amnt	The listed amount of the loan applied for by the borrower. If at some point in time, the credit department reduces the loan amount, then it will be reflected in this value.
max_bal_bc	Maximum current balance owed on all revolving accounts
mo_sin_old_il_acct	Months since oldest bank installment account opened
mo_sin_old_rev_tl_op	Months since oldest revolving account opened
mo_sin_rcnt_rev_tl_op	Months since most recent revolving account opened
mo_sin_rcnt_tl	Months since most recent account opened
mort_acc	Number of mortgage accounts.
mths_since_last_delinq	The number of months since the borrower's last delinquency.
mths_since_last_major_derog	Months since most recent 90-day or worse rating
mths_since_last_record	The number of months since the last public record.
mths_since_rcnt_il	Months since most recent installment accounts opened
mths_since_recent_bc	Months since most recent bankcard account opened.
mths_since_recent_bc_dlq	Months since most recent bankcard delinquency
mths_since_recent_inq	Months since most recent inquiry.
mths_since_recent_revol_delinq	Months since most recent revolving delinquency.
num_accts_ever_120_pd	Number of accounts ever 120 or more days past due
num_actv_bc_tl	Number of currently active bankcard accounts

num_actv_rev_tl	Number of currently active revolving trades
num_bc_sats	Number of satisfactory bankcard accounts
num_bc_tl	Number of bankcard accounts
num_il_tl	Number of installment accounts
num_op_rev_tl	Number of open revolving accounts
num_rev_accts	Number of revolving accounts
num_rev_tl_bal_gt_0	Number of revolving trades with balance >0
num_sats	Number of satisfactory accounts
num_tl_30dpd	Number of accounts currently 30 days past due (updated in past 2 months)
num_tl_90g_dpd_24m	Number of accounts 90 or more days past due in last 24 months
num_tl_op_past_12m	Number of accounts opened in past 12 months
open_acc	The number of open credit lines in the borrower's credit file.
open_acc_6m	Number of open trades in last 6 months
open_il_12m	Number of installment accounts opened in past 12 months
open_il_24m	Number of installment accounts opened in past 24 months
open_act_il	Number of currently active installment trades
open_rv_12m	Number of revolving trades opened in past 12 months
open_rv_24m	Number of revolving trades opened in past 24 months
pct_tl_nvr_dlq	Percent of trades never delinquent
percent_bc_gt_75	Percentage of all bankcard accounts > 75% of limit.
pub_rec	Number of derogatory public records
pub_rec_bankruptcies	Number of public record bankruptcies
purpose	A category provided by the borrower for the loan request.
revol_bal	Total credit revolving balance
revol_util	Revolving line utilization rate, or the amount of credit the borrower is using relative to all available revolving credit.
tax_liens	Number of tax liens
term	The number of payments on the loan. Values are in months and can be either 36 or 60.
tot_coll_amt	Total collection amounts ever owed
tot_cur_bal	Total current balance of all accounts
tot_hi_cred_lim	Total high credit/credit limit

total_acc	The total number of credit lines currently in the borrower's credit file
total_bal_ex_mort	Total credit balance excluding mortgage
total_bal_il	Total current balance of all installment accounts
total_bc_limit	Total bankcard high credit/credit limit
total_cu_tl	Number of finance trades
total_il_high_credit_limit	Total installment high credit/credit limit
total_rev_hi_lim	Total revolving high credit/credit limit
verification_status	Indicates if income was verified by LC, not verified, or if the income source was verified
verified_status_joint	Indicates if the co-borrowers' joint income was verified by LC, not verified, or if the income source was verified
revol_bal_joint	Sum of revolving credit balance of the co-borrowers, net of duplicate balances
sec_app_earliest_cr_line	Earliest credit line at time of application for the secondary applicant
sec_app_inq_last_6mths	Credit inquiries in the last 6 months at time of application for the secondary applicant
sec_app_mort_acc	Number of mortgage accounts at time of application for the secondary applicant
sec_app_open_acc	Number of open trades at time of application for the secondary applicant
sec_app_revol_util	Ratio of total current balance to high credit/credit limit for all revolving accounts
sec_app_open_act_il	Number of currently active installment trades at time of application for the secondary applicant
sec_app_num_rev_accts	Number of revolving accounts at time of application for the secondary applicant
sec_app_chargeoff_within_12_mths	Number of charge-offs within last 12 months at time of application for the secondary applicant
sec_app_collections_12_mths_ex_med	Number of collections within last 12 months excluding medical collections at time of application for the secondary applicant
sec_app_mths_since_last_major_derog	Months since most recent 90-day or worse rating at time of application for the secondary applicant

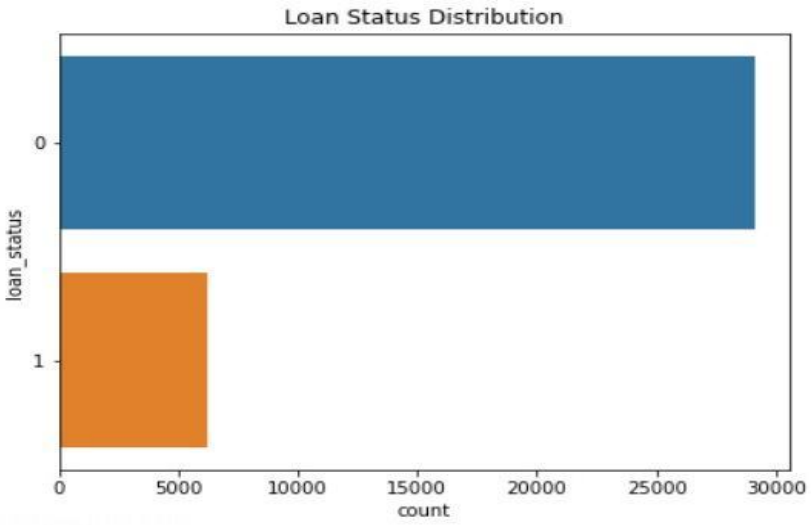
3.3 Feature Categorization:

With the above dataset, feature categorization can be done as below for better understanding of the data for further analysis

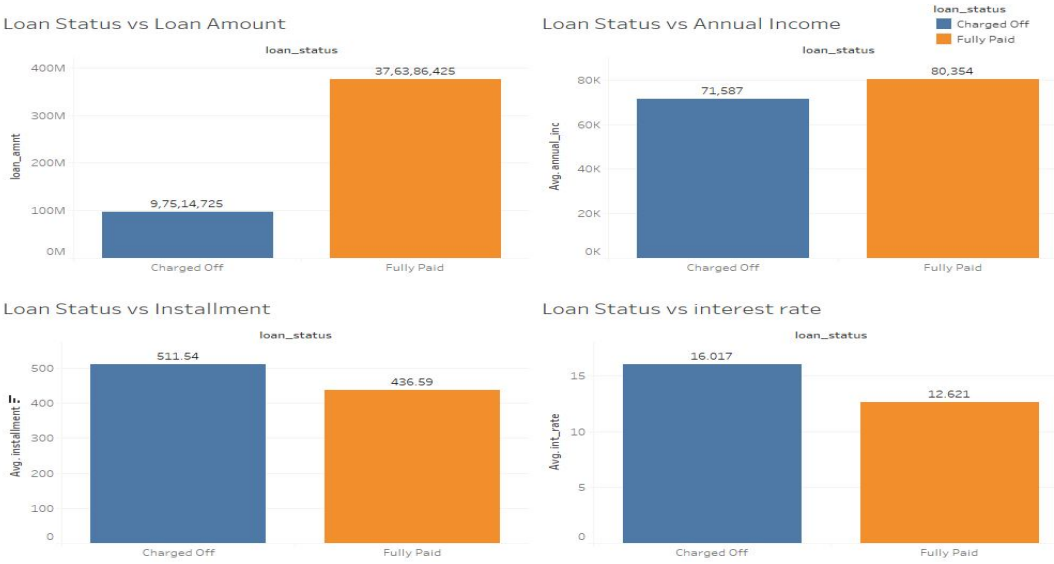
TYPE	No. of Attributes
APPLICATION TYPE DETAIL	1
GEOGRAPHICAL DETAILS	1
INDIVIDUAL PROPERTY DETAILS	1
LISTING STATUS DETAILS	1
INTEREST DETAILS	2
INDIVIDUAL EMPLOYMENT DETAILS	3
INDIVIDUAL INQUIRY DETAILS	4
LOAN DETAILS	5
JOINT ACCOUNT DETAILS	14
INDIVIDUAL - DELINQ/DEROG/RECORD DETAILS	18
INDIVIDUAL ACCOUNT DETAILS	45

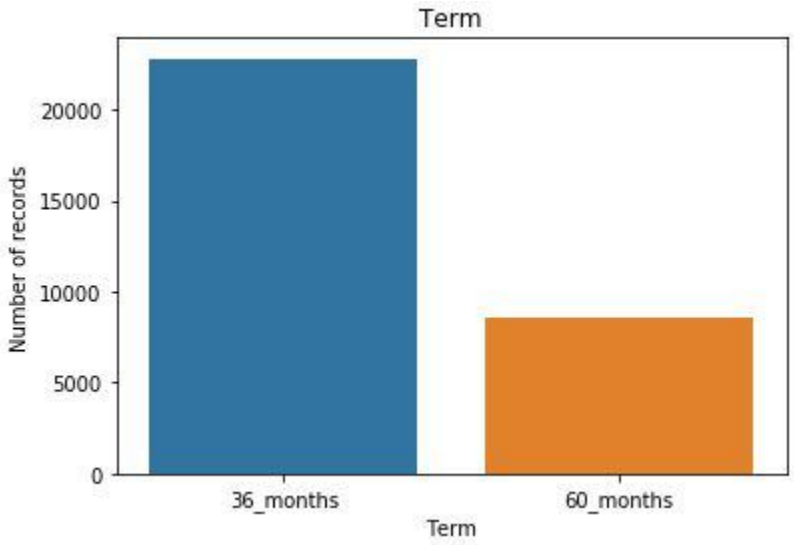
4. Exploratory Data Analysis

4.1 Univariate Analysis

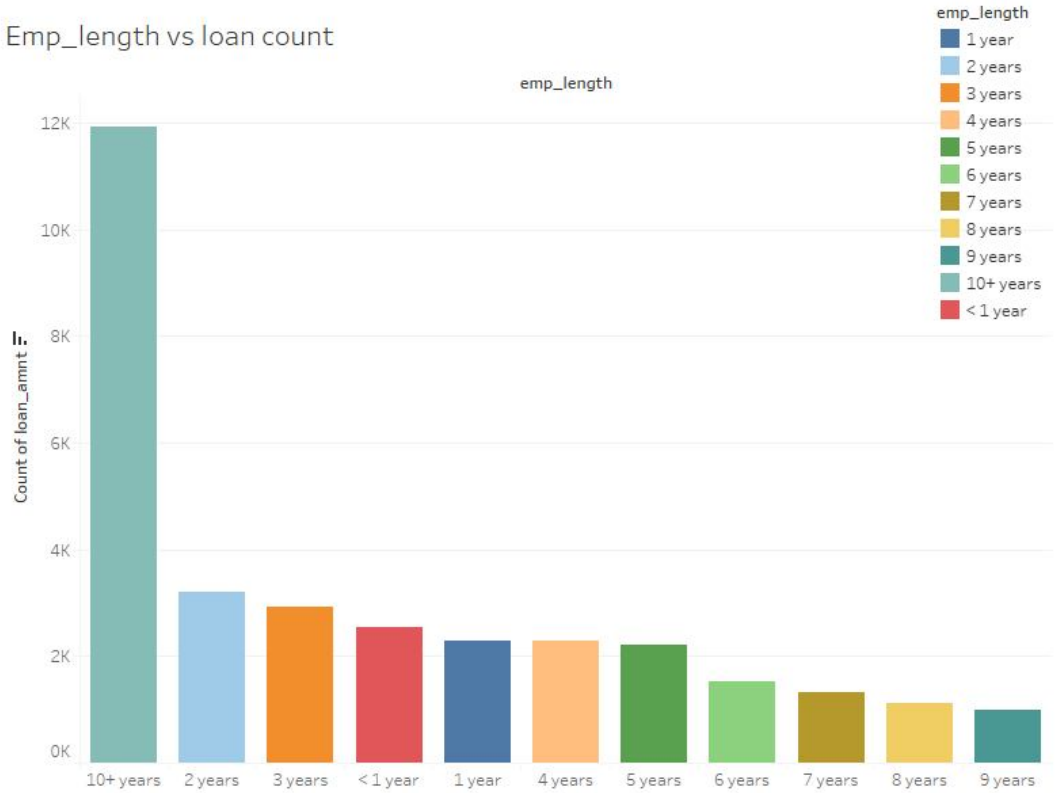


- Loan Status - Target Variable
- 0: Non-Defaulters/Fully paid | 82%
- 1: Defaulters/Charged-off | 18%
- Imbalanced Data
- Calls for Balancing Techniques



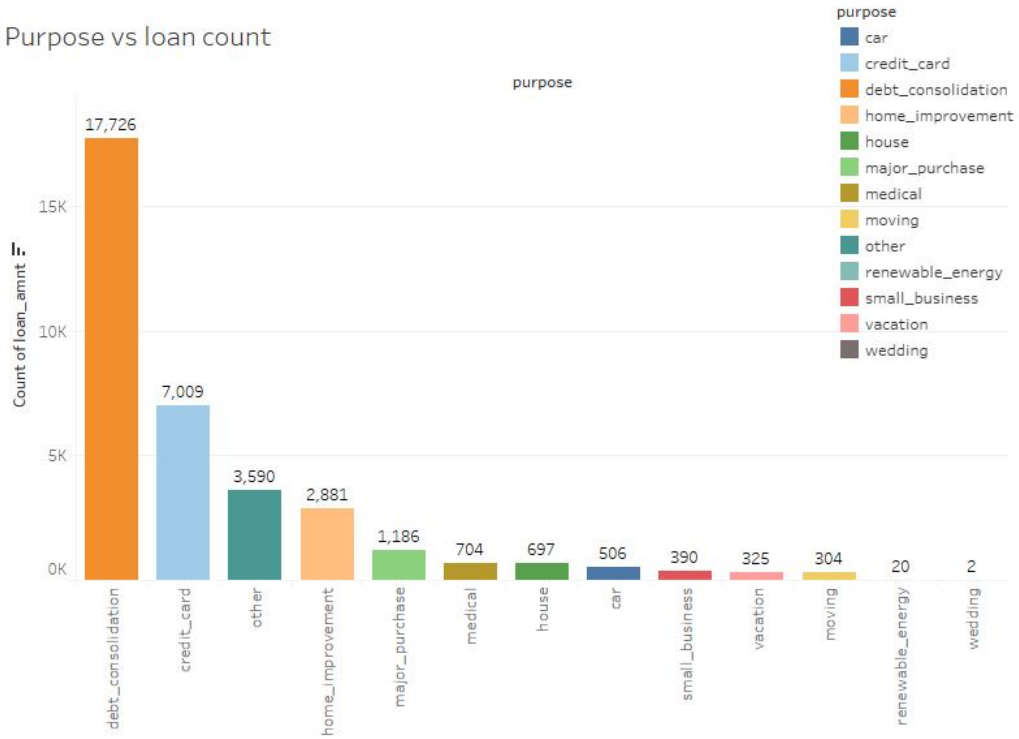


Most of the applicants opt for 3 years' loan period (73%) which looks a good period for repayment and others (27%) opt for 5 years' loan period

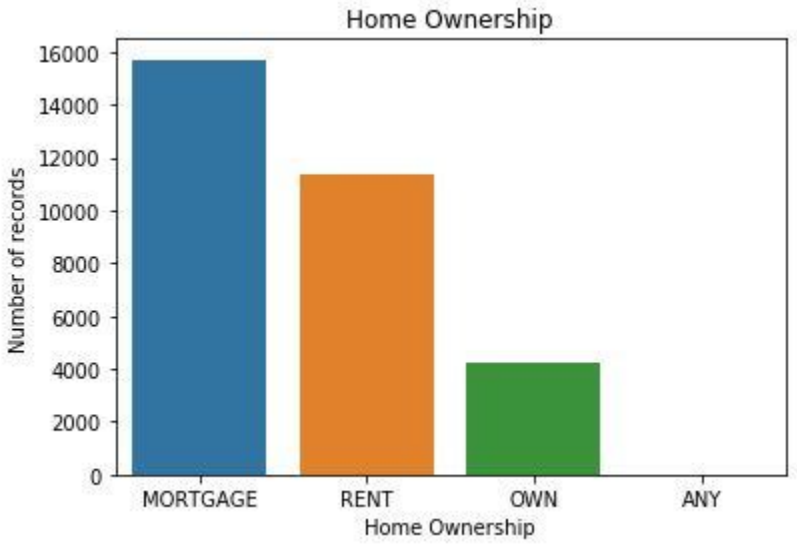


In the above chart we can see that a significant number of our customers have been employed for 10 or more years.

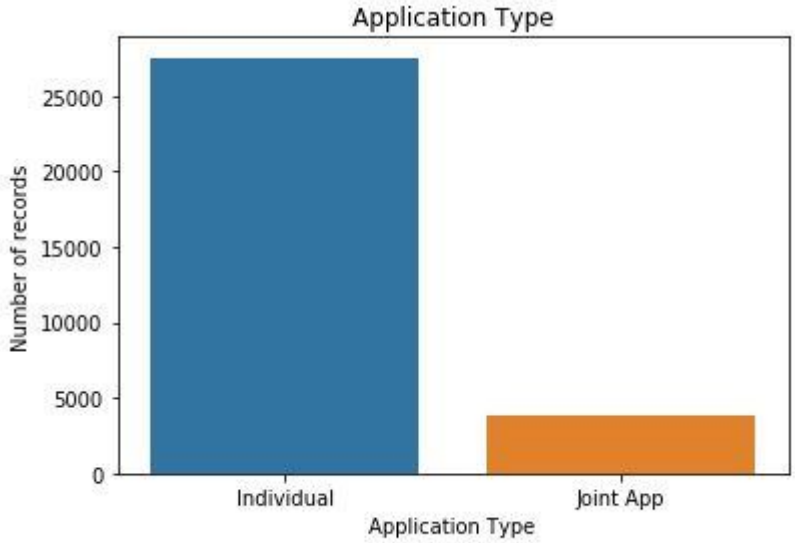
- The majority of the applicants (36%) have been employed for more than 10 years. These applicants can be the people who sincerely repay their complete loan amount and have less chances of defaulting.
- We surprisingly see that people who have been working for more than 5 years but less than 10 years are the ones who haven't applied for the loans.



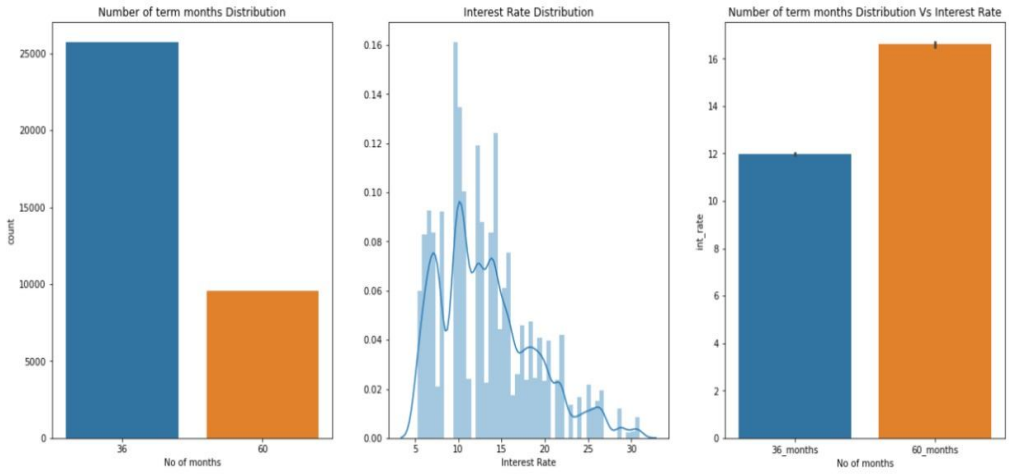
Most of the loans borrowed for debt_consolidation and credit_card billings. Lending club usually charge lower interest rates compared with money provided by traditional banks. So most of the consumers choose to consolidate debt to enjoy lower borrowing costs.



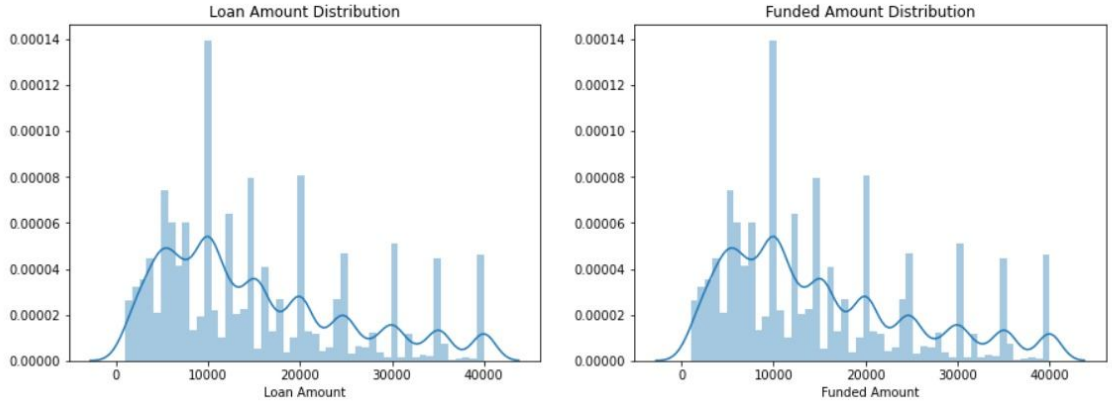
50% of loan applicants have already mortgaged their houses against some or the other loans. This makes them strong contender for not replaying the loan back and hence Defaulters



87% of the applicants are individual applicant and only 12% are joint applicants. We can assume that joint applicants are at a lower risk of defaulting as if one applicant is unable to repay, other can step in to clear the loan

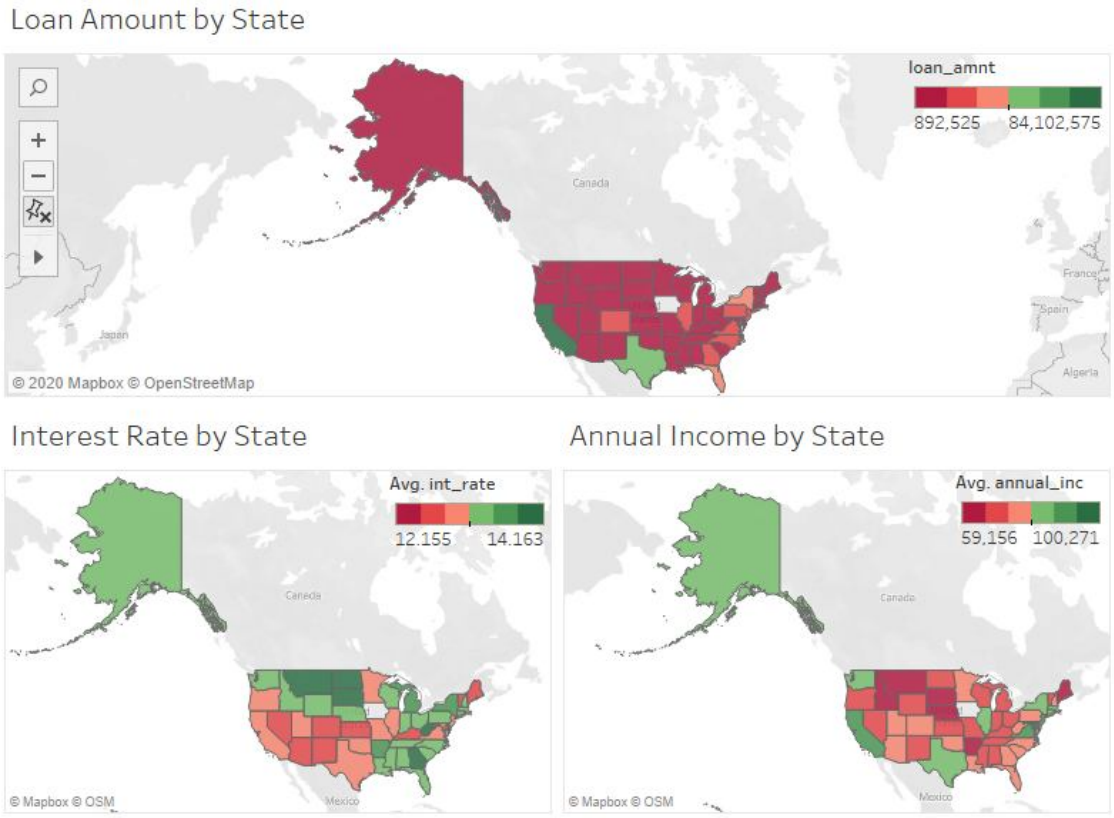


- Most of the loans are borrowed for the period of 36 months
- Maximum number loans were borrowed for 10% interest rate, the data is skewed to right
- It can be easily summarized, more the number of months, higher the interest rate

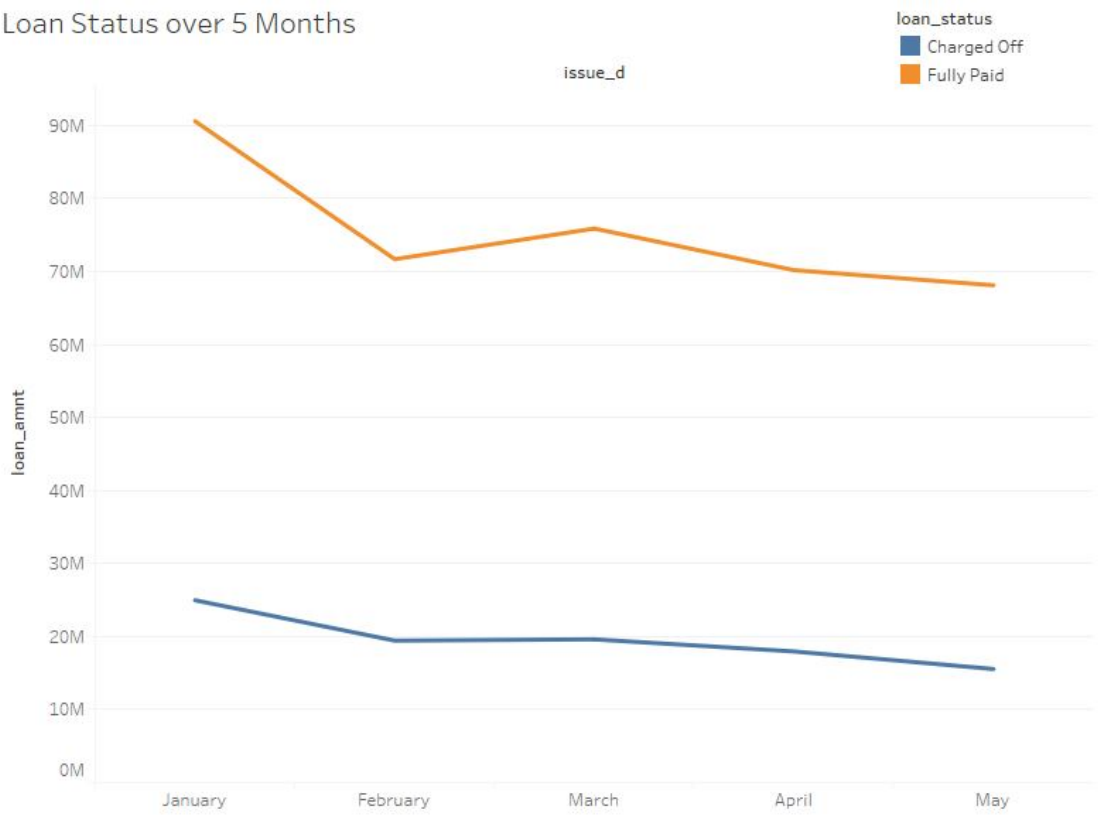


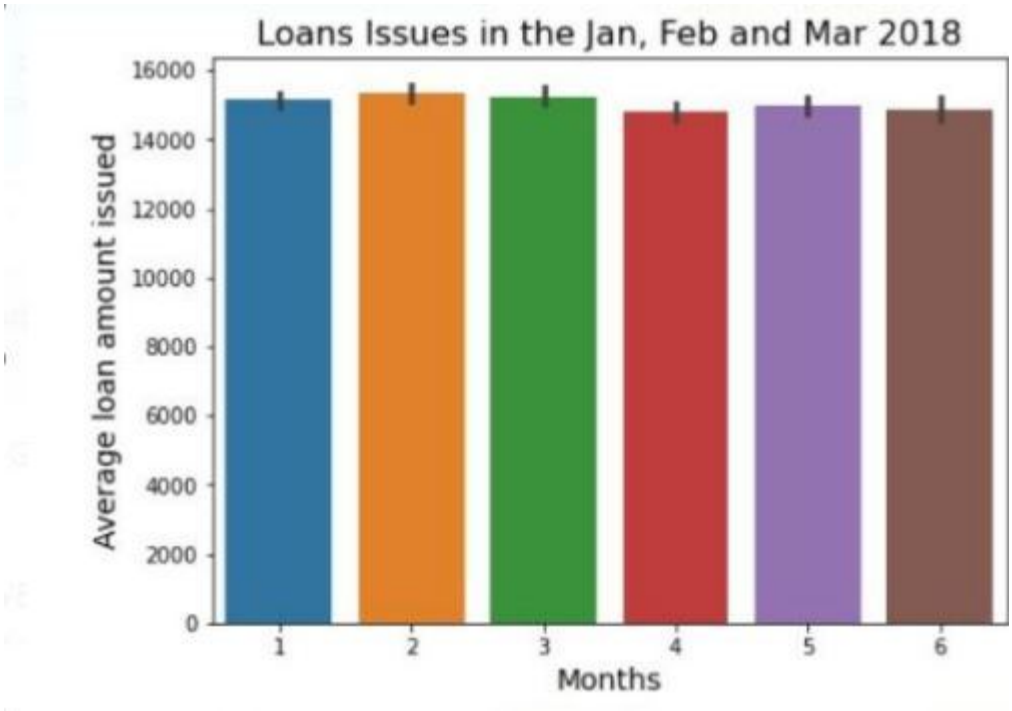
- It looks like Loan Amount and Funded Amount follows the same distribution.
- Most of the loan comes under amount of 7000 USD to 15000 USD and most of the borrowers applied for 10000 USD
- The data is skewed on to the right side

4.2 Bivariate Analysis

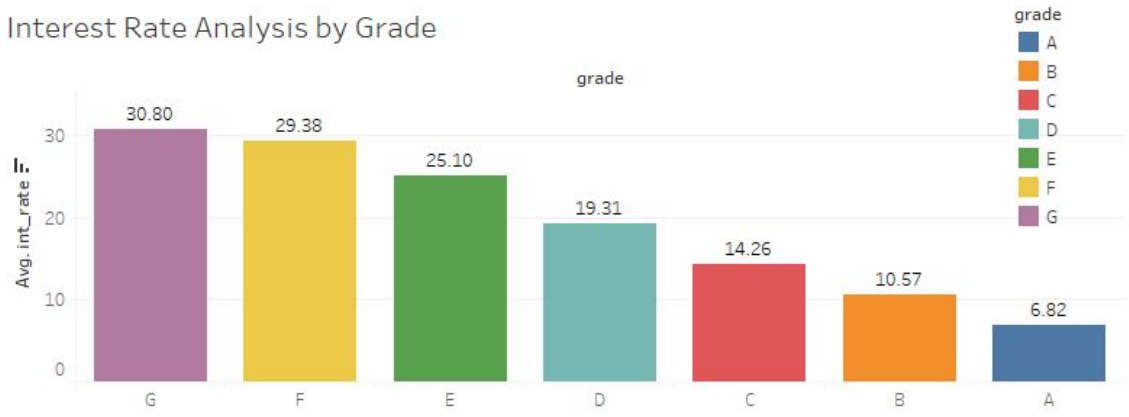


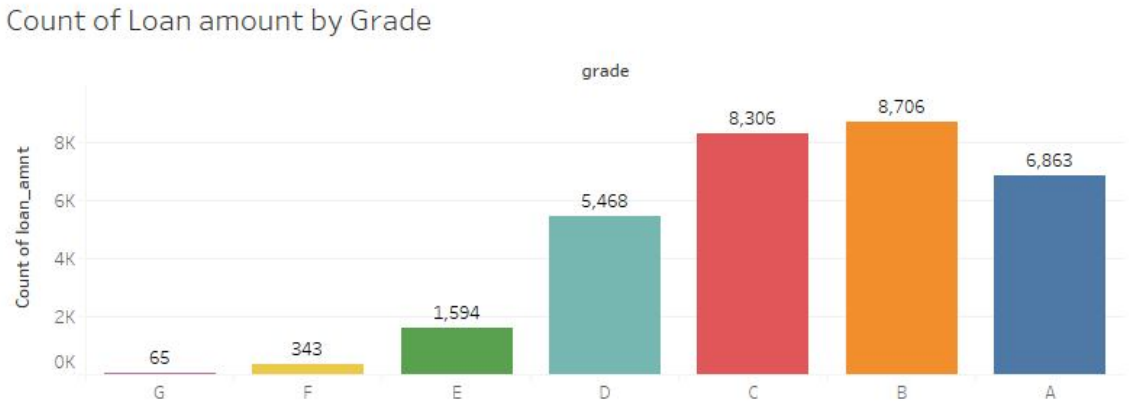
From a geographical perspective, sum of loan amount, avg. annual income and average interest rate differ by state. California, Texas, New York, Florida have the highest loan amount by state. Majority of the applicants (36%) are from south USA. It so happens that most of the working class of USA lives in South and West of USA and also many companies are in California, Texas, etc. Hence the high number of applications. Similarly, as we can see the annual income by state plot, this states have the highest average annual income. but from the interest rate by state geographical plot we conclude that the state has highest loan amount have lowest average interest rate.


Loan Status over 5 Months
loan_status
Charged Off
Fully Paid
issue_d
loan_amnt
90M
80M
70M
60M
50M
40M
30M
20M
10M
0M
January
February
March
April
May

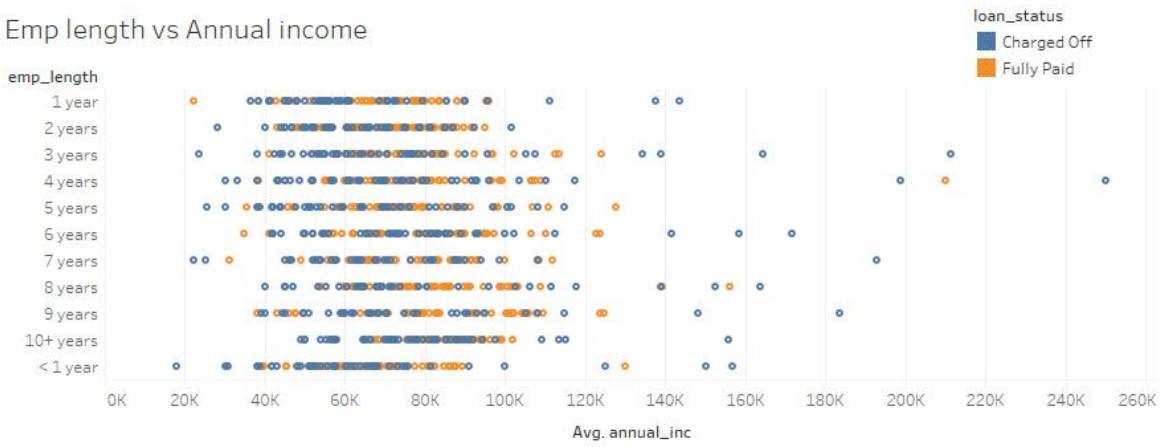

Loans Issues in the Jan, Feb and Mar 2018
Average loan amount issued
16000
14000
12000
10000
8000
6000
4000
2000
0
1
2
3
4
5
6
Months

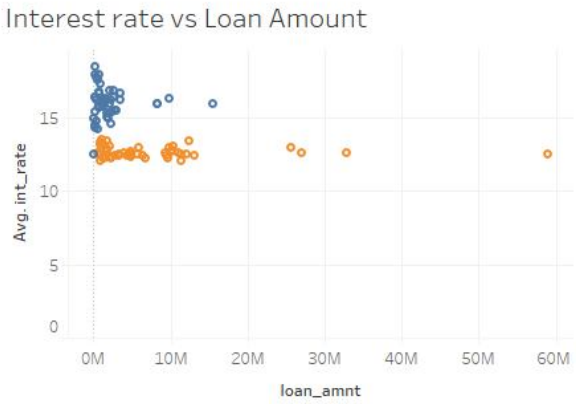
- The very important conclusion from this plot is, higher the loan amount, more the chances of being default.
- As we can see the time series plot the loan amount decreases from Jan to may
- Jan 2018 has highest sum of fully paid loan amount and in the month may 2018 has lesser sum of paid loan amount.
- Average amount of loan issued in the month of Feb are higher.
- Average amount of loan issued in the month of April are lower than other months

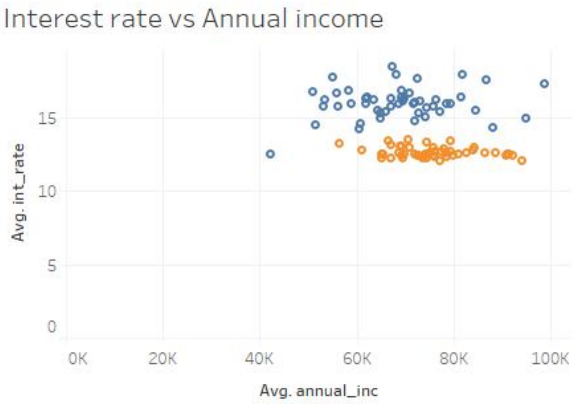




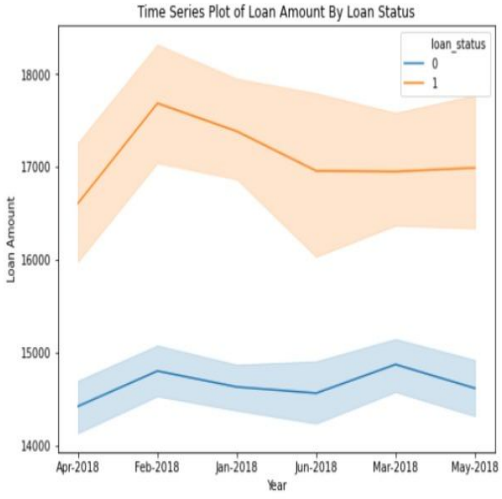
- The lower grade, the higher loan amount loan issued
- Most of the loans borrowed were from Grade B and Grade C Type
- As the loan interest rate increases for grade type, the loans borrowed from Grade Type decreases.
- So from the plot we conclude that the grade G and F type has highest interest rate so the count of loan has lowest among other grade.

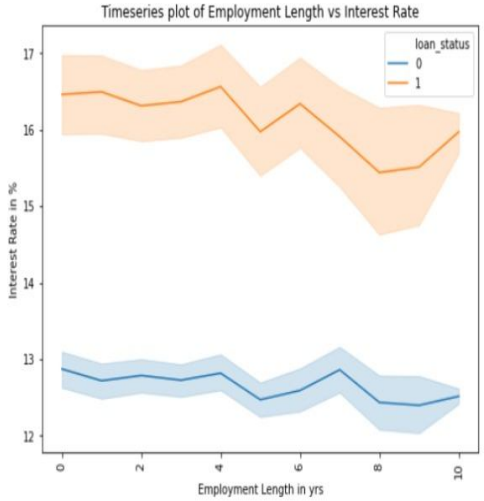




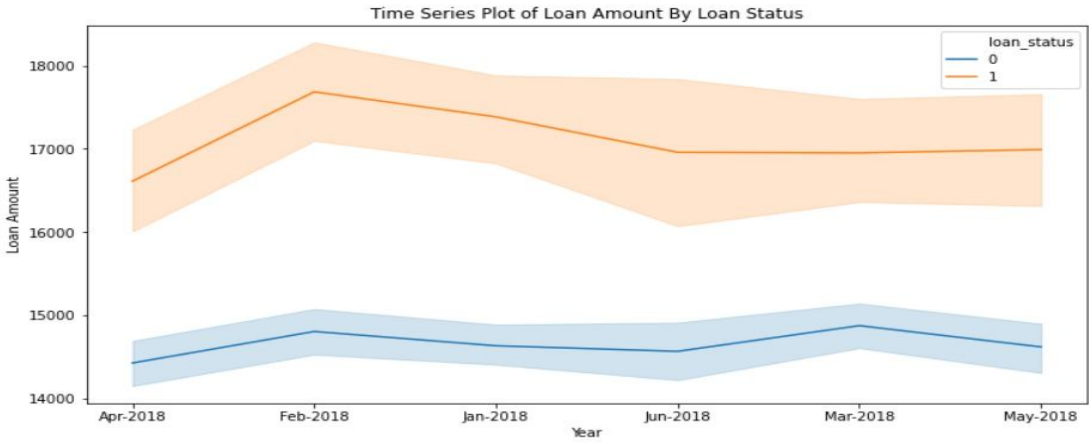


- As we can see straight lines on the plot, there is no relation between “int_rate", "annual_inc"
- As we can see straight line patterns between "loan_amnt", "int_rate" plot, there is no relation between this mentioned features.
- similarly, as we can see straight line patterns between "annual_inc", "emp_length” plot, there is no relation between this mentioned features.



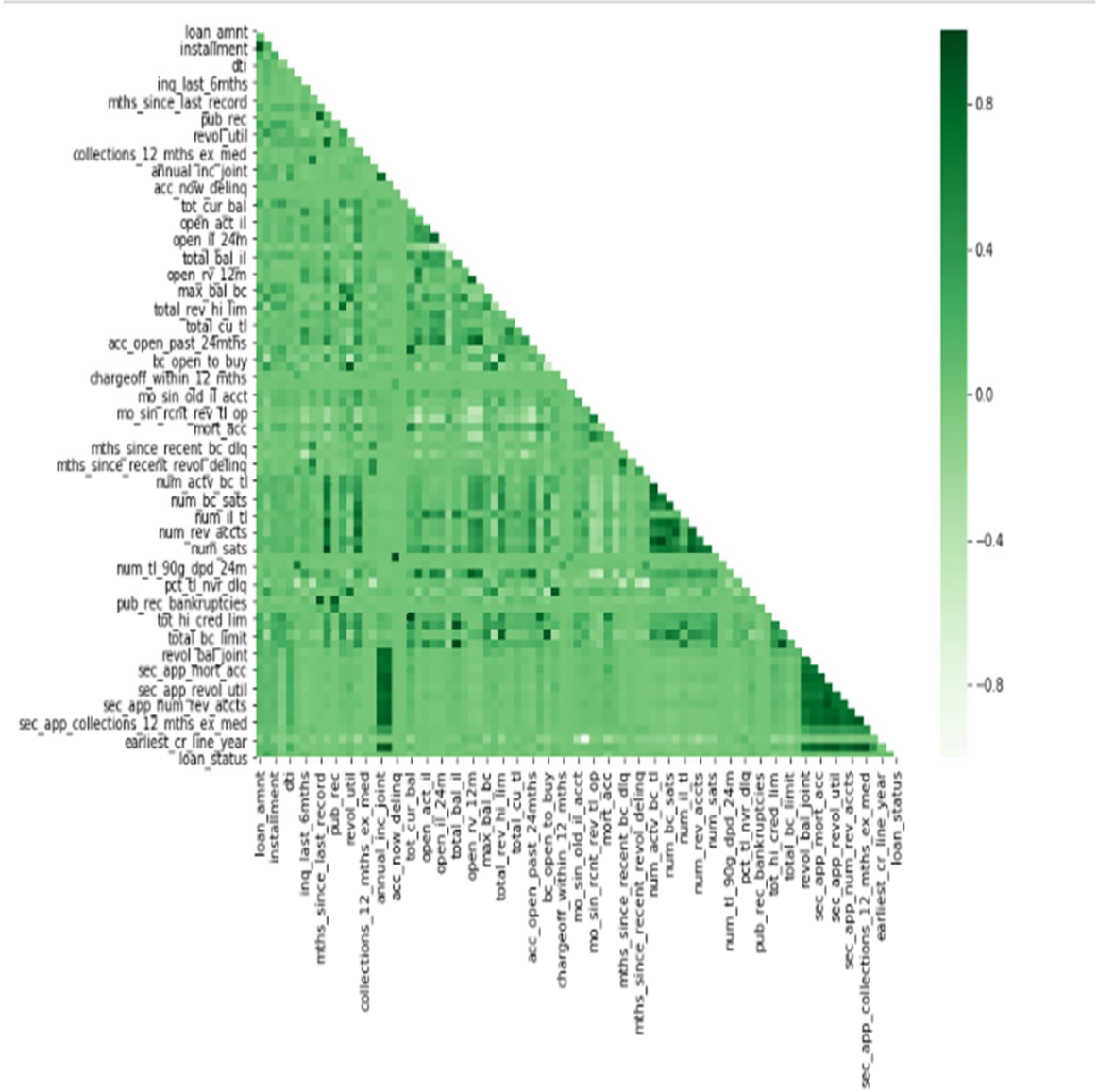


- The very important conclusion from this plot is, higher the loan amount, more the chances of being default.
- Most of the charged off loans borrowed were more than 16000 USD
- June 2018 has less number of fully paid loan count
- For employee length of 0 - 4 years, interest rate of default loan is around 16-16.5% and for non-default loans is around 13%
- While as the employee length years' increases, interest rate also goes down for both loan statuses



- The very important conclusion from this plot is, higher the loan amount, more the chances of being default.
- Most of the charged off loans borrowed were more than 16000 USD
- June 2018 has less number of fully paid loan count

4.3 Multivariate Analysis



1. High correlations are observed between the features pertaining to the joint accounts.
2. High correlations among the number of total accounts, number of active accounts, number of revolving accounts and installment accounts. These features explain the number of various types of accounts the applicant has.
3. High correlations in the credit limits and balance details. This is obvious as balance is the amount owed and credit is the maximum available limit that can be borrowed. As balance increases, credit decreases.
4. High correlation among the features related to number of delinquencies/derogations/records and the time span when they have occurred.
5. The features are not very highly correlated with the target though.

Few inferences for univariate numerical cols:

1. None of the numerical features follow a normal distribution. This is a real time data and for further analysis, we will have to do transformations and scaling
2. Also from the distplots, we observe that most of the features are non-significant. But we cannot depend on the visualizations and shall perform the statistical tests to conclude on the following
3. Few notable observations of some features which we consider to be important in the initial stage:

- a. The loan amount sees a peak at the $10000 for both Defaulters and non-defaulters
- b. The mean interest rate for Non defaulters is approximately 10% and for Defaulters is 15%. Thus, we can say that high interest rate leads to possible default.
- c. The dti ratio, as it should be being low for most of the applicants except a few. There are chances that these are the default cases

- d. The revolving utilization rate mean for Non defaulters is somewhere around 30 while that for defaulters is around 50 because lower the utilization rate better for credit score
- e. Interestingly, the installment account utilization rates and all accounts until rate are almost same for both groups
- f. The curve is uniform for the bankcard utilization rates indicating that the credit limits and the balance for the applicants is almost the same
- g. Two possible groups are observed with regards to oldest accounts opened: before 100 months and after 100 months
- h. Maximum accounts are opened in the year between 1980 to 2018

5. Data Cleaning

5.1 Post EDA dropping the below columns

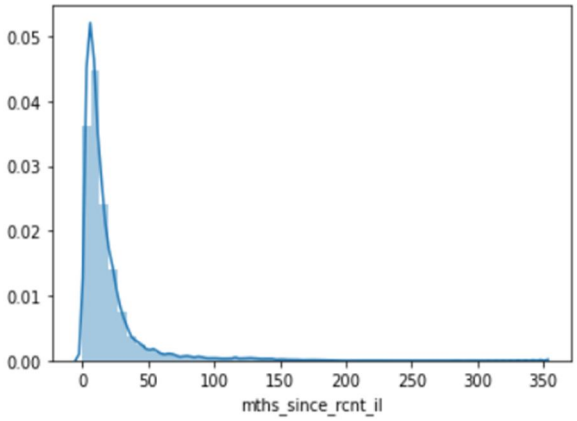
Numerical - earliest_cr_line_year, sec_app_earliest_cr_line_year

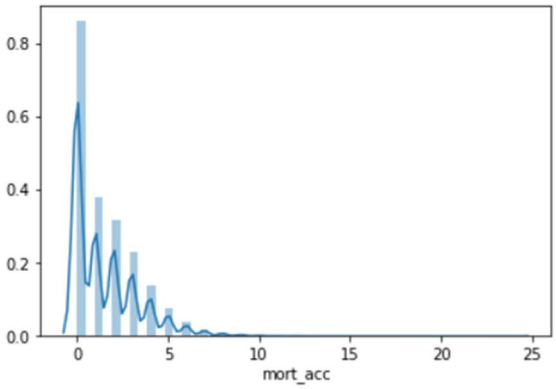
Categorical - issue_d

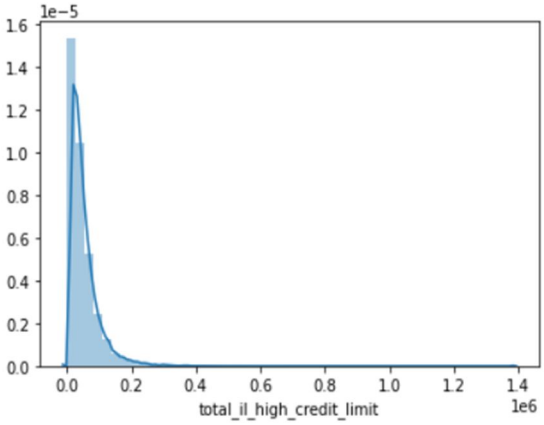
Also dropping grade as it gives details similar to interest rate.

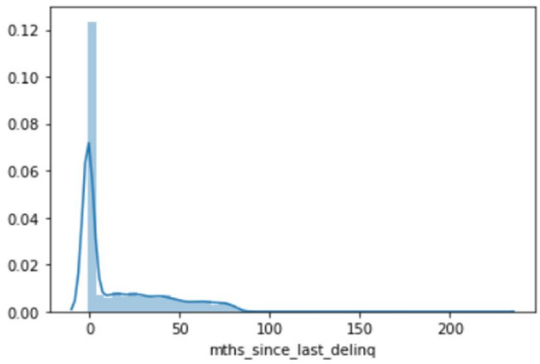
5.2 Checking the outliers and capping

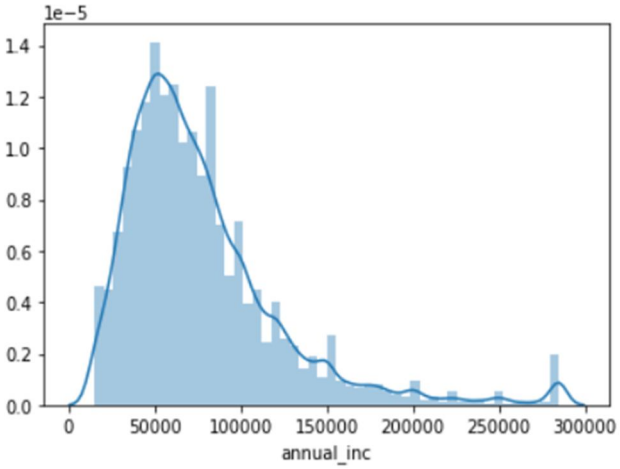
In the train data, we got some columns which are highly skewed due to some outliers and might affect the model performance. The columns having outliers are annual_inc, dti, mths_since_last_delinq, revol_bal, total_bal_il, max_bal_bc, total_rev_hi_lim, inq_last_12m, avg_cur_bal, mths_since_rcnt_il, mort_acc, total_bal_ex_mort, total_il_high_credit_limit. The below visualizations give significance for data skewness.

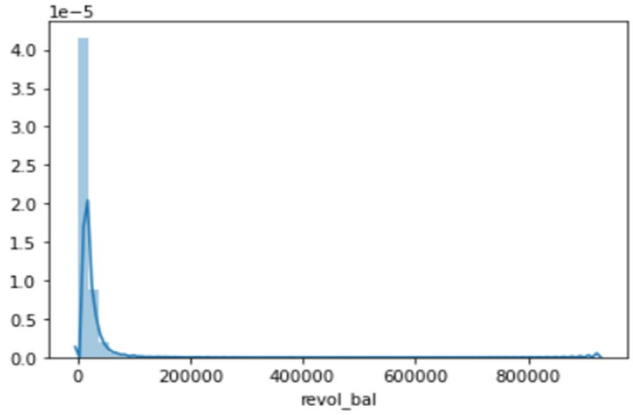

0.05
0.04
0.03
0.02
0.01
0.00
0
50
100
150
200
250
300
350
mths_since_rcnt_il

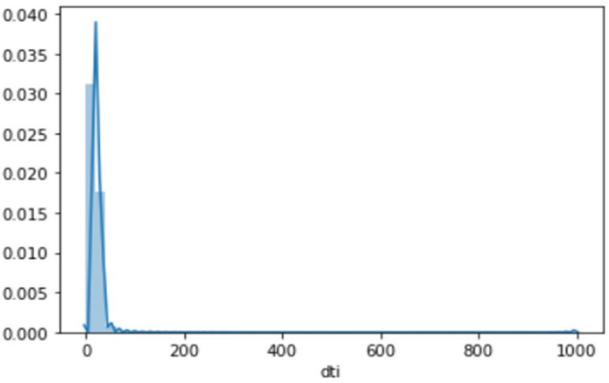

0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
mort_acc

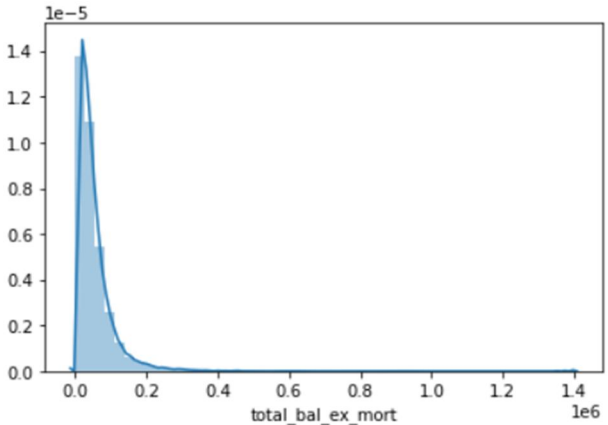

1e−5
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1e6
total_il_high_credit_limit

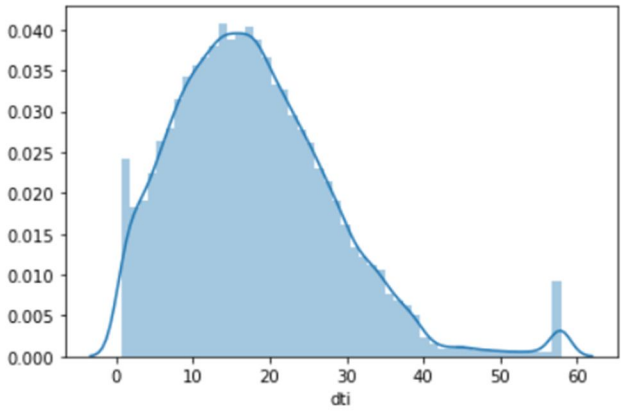

0.12
0.10
0.08
0.06
0.04
0.02
0.00
0
50
100
150
200
mths_since_last_delinq

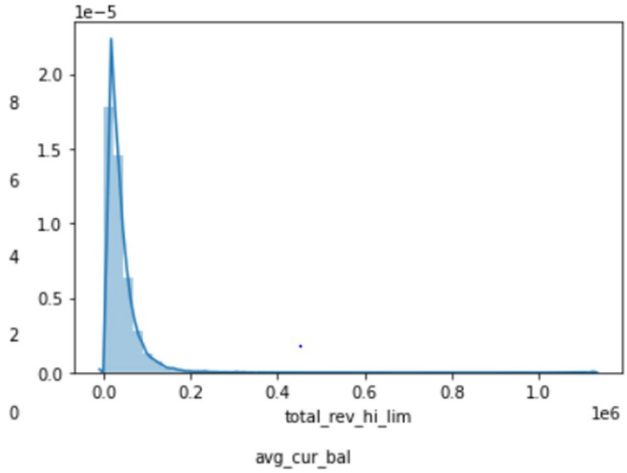

1e−5
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0
50000
100000
150000
200000
250000
300000
annual_inc

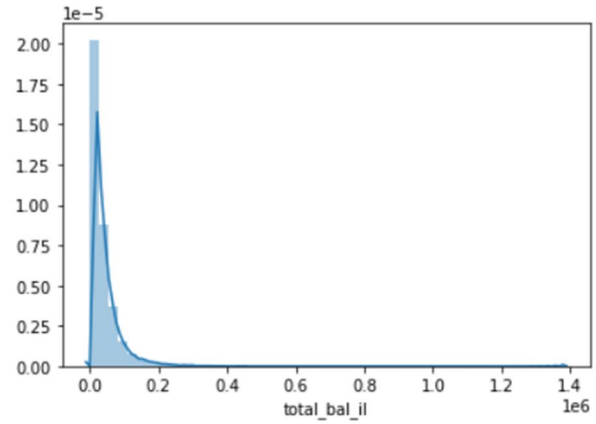

1e−5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
0
200000
400000
600000
800000
revol_bal


0.040
0.035
0.030
0.025
0.020
0.015
0.010
0.005
0.000
0
200
400
600
800
1000
dti


1e−5
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1e6
total_bal_ex_mort

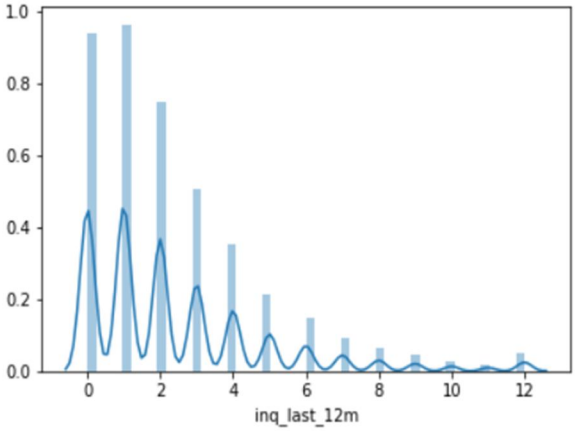


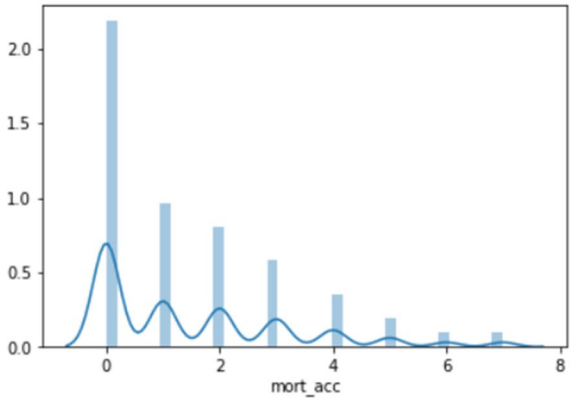


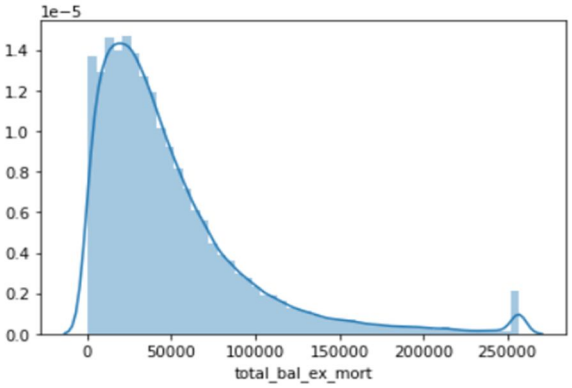


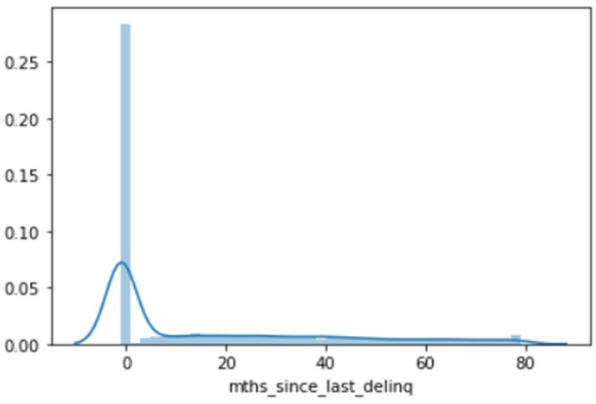
Looking at the skewness due to the outliers, we tried to go for Grubb's test but as the Grubb's test is more effective on Normally distributed data so, we decided to do capping of the outliers between the quantile of 0.01 and 0.99.

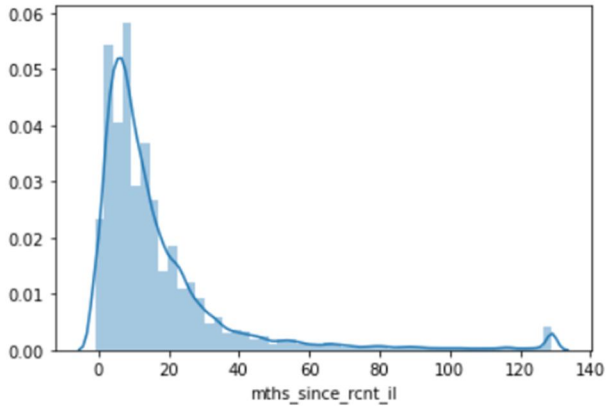
After capping the outliers, we got significant normal distribution among those columns as visualized below:

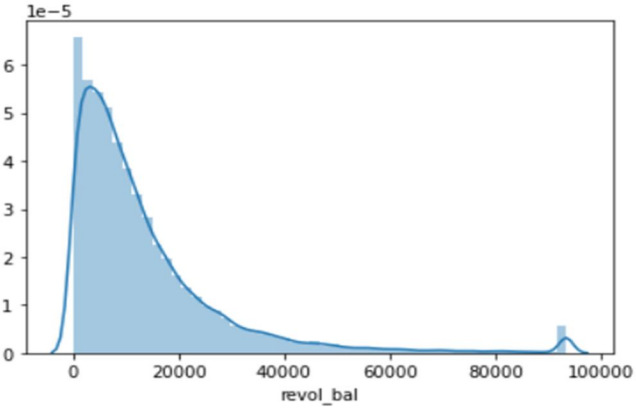

1.0
0.8
0.6
0.4
0.2
0.0
0
2
4
6
8
10
12
inq_last_12m

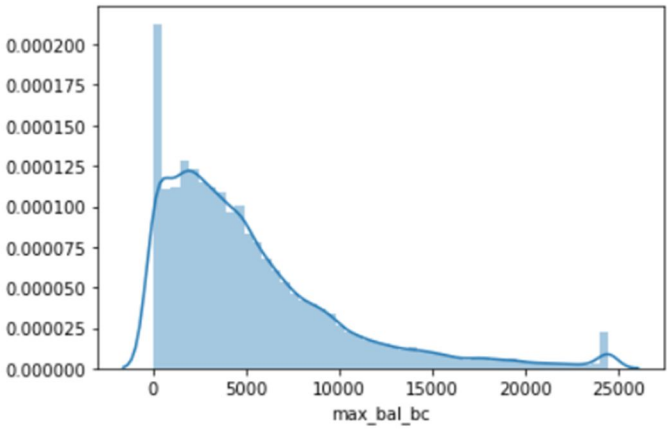

2.0
1.5
1.0
0.5
0.0
0
2
4
6
8
mort_acc

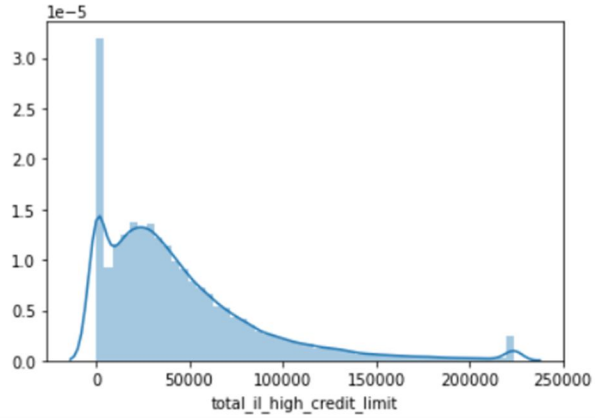

1e−5
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0
50000
100000
150000
200000
250000
total_bal_ex_mort

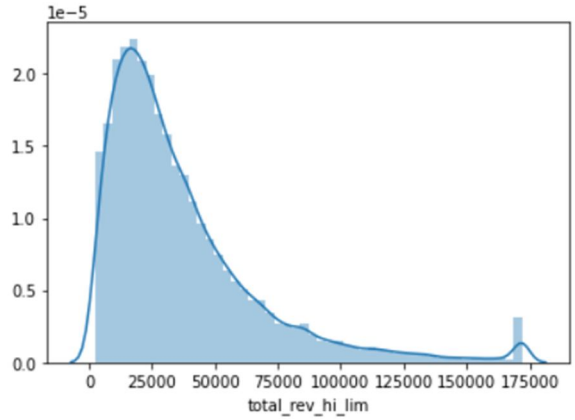

0.25
0.20
0.15
0.10
0.05
0.00
0
20
40
60
80
mths_since_last_delinq

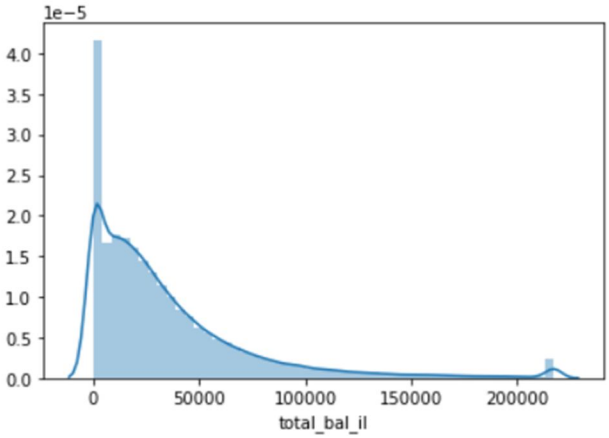

0.06
0.05
0.04
0.03
0.02
0.01
0.00
0
20
40
60
80
100
120
140
mths_since_rcnt_il


1e−5
6
5
4
3
2
1
0
0
20000
40000
60000
80000
100000
revol_bal


0.000200
0.000175
0.000150
0.000125
0.000100
0.000075
0.000050
0.000025
0.000000
0
5000
10000
15000
20000
25000
max_bal_bc


1e−5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
0
50000
100000
150000
200000
250000
total_il_high_credit_limit





5.3 Yeo Johnson Transformation

sklearn.preprocessing.PowerTransformer

class `sklearn.preprocessing.`**`PowerTransformer`**(*method='yeo-johnson'*, *, *standardize=True*, *copy=True*)

Apply a power transform featurewise to make data more Gaussian-like.

Power transforms are a family of parametric, monotonic transformations that are applied to make data more Gaussian-like. This is useful for modeling issues related to heteroscedasticity (non-constant variance), or other situations where normality is desired.

Currently, PowerTransformer supports the Box-Cox transform and the Yeo-Johnson transform. The optimal parameter for stabilizing variance and minimizing skewness is estimated through maximum likelihood.

Box-Cox requires input data to be strictly positive, while Yeo-Johnson supports both positive or negative data.

By default, zero-mean, unit-variance normalization is applied to the transformed data.

Due to negative values present in a few columns, this transformation technique was preferred, after application of which, near to normal was achieved.

5.4 Correlation Check

Objective:

Remove collinear features in a dataframe with a correlation coefficient greater than the threshold. Removing collinear features can help a model to generalize and improves the interpretability of the model.

Inputs:

threshold: any features with correlations greater than this value are removed

Output:

dataframe that contains only the non-highly-collinear features

5.5 Encoding the Categorical Variables

Emp_length is ordinal variables and hence we can do manual / label encoding.
Remaining are nominal and hence we use dummies / manual / one hot encoding
For ordinal data, mapping was used to assign an order or rank to it regarding the business importance and for nominal data, one hot encoder was used. Encode categorical features as a one-hot numeric array. The input to this transformer should be an array-like of integers or strings, denoting the values taken on by categorical (discrete) features. The features are encoded using a one-hot (aka 'one-of-K' or 'dummy') encoding scheme. This creates a binary column for each category and returns a sparse matrix or dense array (depending on the sparse parameter).
Now, the encoding has resulted into 32 categorical features.

5.6 Missing Values Imputation

The missing values in the dataset range from 0.01% to 100 %. There are in all 39 features which have missing values.

Sl No.	Feature_Name	No. of missing values	Percent of missing values
1	id	35648	100
2	member_id	35648	100
3	emp_title	3155	8.85
4	emp_length	3104	8.71
5	url	35648	100
6	desc	35648	100
7	dti	88	0.25
8	mths_since_last_delinq	19759	55.43
9	mths_since_last_record	30204	84.73
10	revol_util	50	0.14
11	next_pymnt_d	35648	100
12	mths_since_last_major_derog	27023	75.81
13	annual_inc_joint	31104	87.25

14	dti_joint	31104	87.25
15	verification_status_joint	31176	87.46
16	mths_since_rcnt_il	1260	3.53
17	il_util	6014	16.87
18	all_util	11	0.03
19	avg_cur_bal	3	0.01
20	bc_open_to_buy	580	1.63
21	bc_util	604	1.69
22	mo_sin_old_il_acct	1260	3.53
23	mths_since_recent_bc	539	1.51
24	mths_since_recent_bc_dlq	28317	79.44
25	mths_since_recent_inq	2985	8.37
26	mths_since_recent_revol_delinq	25134	70.51
27	pct_tl_nvr_dlq	1	0
28	percent_bc_gt_75	580	1.63
29	revol_bal_joint	31104	87.25
30	sec_app_earliest_cr_line	31104	87.25
31	sec_app_inq_last_6mths	31104	87.25
32	sec_app_mort_acc	31104	87.25
33	sec_app_open_acc	31104	87.25
34	sec_app_revol_util	31202	87.53
35	sec_app_open_act_il	31104	87.25
36	sec_app_num_rev_accts	31104	87.25
37	sec_app_chargeoff_within_12_mths	31104	87.25
38	sec_app_collections_12_mths_ex_med	31104	87.25
39	sec_app_mths_since_last_major_derog	34029	95.46

Treatment of the missing values:

a. Missing values with 100% - Drop the features - 5
b. Imputing the null values for the features corresponding to the Joint applicant as they are NON Random NULL VALUES: Doing imputation for the details of the secondary applicants. In here, every null value will be valid null. The data which is null is pertaining to the individual applicants. Also it cannot be imputed with 0 as 0 has different meaning with respect to different feature. Thus, imputation with -1 for numerical variables and NOT APPLICABLE for categorical variables

c. Imputing the null values of the features like mnth_since_delinq/derog/collection. They are all NON RANDOM NULL VALUES: Imputation with any constant (-1) as they are Non Random Nulls. Imputing with 0 is not a good option because zero means that person has entered into that category recently. All the columns here will be numerical only as they are the month column.
d. Missing Values less than 0.5% - Drop the rows
e. Features with high unique values (e.g. Emp_title) – drop
f. Imputing the null values of the features like mo_sin and other features from missing values table. They are all RANDOM NULL VALUES. Imputation with KNN imputer is good option. This can be done after the EDA.

sklearn.impute.KNNImputer

class sklearn.impute.**KNNImputer**(*, *missing_values=nan*, *n_neighbors=5*, *weights='uniform'*, *metric='nan_euclidean'*, *copy=True*, *add_indicator=False*)

Imputation for completing missing values using k-Nearest Neighbors.

Each sample's missing values are imputed using the mean value from n_neighbors nearest neighbors found in the training set. Two samples are close if the features that neither is missing are close.

After missing value imputation, the data set size now is 35340 rows and 96 features

6. Statistical Tests

Statistical tests are performed to check the feature significance with the Target Variable.

6.1 Chi2 for categorical

Chi2 test is done for categorical variables.

H0: There is no relation between the feature and the target

H1: There is relation between the feature and the target

Level of significance – alpha – 0.05.

The features which have pvalue less than 0.05 are termed as significant feature.

In the given dataset, all the 9 categorical features turn out to be significant.

6.2 Manwhitneyu and T-test for continuous

a. For continuous variables, test for normality is performed using Jarque Bera test as the sample size is greater than 2000.

   H0: Data is normal

   H1: Data is not normal

   With the above test, it pointed out that 6 variables have normal distribution and 76 are not normal.

b. The next test is test for equal variances and this is tested using Levene's test.

   H0: Equal variances

   H1: Not equal variances

   With above test, it pointed out that 23 features have equal variances and 59 features do not have equal variances

c. From the above tests, it is noted that the distributions of 6 features are normal. And hence T-test of independence is used.

   H0: Means are not significant

   H1: Means are significant

   With above test, it is pointed that all 6 normal features are significant

   For remaining 76 features, Manwhitneyu test is done.

   H0: Means are not significant

   H1: Means are significant

   With above test, it is pointed that all 66 features are significant.

6.3 Feature Significance

Out of 9 categorical features, all 9 found to be significant

Out of 82 continuous features, 72 found to be significant

7. Base Model Fitting

After the basic EDA, Linear (Logistic Regression) and Non-Linear (Random Forest) base models were fit on the train dataset of size (31345,114)

Below are the results:

Linear Model:

Feature Selection Technique	Model	F1 weighted	Variance Error
Base Models	LR	0.76	0.0000187

Non-Linear Model:

Feature Selection Technique	Model	F1 weighted	Variance Error
Base Models	RFC	0.75	0.0000259

8. Feature Selection Techniques

8.1 VIF for numerical variables.

Variance Inflation Factor (VIF) is used to detect the presence of multi-collinearity. Variance inflation factors (VIF) measure how much the variance of the estimated regression coefficients are inflated as compared to when the predictor variables are not linearly related.

- As a measure to check for existence of multi-collinearity below:

Feature	VIF
loan_amnt	1.40
int_rate	1.81
annual_inc	3.29
dti	3.81
delinq_2yrs	1.80
inq_last_6mths	1.71
mths_since_last_delinq	3.69
mths_since_last_record	1.21
open_acc	33.56
revol_bal	19.12
revol_util	14.69
total_acc	38.86
collections_12_mths_ex_med	1.09
mths_since_last_major_derog	4.87
annual_inc_joint	24330.33
dti_joint	24347.59
acc_now_delinq	1.41
tot_coll_amt	1.12
tot_cur_bal	70.14
open_acc_6m	3.44
open_act_il	11.19
open_il_12m	6.44
open_il_24m	9.36
mths_since_rcnt_il	1.90
total_bal_il	15.77
il_util	2.80
open_rv_12m	13.10

open_rv_24m	16.25
max_bal_bc	8.08
all_util	5.47
total_rev_hi_lim	20.42
inq_fi	1.66
total_cu_tl	1.29
inq_last_12m	2.30
acc_open_past_24mths	23.86
avg_cur_bal	57.87
bc_open_to_buy	10.72
bc_util	13.83
chargeoff_within_12_mths	1.12
delinq_amnt	1.41
mo_sin_old_il_acct	1.66
mo_sin_old_rev_tl_op	1.62
mo_sin_rcnt_rev_tl_op	6.90
mo_sin_rcnt_tl	5.47
mort_acc	3.03
mths_since_recent_bc	2.07
mths_since_recent_bc_dlq	2.82
mths_since_recent_inq	1.29
mths_since_recent_revol_delinq	3.82
num_accts_ever_120_pd	4.38
num_actv_bc_tl	10.84
num_actv_rev_tl	9.98
num_bc_sats	13.20
num_bc_tl	9.94
num_il_tl	16.62
num_op_rev_tl	29.83
num_rev_accts	26.12
num_tl_90g_dpd_24m	1.66
num_tl_op_past_12m	19.00
pct_tl_nvr_dlq	4.62
percent_bc_gt_75	2.84
tax_liens	1.12
total_bal_ex_mort	15.68
total_bc_limit	20.20
sec_app_mths_since_last_major_derog	1.50

- 24 features exhibited high multi-collinearity, above 10.

- Post dealing with multi-collinearity, left with 48 features and results were:

Feature	VIF
loan_amnt	1.394
int_rate	1.776
annual_inc	2.573
dti	2.701
delinq_2yrs	1.791
inq_last_6mths	1.682
mths_since_last_delinq	3.670
mths_since_last_record	1.173
open_acc	4.893
collections_12_mths_ex_med	1.087
mths_since_last_major_derog	4.833
annual_inc_joint	1.687
acc_now_delinq	1.403
tot_coll_amt	1.108
open_acc_6m	3.299
open_act_il	3.795
open_il_12m	2.877
open_il_24m	3.407
mths_since_rcnt_il	1.712
il_util	2.151
open_rv_12m	3.203
open_rv_24m	3.243
max_bal_bc	2.168
all_util	3.489
inq_fi	1.641
total_cu_tl	1.224
inq_last_12m	2.155
avg_cur_bal	2.681
bc_open_to_buy	3.555
chargeoff_within_12_mths	1.115
delinq_amnt	1.407
mo_sin_old_il_acct	1.642
mo_sin_old_rev_tl_op	1.448
mo_sin_rcnt_tl	3.069
mort_acc	2.246
mths_since_recent_bc	1.713

mths_since_recent_bc_dlq	2.780
mths_since_recent_inq	1.273
mths_since_recent_revol_delinq	3.728
num_accts_ever_120_pd	4.339
num_actv_rev_tl	2.825
num_bc_tl	2.593
num_il_tl	3.267
num_tl_90g_dpd_24m	1.656
pct_tl_nvr_dlq	4.567
percent_bc_gt_75	2.333
tax_liens	1.113
sec_app_mths_since_last_major_derog	1.503

8.2 PCA

Principal Component Analysis or PCA is a widely used technique for dimensionality reduction of the large data set. Reducing the number of components or features costs some accuracy and on the other hand, it makes the large data set simpler, easy to explore and visualize. Also, it reduces the computational complexity of the model which makes machine learning algorithms run faster.

Steps Involved in PCA:

*Standardize the data. (with mean =0 and variance = 1)

*Compute the Covariance matrix of dimensions.

*Obtain the Eigenvectors and Eigenvalues from the covariance matrix (we can also use correlation matrix or even Single value decomposition, however in this post will focus on covariance matrix).

*Sort eigenvalues in descending order and choose the top k Eigenvectors that correspond to the k largest eigenvalues (k will become the number of dimensions of the new feature subspace $k \leq d$, d is the number of original dimensions).

*Construct the projection matrix W from the selected k Eigenvectors.

*Transform the original data set X via W to obtain the new k-dimensional feature subspace Y.

Assumptions:

*PCA is based on the Pearson correlation coefficient framework and inherits similar assumptions.

*Sample size: Minimum of 150 observations and ideally a 5:1 ratio of observation to features (Pallant, 2010)

*Correlations: The feature set is correlated, so the reduced feature set effectively represents the original data space.

*Linearity: All variables exhibit a constant multivariate normal relationship, and principal components are a linear combination of the original features.

*Outliers: No significant outliers in the data as these can have a disproportionate influence on the results.

*Large variance implies more structure: high variance axes are treated as principal components, while low variance axes are treated as noise and discarded.

PCA Limitations:

*Model performance: PCA can lead to a reduction in model performance on datasets with no or low feature correlation or does not meet the assumptions of linearity.

*Classification accuracy: Variance based PCA framework does not consider the differentiating characteristics of the classes. Also, the information that distinguishes one class from another might be in the low variance components and may be discarded.

*Outliers: PCA is also affected by outliers, and normalization of the data needs to be an essential component of any workflow.

*Interpretability: Each principal component is a combination of original features and does not allow for the individual feature importance to be recognized.

Below are the results for 95% captured variance, which is 38 PCs for PCA applied to 82 numerical features:

PC No.	Eigenvalues	Proportion Explained	Cumulative Prop explained
1	12.516	0.153	0.153
2	11.670	0.142	0.295
3	6.676	0.081	0.377
4	6.117	0.075	0.451
5	4.706	0.057	0.509
6	4.188	0.051	0.560
7	2.923	0.036	0.595
8	2.623	0.032	0.627
9	2.405	0.029	0.657
10	1.878	0.023	0.680
11	1.681	0.021	0.700
12	1.463	0.018	0.718
13	1.428	0.017	0.735
14	1.293	0.016	0.751
15	1.273	0.016	0.767
16	1.119	0.014	0.780
17	1.079	0.013	0.793
18	0.962	0.012	0.805
19	0.917	0.011	0.816
20	0.896	0.011	0.827
21	0.850	0.010	0.838
22	0.814	0.010	0.848
23	0.785	0.010	0.857
24	0.730	0.009	0.866
25	0.700	0.009	0.875
26	0.663	0.008	0.883
27	0.639	0.008	0.891
28	0.624	0.008	0.898
29	0.597	0.007	0.905

30	0.589	0.007	0.913
31	0.535	0.007	0.919
32	0.523	0.006	0.926
33	0.478	0.006	0.931
34	0.470	0.006	0.937
35	0.445	0.005	0.943
36	0.425	0.005	0.948
37	0.396	0.005	0.953

The total number of features after PCA were 38 (numerical) and 32 (categorical) features.

8.3 SelectKBest - Feature Selection Method

sklearn.feature_selection.SelectKBest

class `sklearn.feature_selection.`**`SelectKBest`**(*score_func=<function f_classif>*, *, *k=10*)

Parameters

score_func*callable*
Function taking two arrays X and y, and returning a pair of arrays (scores, pvalues) or a single array with scores. Default is f_classif (see below “See also”). The default function only works with classification tasks.

k*int or “all”, optional, default=10*
Number of top features to select. The “all” option bypasses selection, for use in a parameter search.

Attributes

scores_*array-like of shape (n_features,)*
Scores of features.

pvalues_*array-like of shape (n_features,)*
p-values of feature scores, none if score_func returned only scores.

- Models tried with different features’ numbers like 60,70,80,100.

8.4 RFE

sklearn.feature_selection.RFE

class sklearn.feature_selection.**RFE**(*estimator*, *, *n_features_to_select=None*, *step=1*, *verbose=0*)

Feature ranking with recursive feature elimination.

Given an external estimator that assigns weights to features (e.g., the coefficients of a linear model), the goal of recursive feature elimination (RFE) is to select features by recursively considering smaller and smaller sets of features. First, the estimator is trained on the initial set of features and the importance of each feature is obtained either through a coef_ attribute or through a feature_importances_ attribute. Then, the least important features are pruned from current set of features. That procedure is recursively repeated on the pruned set until the desired number of features to select is eventually reached.

- Models tried with different features' numbers like 60,70,80,100.

9. Machine Learning Model

- Different models tried with different features selection techniques and model with 114 features.

- ***Linear Models***

Feature Selection Technique	Model	F1 weighted	Variance Error
Base Models	LR	0.76	0.0000187
KBEST (100 features)	LR	0.82	0.0000135
KBEST (80 features)	**LR**	**0.82**	**0.0000091**
RFE (57 features)	LR	0.76	0.0000185
Ridge Penalty	LR	0.76	0.0000218
KBEST (57 features)	LR	0.82	0.0000141
Multicollinearity treated (VIF)	LR	0.76	0.0000277
RFE (100 features)	LR	0.76	0.0000174
PCA - 0.95 variance explained	LR	0.76	0.0000236
VIF - Kbest(80 features)	LR	0.76	0.0000187
VIF-RFE(DecisionTree)	LR	0.76	0.0000188

- ***Non Linear Models***

Feature Selection Technique	Model	F1 weighted	Variance Error
Base Models	RFC	0.75	0.0000259
PCA - 0.95 variance explained	RFC	0.74	0.0000131
RFE - (100 features)	ADA	0.77	0.000025
RFE - (100 features)	GB	0.76	0.000048
RFE - (100 features)	BAG	0.76	0.000017
RFE - (100 features)	DTC	0.76	0.000009
RFE - (80 features)	KNN	0.76	0.000003
RFE - (100 features)	KNN	0.76	0.000019
RFE - (100 features)	RFC	0.75	0.00006
RFE - (80 features)	DTC	0.76	0.000017
RFE - (80 features)	RFC	0.76	0.000039
RFE - (80 features)	BAG	0.76	0.000019
RFE - (80 features)	ADA	0.77	0.000035
RFE - (80 features)	GB	0.76	0.000052
RFE - (70 features)	KNN	0.76	0.000023

RFE - (70 features)	DTC	0.76	0.000012
RFE - (70 features)	RFC	0.76	0.000026
RFE - (70 features)	BAG	0.76	0.00004
RFE - (70 features)	ADA	0.77	0.000047
RFE - (70 features)	GB	0.76	0.000044
RFE - (60 features)	KNN	0.76	0.000023
RFE - (60 features)	DTC	0.76	0.000002
RFE - (60 features)	RFC	0.75	0.000025
RFE - (60 features)	BAG	0.76	0.000027
RFE - (60 features)	ADA	0.77	0.000024
RFE - (60 features)	GB	0.76	0.000036
KBEST - 100	KNN	0.76	0.000017
KBEST - 100	DTC	0.76	0.000004
KBEST - 100	RFC	0.75	0.000042
KBEST - 100	BAG	0.76	0.000013
KBEST - 100	ADA	0.77	0.000032
KBEST - 100	GB	0.76	0.000043
KBEST - 80	KNN	0.76	0.000018
KBEST - 80	DTC	0.76	0.000014
KBEST - 80	RFC	0.76	0.000056
KBEST - 80	BAG	0.76	0.000009
KBEST - 80	ADA	0.77	0.000027
KBEST - 80	GB	0.76	0.000062
KBEST - 60	KNN	0.76	0.000014
KBEST - 60	DTC	0.76	0.000021
KBEST - 60	RFC	0.76	0.000036
KBEST - 60	BAG:	0.76	0.00003
KBEST - 60	ADA:	0.77	0.00003
KBEST - 60	GB:	0.76	0.000057

- The highlighted model was best performing model:

    * Feature Selection Technique: select KBest (80 features)

    * Model: Logistic Regression

- Below are the list of 80 features:

Specs	Score
int_rate	1956.434376
tot_hi_cred_lim	592.929374
term	498.144199
tot_cur_bal	483.073531
mort_acc	471.559835
x0_MORTGAGE	424.312876
avg_cur_bal	401.863466
x0_RENT	376.807715
installment	351.19463
x1_Verified	323.258693
loan_amnt	310.00608
bc_open_to_buy	307.658759
total_acc	304.431814
num_il_tl	257.87714
x1_Not Verified	252.179219
total_rev_hi_lim	245.981195
annual_inc	224.544043
total_bc_limit	221.228648
mo_sin_old_il_acct	159.138632
inq_last_6mths	153.073658
sec_app_mths_since_last_major_derog	118.14517
num_rev_accts	112.615241
revol_util	101.504131
num_sats	99.814368
bc_util	99.324699
open_acc	98.512524
num_bc_tl	96.645636
total_il_high_credit_limit	92.625603
mo_sin_old_rev_tl_op	90.3995
mths_since_recent_inq	88.27972
total_bal_ex_mort	85.650756
x2_credit_card	84.786589
open_il_24m	81.650727
total_bal_il	78.370221
percent_bc_gt_75	73.527568
sec_app_open_act_il	72.486266
sec_app_inq_last_6mths	72.397273
sec_app_revol_util	72.371428
dti_joint	72.243471
all_util	72.097842

revol_bal_joint	72.028568
sec_app_collections_12_mths_ex_med	71.872815
sec_app_chargeoff_within_12_mths	71.857158
application_type	71.848329
x3_NA	71.848329
annual_inc_joint	71.801888
sec_app_open_acc	71.605267
sec_app_num_rev_accts	71.368253
sec_app_mort_acc	68.932992
open_act_il	58.831846
x2_small_business	55.639336
total_cu_tl	55.197031
max_bal_bc	52.665428
dti	51.033965
emp_length	45.742598
x3_Verified	34.804189
x3_Source Verified	34.068188
open_il_12m	29.144671
mo_sin_rcnt_rev_tl_op	28.324466
num_op_rev_tl	27.328947
x4_West	26.397733
inq_last_12m	23.99168
revol_bal	20.058075
open_rv_12m	19.679548
num_bc_sats	17.821092
initial_list_status	16.353479
num_actv_rev_tl	14.225736
x2_medical	14.164343
open_rv_24m	13.521271
acc_open_past_24mths	12.803128
x2_major_purchase	12.535832
x4_Northeast	11.929872
mths_since_recent_bc	11.256434
num_tl_90g_dpd_24m	11.248036
num_rev_tl_bal_gt_0	10.881942
x2_other	10.056425
x2_home_improvement	9.965966
collections_12_mths_ex_med	9.744764
x2_moving	9.72147
open_acc_6m	9.077912

- Hyper parameter tuning

It is observed that f1 weighted score of above model on application of cross – validation is 82% and test f1 weighted score is 83% but the recall value on both train and test is very low i.e. 0.09 and 0.1 for train and test respectively.

The business requirement being to correctly predict minority class (Non-defaulters), the recall value should be high. Hence methods like down sampling and over sampling were considered.

The down sampled data has only 5688 rows. Though this data is realistic, the basic assumption, sample size (5688) equals features' square (80*80) fails.

Thus, over sampling is preferred. The model trains on 25657 samples. It is observed that though f1 weighted score for train is 68% and for test is 73%, i.e. the reduction in f1 weighted score, the recall for train and test improved to a great deal, i.e. 0.69 for both train and test.

So, this is considered to be a better model as it fulfills the basic business requirements.

10. Business Recommendations & Model Interpretability

Overall work on this data suggests that business should go ahead with linear models, which performed better given weighted f1 score and recall.

A few of top influencing features are interest rate, term, employment length, loan amount.

Below are a few interpretations from statistical model:

* As the loan amount increases, the chances of an applicant might turn out to a defaulter is high

* Higher the interest rate, possibility of applicant to become a default is high

* Lower the annual income, higher the chances of defaulters

* Higher the debt to income ratio, there are the chances that applicants might turn out to a defaulter is high

* As the balance to credit limit ratio increases, possibility of applicant to become a default is high

* More the recent inquiries, more the chance of defaulting.

* More the number of taxes missed, higher the possibility of loan default

* Higher the employment length of the applicant, lower the possibility of being default.

*Higher the term, more the chance of defaulting.

11. Future Work

- Accuracy improvement through fine tuning of model

- End to end Deployment of model using pipeline and web developments frameworks.