**Lending Club Issued Loans Analysis Protocol**

*Authors: Feldman Nadav   
 Burkis Konstantin*

Table of contents

[1. Introduction 3](#_Toc2715538)

[2. Methodology 5](#_Toc2715539)

[2.1 Data 5](#_Toc2715540)

[2.1.1 Data description 5](#_Toc2715541)

[2.1.2 Data preparation 5](#_Toc2715542)

[2.1.3 Outcome variable definition 6](#_Toc2715543)

[2.1.4 Data enrichment 9](#_Toc2715544)

[2.1.5 Data exploratory analysis 10](#_Toc2715545)

[2.1.6 Data cleansing 21](#_Toc2715546)

[2.1.7 Feature Selection 28](#_Toc2715547)

[2.2 Models 30](#_Toc2715548)

[2.2.1 Train, dev (cross validation) and test 30](#_Toc2715549)

[2.2.2 Treating the Imbalance 30](#_Toc2715550)

[2.2.3 Model selection 31](#_Toc2715551)

[3. Results 33](#_Toc2715552)

[3.1 Data results 33](#_Toc2715553)

[3.2 Model Results 34](#_Toc2715554)

[3.2.1 Model Selection 34](#_Toc2715555)

[3.2.2 Fine tuning 35](#_Toc2715556)

[3.2.3 Final model assessment 39](#_Toc2715557)

[4. Conclusion 40](#_Toc2715558)

[5. Bibliography 43](#_Toc2715559)

[6. Appendix 44](#_Toc2715560)

[6.1 Data Dictionary by Lending Club 44](#_Toc2715561)

[6.2 Table 4 - Table one 46](#_Toc2715562)

# Introduction

In this project we analyzed Landing Club[[1]](#footnote-1) Issued Loans data with the aim of estimating the credit risk of each loan, independent of LC's estimation.

Although there are several approaches to estimate credit risk, such as estimating the exposure at default (EAD) and the loss given default (LGD), in this work we studied only the probability of defaulted (PD) method, meaning estimating the probability to default of each given loan, by using classification algorithms to score each loan.

Credit risk problems have great importance for credit providers such as banks or credit cards companies and credit risk rating models are at the heart of their business model, evidence of this can also be seen in the constant preoccupation of the regulator with this subject.

Therefore, there is no wander that credit risk problems have been studied extensively, both in industry and academy and in particular, Lending Club data have been studied by many.

From the algorithmic point of view, there is a lot to studies from (Baesens et al., 2003)[[2]](#footnote-2) who set up a benchmarking state of the art classification algorithms for credit scoring by applied various state of the art classification algorithms on to eight real-life credit scoring data sets. they found that both the LS-SVM and neural network classifiers yield a very good performance, but also simple classifiers such as logistic regression and linear discriminant analysis perform very well for credit scoring.

Another great resource, and much up to date, are (Lessmann & e.g., 2015)[[3]](#footnote-3) whom updated the study of Baesens et al. and compared several novel classification algorithms to the state-of-the-art in credit scoring and found some advanced methods to perform extremely well on their credit scoring data sets, but never observe the most recent classifiers to excel.

Also (Teplý &Polena, 2019) [[4]](#footnote-4) did tried 10 different classification techniques algorithms on the Lending Club data set, in order to create a robust ranking of those 10 algorithms.[[5]](#footnote-5)

There results show that logistic regression, artificial neural networks, and linear discriminant analysis are the three best algorithms based on the Lending Club data.

From previous researches we can also study about important variables which are known to influence the probability to default, such as (Emekter, Tu, Jirasakuldech & Lu, 2015)[[6]](#footnote-6) , which analyst Lending Club data[[7]](#footnote-7) and found that credit grade (given by lending club), debt-to-income ratio, FICO score and revolving line utilization play an important role in loan defaults and that higher interest rates charged on the high-risk borrowers are not enough to compensate for higher probability of the loan default.

It should be noted that others also found loan purpose to be an important variable.

As mentioned, in this project we aimed to estimate the probability of defaulted independently to Lending Club estimation, hence we decided not to use any variable related to this estimation, including Lending Club credit grade.

In addition, while a proper definition of default is a case which a debtor has passed the payment deadline on a debt they were due to pay, in aim of making a better and more precise credit risk estimation and in order to decrease noises, we defined it as a case which the borrower failed to pay his debt. It should be noted that this definition is also consistent with definitions made in previous studies.

Finally, In order to create the best predicative model for probability of defaulted estimation, as defined above, we have set number of baselines for comparison.

First, due to the unbalance nature of our problem and data (most of loans are paid), we would like to be more successful than if we had guessed that all the loans would be paid.

Second we would like to be more successful than previous work[[8]](#footnote-8) done on the same Lending Club data set.[[9]](#footnote-9)

Although not only that this subject has been studied extensively in the academia, but also much work has been done on this issue also in less formal frameworks, we hoped that we would be able to create a good model against the baselines that we have set.

This was because we operate according to academic standards, but unlike academic studies, our sole goal was to produce as successful predictive model as possible, based on our data, and we were willing to try an extensive toolkit to achieve this goal.

# Methodology

## Data

### Data description

In our project we are using a dataset which was posted in Kaggle, [Lending Club Loan Data.](https://www.kaggle.com/wendykan/lending-club-loan-data)

It seems that the data can be downloaded directly from Lending Club Webpage <https://www.lendingclub.com/info/download-data.action>, but it may differ. Also as stated in Lending Club Webpage the data is not full, full data can be downloaded only after registration to the site therefore our dataset have a lot of missing values and some features are missing at all.

The downloaded dataset from Kaggle comes with a form of one .csv file or sqldb file. The file contain loan data for all loans issued through 2007-2015, including the current loan status (Current, Late, Fully Paid, etc.) and latest payment information. Additional features include credit scores, number of finance inquiries, address including zip codes, and state, and collections among others. The file contains 887,382 observations and 75 variables. A data dictionary is provided by Lending Club in a separate file in appendix (lcdatadictionary.xlsx).

### Data preparation

In order to make the data suitable to the course project requirements we divided the file into 6 tables which describe the loans details and the loaner's details. The final ER Diagram of the database that we created is shown below.

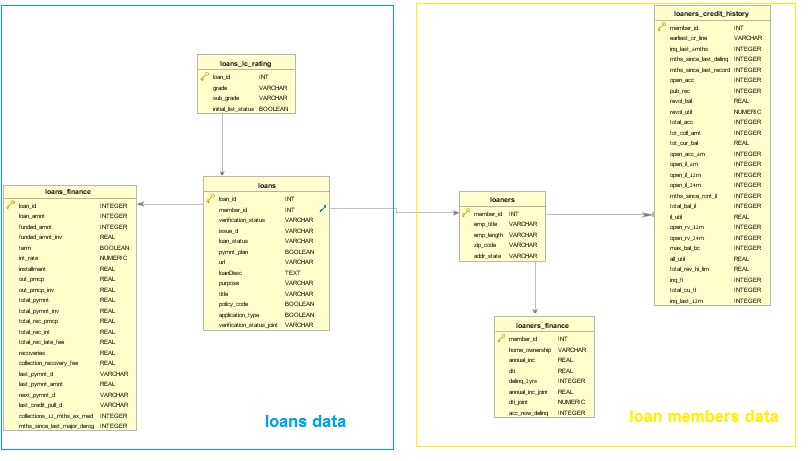


Figure 1 - ER Diagram of database

One of the possible external data that we thought and that may enrich our data considering that we have the state of each loan is the poverty rate of the state in that time.  
We downloaded poverty rates by states in the years 2007 – 2015 of loans from

<https://www.census.gov/cps/data/cpstablecreator.html>.

### Outcome variable definition

Our ultimate goal is the prediction of borrower failed to pay his debt, which we defined as Defaulted. The variable loan status seems to be an indicator of the current state a particular loan is in.

The different loan statuses in our data and their spread shown in Table 1 - Loan statuses statistics and Figure 2 - Loans statistics.

| **loan\_status** | **count** | **rel\_count** |
| --- | --- | --- |
| Charged Off | 45248 | 0.0510198214 |
| Current | 601340 | 0.6780467509 |
| Default | 1219 | 0.0013744953 |
| Does not meet the credit policy. Status:Charged Off | 761 | 0.0008580729 |
| Does not meet the credit policy. Status:Fully Paid | 1988 | 0.0022415887 |
| Fully Paid | 207723 | 0.2342200839 |
| In Grace Period | 6250 | 0.0070472481 |
| Issued | 8396 | 0.0094669913 |
| Late (16-30 days) | 2357 | 0.0026576582 |
| Late (31-120 days) | 11589 | 0.0130672894 |

Table 1 - Loan statuses statistics

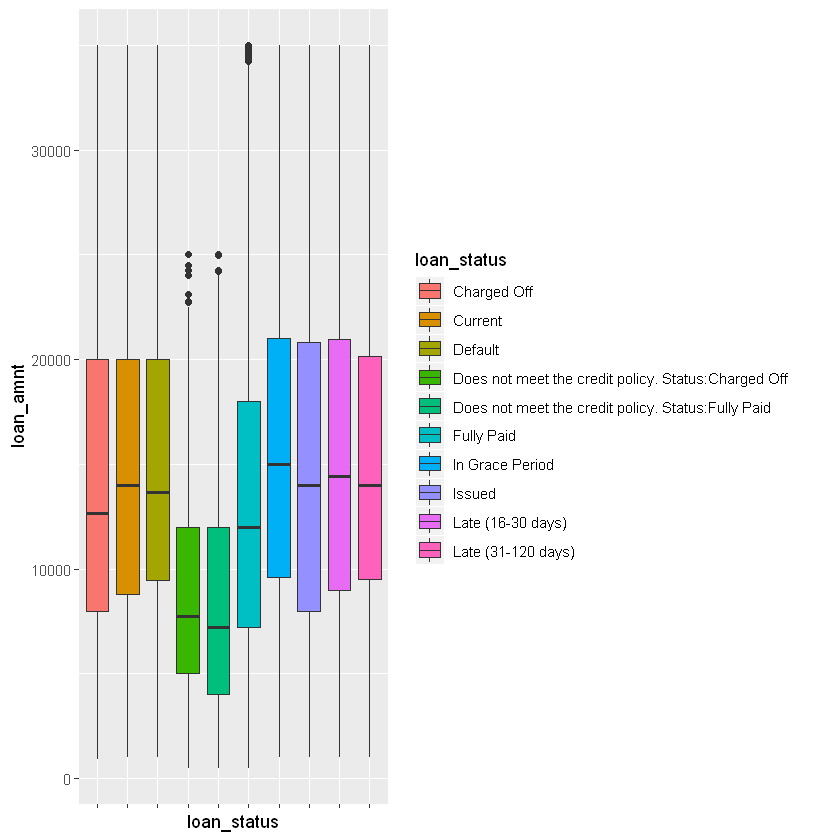


Figure 2 - Loans statistics

It is not immediately obvious what the different values stand for, so we refer to Lending Club’s documentation about “[What do the different Note statuses mean?](https://help.lendingclub.com/hc/en-us/articles/215488038-What-do-the-different-Note-statuses-mean-)”

* **Fully Paid:** Loan has been fully repaid, either at the expiration of the 3- or 5-year year term or as a result of a prepayment.
* **Current:** Loan is up to date on all outstanding payments.
* **Does not meet the credit policy. Status:Fully Paid:** No explanation but see “fully paid”.
* **Issued:** New loan that has passed all Lending Club reviews, received full funding, and has been issued.
* **Charged Off:** Loan for which there is no longer a reasonable expectation of further payments. Generally, Charge Off occurs no later than 30 days after the Default status is reached. Upon Charge Off, the remaining principal balance of the Note is deducted from the account balance. Learn more about the [difference between “default” and “charge off”](https://help.lendingclub.com/hc/en-us/articles/216127747).
* **Does not meet the credit policy. Status:Charged Off:** No explanation but see “Charged Off”
* **Late (31-120 days):** Loan has not been current for 31 to 120 days.
* **In Grace Period:** Loan is past due but within the 15-day grace period.
* **Late (16-30 days):** Loan has not been current for 16 to 30 days.
* **Default:** Loan has not been current for 121 days or more.

Given above information, we will define a default (**Outcome**) as follows:

**Defaulted**loans are in status:

1. Charged Off
2. Does not meet the credit policy. Status:Charged Off

**Fully Paid** loans are in status:

1. Fully Paid
2. Does not meet the credit policy. Status:Fully Paid

For the other possible statuses we don't have a definite outcome of the loan therefore we will exclude them from our dataset.

We have these variables in our dataset which are indications of the outcome - determined by Lending Club after the company evaluations of the credit risk.

* grade
* sub\_grade
* int\_rate

These are confounder variables that may affect the outcome so we drop them.

We also transform the variable ***installment*** as it was calculated with the interest rate that LC is calculated for the loan. New variable ***loan\_installment*** was calculated as the "clean" calculation of.

After definition of our ***outcome*** value and definition of our question we can remove all columns that are not defined at the beginning of the loan evaluation process:

* last\_pymnt\_d
* next\_pymnt\_d
* collection\_recovery\_fee
* last\_pymnt\_amnt
* out\_prncp
* out\_prncp\_inv
* recoveries
* total\_pymnt
* total\_pymnt\_inv
* total\_rec\_int
* total\_rec\_late\_fee
* total\_rec\_prncp

By their definitions provided in the data dictionary, they all relate to later or current stages of the loan therefore are not relevant to the initiate state of the loan.

In the context of fear of bias, in our database there are two main reasons for concern:

1. The fact that the database contains only information about people who received a loan from the LC. As a result, it is reasonable to assume that we will find in the data fewer people who are dangerous relative to their existence in the general distribution of the population, i.e. it may lead to a **selection bias**. For example, it is nice to see that the max dti (debt to income ratio) is 39.9, while in Lending Club web site it is noted that: "**Ideally, your debt-to-income ratio would be lower than 40%. That’s generally the threshold used across the industry. If your DTI is higher than 40%, your loan application will likely be denied."**

Therefore, we must remember that our model is only good for assessing people who have passed the baseline criteria of LC.

This means that in order to correctly use our model, the user must first run filtering according to LC base criteria.

1. There is a concern that the **distribution of data from past years is different from the distribution in later years, and in particular in the current time, in which we want the model to work.** That because LC experienced a significant increase in its activity during the measured time and there is a danger that the mix of borrowers operating through LC has changed over time.

To deal with Potential bias number 2, we consider excluding some past years.

We did tested the different distribution of the Defaulted variable, which is our outcome variable, in each different year.

Also we checked a bout lending club history business model both in LC website and Wikipedia.

We didn't found any clues that Lending Club changed its business model, but we did identified that issued year variable effect on defaults rates was significant, as the reader can see in section 5, data exploration.

However, deleting the first few years also did not succeed in concealing the effect.

### Data enrichment

In order to enrich our dataset we found 2 pairs of variables that interaction between them may provide additional information.

* open\_acc and total\_acc describing the total and opened number of credits lines in the borrower's credit file we can divide them to show a ratio between them as an indicator of the borrower's activity.
* earliest\_cr\_line and last\_credit\_pull\_d describing the date of the first credit line and the last pull date that was made by Lending Club so if we calculate the difference we can see the length or age of credit history.

For better visualizations and to properly merge the poverty rate data we transform the addr\_state variable that contain the states in their abbreviated form into the full state name.

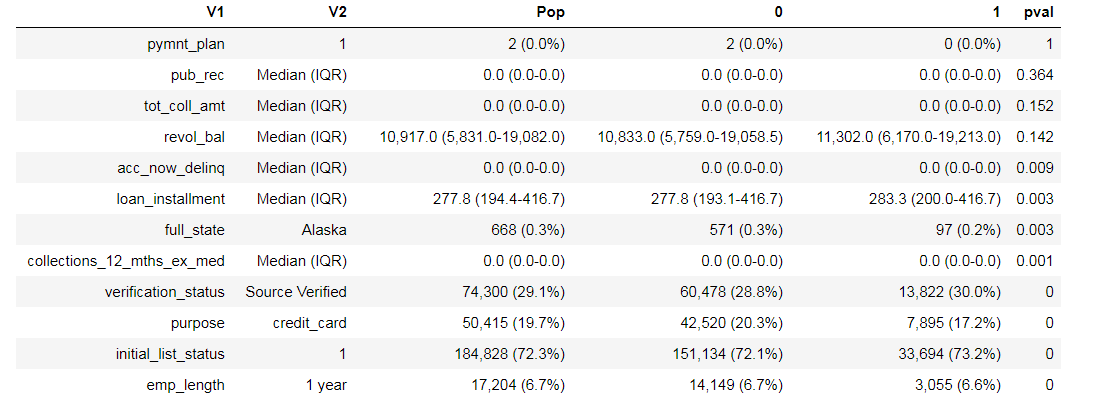
As stated before we also enriched the dataset with external poverty rate data. We created a variable that will show the poverty rate in the specific state in the time of the loan issue date.

### Data exploratory analysis

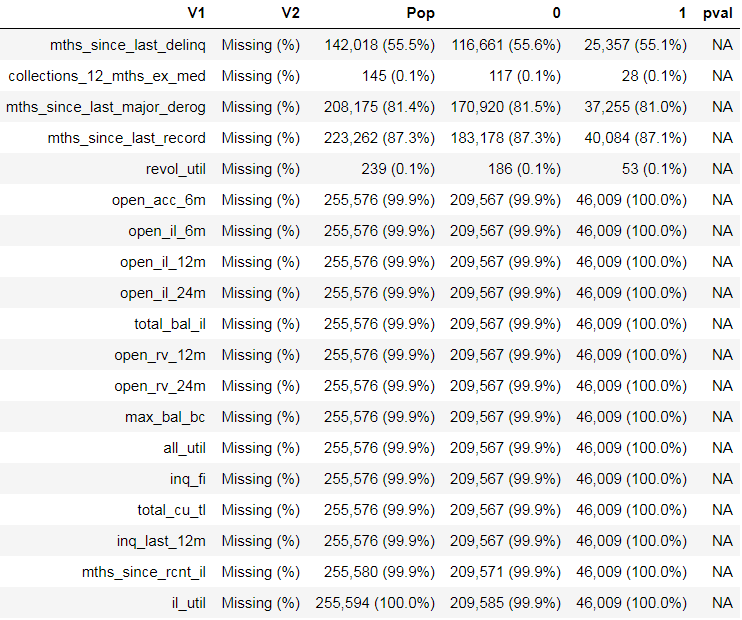
#### **Descriptive statistic and Table 1**

The exploratory stage we begin with checking statistical analysis of the variables, with summary function and Table1 (Table 4 - Table one) we check the mean, median, Iqr, min, max SD values.

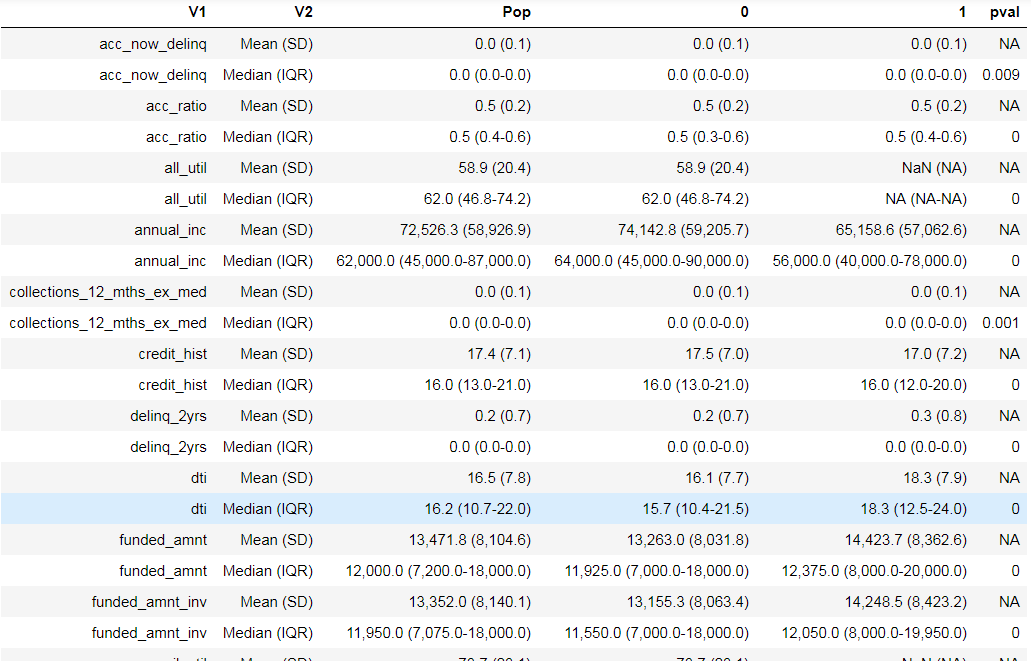
For each variable and we also checking the NA and unique ratios to fill the data retrieval protocol.

In Table 1(Table 4 - Table one) we can already see some variables that have p-value>0.05 for example pub\_rec,pymnt\_plan… and variables that have p-value <0.05 for example acc\_now\_delinq,loan\_installment. 

We can also see that we have a lot of variables with a very high missing rate like open\_acc\_6m and total\_bal\_il…



In another outlook of the table we can see which variables have median that is similar to their mean, which can be an indicator to a normal distribution. From the other hand, we can see which variables have median that differ from their median, and there for not normally distributed.

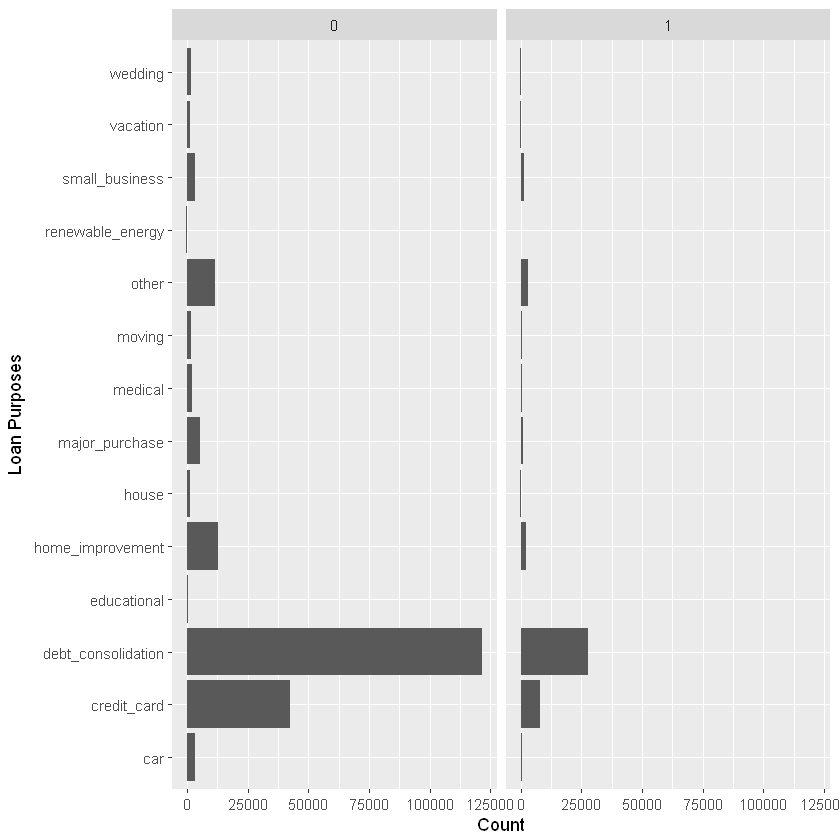


#### **Variables graphical visualizations and statistical analysis**

We continue the exploration with graphics visualizations of our variables. For each type of variable we will create specific plots:

● Categorical variables: barplot  
● Numeric Variables: histogram (distplot)   
● Numeric vs. Categorical: boxplot  
● Numeric vs. Numeric: scatter plot  
● Pairs

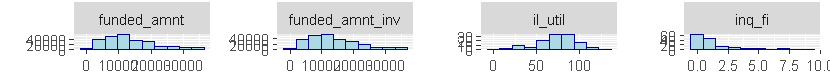
In the graph below, we can see the number of loans issued to the different purposes, divided to loans that had paid successfully and loan that defaulted. We can see that debt consolidation is the most common purpose and credit card is the second most common. From this graph we didn’t identified any purpose that has higher defaulted ratio then others, even though with the help of table one above, we could identified that Purpose is a good variable at separating between the paid loans to the defaulted.



In the plot below we can see the histogram of some the numerical variables. It should be noted that some of the variables are count variables.

From the histograms we can get another clues about the data distribution.

For example, founded\_amnt and funded\_amnt\_inv distributions seems very similar to each other and a bit close to normality distribution with a little bias towards the lowers values.



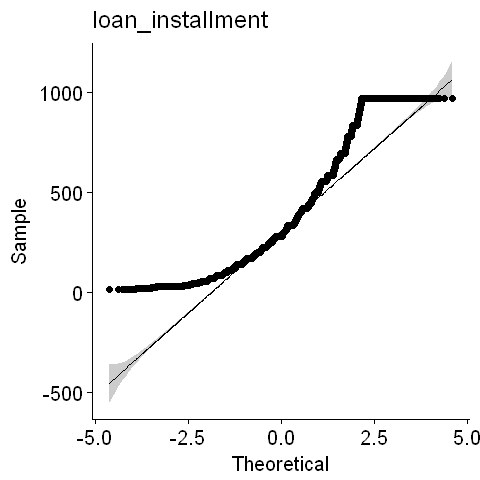
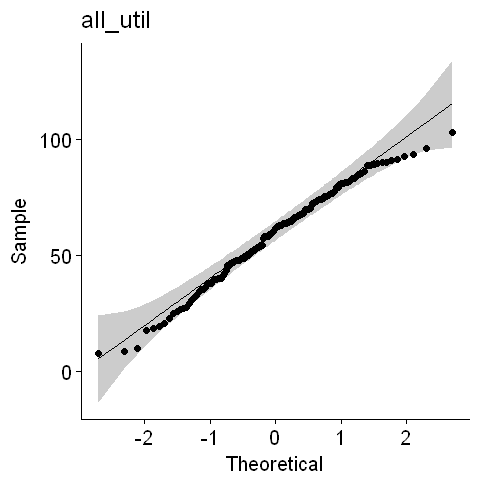
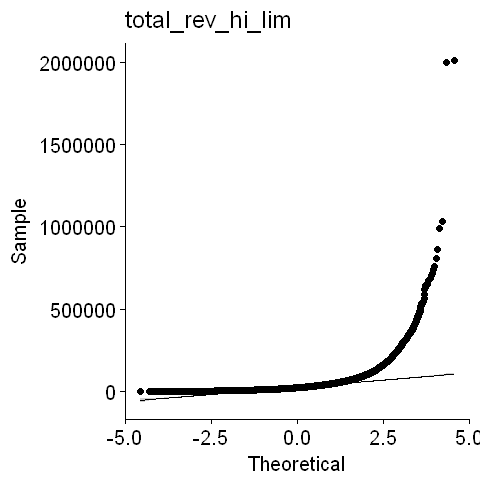
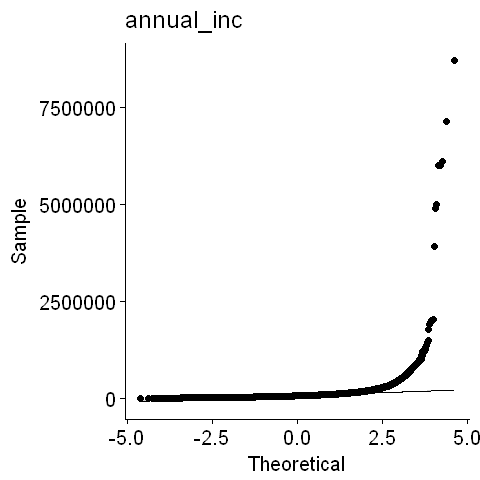
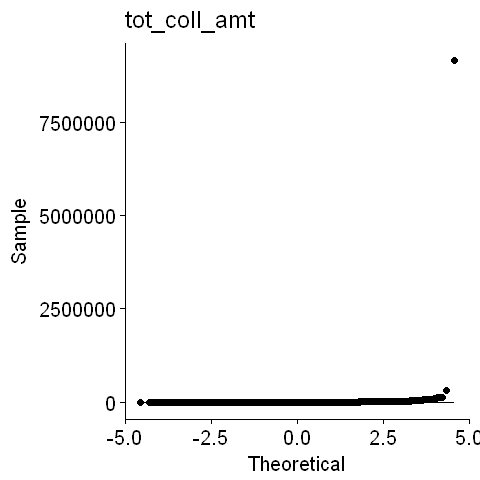
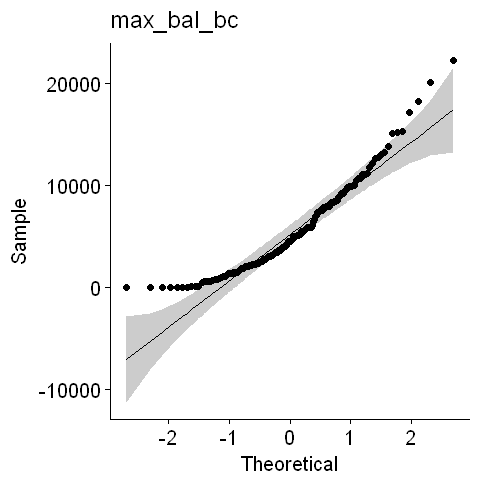
We also performed a normality distribution test to continuous variables using Kolmogorov-Smirnov normality test[[10]](#footnote-10) and a normality distribution test to all of numeric variables using Anderson-Darling normality test[[11]](#footnote-11) .

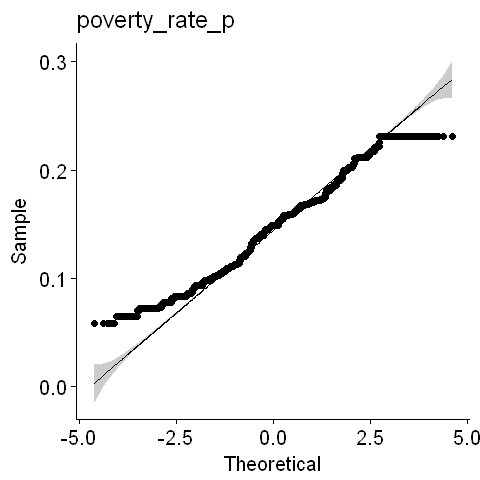
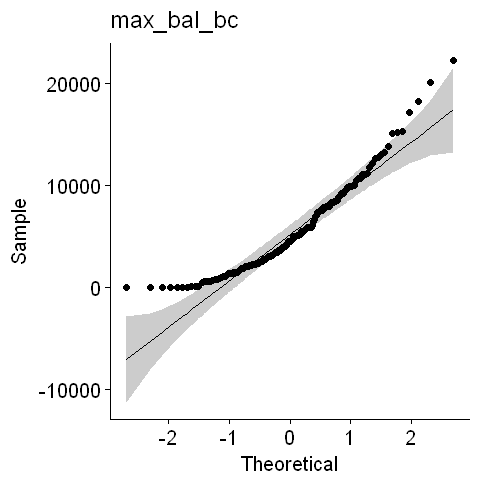
We did used it to check our variables distributions against theoretical normal distribution. We also did plot a Q-Q[[12]](#footnote-12) plots of our variables distributions against theoretical normal distribution.

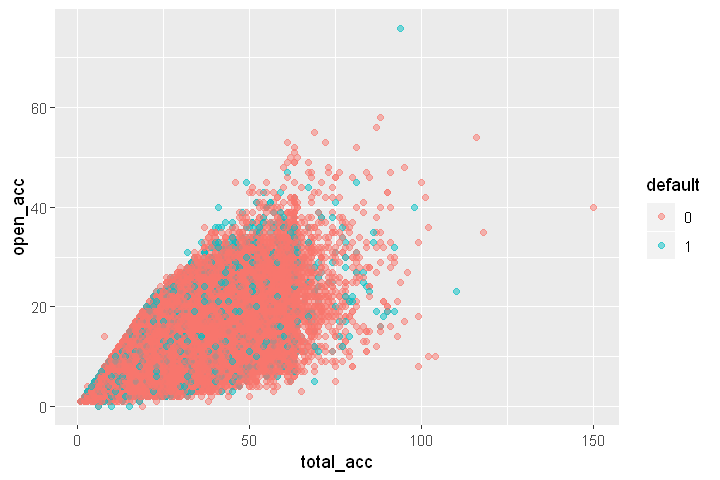
According to the KS test, not even one of the continuous variables is normally distributed[[13]](#footnote-13).

According to the AD test, the only numerical variable which normally distributed is all\_util[[14]](#footnote-14).

Below you can see some of the Q-Q graphs (the first one is all\_util ):

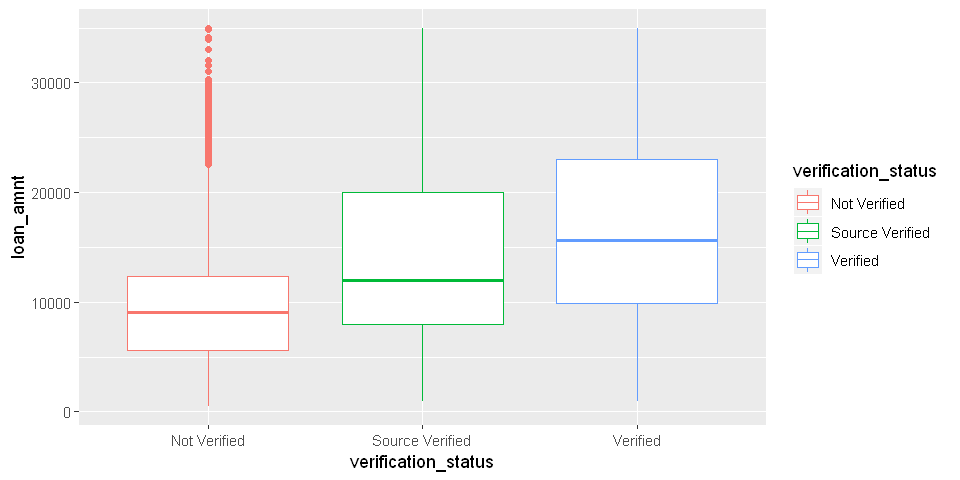




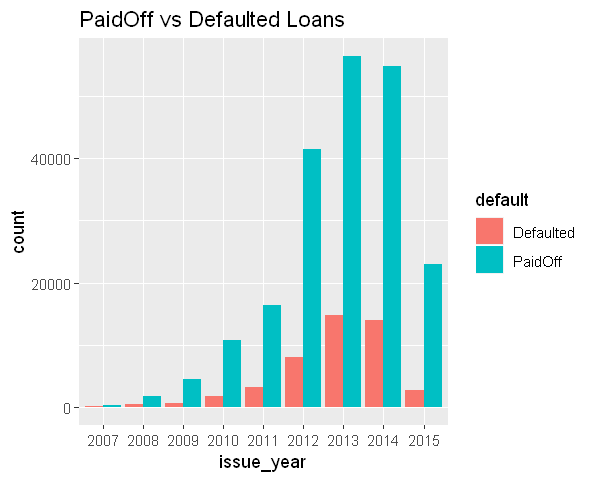
From the graphs above we can study that despite of the KS and the AD tests, for practical work, there are some variables that may be sufficiently close to the normal distribution[[15]](#footnote-15).

|  |  |
| --- | --- |
| Below there is a scatter plot of open\_acc vs. total\_acc. We can see that this pairs of variables are positively correlated. Throw that plot we also can identify some potentially outliers. These two variables are interesting to see in the same plot because we use the ratio between them to create a new variable. We can identify an area with an highly density of defaulted loans around the (50,40) coordinates. | Below there is a scatter plot of dti vs mnth since last major derog. In that plot its stands out that most of samples have less than 80 months since last major derog and above this value it should be suspected as an outlier. We also can see that most of defaulted loans are accrued when dti is higher than 25. |
|  |  |

In the plot below we can see that the loan amount is depends in the verified status, while the not verified loans are characterized by lower loan amount than the source verified which are characterized by lower loan amount than the verified loans.

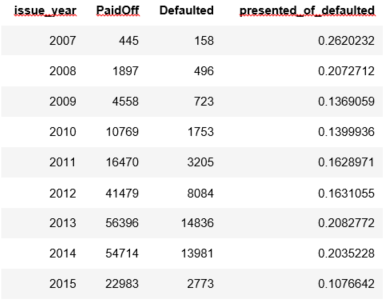


Even though the issue date variable is not going to be an exploratory variable in our final model[[16]](#footnote-16), it is an extremely important check[[17]](#footnote-17) in order to be careful from data biases. In the plot below, we can see the segmentation of fully paid and defaulted loans over every year in our data.



We can see, that the loans numbers in growth by the years and decrees at 2015. This dynamic is an outcome of the growth in Lending Club activity over the years and of the fact we have dataset till 2015 and we exclude loans with indefinite status, such as "current" status.

From this plot, we can't identify any meaningful different between the defaulted to paid ratios over the years. Even though with the help of ratio calculation and checking the Pearson's Chi-squared test we can see that year variable effect on the defaults is significant.



Pearson's Chi-squared test

data: loans$issue\_year and loans$default

X-squared = 1933.7, df = 8, p-value < 2.2e-16

We see that in years 2007- 2008 the ratio is higher than in other years, so we tried to exclude those years from the dataset and rerun the Chi-squared test, but that still shows that the variable is significant.

Pearson's Chi-squared test

data: loans\_ex\_2008$issue\_year and loans\_ex\_2008$default

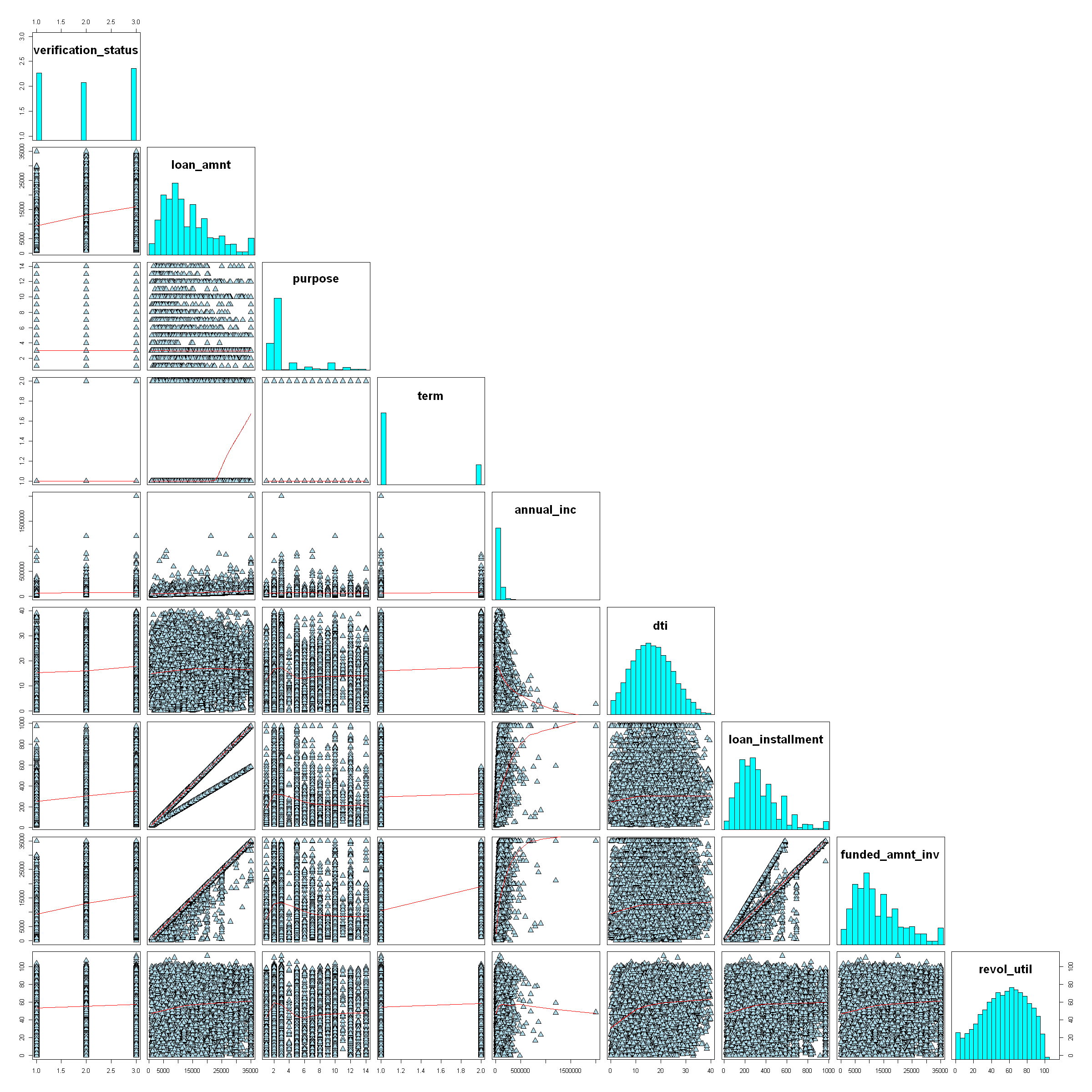
X-squared = 1897.5, df = 6, p-value < 2.2e-16

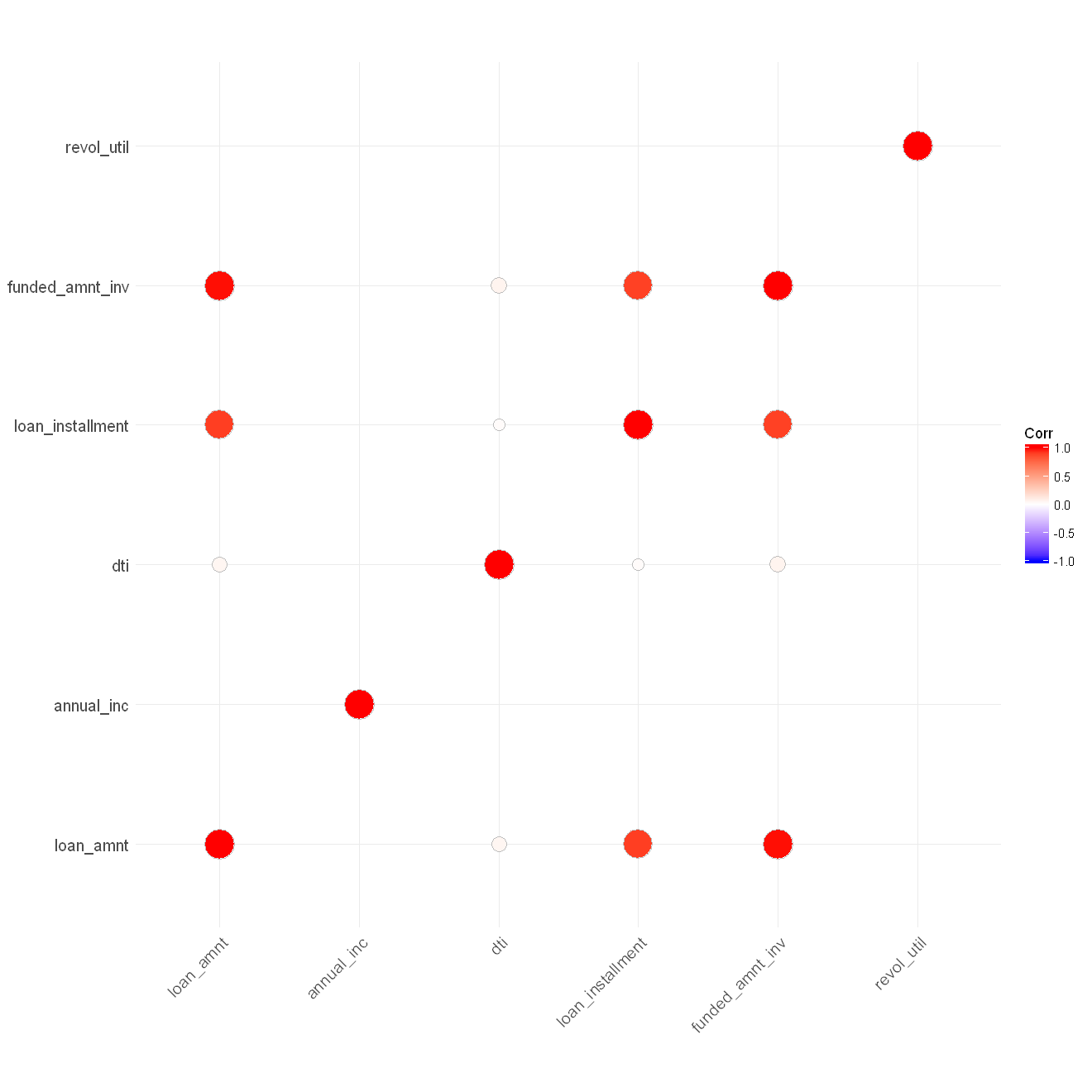
Therefore we did not excluded those loans from our dataset.

From the pairs plot Figure 3 - Pairs plot we can study that:

1. There is a positive and high correlation between loan\_amnt, funded\_amnt\_inv and loan installment. Furthermore, It is seems like funded\_amnt\_inv are equal to loan amount for most of the loans. Even though, there are some cases when they differ from each other.
2. We can identify tow lines in the scatter plot of loan installment and loans amount. Those lines are an outcome of the two possible different time terms of the loans (36 month of 60 month). That because the installment variable is an outcome of the loan amount variable divided by the loan time term.
3. It is very hard to see any correlation between dti and the others variables.
4. It is seems like revol\_util is not strongly correlated with any other variable. Although, it may correlated with dti.
5. We can identify some samples that are suspects as an outliers in the loan\_installment and the funded\_inv scatter plots.
6. We can see that in 36 month term period the loan installment is higher than in the 60 month term period.
7. It is seems like annual\_inc distribute exponential with a lot of samples in the low part of the distribution.
8. It seems like when term is 36 month the average loan amount is lower than if term is 60 month.

Figure 3 - Pairs plot



The outcome of the correlation table is consistent with our conclusions from the pairs plot.

For additional visualizations and analysis there are EDA notebook (3-Exploratory Data Analysis.ipynb) and a report (report.html) in their designated folders.

#### **creating more variables based on transformations and grouping**

As consequents of the data exploratory analysis, we created more variables based on transformations and grouping to categories. Some of the originals variables show skewed distributions so we try to transform them with logarithmic and square root transformations.

After we also performed a normality test to the new we created.

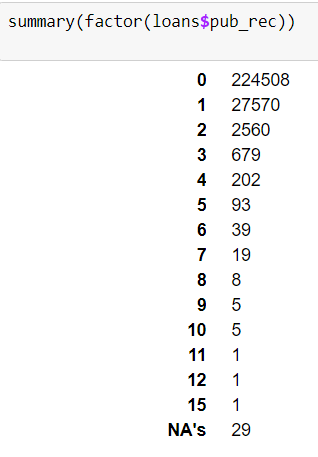
Even though the transformations helped to create more normality distributed alike variables, not even one of the new variables passed the Anderson–Darling test.

For example the variable ***open\_acc***:

|  |  |
| --- | --- |
| Open\_acc | Open\_acc with root transformation |
|  |  |
|  |  |

Some of our variables are count variables and have a small range of values and high percentage of the rows are around 0 and 1 so we group them into a categorical variable.

For example the variable ***pub\_rec***:



The variable has 14 different values but almost 90% of the rows are 0, so we creating a grouping variable which indicating if it's 0 or 1 and above.

### Data cleansing

#### **Dealing with outliers**

We started with identification of univariate outliers in our data.

As we saw in the EDA article, according to the AD test, the only numerical variable which normally distributed is ***all\_util***.

Therefore, in this work, we decided to use median and Interquartile range (IQR method), as showed by Tukey[[18]](#footnote-18) (Tukey, 1977), by using boxplot for all variables.

Additionally we perform the Z score test only for ***all\_util*** variable[[19]](#footnote-19), by using function written by us. That due to the fact that use median and Interquartile range is more robust to un-normal distribution.

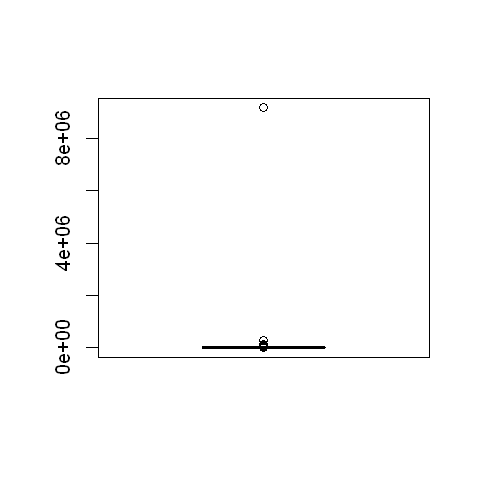
In the end of the EDA part we tried to perform a transformation on variables with far from normal distribution in order to make them more normally distributed alike.

In this stage of outliers detection we also demonstrated that the transformations indeed helped us detect outliers more precisely and therefore labeled less samples as an outliers.

During this check we did try to find a samples that are clearly a mistakes, in purpose of replacing them with NAs. We did that both by using data analysis methods and by investigating about lending club activity.

For an example to such case one can take a look on our findings about the ***tot\_coll\_amt*** variable. This variable giving an indication of what is the total collection amounts ever owed.

We found one sample with over than 9 million dollars that collected from him over all times, 30 times more than the second high amount collected!



Who is this individual?

We found that this person have a low dti, nice but not very high annual income and a mortgage.   
For a person that don't own a home and his annual income less than 150,000 $, more than 9 million dollars paid in the past looks almost unreasonable!   
Therefore, it was reasonable for us to assume that this outlier is an outcome of an error and we replaced it with a NA.

We also labeled some interesting outliers that we found by creating a new binary variables.

For example we did created new variable that take the value 1 for all loans amount that is equal to 35,000$, which is the highest possible amount for loan amount that taken in LC platform, and 0 for all others amount.

It should be noted that we were careful when decided that some sample is an error.

As evidence, you can look at revol\_util variable. As noted before, this variable is a rate representing the amount of the credit line used relative to total credit available. Hence, we accepted all is value to be less than 100, but for our surprise we find some of the samples value's to be more than 100. After we did investigate it, we found that this actually can happened, and there are cases when people can take more loans than the maximum official celling[[20]](#footnote-20).

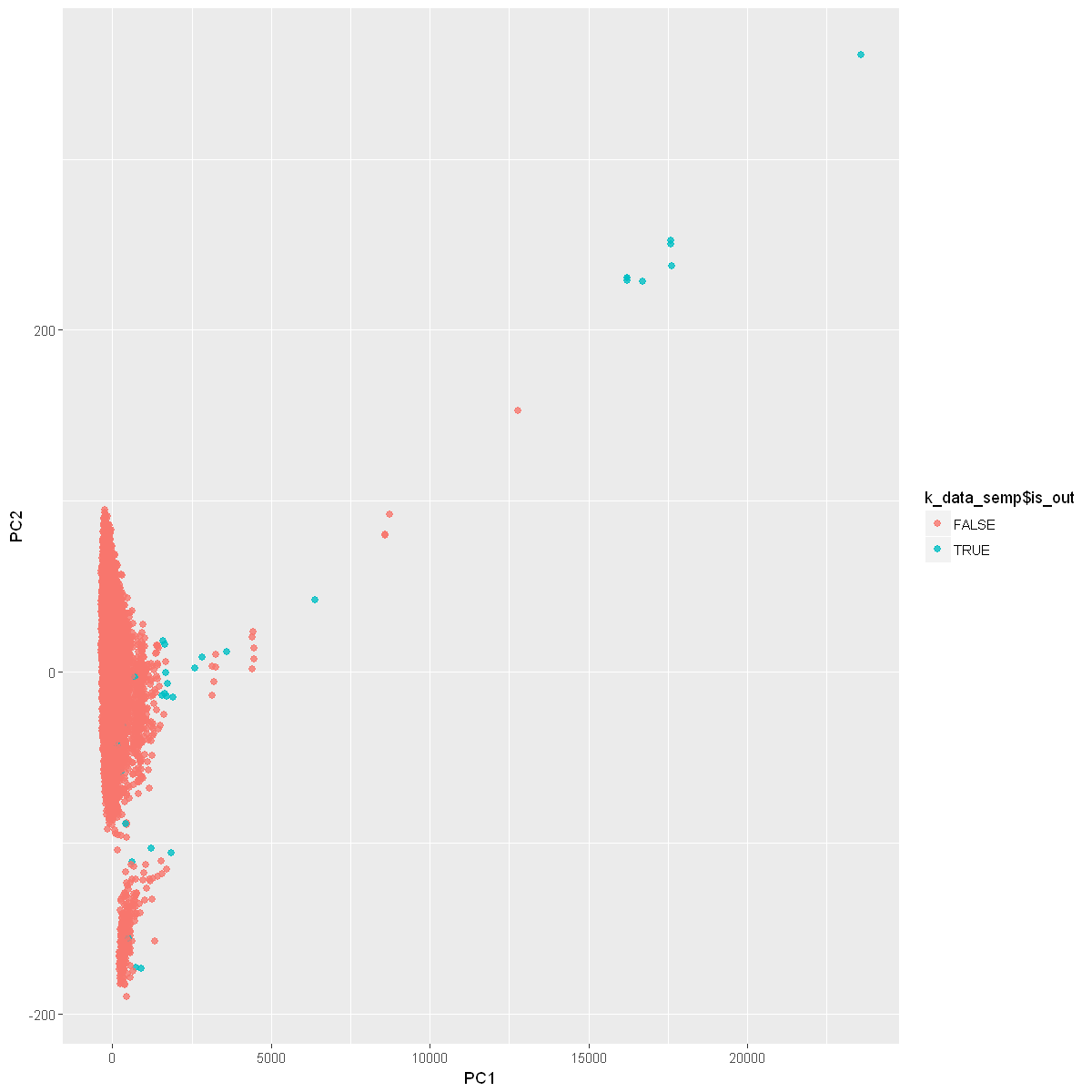
After the univariate detection we proceed our outlier detection with multivariate outlier's detection. At first we used DBSCAN[[21]](#footnote-21) algorithm to cluster the data and after we did labeled the cluster which did not contain as many loan samples as an outliers[[22]](#footnote-22). Additionally, we did perform a PCA in aiming to visualize the data.

In the plots below, you can see a sample of the data set labeled to clusters after the DBSCAN and labeled to outliers after our labeling base on the DBSCAN outcomes.

PCA visualization of the different clusters:



PCA visualization of the outliers:



After labeling the outliers, in order to decide if we want to exclude the outliers or just to label them, we did checked if there is a relation between the "defaulted" variable, which is our outcome variable, to the ***is\_out*** variable.

We did that by using Chi squared test to see if there is a correlation between the outliers and the outcome variable.

We found that is no significant correlation between the ***is\_out*** variable and the default variable. But it is not so far from being significant, as you can see in below:

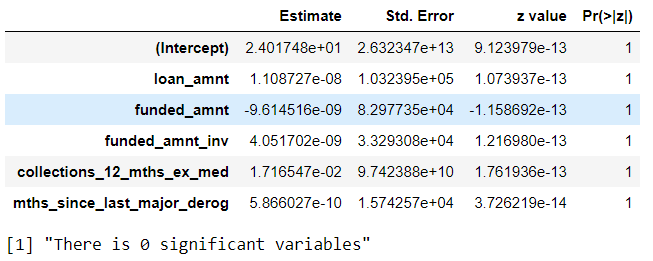
Pearson's Chi-squared test

data: k\_data$is\_out and k$default

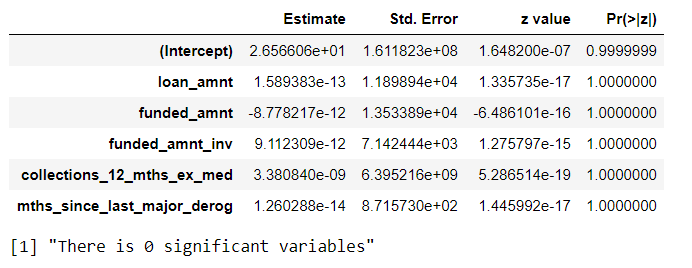
X-squared = 2.7631, df = 1, p-value = 0.09646

We also did checked if the relationship between the loans that labeled as outliers to our outcome variable are differ from the relation between the other loans to our outcome variable and from the whole data set that include both the outliers and not outliers. We did that by using logistic regression on the three cases:

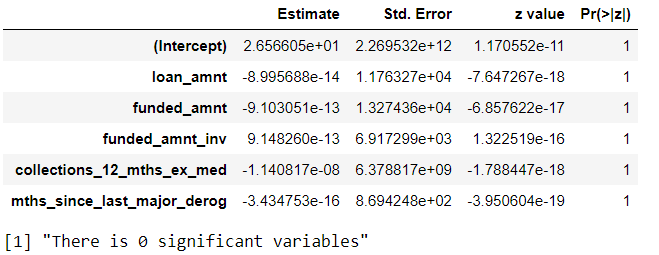
First, on the loans labeled as outliers versus the outcome variable,



Second, on the data that didn’t labeled as outliers versus the outcome variable,



And third, on the whole data set.

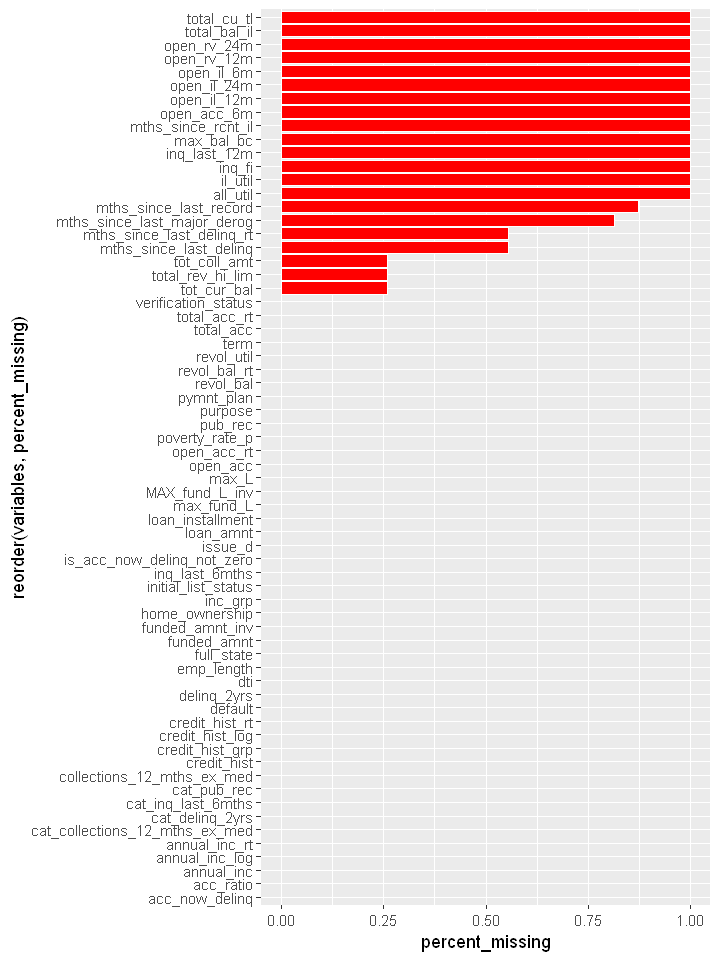


We didn’t find any significant coefficients in all cases that described above, hence we concluded that excluding the outliers don’t changed the relations between the outcome variable and the others variables and we drop those rows from our dataset.

#### **Dealing with missing values**

We started our treatment by getting all the variables which have missing rows and find which variables are full.

Figure 4 - Missing Variables



For each missing values variable we determined what type of missing mechanism it belongs. To do that we generated dummy variable for each of the missing values variables that will indicate if it is a full or missing. Then we ran a logistic regression with all the full data (data with only the full variables) against this dummy variable as outcome (y).

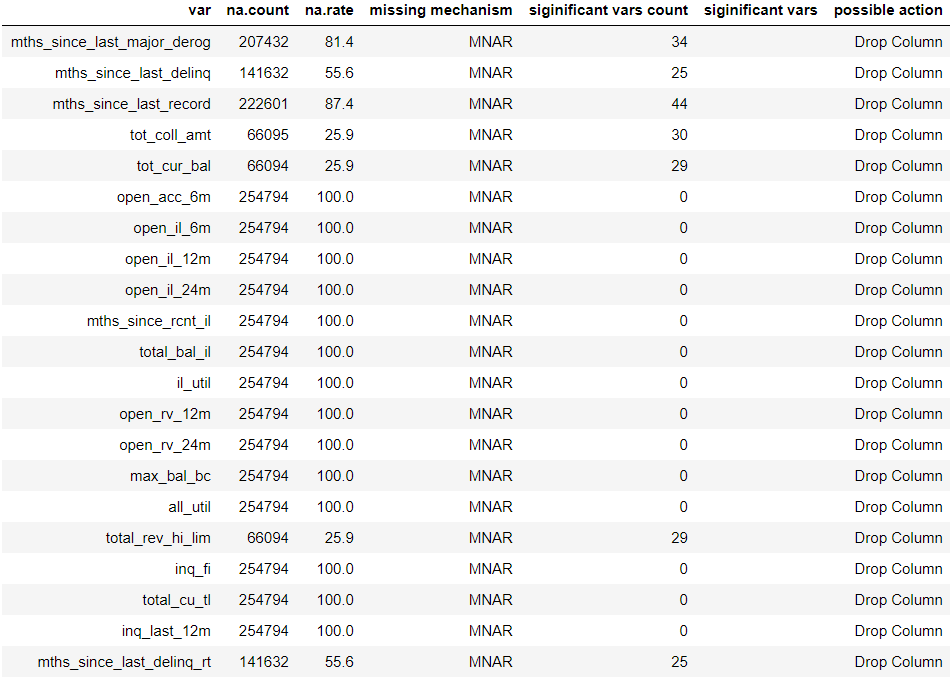
In the result model we checked the coefficients and see how much of them are significant (p-value<0.05).

If there are 0 significant coefficients than it is MCAR mechanism, if there are more than 1 then we say it is MNAR.

In case of only one significant coefficient we will need to examine further the relationship of this variable with the missing variable and determine if it is a MCAR or MAR mechanism.

After the determination of the mechanism we decided on the treatment. For variables of MNAR mechanism we checked the NA ratio of the data. If it is lower than 2-3% then we deleted the rows with NA values, otherwise we removed this variable from our dataset.

We wrote and ran a function that implement the process described above and put it in table format with the results for each variable.



We saw that there are variables that have 0 significant variables defined as MNAR , that because they have a 100% NA rate so we can't do imputation on them, therefore we drop all the columns that have missing values.

### Feature Selection

After the cleansing stages we have a dataset with all relevant features, original and engineered and doesn’t contain missing values. To prepare it further for modeling we encoded the categorical variables. For categories with more than 2 values we used one hot encoding which creates for each category variable dummy columns indicating which category it belongs to. After the encoding we ended with 138 columns.

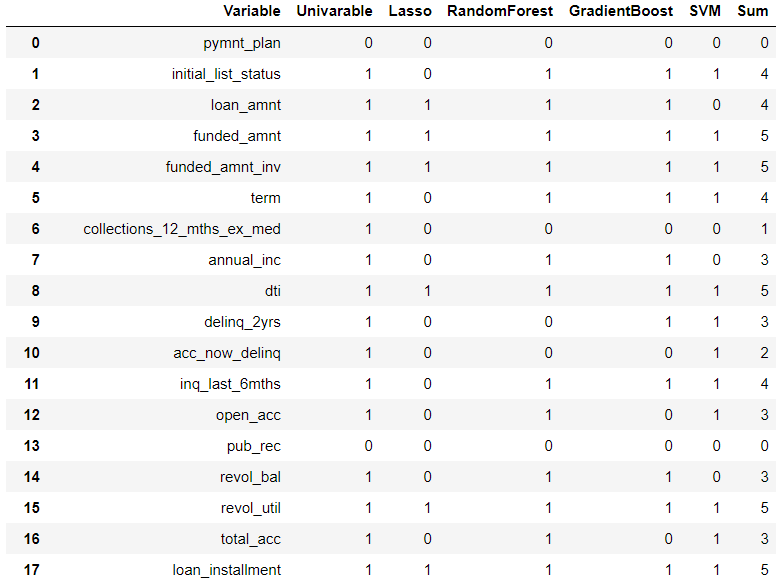
For the process of feature selection we did univariate and multivariate analysis. We created  
a table with all the variables on the dataset and indicate the recommended variables for each  
technique analysis.

For the univariate analysis we used again the table one function which show us the variable p-values. For each variable with p-value lower than 0.05 we set 1 on the variable in the table.

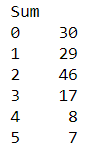
For the multivariate analysis we used 4 different predictive models that recommend the features which have influence in the model.

1. LASSO (L1 penalization)
2. Random Forest
3. Gradient Boosting classification
4. SVM classification

We ended with the table of variables and indication for each technique analysis and summarization column that sums all the "voting" for each variable.



By this table we saw that we have a majority of variables that got 2 votes.



Therefore we selected a threshold of 2 votes and on this basis we select the variables that will be used as our final dataset for the model. We left 78 variables that are listed in the LendingClub Data Retrieval Protocol.xlsx in the appendix.

## Models

### Train, dev (cross validation) and test

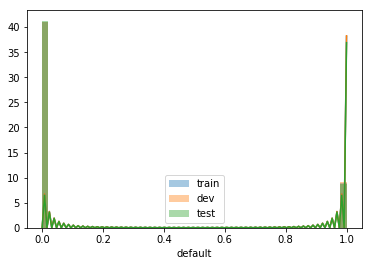
In order to develop predictive models we split our dataset into 3 subsets.

We split the data to:

* **train**, sample of 80% of our data to train and fit the models
* **dev**, sample of 10% of our data to evaluate our train model and allow us to tune the model using hyper parameters of the specific model.
* **test**, sample of 10% of our data to evaluate our final model.

Those 3 datasets have to be divided randomly and have the same distributions. To guaranty it we used shuffling of our index on the range of our dataset and random seed number which is saved afterwards to enable reproducing the split.

For each split we tested the 3 datasets with table one function and checking if there is no significant variables in each dataset that can cause bias. In our case we needed 4 tries to reach balanced split. To make sure we have good split we put our outcome variable (y) for each subset in distplot and see their distribution. We see that there is a good overlap between the 3 outcomes.



### Treating the Imbalance

From the EDA stage we already know that we have imbalance in our data. In our train dataset the proportion between 0 and 1 is one to 4.5.

|  |  |
| --- | --- |
|  |  |

So we decided to check the performance of the dataset with simple logistic regression and then check the performance of the model of datasets that was enhanced with resampling methods.

In the following table there are the results of those tests.

|  |  |  |  |
| --- | --- | --- | --- |
| Sample | Outcome 0 | Outcome 1 | Model Score |
| Original Train | 167,067 | 36,769 | 0.655611068904266 |
| Random Under Sampling | 36,769 | 36,769 | 0.65048343093457 |
| Random Over Sampling | 167,067 | 167,067 | 0.628807012113266 |
| Tomek links Under Sampling | 101,941 | 101,895 | 0.630247736915615 |
| SMOTE Over Sampling | 101,941 | 101,895 | 0.646898756617175 |

We saw that the original train dataset has the best score therefore we continued working without applying any of the resampling techniques.

### Model selection

By our outcome definition, as described in previous articles, we have a binary classification problem.

Hence we used machine learning models that fit to classifications, in order of solving it.

We tried the following models:

1. Logistic regression
2. Naïve Bayes
3. Conditional decision tree
4. Random Forest
5. AdaBoost
6. XGBoost
7. Linear SVM

We trained each model on the train data and calculating their AUC based on the dev (cross-validation) data.

After, we compared all the algorithms by their AUC score, in order to choose the best fit model.

It should be noted that we choose the AUC as our comparison index because of three primary reasons:

* 1. Our problem is a binary classification problem and the AUC is well suited for this type of problems.
  2. We are interesting in producing model that perform well producing probabilities and not just a binary outcomes. The AUC is given us a clue about that performance.
  3. The AUC is not affected by class imbalance (Fawcett, 2006)[[23]](#footnote-23), which our data is suffering from.

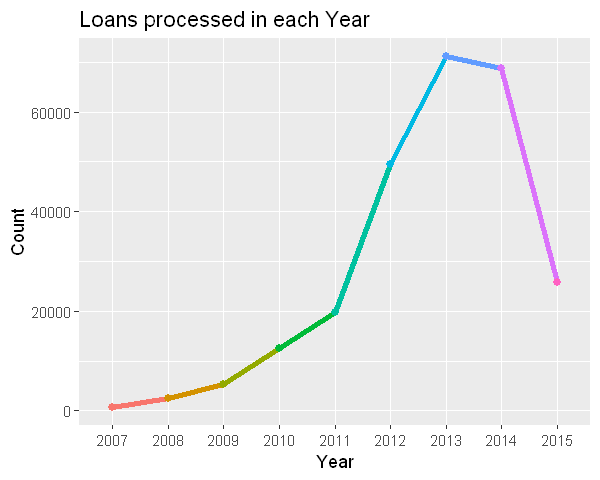
It is also nice to mention that when using normalized units (like in our case), the AUC equals the probability that a randomly chosen positive case receives a score higher than a randomly chosen negative case[[24]](#footnote-24).

# Results

## Data results

We started our process with 887,382 observations and 75 variables, as we gone through the processes of data exploration and definition of our outcome variable we left with 255,720 observations and 48 variables.

Our dataset distribution of loans over the year 2007 – 2015 as shown in the graph below:



In the feature engineering stage we enriched out dataset with additional variables by way of transformation, grouping and external data ending with 62 variables.

We treated outliers on 2 stages, one for univariate analysis and second multivariate analysis with DBScan algorithm. We removed 926 observations that was identified as outliers, added some more transformed variables ended with 254,794 observations and 66 variables.

We already noted that we started with a missing dataset therefore in missing treatment stage we saw that all of the variables that have missing values are belong to MNAR mechanism and we removed 21 variables from our dataset.

As we reached the feature selection stage we scaled and transformed our categorical variables in order to be more fitted for modeling.

After that we used numerous methods to find the significant variables that we used for the final model assessment. All the stages described in detail in previous chapters and the descriptive statistics of the variables can be found in the data retrieval protocol (LendingClub Data Retrieval Protocol.xlsx).

In Summary:

|  |  |  |
| --- | --- | --- |
| Stage | # of observations | # of variables |
| Raw data | 887,382 | 75 |
| EDA | 255,720 | 48 |
| Feature Engineering | 255,720 | 62 |
| Outliers | 254,794 | 66 |
| Missing | 254,794 | 45 |
| Encoding | 254,794 | 138 |
| Feature Selection | 254,794 | 79 |

With our final dataset we continued and split the dataset into 3 subsets for train, dev and test.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Subset | # of observations | # of fully paid (default =0) | # of defaulted (default =1) | Default rate |
| Train | 203,836 | 167,067 | 36,769 | 0.18 |
| Dev | 25,479 | 20,869 | 4,610 | 0.18 |
| Test | 25,479 | 21,035 | 4,444 | 0.17 |

## Model Results

### Model Selection

In order to find our best predictive model we tried to train 7 different models.

All the models was trained with their default parameters and was validated on the dev dataset.

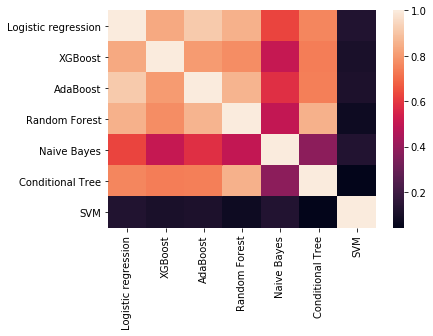
The performance measure was an AUC score.

The outcomes of the process described in the table below:

|  |  |  |
| --- | --- | --- |
| Ranking | Model classifier | AUC Score |
| 1 | Logistic regression | 0.689490 |
| 2 | XGBoost | 0.683265 |
| 3 | AdaBoost | 0.682977 |
| 4 | Random Forest | 0.673409 |
| 5 | Conditional Tree | 0.650171 |
| 6 | Naive Bayes | 0.624383 |
| 7 | Linear SVM | 0.521776 |

From the table above, on can see that logistic regression, XGBoost and AdaBoost algorithms has the highest AUC score.

We did consider to ensemble our models, therefore we did a correlation check between the performances of the different algorithms, as described in the correlation[[25]](#footnote-25) table below.



Because of the high correlation between the high-ranking models[[26]](#footnote-26) and due to cost-benefit[[27]](#footnote-27) considerations and time constraints, we decided not to ensemble models and leave it to a future work.

Instead, we moved on to perform a fine tuning to the highest rated models, XG and LR.

### Fine tuning

In this article we fine-tuned our best two models, LR and XGBoost.

#### Logistic Regression

We started with the Logistic Regression, which at the model selection stage gave use AUC of 0.6894901040048504, when testing it on the dev data with C=1 and penalty = L2.

The AUC score on the train data where 0.6942647732514526, hence the model is a bit over fit and not under fit.

We used greed search in order to try a different regulation types and values by changing the penalty (L1 and L2) hyper parameter and the C (0.001-100) hyper parameter, respectively.

It should be noted that even though the greed search function does divided the data to 5 folds, we use it on the train data and not on the whole data.

We did so because we were very careful when we divide the data to train, dev and test and we couldn’t be sure that the greed search function would produce the same quality of division.

As consequence of using the greed search, we found the best hyper parameter to be C=0.01 and penalty=L1.

We tried those hyper parameters on the train and the dev but unfortunately, we found that although the AUC result on the train improved with respect to the original model and stood at 0.6907755334747212, the result on the dev was only 0.6858102122225318, so that the over fitting problem only worsened and therefore the base model, was preferable.

We also tried to hyper tune the hyper parameters manually, by training models with different hyper parameters on the train data set and checking their result on the dev data set.

In this stage we also tried to change the different regulation types and values by changing the penalty (L1 and L2) hyper parameter and the C (1e-07-1000) hyper parameter, respectively.

As consequence of this stage we conclude that the best model is model that take l2 penalty and C=100.

This model AUC score on the dev was AUC: 0.689506579053363, which was the highest AUC score we got on the dev.

This model AUC score on the train was 0.6942683124670835, which is very similar to the original model AUC score on the train. Form that we can conclude that this fine-tuned model is a better model mainly due to the fact that it is less overfitting.

#### XGBoost

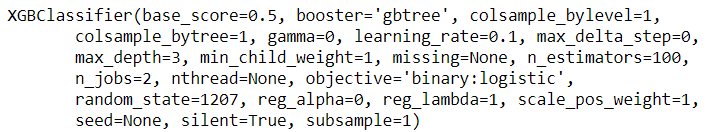
After, we finished tuning the LR we moved to fine tune the XGB.

The XGB gave use at the model selection stage AUC of 0.6832650199171383 on the dev data set. These result was given with the general direct library of XGB.

In order to tune more hyper parameters we began using XGBClassifier implementation of XGB.

The base model with XGBClassifier that has more hyper parameters already gave us a better results, AUC of 0.690246 on the dev data set therefore we continued to tuning in order to achieve better score.

Base model parameters:



In the beginning of the tuning process we ran on the base model parameters and tried to find the optimal number of trees (n\_estimators) and the learning rate for our train data.

We got the best results for learning rate = 0.1 and number of estimators in the range 300-1000 so we chose 500 as our optimal number.

We continued with the **tune of tree-specific parameters**:

* max\_depth & min\_child\_weight

For max\_depth we chose the range from 3 to 8 and for min\_child\_weight 1 to 4.

Those parameters controls the depth of the tree and the minimum number of instances required in a child node. They are important to prevent overfitting.

The best result was for the max\_depth=6 and min\_child\_weight=2 but we noticed that for the values max\_depth=4 and min\_child\_weight=3 there as slight decline on the auc score but the difference between train and dev is much smaller than the best combination that might indicate overfitting so we decided to choose max\_depth=4 and min\_child\_weight=3 as the paramters for the model.

* Subsample & colsample\_bytree

The values for this parameters controls the number of samples (observations) and the number of features (variables) supplied to a tree.  
Typically, its values lie between (0.5-0.8) for the subsample and (0.5-0.9) for colsample\_bytree.

We also added the value 1 as we using the whole train data set.

We choose to use subsample =0.8 and colsample\_bytree=1 because it shows smaller difference between train and dev.

* gamma

Minimum loss reduction required to make a further partition on a leaf node of the tree. We tried some small values around 0 and some higher values like 1, 5 and 10. We got good result with gamma=0.2.

We also try to **tune** **the regularization parameters**:

* reg\_alpha

It controls L1 regularization (equivalent to Lasso regression) on weights and the default value is 0, so we tried some small values around 0 and some higher values like 1, 5 and 10.

* reg\_lambda

It controls L2 regularization (equivalent to Ridge regression) on weights and the default value is 1, so we tried some small values around 0 and some higher values like 1, 5 and 10.

For both parameters we chose the default values because they show lower auc difference between train and test.

After we combined the tuned parameters for the final model we reached an AUC score of 0.7041575954287301 on the dev dataset

#### XGBoost calibration

After the fine tuning of both the XGBoost and the LR, XGBoost gives the highest AUC score.

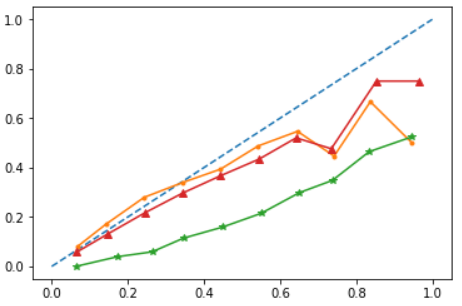
But, at the same time, it is known that while theoretically logistic regression produces a forecast that is reliable, XGB does not necessarily produce such a forecast.

Since the purpose of our project is ultimately to predict probabilities in order to estimate risk and not to execute a binary forecast[[28]](#footnote-28), it was important for us to make the forecast produced by the XGB more reliable.

For that reason, we did use a sigmoid and an isotonic methods, trying to make the basic XGBoost more reliable.

The following is a comparison between the model prior to calibrations, after calibrating with the isotonic method and the sigmoid method and the logistic regression results.

|  |  |  |
| --- | --- | --- |
|  | AUC score | Brier score |
| XGB \_ prior to calibration | 0.7041575954287301 | 0.1361149630695373 |
| XGB "sigmoid" calibration | 0.6997257761956649 | 0.14222660711065233 |
| XGB "isotonic" calibration | 0.7016643540965025 | 0.13933904717771425 |
| LR | 0.6895065790533635 | 0.13795208331363348 |



^ - XGB prior to calibration

. - XGB with "sigmoid" calibration

\* - XGB with "isotonic" calibration

We can see that the sigmoid and the isotonic gave very similar results.

Both models do give us less reliable results than the XGB prior to calibration, with a higher Brier score, and also showed a small decrease in AUC. Despite this decline, XGB still gave better results than the LR.

Therefore we chose to use the original XGB model, prior to calibration, to be our final model.

### Final model assessment

In this section, we ran the tuned XGB model that we determine as the final on our test data set.

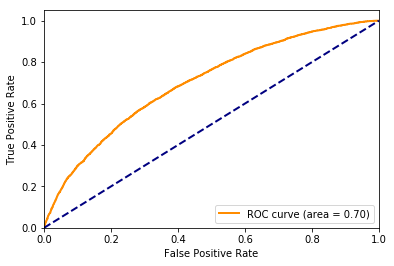
This, in order to evaluate our final results.

#### Model AUC score

Using the XGB chosen model on the test data, we got an AUC of 0.6988406393527397

Which is a small decline from the AUC of the dev data, which was 0.7041575954287301.

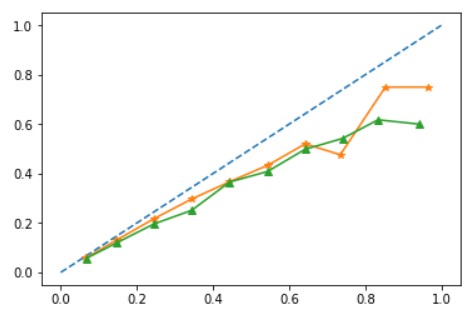
Is should be mentioned that there is a smaller gap between the AUC calculated on the dev to AUC calculated on test then between the AUC calculated on the dev to AUC calculated on train.



#### Model calibration

Using the XGB chosen model on the test data, we got an even more calibrated model with a Brayer score of 0.13272216490039845, less than the Brayer score calculated on the dev which was 0.1361149630695373.

In the graph below you can see how well the model is calibrated on the dev and test datasets.

****

\* - XGB on dev dataset

^ - XGB on test dataset

# Conclusion

In this project we analyzed Landing Club Issued Loans data with the aim of estimating the credit risk of each loan, independent of LC's estimation.

Although there are several approaches to estimate credit risk, in this work we studied only the probability of defaulted[[29]](#footnote-29) (PD) method, meaning estimating the probability to default of each given loan, by using classification algorithms to score each loan.

For that, we also defined set of baselines for comparison, such as being more successful than if we had guessed that all the loans would be paid and being more successful than previous works done on the same Lending Club data set.

We hoped that we would be able to create a good model against the baselines that we have set.

In aiming to achieve that goal, we first prepared and cleaned the data.

When doing so, we had to overcome number of challenges.

Perhaps the main challenge we had to deal with was the fact that the data we were working with was lacking compared to the full Landing Club data, including a lack of important features.

In particular, our data did not include data on FICO SCORE variable, which has been found in previous studies as highly predictive.

To handle this we did tried to enrich the data, both with an external data, like states poverty rates, and both with new features grouping and transformations, like application of a logistic transformation to the income variable.

Our data also included a lot of missing values relative to the full Lending Club data, but after checking the missing-ness mechanism we found it to be missing completely at random, hence we excluded all the missing data.

Another topic we had to relate to with respect to LC data is the imbalance problem which by their nature is suffering from.

We tried a various methods of balancing the data but after checking on the dev set, we found the original data to give the best results.

After the preparation and data cleansing, we tried 7 different models by training them on the train set and checking them on the dev set. After we did fine tuning of our two best models, XGBoost and Logistic Regression. As consequent we found XGBoost to be the best model with AUC=0.69 score.

This AUC score is clearly a success compering to the first baseline we have set, which was handling the imbalance and being more successful that if we had guessed that all the loans would be paid[[30]](#footnote-30).

This AUC score is a decent compering to the second baseline we have set, which was making good prediction comparing to some previous studies.

For example, (Michal Polena, 2016)[[31]](#footnote-31) achieved AUC of 0.684 but on more imbalance data (15.9% defaulted percent), Polena also mention Malekipirbazari & Aksakalli (2015)[[32]](#footnote-32) who achieved AUC of 0.71. Non-academic works also achieved respectable performance, as Imad Dabbura[[33]](#footnote-33) in his article at Towards Data Science, who got AUC=0.704, while working on the complete data set.

With that being said, it should be noted that when comparing our results to other works results, it is very important to pay attention to the fact that different studies based not on exactly the exact same data sets and also varies by defining the outcome variable.

For example, Michal Polena also defined loans with more than 90[[34]](#footnote-34) days late as defaulted, and not just loans with final status of Charged off.

And all the three studies mentioned above use the whole data from lending club site, data that also include FICO SCORE variable.

Therefore the comparison to other works should be used only to obtain an order of magnitude to our model's performance.

We believe that we succeed at getting fine results even though we had a lacking data, mainly due to 3 reasons:

1. We enriched the data with an external data.
2. We used XGBoost who gave use a bit better score from Logistic regression, while most academic researchers prefer using other algorithms and in many other works it didn’t considered to be used. That probably due to the fact it is a relative new algorithm.
3. We did clean the data from outliers, while many others studies didn’t.

Despite the fine result, it should be remembered that the model we created also has several limitations:

1. This model is based on data contains only information about people who received a loan from the LC. As a result, it is reasonable to assume that we will find in the data fewer people who are dangerous relative to their existence in the general distribution of the population.

Therefore, we must remember that our model is only good for assessing people who have passed the baseline criteria of LC.

This means that in order to correctly use our model, the user must first run filtering according to LC base criteria.

1. We saw that defaulted rates distribution from past years is different from the distribution in later years, even though it can happened from various reasons, we recommend to update this model frequently, and at least once in two years.
2. This model is not built for handling with missing data or with an outliers.

Hence, in case of missing required values or an outlier, as defined by the DBscan model that we created for outlier's detection, we recommend that the customer will not be evaluated by this model.

# Bibliography

Baesens, Bart & Van Gestel, Tony & Viaene, Stijn & STEPANOVA, M & Suykens, Johan & Vanthienen, Jan. (2003). Benchmarking state-of-the-art classification algorithms for credit scoring. Journal of the Operational Research Society. 54. 10.1057/palgrave.jors.2601545.

Lessmann, Stefan & Baesens, Bart & Seow, Hsin-Vonn & Thomas, Lyn. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. European Journal of Operational Research. (doi:10.1016/j.ejor.2015.

1. Riza Emekter, Yanbin Tu, Benjamas Jirasakuldech & Min Lu (2015) Evaluating credit risk and loan performance in online Peer-to-Peer (P2P) lending, Applied Economics, 47:1, 54-70, DOI: [10.1080/00036846.2014.962222](https://doi.org/10.1080/00036846.2014.962222)
2. Teplý, Petr & Polena, Michal. (2019). Best Classification Algorithms in Peer-to-Peer Lending. The North American Journal of Economics and Finance. 10.1016/j.najef.2019.01.001.
3. Riza Emekter, Yanbin Tu, Benjamas Jirasakuldech & Min Lu (2015) Evaluating credit risk and loan performance in online Peer-to-Peer (P2P) lending, Applied Economics, 47:1, 54-70, DOI: 10.1080/00036846.2014.962222.
4. Michal Polena, 2016, Performance Analysis of Credit Scoring Models on Lending Club Data, MASTER’S THESIS Supervise by Petr Tepl´y, Charles University ,Faculty of Social Sciences, Institute of Economic Studies.
5. Malekipirbazari, M. & V. Aksakalli (2015): ”Risk assessment in social lending via random forests.” Expert Systems with Applications 42(10): pp. 4621-4631.
6. https://towardsdatascience.com/predicting-loan-repayment-5df4e0023e92.

# Appendix

## Data Dictionary by Lending Club

|  |  |
| --- | --- |
| **LoanStatNew** | **Description** |
| acc\_now\_delinq | The number of accounts on which the borrower is now delinquent. |
| 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 |
| collection\_recovery\_fee | post charge off collection fee |
| 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 |
| desc | Loan description provided by the borrower |
| 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. |
| emp\_title | The job title supplied by the Borrower when applying for the loan.\* |
| funded\_amnt | The total amount committed to that loan at that point in time. |
| funded\_amnt\_inv | The total amount committed by investors for that loan at that point in time. |
| grade | LC assigned loan grade |
| home\_ownership | The home ownership status provided by the borrower during registration. Our values are: RENT, OWN, MORTGAGE, OTHER. |
| id | A unique LC assigned ID for the loan listing. |
| 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 |
| is\_inc\_v | Indicates if income was verified by LC, not verified, or if the income source was verified |
| issue\_d | The month which the loan was funded |
| last\_credit\_pull\_d | The most recent month LC pulled credit for this loan |
| last\_pymnt\_amnt | Last total payment amount received |
| last\_pymnt\_d | Last month payment was received |
| 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. |
| loan\_status | Current status of the loan |
| max\_bal\_bc | Maximum current balance owed on all revolving accounts |
| member\_id | A unique LC assigned Id for the borrower member. |
| 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 |
| next\_pymnt\_d | Next scheduled payment date |
| 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\_il\_6m | 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 |
| out\_prncp | Remaining outstanding principal for total amount funded |
| out\_prncp\_inv | Remaining outstanding principal for portion of total amount funded by investors |
| policy\_code | publicly available policy\_code=1 new products not publicly available policy\_code=2 |
| pub\_rec | Number of derogatory public records |
| purpose | A category provided by the borrower for the loan request. |
| pymnt\_plan | Indicates if a payment plan has been put in place for the loan |
| recoveries | post charge off gross recovery |
| 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. |
| sub\_grade | LC assigned loan subgrade |
| term | The number of payments on the loan. Values are in months and can be either 36 or 60. |
| title | The loan title provided by the borrower |
| tot\_coll\_amt | Total collection amounts ever owed |
| tot\_cur\_bal | Total current balance of all accounts |
| total\_acc | The total number of credit lines currently in the borrower's credit file |
| total\_bal\_il | Total current balance of all installment accounts |
| total\_cu\_tl | Number of finance trades |
| total\_pymnt | Payments received to date for total amount funded |
| total\_pymnt\_inv | Payments received to date for portion of total amount funded by investors |
| total\_rec\_int | Interest received to date |
| total\_rec\_late\_fee | Late fees received to date |
| total\_rec\_prncp | Principal received to date |
| total\_rev\_hi\_lim | Total revolving high credit/credit limit |
| url | URL for the LC page with listing data. |
| verified\_status\_joint | Indicates if the co-borrowers' joint income was verified by LC, not verified, or if the income source was verified |
| zip\_code | The first 3 numbers of the zip code provided by the borrower in the loan application. |

## Table - Table one

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| "V1" | "V2" | "Pop" | "0" | "1" | "pval" |
| "Individuals" | "n" | "255720" | "209711" | "46009" | NA |
| "verification\_status" | "Not Verified" | "87,897 (34.4%)" | "75,178 (35.8%)" | "12,719 (27.6%)" | NA |
| "verification\_status" | "Source Verified" | "74,300 (29.1%)" | "60,478 (28.8%)" | "13,822 (30.0%)" | "0" |
| "verification\_status" | "Verified" | "93,523 (36.6%)" | "74,055 (35.3%)" | "19,468 (42.3%)" | NA |
| "pymnt\_plan" | "1" | "2 (0.0%)" | "2 (0.0%)" | "0 (0.0%)" | "1" |
| "purpose" | "car" | "3,710 (1.5%)" | "3,249 (1.5%)" | "461 (1.0%)" | NA |
| "purpose" | "credit\_card" | "50,415 (19.7%)" | "42,520 (20.3%)" | "7,895 (17.2%)" | "0" |
| "purpose" | "debt\_consolidation" | "149,464 (58.4%)" | "121,573 (58.0%)" | "27,891 (60.6%)" | NA |
| "purpose" | "educational" | "422 (0.2%)" | "334 (0.2%)" | "88 (0.2%)" | NA |
| "purpose" | "home\_improvement" | "15,143 (5.9%)" | "12,803 (6.1%)" | "2,340 (5.1%)" | NA |
| "purpose" | "house" | "1,696 (0.7%)" | "1,399 (0.7%)" | "297 (0.6%)" | NA |
| "purpose" | "major\_purchase" | "6,388 (2.5%)" | "5,491 (2.6%)" | "897 (1.9%)" | NA |
| "purpose" | "medical" | "2,912 (1.1%)" | "2,321 (1.1%)" | "591 (1.3%)" | NA |
| "purpose" | "moving" | "2,074 (0.8%)" | "1,634 (0.8%)" | "440 (1.0%)" | NA |
| "purpose" | "other" | "14,701 (5.7%)" | "11,644 (5.6%)" | "3,057 (6.6%)" | NA |
| "purpose" | "renewable\_energy" | "270 (0.1%)" | "215 (0.1%)" | "55 (0.1%)" | NA |
| "purpose" | "small\_business" | "4,907 (1.9%)" | "3,464 (1.7%)" | "1,443 (3.1%)" | NA |
| "purpose" | "vacation" | "1,607 (0.6%)" | "1,331 (0.6%)" | "276 (0.6%)" | NA |
| "purpose" | "wedding" | "2,011 (0.8%)" | "1,733 (0.8%)" | "278 (0.6%)" | NA |
| "initial\_list\_status" | "1" | "184,828 (72.3%)" | "151,134 (72.1%)" | "33,694 (73.2%)" | "0" |
| "emp\_length" | "< 1 year" | "21,358 (8.4%)" | "17,395 (8.3%)" | "3,963 (8.6%)" | NA |
| "emp\_length" | "1 year" | "17,204 (6.7%)" | "14,149 (6.7%)" | "3,055 (6.6%)" | "0" |
| "emp\_length" | "10+ years" | "77,351 (30.2%)" | "64,062 (30.5%)" | "13,289 (28.9%)" | NA |
| "emp\_length" | "2 years" | "23,910 (9.4%)" | "19,794 (9.4%)" | "4,116 (8.9%)" | NA |
| "emp\_length" | "3 years" | "20,646 (8.1%)" | "17,040 (8.1%)" | "3,606 (7.8%)" | NA |
| "emp\_length" | "4 years" | "16,402 (6.4%)" | "13,571 (6.5%)" | "2,831 (6.2%)" | NA |
| "emp\_length" | "5 years" | "18,231 (7.1%)" | "14,978 (7.1%)" | "3,253 (7.1%)" | NA |
| "emp\_length" | "6 years" | "14,897 (5.8%)" | "12,159 (5.8%)" | "2,738 (6.0%)" | NA |
| "emp\_length" | "7 years" | "14,185 (5.5%)" | "11,551 (5.5%)" | "2,634 (5.7%)" | NA |
| "emp\_length" | "8 years" | "11,956 (4.7%)" | "9,770 (4.7%)" | "2,186 (4.8%)" | NA |
| "emp\_length" | "9 years" | "9,649 (3.8%)" | "7,851 (3.7%)" | "1,798 (3.9%)" | NA |
| "emp\_length" | "n/a" | "9,931 (3.9%)" | "7,391 (3.5%)" | "2,540 (5.5%)" | NA |
| "loan\_amnt" | "Mean (SD)" | "13,514.2 (8,127.1)" | "13,303.8 (8,053.3)" | "14,473.1 (8,388.9)" | NA |
| "loan\_amnt" | "Median (IQR)" | "12,000.0 (7,200.0-18,200.0)" | "12,000.0 (7,075.0-18,000.0)" | "12,500.0 (8,000.0-20,000.0)" | "0" |
| "funded\_amnt" | "Mean (SD)" | "13,471.8 (8,104.6)" | "13,263.0 (8,031.8)" | "14,423.7 (8,362.6)" | NA |
| "funded\_amnt" | "Median (IQR)" | "12,000.0 (7,200.0-18,000.0)" | "11,925.0 (7,000.0-18,000.0)" | "12,375.0 (8,000.0-20,000.0)" | "0" |
| "funded\_amnt\_inv" | "Mean (SD)" | "13,352.0 (8,140.1)" | "13,155.3 (8,063.4)" | "14,248.5 (8,423.2)" | NA |
| "funded\_amnt\_inv" | "Median (IQR)" | "11,950.0 (7,075.0-18,000.0)" | "11,550.0 (7,000.0-18,000.0)" | "12,050.0 (8,000.0-19,950.0)" | "0" |
| "term" | "1" | "56,624 (22.1%)" | "40,347 (19.2%)" | "16,277 (35.4%)" | "0" |
| "collections\_12\_mths\_ex\_med" | "Mean (SD)" | "0.0 (0.1)" | "0.0 (0.1)" | "0.0 (0.1)" | NA |
| "collections\_12\_mths\_ex\_med" | "Median (IQR)" | "0.0 (0.0-0.0)" | "0.0 (0.0-0.0)" | "0.0 (0.0-0.0)" | "0.001" |
| "collections\_12\_mths\_ex\_med" | "Missing (%)" | "145 (0.1%)" | "117 (0.1%)" | "28 (0.1%)" | NA |
| "mths\_since\_last\_major\_derog" | "Mean (SD)" | "43.4 (21.6)" | "43.6 (21.6)" | "42.6 (21.8)" | NA |
| "mths\_since\_last\_major\_derog" | "Median (IQR)" | "43.0 (26.0-60.0)" | "43.0 (27.0-60.0)" | "42.0 (25.0-60.0)" | "0" |
| "mths\_since\_last\_major\_derog" | "Missing (%)" | "208,175 (81.4%)" | "170,920 (81.5%)" | "37,255 (81.0%)" | NA |
| "home\_ownership" | "ANY" | "1 (0.0%)" | "1 (0.0%)" | "0 (0.0%)" | NA |
| "home\_ownership" | "MORTGAGE" | "126,100 (49.3%)" | "105,874 (50.5%)" | "20,226 (44.0%)" | "0" |
| "home\_ownership" | "NONE" | "48 (0.0%)" | "40 (0.0%)" | "8 (0.0%)" | NA |
| "home\_ownership" | "OTHER" | "179 (0.1%)" | "141 (0.1%)" | "38 (0.1%)" | NA |
| "home\_ownership" | "OWN" | "22,173 (8.7%)" | "18,099 (8.6%)" | "4,074 (8.9%)" | NA |
| "home\_ownership" | "RENT" | "107,219 (41.9%)" | "85,556 (40.8%)" | "21,663 (47.1%)" | NA |
| "annual\_inc" | "Mean (SD)" | "72,526.3 (58,926.9)" | "74,142.8 (59,205.7)" | "65,158.6 (57,062.6)" | NA |
| "annual\_inc" | "Median (IQR)" | "62,000.0 (45,000.0-87,000.0)" | "64,000.0 (45,000.0-90,000.0)" | "56,000.0 (40,000.0-78,000.0)" | "0" |
| "annual\_inc" | "Missing (%)" | "4 (0.0%)" | "4 (0.0%)" | NA | NA |
| "dti" | "Mean (SD)" | "16.5 (7.8)" | "16.1 (7.7)" | "18.3 (7.9)" | NA |
| "dti" | "Median (IQR)" | "16.2 (10.7-22.0)" | "15.7 (10.4-21.5)" | "18.3 (12.5-24.0)" | "0" |
| "delinq\_2yrs" | "Mean (SD)" | "0.2 (0.7)" | "0.2 (0.7)" | "0.3 (0.8)" | NA |
| "delinq\_2yrs" | "Median (IQR)" | "0.0 (0.0-0.0)" | "0.0 (0.0-0.0)" | "0.0 (0.0-0.0)" | "0" |
| "delinq\_2yrs" | "Missing (%)" | "29 (0.0%)" | "26 (0.0%)" | "3 (0.0%)" | NA |
| "acc\_now\_delinq" | "Mean (SD)" | "0.0 (0.1)" | "0.0 (0.1)" | "0.0 (0.1)" | NA |
| "acc\_now\_delinq" | "Median (IQR)" | "0.0 (0.0-0.0)" | "0.0 (0.0-0.0)" | "0.0 (0.0-0.0)" | "0.009" |
| "acc\_now\_delinq" | "Missing (%)" | "29 (0.0%)" | "26 (0.0%)" | "3 (0.0%)" | NA |
| "inq\_last\_6mths" | "Mean (SD)" | "0.9 (1.2)" | "0.9 (1.1)" | "1.0 (1.3)" | NA |
| "inq\_last\_6mths" | "Median (IQR)" | "1.0 (0.0-1.0)" | "1.0 (0.0-1.0)" | "1.0 (0.0-2.0)" | "0" |
| "inq\_last\_6mths" | "Missing (%)" | "29 (0.0%)" | "26 (0.0%)" | "3 (0.0%)" | NA |
| "mths\_since\_last\_delinq" | "Mean (SD)" | "35.1 (21.9)" | "35.3 (21.8)" | "34.1 (22.1)" | NA |
| "mths\_since\_last\_delinq" | "Median (IQR)" | "32.0 (17.0-51.0)" | "33.0 (17.0-51.0)" | "31.0 (15.0-50.0)" | "0" |
| "mths\_since\_last\_delinq" | "Missing (%)" | "142,018 (55.5%)" | "116,661 (55.6%)" | "25,357 (55.1%)" | NA |
| "mths\_since\_last\_record" | "Mean (SD)" | "74.4 (31.1)" | "74.0 (30.7)" | "76.0 (32.5)" | NA |
| "mths\_since\_last\_record" | "Median (IQR)" | "78.0 (54.0-101.0)" | "76.0 (53.0-101.0)" | "84.0 (55.0-103.0)" | "0" |
| "mths\_since\_last\_record" | "Missing (%)" | "223,262 (87.3%)" | "183,178 (87.3%)" | "40,084 (87.1%)" | NA |
| "open\_acc" | "Mean (SD)" | "10.9 (4.9)" | "10.9 (4.9)" | "11.0 (4.9)" | NA |
| "open\_acc" | "Median (IQR)" | "10.0 (7.0-14.0)" | "10.0 (7.0-13.0)" | "10.0 (8.0-14.0)" | "0" |
| "open\_acc" | "Missing (%)" | "29 (0.0%)" | "26 (0.0%)" | "3 (0.0%)" | NA |
| "pub\_rec" | "Mean (SD)" | "0.1 (0.4)" | "0.1 (0.4)" | "0.1 (0.4)" | NA |
| "pub\_rec" | "Median (IQR)" | "0.0 (0.0-0.0)" | "0.0 (0.0-0.0)" | "0.0 (0.0-0.0)" | "0.364" |
| "pub\_rec" | "Missing (%)" | "29 (0.0%)" | "26 (0.0%)" | "3 (0.0%)" | NA |
| "revol\_bal" | "Mean (SD)" | "15,299.6 (19,709.8)" | "15,326.0 (19,776.9)" | "15,179.0 (19,400.6)" | NA |
| "revol\_bal" | "Median (IQR)" | "10,917.0 (5,831.0-19,082.0)" | "10,833.0 (5,759.0-19,058.5)" | "11,302.0 (6,170.0-19,213.0)" | "0.142" |
| "revol\_util" | "Mean (SD)" | "54.3 (24.8)" | "53.2 (25.0)" | "59.4 (23.6)" | NA |
| "revol\_util" | "Median (IQR)" | "55.8 (36.2-73.9)" | "54.5 (34.8-72.8)" | "61.6 (43.1-78.0)" | "0" |
| "revol\_util" | "Missing (%)" | "239 (0.1%)" | "186 (0.1%)" | "53 (0.1%)" | NA |
| "total\_acc" | "Mean (SD)" | "25.0 (11.8)" | "25.2 (11.8)" | "24.2 (11.6)" | NA |
| "total\_acc" | "Median (IQR)" | "23.0 (16.0-32.0)" | "24.0 (17.0-32.0)" | "23.0 (16.0-31.0)" | "0" |
| "total\_acc" | "Missing (%)" | "29 (0.0%)" | "26 (0.0%)" | "3 (0.0%)" | NA |
| "tot\_coll\_amt" | "Mean (SD)" | "203.8 (21,102.1)" | "219.8 (23,382.5)" | "133.9 (1,316.1)" | NA |
| "tot\_coll\_amt" | "Median (IQR)" | "0.0 (0.0-0.0)" | "0.0 (0.0-0.0)" | "0.0 (0.0-0.0)" | "0.152" |
| "tot\_coll\_amt" | "Missing (%)" | "66,457 (26.0%)" | "55,677 (26.5%)" | "10,780 (23.4%)" | NA |
| "tot\_cur\_bal" | "Mean (SD)" | "138,331.2 (152,436.7)" | "143,791.9 (157,054.2)" | "114,454.8 (127,626.4)" | NA |
| "tot\_cur\_bal" | "Median (IQR)" | "81,002.0 (28,372.5-208,229.0)" | "87,779.0 (29,146.2-216,689.8)" | "59,738.0 (25,670.0-172,573.0)" | "0" |
| "tot\_cur\_bal" | "Missing (%)" | "66,457 (26.0%)" | "55,677 (26.5%)" | "10,780 (23.4%)" | NA |
| "open\_acc\_6m" | "Mean (SD)" | "1.4 (1.3)" | "1.4 (1.3)" | "NaN (NA)" | NA |
| "open\_acc\_6m" | "Median (IQR)" | "1.0 (0.0-2.0)" | "1.0 (0.0-2.0)" | "NA (NA-NA)" | "0" |
| "open\_acc\_6m" | "Missing (%)" | "255,576 (99.9%)" | "209,567 (99.9%)" | "46,009 (100.0%)" | NA |
| "open\_il\_6m" | "Mean (SD)" | "3.1 (3.2)" | "3.1 (3.2)" | "NaN (NA)" | NA |
| "open\_il\_6m" | "Median (IQR)" | "2.0 (1.0-4.0)" | "2.0 (1.0-4.0)" | "NA (NA-NA)" | "0" |
| "open\_il\_6m" | "Missing (%)" | "255,576 (99.9%)" | "209,567 (99.9%)" | "46,009 (100.0%)" | NA |
| "open\_il\_12m" | "Mean (SD)" | "0.9 (1.2)" | "0.9 (1.2)" | "NaN (NA)" | NA |
| "open\_il\_12m" | "Median (IQR)" | "0.0 (0.0-1.0)" | "0.0 (0.0-1.0)" | "NA (NA-NA)" | "0" |
| "open\_il\_12m" | "Missing (%)" | "255,576 (99.9%)" | "209,567 (99.9%)" | "46,009 (100.0%)" | NA |
| "open\_il\_24m" | "Mean (SD)" | "1.9 (2.1)" | "1.9 (2.1)" | "NaN (NA)" | NA |
| "open\_il\_24m" | "Median (IQR)" | "1.0 (1.0-3.0)" | "1.0 (1.0-3.0)" | "NA (NA-NA)" | "0" |
| "open\_il\_24m" | "Missing (%)" | "255,576 (99.9%)" | "209,567 (99.9%)" | "46,009 (100.0%)" | NA |
| "mths\_since\_rcnt\_il" | "Mean (SD)" | "20.7 (27.7)" | "20.7 (27.7)" | "NaN (NA)" | NA |
| "mths\_since\_rcnt\_il" | "Median (IQR)" | "12.5 (4.0-21.0)" | "12.5 (4.0-21.0)" | "NA (NA-NA)" | "0" |
| "mths\_since\_rcnt\_il" | "Missing (%)" | "255,580 (99.9%)" | "209,571 (99.9%)" | "46,009 (100.0%)" | NA |
| "total\_bal\_il" | "Mean (SD)" | "36,462.3 (38,137.6)" | "36,462.3 (38,137.6)" | "NaN (NA)" | NA |
| "total\_bal\_il" | "Median (IQR)" | "23,605.0 (10,763.8-50,835.8)" | "23,605.0 (10,763.8-50,835.8)" | "NA (NA-NA)" | "0" |
| "total\_bal\_il" | "Missing (%)" | "255,576 (99.9%)" | "209,567 (99.9%)" | "46,009 (100.0%)" | NA |
| "il\_util" | "Mean (SD)" | "73.7 (23.1)" | "73.7 (23.1)" | "NaN (NA)" | NA |
| "il\_util" | "Median (IQR)" | "77.0 (63.3-88.2)" | "77.0 (63.3-88.2)" | "NA (NA-NA)" | "0" |
| "il\_util" | "Missing (%)" | "255,594 (100.0%)" | "209,585 (99.9%)" | "46,009 (100.0%)" | NA |
| "open\_rv\_12m" | "Mean (SD)" | "1.7 (1.6)" | "1.7 (1.6)" | "NaN (NA)" | NA |
| "open\_rv\_12m" | "Median (IQR)" | "1.0 (0.0-2.0)" | "1.0 (0.0-2.0)" | "NA (NA-NA)" | "0" |
| "open\_rv\_12m" | "Missing (%)" | "255,576 (99.9%)" | "209,567 (99.9%)" | "46,009 (100.0%)" | NA |
| "open\_rv\_24m" | "Mean (SD)" | "3.6 (2.9)" | "3.6 (2.9)" | "NaN (NA)" | NA |
| "open\_rv\_24m" | "Median (IQR)" | "3.0 (2.0-5.0)" | "3.0 (2.0-5.0)" | "NA (NA-NA)" | "0" |
| "open\_rv\_24m" | "Missing (%)" | "255,576 (99.9%)" | "209,567 (99.9%)" | "46,009 (100.0%)" | NA |
| "max\_bal\_bc" | "Mean (SD)" | "5,517.3 (4,507.6)" | "5,517.3 (4,507.6)" | "NaN (NA)" | NA |
| "max\_bal\_bc" | "Median (IQR)" | "4,511.5 (2,091.2-8,212.8)" | "4,511.5 (2,091.2-8,212.8)" | "NA (NA-NA)" | "0" |
| "max\_bal\_bc" | "Missing (%)" | "255,576 (99.9%)" | "209,567 (99.9%)" | "46,009 (100.0%)" | NA |
| "all\_util" | "Mean (SD)" | "58.9 (20.4)" | "58.9 (20.4)" | "NaN (NA)" | NA |
| "all\_util" | "Median (IQR)" | "62.0 (46.8-74.2)" | "62.0 (46.8-74.2)" | "NA (NA-NA)" | "0" |
| "all\_util" | "Missing (%)" | "255,576 (99.9%)" | "209,567 (99.9%)" | "46,009 (100.0%)" | NA |
| "total\_rev\_hi\_lim" | "Mean (SD)" | "29,709.1 (29,517.3)" | "30,473.2 (30,332.8)" | "26,368.6 (25,380.7)" | NA |
| "total\_rev\_hi\_lim" | "Median (IQR)" | "22,300.0 (13,300.0-36,800.0)" | "22,700.0 (13,500.0-37,800.0)" | "20,500.0 (12,400.0-33,200.0)" | "0" |
| "total\_rev\_hi\_lim" | "Missing (%)" | "66,457 (26.0%)" | "55,677 (26.5%)" | "10,780 (23.4%)" | NA |
| "inq\_fi" | "Mean (SD)" | "1.3 (1.8)" | "1.3 (1.8)" | "NaN (NA)" | NA |
| "inq\_fi" | "Median (IQR)" | "1.0 (0.0-1.0)" | "1.0 (0.0-1.0)" | "NA (NA-NA)" | "0" |
| "inq\_fi" | "Missing (%)" | "255,576 (99.9%)" | "209,567 (99.9%)" | "46,009 (100.0%)" | NA |
| "total\_cu\_tl" | "Mean (SD)" | "2.1 (3.9)" | "2.1 (3.9)" | "NaN (NA)" | NA |
| "total\_cu\_tl" | "Median (IQR)" | "0.0 (0.0-2.0)" | "0.0 (0.0-2.0)" | "NA (NA-NA)" | "0" |
| "total\_cu\_tl" | "Missing (%)" | "255,576 (99.9%)" | "209,567 (99.9%)" | "46,009 (100.0%)" | NA |
| "inq\_last\_12m" | "Mean (SD)" | "2.6 (4.1)" | "2.6 (4.1)" | "NaN (NA)" | NA |
| "inq\_last\_12m" | "Median (IQR)" | "2.0 (1.0-4.0)" | "2.0 (1.0-4.0)" | "NA (NA-NA)" | "0" |
| "inq\_last\_12m" | "Missing (%)" | "255,576 (99.9%)" | "209,567 (99.9%)" | "46,009 (100.0%)" | NA |
| "loan\_installment" | "Mean (SD)" | "327.1 (190.1)" | "327.6 (192.0)" | "324.9 (180.8)" | NA |
| "loan\_installment" | "Median (IQR)" | "277.8 (194.4-416.7)" | "277.8 (193.1-416.7)" | "283.3 (200.0-416.7)" | "0.003" |
| "full\_state" | "Alabama" | "3,179 (1.2%)" | "2,509 (1.2%)" | "670 (1.5%)" | NA |
| "full\_state" | "Alaska" | "668 (0.3%)" | "571 (0.3%)" | "97 (0.2%)" | "0.003" |
| "full\_state" | "Arizona" | "6,128 (2.4%)" | "5,061 (2.4%)" | "1,067 (2.3%)" | NA |
| "full\_state" | "Arkansas" | "1,769 (0.7%)" | "1,426 (0.7%)" | "343 (0.7%)" | NA |
| "full\_state" | "California" | "43,434 (17.0%)" | "36,001 (17.2%)" | "7,433 (16.2%)" | NA |
| "full\_state" | "Colorado" | "5,678 (2.2%)" | "4,881 (2.3%)" | "797 (1.7%)" | NA |
| "full\_state" | "Connecticut" | "3,743 (1.5%)" | "3,117 (1.5%)" | "626 (1.4%)" | NA |
| "full\_state" | "Delaware" | "689 (0.3%)" | "564 (0.3%)" | "125 (0.3%)" | NA |
| "full\_state" | "District of Columbia" | "847 (0.3%)" | "758 (0.4%)" | "89 (0.2%)" | NA |
| "full\_state" | "Florida" | "17,776 (7.0%)" | "14,180 (6.8%)" | "3,596 (7.8%)" | NA |
| "full\_state" | "Georgia" | "8,118 (3.2%)" | "6,723 (3.2%)" | "1,395 (3.0%)" | NA |
| "full\_state" | "Hawaii" | "1,485 (0.6%)" | "1,207 (0.6%)" | "278 (0.6%)" | NA |
| "full\_state" | "Idaho" | "9 (0.0%)" | "8 (0.0%)" | "1 (0.0%)" | NA |
| "full\_state" | "Illinois" | "9,400 (3.7%)" | "7,822 (3.7%)" | "1,578 (3.4%)" | NA |
| "full\_state" | "Indiana" | "2,810 (1.1%)" | "2,177 (1.0%)" | "633 (1.4%)" | NA |
| "full\_state" | "Iowa" | "13 (0.0%)" | "10 (0.0%)" | "3 (0.0%)" | NA |
| "full\_state" | "Kansas" | "2,109 (0.8%)" | "1,748 (0.8%)" | "361 (0.8%)" | NA |
| "full\_state" | "Kentucky" | "2,300 (0.9%)" | "1,854 (0.9%)" | "446 (1.0%)" | NA |
| "full\_state" | "Louisiana" | "2,977 (1.2%)" | "2,406 (1.1%)" | "571 (1.2%)" | NA |
| "full\_state" | "Maine" | "13 (0.0%)" | "13 (0.0%)" | "0 (0.0%)" | NA |
| "full\_state" | "Maryland" | "6,077 (2.4%)" | "4,954 (2.4%)" | "1,123 (2.4%)" | NA |
| "full\_state" | "Massachusetts" | "6,234 (2.4%)" | "5,193 (2.5%)" | "1,041 (2.3%)" | NA |
| "full\_state" | "Michigan" | "6,070 (2.4%)" | "4,901 (2.3%)" | "1,169 (2.5%)" | NA |
| "full\_state" | "Minnesota" | "4,496 (1.8%)" | "3,682 (1.8%)" | "814 (1.8%)" | NA |
| "full\_state" | "Mississippi" | "428 (0.2%)" | "338 (0.2%)" | "90 (0.2%)" | NA |
| "full\_state" | "Missouri" | "4,033 (1.6%)" | "3,226 (1.5%)" | "807 (1.8%)" | NA |
| "full\_state" | "Montana" | "747 (0.3%)" | "646 (0.3%)" | "101 (0.2%)" | NA |
| "full\_state" | "Nebraska" | "44 (0.0%)" | "37 (0.0%)" | "7 (0.0%)" | NA |
| "full\_state" | "Nevada" | "3,838 (1.5%)" | "3,019 (1.4%)" | "819 (1.8%)" | NA |
| "full\_state" | "New Hampshire" | "1,164 (0.5%)" | "1,005 (0.5%)" | "159 (0.3%)" | NA |
| "full\_state" | "New Jersey" | "9,734 (3.8%)" | "7,867 (3.8%)" | "1,867 (4.1%)" | NA |
| "full\_state" | "New Mexico" | "1,392 (0.5%)" | "1,120 (0.5%)" | "272 (0.6%)" | NA |
| "full\_state" | "New York" | "21,586 (8.4%)" | "17,405 (8.3%)" | "4,181 (9.1%)" | NA |
| "full\_state" | "North Carolina" | "6,960 (2.7%)" | "5,642 (2.7%)" | "1,318 (2.9%)" | NA |
| "full\_state" | "North Dakota" | "8 (0.0%)" | "8 (0.0%)" | "0 (0.0%)" | NA |
| "full\_state" | "Ohio" | "7,841 (3.1%)" | "6,351 (3.0%)" | "1,490 (3.2%)" | NA |
| "full\_state" | "Oklahoma" | "2,148 (0.8%)" | "1,724 (0.8%)" | "424 (0.9%)" | NA |
| "full\_state" | "Oregon" | "3,370 (1.3%)" | "2,820 (1.3%)" | "550 (1.2%)" | NA |
| "full\_state" | "Pennsylvania" | "8,531 (3.3%)" | "6,931 (3.3%)" | "1,600 (3.5%)" | NA |
| "full\_state" | "Rhode Island" | "1,096 (0.4%)" | "903 (0.4%)" | "193 (0.4%)" | NA |
| "full\_state" | "South Carolina" | "2,835 (1.1%)" | "2,382 (1.1%)" | "453 (1.0%)" | NA |
| "full\_state" | "South Dakota" | "547 (0.2%)" | "456 (0.2%)" | "91 (0.2%)" | NA |
| "full\_state" | "Tennessee" | "2,441 (1.0%)" | "1,874 (0.9%)" | "567 (1.2%)" | NA |
| "full\_state" | "Texas" | "19,524 (7.6%)" | "16,434 (7.8%)" | "3,090 (6.7%)" | NA |
| "full\_state" | "Utah" | "2,131 (0.8%)" | "1,777 (0.8%)" | "354 (0.8%)" | NA |
| "full\_state" | "Vermont" | "431 (0.2%)" | "360 (0.2%)" | "71 (0.2%)" | NA |
| "full\_state" | "Virginia" | "8,021 (3.1%)" | "6,567 (3.1%)" | "1,454 (3.2%)" | NA |
| "full\_state" | "Washington" | "5,961 (2.3%)" | "4,959 (2.4%)" | "1,002 (2.2%)" | NA |
| "full\_state" | "West Virginia" | "1,147 (0.4%)" | "986 (0.5%)" | "161 (0.3%)" | NA |
| "full\_state" | "Wisconsin" | "3,134 (1.2%)" | "2,584 (1.2%)" | "550 (1.2%)" | NA |
| "full\_state" | "Wyoming" | "606 (0.2%)" | "524 (0.2%)" | "82 (0.2%)" | NA |
| "acc\_ratio" | "Mean (SD)" | "0.5 (0.2)" | "0.5 (0.2)" | "0.5 (0.2)" | NA |
| "acc\_ratio" | "Median (IQR)" | "0.5 (0.4-0.6)" | "0.5 (0.3-0.6)" | "0.5 (0.4-0.6)" | "0" |
| "acc\_ratio" | "Missing (%)" | "29 (0.0%)" | "26 (0.0%)" | "3 (0.0%)" | NA |
| "credit\_hist" | "Mean (SD)" | "17.4 (7.1)" | "17.5 (7.0)" | "17.0 (7.2)" | NA |
| "credit\_hist" | "Median (IQR)" | "16.0 (13.0-21.0)" | "16.0 (13.0-21.0)" | "16.0 (12.0-20.0)" | "0" |
| "credit\_hist" | "Missing (%)" | "51 (0.0%)" | "42 (0.0%)" | "9 (0.0%)" | NA |
| "poverty\_rate\_p" | "Mean (SD)" | "0.1 (0.0)" | "0.1 (0.0)" | "0.1 (0.0)" | NA |
| "poverty\_rate\_p" | "Median (IQR)" | "0.1 (0.1-0.2)" | "0.1 (0.1-0.2)" | "0.1 (0.1-0.2)" | "0" |

1. Lending Club is a P2P lending company whom connected between people wishing to lend money to people who wish to borrow it, in the American market. Lending Club established at 2007 in California, USA ([Company website](https://www.lendingclub.com/company/about-us), [wiki page](https://en.wikipedia.org/wiki/Lending_Club) ). [↑](#footnote-ref-1)
2. Baesens, Bart & Van Gestel, Tony & Viaene, Stijn & STEPANOVA, M & Suykens, Johan & Vanthienen, Jan. (2003). Benchmarking state-of-the-art classification algorithms for credit scoring. Journal of the Operational Research Society. 54. 10.1057/palgrave.jors.2601545. [↑](#footnote-ref-2)
3. Lessmann, Stefan & Baesens, Bart & Seow, Hsin-Vonn & Thomas, Lyn. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. European Journal of Operational Research. (doi:10.1016/j.ejor.2015. [↑](#footnote-ref-3)
4. Teplý, Petr & Polena, Michal. (2019). Best Classification Algorithms in Peer-to-Peer Lending. The North American Journal of Economics and Finance. 10.1016/j.najef.2019.01.001. [↑](#footnote-ref-4)
5. This data is not exactly the same data set which we make use of. [↑](#footnote-ref-5)
6. Riza Emekter, Yanbin Tu, Benjamas Jirasakuldech & Min Lu (2015) Evaluating credit risk and loan performance in online Peer-to-Peer (P2P) lending, Applied Economics, 47:1, 54-70, DOI: [10.1080/00036846.2014.962222](https://doi.org/10.1080/00036846.2014.962222) [↑](#footnote-ref-6)
7. This data is not exactly the same data set which we make use of. [↑](#footnote-ref-7)
8. [↑](#footnote-ref-8)
9. This is a difficult task considering that we do not have access to a number of very important variables used by Lending Club. [↑](#footnote-ref-9)
10. The Kolmogorov–Smirnov test (K–S test or KS test) is a nonparametric test of the equality of continuous, one-dimensional probability distributions that can be used to compare a sample with a reference probability distribution. [↑](#footnote-ref-10)
11. Anderson–Darling test is a statistical test of whether a given sample of data is drawn from a given probability distribution. [↑](#footnote-ref-11)
12. A Q–Q plot is a plot of the quintiles of two distributions against each other, or a plot based on estimates of the quintiles. [↑](#footnote-ref-12)
13. All the P values are less than 0.05, which mean that the null hypothesis of normal distribution is declined. [↑](#footnote-ref-13)
14. With p-value = 0.2293. [↑](#footnote-ref-14)
15. Because of that our data is a big data set and because of the null hypothesis of the KS and the AD tests, it can be difficult to show normality using those tests. [↑](#footnote-ref-15)
16. Those due to the fact that our model is going to be use in "future" years which are not visible in our data. [↑](#footnote-ref-16)
17. See also section 3 " Outcome variable definition". [↑](#footnote-ref-17)
18. Tukey, JW. Exploratory data analysis. Addison-Wesely, 1977. [↑](#footnote-ref-18)
19. Although this variables has more than 95% of missing values, we did the Z score test for an educational reasons. [↑](#footnote-ref-19)
20. As mentioned in the following sources:

    https://www.thebalance.com/can-you-go-over-your-credit-limit-961095

    https://www.creditcards.com/credit-card-news/credit-limit-how-far-over-before-card-declined.php

    https://www.valuepenguin.com/what-happens-if-you-go-over-your-credit-limit [↑](#footnote-ref-20)
21. It should be noted that we choose the eps hyper parameter by using KNNdistance parameter to calculate the mean distance between samples in our data set. [↑](#footnote-ref-21)
22. DBSCAN algorithm can label the outliers by himself, while the all samples that are not associated by the algorithm to a cluster label by it as an outlier. Because our goal using DBSCAN was not to clustering the data but to identify outliers only, we chose to set the min point's hyper parameter to 1 and watch the different clusters size first and only then labeling the very little clusters as an outliers. [↑](#footnote-ref-22)
23. Fawcett, T. (2006). An introduction to ROC analysis. Pattern Recognition Letters, 27, 861-874. [↑](#footnote-ref-23)
24. <https://en.wikipedia.org/wiki/Receiver_operating_characteristic#Area_under_the_curve> and Lessmann & e.g., 2015. [↑](#footnote-ref-24)
25. For the correlations values, see notebook number 9 " 9-Model-Selection.ipynb". [↑](#footnote-ref-25)
26. XGboost, LR and ADAboost. Additionally, it should be noted that the SVM algorithm AUC outcome was close to 0.5 [↑](#footnote-ref-26)
27. [↑](#footnote-ref-27)
28. Lessmann & e.g., 2015, also mention that " financial institutions require PD estimates that are not only accurate but also well calibrated" [↑](#footnote-ref-28)
29. As described at the introduction, we defined default as a case which the borrower failed to pay his debt. [↑](#footnote-ref-29)
30. AUC is higher than 0.5. [↑](#footnote-ref-30)
31. Michal Polena, 2016, Performance Analysis of Credit Scoring Models on Lending Club Data, MASTER’S THESIS Supervise by Petr Tepl´y, Charles University ,Faculty of Social Sciences, Institute of Economic Studies. [↑](#footnote-ref-31)
32. Malekipirbazari, M. & V. Aksakalli (2015): ”Risk assessment in social lending

    via random forests.” Expert Systems with Applications 42(10): pp. 4621-4631. [↑](#footnote-ref-32)
33. <https://towardsdatascience.com/predicting-loan-repayment-5df4e0023e92>, [↑](#footnote-ref-33)
34. Acording to Polena, he did so due to the fact that: " According to the Lending Club statistics, loans that are more than 90 days past due have 85% chance of not being paid back at all." [↑](#footnote-ref-34)