Analysis of Lending Club Loan Data to Predict Delinquency Likelihood

MSIS 5223 – Programming for Data Science

Project Report

Oklahoma State University

05-05-2018

MEMBERS

ArunPrakash Elavarasan - A20103082

Manideep Mellachervu - A20036160

Paritosh Mehta - A20109589

SathissKumar UdayaKumar - A20102226

Table of Contents

Executive Summary

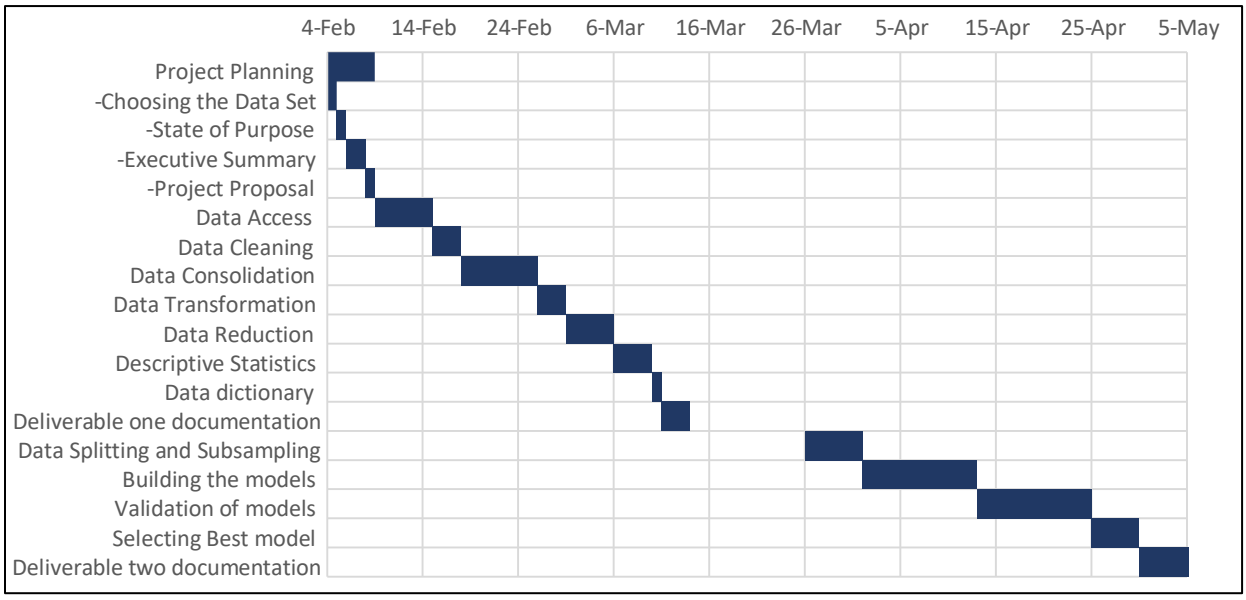
Lending Club, Inc. is a US based company which offers platform for borrowers to obtain loans from investors, where the investors receive interest payments based on invested loans. As with any lending practices, there is always a certain percent of loans that get defaulted/charged off. The confidence in lenders to invest in loans through a platform largely depends on the amount of risk there exists for a loan getting charged off. Thus, the chance of a loan being defaulted by a borrower is an extremely important attribute that could help in increasing investor confidence and consequently higher loan investment through the platform, profiting the company.

The project aims to obtain the likelihood of a loan being charged off by identifying various factors related to the past borrowers by building a predictive model. Further, we also plan to estimate the amount of loan that is reasonable to give to a borrower based on said characteristics. Through these predictions, investor confidence and reliance on the company could increase due to the notion of minimized risk of loan delinquency.

Project Schedule

The project is estimated to finish in a span of three months starting 02/04/2018. The members of the project met every week to consolidate the individually completed tasks and then to divide the work accordingly. Team meetings held every Sunday and were given approximately 3 hours. We took a break during the spring break holidays.

Gantt chart:



	Deliverable two documentation	Selecting Best model	Validation of models	Building the models	Data Splitting and Subsampling	Deliverable one documentation	Data dictionary	Descriptive Statistics	Data Reduction	Data Transformation	Data Consolidation	Data Cleaning	Data Access	-Project Proposal	-Executive Summary	-State of Purpose	-Choosing the Data Set	Project Planning
Start Date	30-Apr	25-Apr	13-Apr	1-Apr	26-Mar	11-Mar	10-Mar	6-Mar	1-Mar	26-Feb	18-Feb	15-Feb	9-Feb	8-Feb	6-Feb	5-Feb	4-Feb	4-Feb
Duration	7	5	12	12	6	3	1	4	5	3	8	3	6	1	2	1	1	5

Project Allocation:

Activity	Members
Project planning	All
Choosing dataset	All
Statement of purpose	Sathisskumar Udayakumar / ArunPrakash Elavarasan
Executive summary	Paritosh Mehta / Manideep Mellachervu
Project proposal	All
Data access	Manideep Mellachervu / Sathisskumar Udayakumar
Data cleaning	Sathisskumar Udayakumar / ArunPrakash Elavarasan
Data consolidation	Manideep Mellachervu / Paritosh Mehta
Data transformation	ArunPrakash Elavarasan/ Paritosh Mehta
Data reduction	Manideep Mellachervu / Sathisskumar Udayakumar
Descriptive statistics	All
Data dictionary	ArunPrakash Elavarasan/ Paritosh Mehta
Identifying models	Manideep Mellachervu / Sathisskumar Udayakumar
Model creation	All
Reporting the results	All
Model Assessment	ArunPrakash Elavarasan/ Paritosh Mehta
Documentation	All

Statement of Scope

Objectives:

The project's goal is to predict the likelihood of a borrower to default a loan based on the evaluation of various factors associated with the past borrowers. The data is based on loan issuances from 2015 to 2017 which includes a total of 858,243 borrowers. We built the model based on sample size of approximately 200,000 and generalize it to population of around 800,000. After analyzing attributes related to past borrowers including, but not limited to, geography, financial history, loan status and purpose as well as evaluating the data to distinguish the factors that determine the change

in the target variable, we proceed to predict the dependent variables related to potential loan delinquency and funded amount.

Target and Predictor Variables:

The objective for our study is to predict the chance of delinquency of potential loan borrowers using aforementioned attributes related to past borrowers. Here, target variable of our analysis is the loan status and the predictor variables include loan term, interest rate, region, grade, loan amount, annual income. The loan status is classified into three categories: Fully Paid, Charged off and Current. Based on the predictor variables, we can estimate the chance of delinquency of a particular loan that is classified as 'Current' in the loan status i.e. if the loan will be paid or charged off. We would also predict the loan amount that is reasonable to be funded to a potential loan borrower based on the borrower's details. Here target variable is funded amount and predictor variables are selected based on managerial importance.

Data Preparation

1. Data Access

The data for the project is obtained from the Kaggle website (https://www.kaggle.com/wordsforthewise/lending-club/data). The data consists of the geographical, professional and financial history of the previous loan borrowers. There are three files, each associated with the year of loan issuance: 2015, 2016 and 2017. The files are CSV in format which have a total size of 1GB. Each dataset has approximately 400,000 records with 150 attributes. We planned to use all the data initially for cleaning and for analysis we used data by removing unrelated and unnecessary attributes.

2. Data Consolidation

The dataset we found contained information from 2007 to 2017, but our analysis is limited to 3 years from 2015 to 2017. Hence, we extracted the yearly information from master dataset into 3 separate datasets and then consolidated them into a single dataset. We used R Studio to extract the data sets with the help of sqldf package. Since all the attributes were same for all the datasets, Rbind function was used for consolidation. The following is the code used for data consolidation.

```
install.packages('sqldf')
library(sqldf)
getwd()
wd<-'E:\\Programming DS with R and python -MSIS 5223\\Project\\MSIS_Project_data\\'
setwd(wd)
#Data Consolidation
#Reading accepted_2007_to_2017.csv
loan_data<-paste(wd,'accepted_2007_to_2017.csv',sep='')
loan_data<-read.csv(loan_data,sep=',',header=F,row.names=NULL,encoding='UTF')
colnames(loan_data)=as.character(unlist(loan_data[1,]))
loan_data<-loan_data[-1,]
loan_data<-loan_data[-c(nrow(loan_data),nrow(loan_data)-1),]
rownames(loan_data)<-NULL
loan_data_2015<-sqldf('select * from loan_data where issue_d like "%2015"')
loan_data_2016<-sqldf('select * from loan_data where issue_d like "%2016"')
loan_data_2017<-sqldf('select * from loan_data where issue_d like "%2017"')

ld_2015<-paste(wd,'loan_data_2015.csv',sep='')
write.csv(loan_data_2015,ld_2015,sep=",",row.names=F)
loan_data_2015_final<-read.csv(ld_2015,sep=',',header=T,encoding='UTF-8')

ld_2016<-paste(wd,'loan_data_2016.csv',sep='')
write.csv(loan_data_2016,ld_2016,sep=",",row.names=F)
loan_data_2016_final<-read.csv(ld_2016,sep=',',header=T,encoding='UTF-8')

ld_2017<-paste(wd,'loan_data_2017.csv',sep='')
write.csv(loan_data_2017,ld_2017,sep=",",row.names=F)
loan_data_2017_final<-read.csv(ld_2017,sep=',',header=T,encoding='UTF-8')

loan_data_2015to2017<-rbind(loan_data_2015_final,loan_data_2016_final,loan_data_2017_final)
ld_2015to2017<-paste(wd,'loan_data_2015to2017.csv',sep='')
write.csv(loan_data_2015to2017,ld_2015to2017,sep=",",row.names=F)
```

3. Data Cleaning

a) Columns with missing values:

On preliminary analysis of data, about 80 columns had missing values and 40 others were irrelevant with respect to the target variable. Hence, we used the following code to remove the missing and irrelevant columns.

```
loan_data_2015to2017_final_cleaned<-subset(loan_data_2015to2017_final,select=-c(2,10,11,19,21,22,25:39,44:50,53,54,55,57:62,63:78,80:150))
colnames(loan_data_2015to2017_final_cleaned)
```

b) Elimination of erroneous data:

As per our observation, we didn't find any erroneous data in our dataset.

c) Detection of Outliers:

We plotted box plots for all numerical columns taken into our analysis and in order to reduce the number of observations from 1,200,000 observations to 900,000 observations, we removed outliers from the dataset.

d) Records with missing values:

There are very few records in our data set with missing values, approximately 1000 records out of total 1,200,000 records. Removal of these observations only accounted for 0.08% of the data, not having a significant impact on our final result. Hence, we removed observations with missing values and finally considered the complete cases for our analysis.

```
loan_data_analysis_2015to2017<-loan_data_analysis_2015to2017[complete.cases(loan_data_analysis_2015to2017),]
```

e) Adjustments to data types:

Our dataset included variables of numeric and character datatypes and we haven't found any need to adjust the data type, so proceeded with our analysis without making any changes concerned in this step.

4. Data Transformations

a) Aggregating the data:

Our dataset required binning of dti variable (debt to income ratio) as it is a continuous variable where there is a possibility of various values for the analysis. The code used is:

```
#binning dti
bin_interval = c(-1,12,18,24,1000)
table(cut(data_with_regions$dti, bin_interval, right = F))
dti_vector = car::recode(data_with_regions$dti, "-1:12='0-12'; 12:18='13-18';
                  18:24='19-24'; 24:1000='24-1000'")
data_with_region$dti_categ = dti_vector
summary(data_without_outliers$dti)
```

b) Constructing new attributes:

Our dataset required construction of three new variables: regions, quarters and loan status. We combined levels of state variable and created region variable whereas quarters variable is obtained by combining levels of issue_date variable and we combined different levels of loan status into three distinct levels as shown.

Code:

```
#Aggregatingregions
region_bins = tribble(~addr_state, ~region,'CT','NorthEast','ME','NorthEast','MA','NorthEast','NH','NorthEast',
                      'RI','NorthEast','VT','NorthEast','NY','NorthEast','NJ','NorthEast','PA','NorthEast',
                      'IL','MidWest','IN','MidWest','MI','MidWest','OH','MidWest','WI','MidWest','IA','MidWest',
                      'KS','MidWest','MN','MidWest','MO','MidWest','NE','MidWest','ND','MidWest','SD','MidWest',
                      'DE','South','FL','South','GA','South','MD','South','NC','South','SC','South','VA','South',
                      'WV','South','DC','South','AL','South','KY','South','MS','South','TN','South','AR','South',
                      'LA','South','OK','South','TX','South','AZ','West','CO','West','ID','West','MT','West','NV',
                      'West','NM','West','UT','West','WY','West','AK','West','CA','West','HI','West','OR','West','WA','West')
data_with_regions = left_join(data_without_outliers, region_bins, by='addr_state')
```

```
#Aggrregating Issue date
Month = tribble(~issue_d,~Quarter,'Jan-2015','2015-Q1','Feb-2015','2015-Q1','Mar-2015','2015-Q1','Apr-2015',
                '2015-Q2','May-2015','2015-Q2','Jun-2015','2015-Q2','Jul-2015','2015-Q3','Aug-2015','2015-Q3',
                'Sep-2015','2015-Q3','Oct-2015','2015-Q4','Nov-2015','2015-Q4','Dec-2015','2015-Q4',
                'Jan-2016','2016-Q1','Feb-2016','2016-Q1','Mar-2016','2016-Q1','Apr-2016','2016-Q2',
                'May-2016','2016-Q2','Jun-2016','2016-Q2','Jul-2016','2016-Q3','Aug-2016','2016-Q3',
                'Sep-2016','2016-Q3','Oct-2016','2016-Q4','Nov-2016','2016-Q4','Dec-2016','2016-Q4',
                'Jan-2017','2017-Q1','Feb-2017','2017-Q1','Mar-2017','2017-Q1','Apr-2017','2017-Q2',
                'May-2017','2017-Q2','Jun-2017','2017-Q2','Jul-2017','2017-Q3','Aug-2017','2017-Q3',
                'Sep-2017','2017-Q3','Oct-2017','2017-Q4','Nov-2017','2017-Q4','Dec-2017','2016-Q4')
data_with_regions_loanstatus_issuedate = left_join(data_with_regions_loanstatus, Month, by='issue_d')
```

```
#binning loan status
Loan_Status_ =tribble(~loan_status, ~Loan_Status,'Fully Paid','Fully_Paid','Late (31-120 days)','Fully_Paid','Late (16-30 days)',
                      'Fully_Paid','Default','Defaulted','Charged Off','Defaulted',
                     'Current','current','In Grace Period','current')
data_with_regions_loanstatus = left_join(data_with_regions, Loan_Status, by='loan_status')
```

5. Data Reduction

In our dataset, we observed variables which are redundant and we performed principal component analysis on those variables to reduce the number of columns for our analysis. Variables used for data reduction are funded amount, loan amount, funded amount by investors, total payment, total payment investor, total received interest to date and total received principal amount to date.

Code:

```
pca_data=subset(loan_data_2015to2017_final_cleaned,select=c(2,3,4,19:22))
colnames(pca_data)
str(pca_data)
pca = princomp(pca_data,cor=TRUE)
pca$sdev^2
plot(pca,main="Attribute")
FA = factanal(pca_data,rotation="varimax", factors=3, scores="none")
FA
```

6. Descriptive statistics

The following table shows the basic descriptive statistics (Mean, Median, Standard Deviation, Minimum and Maximum) for numeric variables in our dataset.

Attribute	Mean	Median	Standard Deviation	Minimum	Maximum
Funded Amount($)	12,345	11150	6931.06	1000	38000
Interest Rate (%)	12.18	11.99	3.96	5.32	24.74
Installment ($)	368.77	329.97	198.29	14.01	1090.18
Annual Income($)	66709	60000	30212	600	166300
Debt to Income	18.88	18.44	8.37	-1	43.85
Last Fico Range high	699.3	699	49.97	569	824
Last Fico Range Low	695.3	695	49.97	565	820

From above table, we can observe that,

- Mean funded amount given to the customers as part of loan by lending club is $12,345.

- 24.74% is the maximum interest rate that can be provided to the borrowers.
- $600 is the minimum annual income of the customers who have taken loan from lending club platform.
- The median installment value paid by customers to Lending club is 329.97

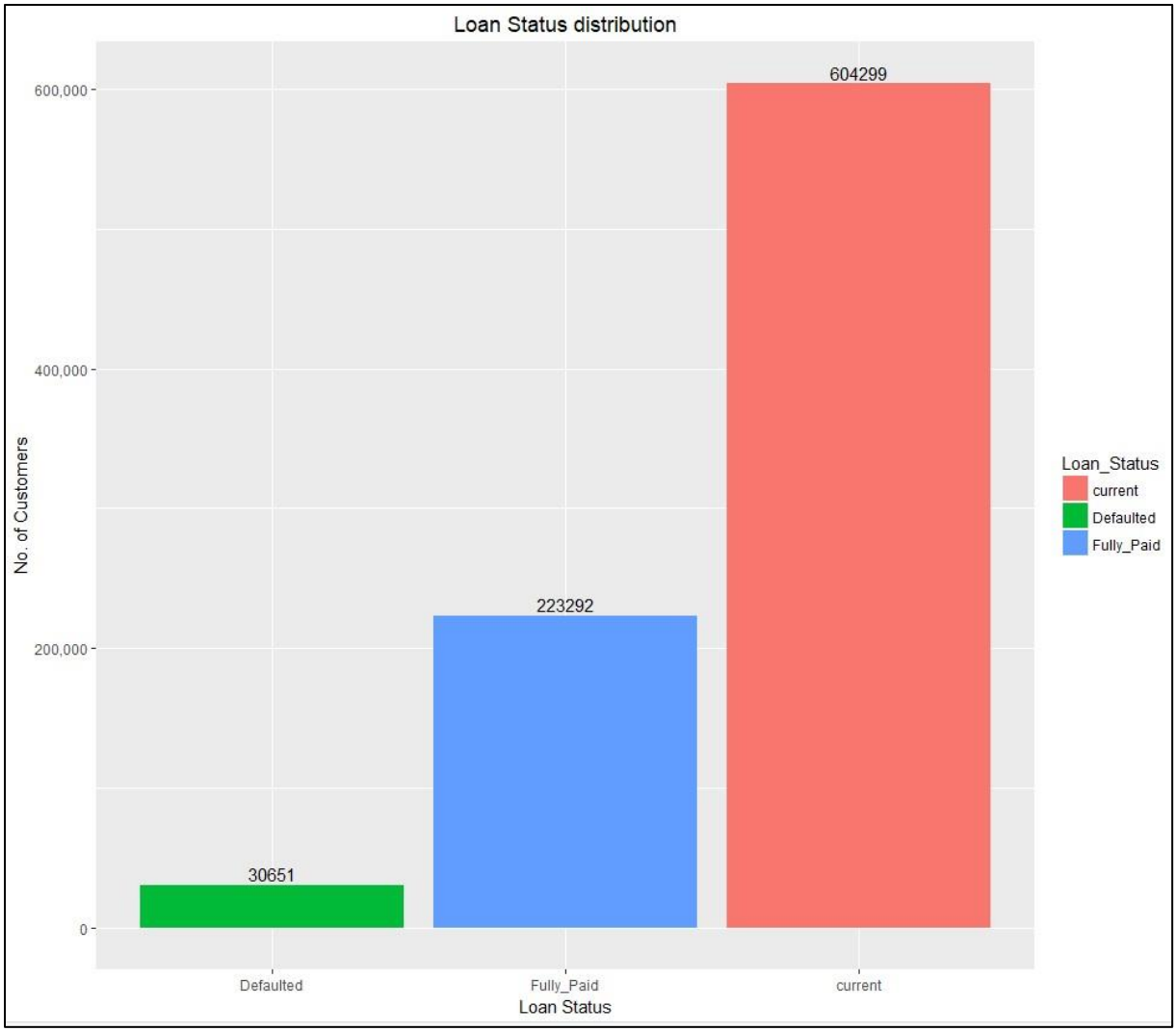
Graphs and Plots:

In our descriptive analysis, we created following plots which involve target variable as Loan status and other predictor variables such as dti ratio, verification status and loan issued date.

We further plotted some of the important variables in our dataset against number of customers.

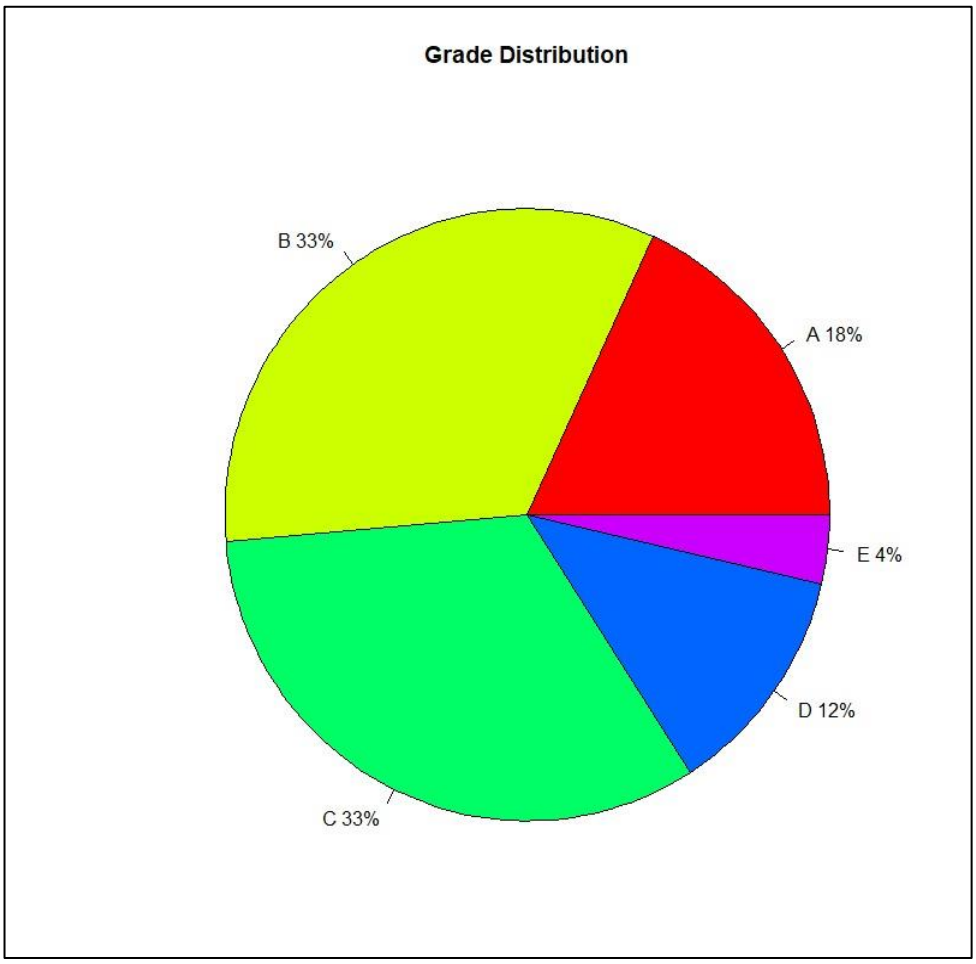
1. Loan status vs No. of customers: The below plot shows the distribution of our target variable, Loan status across the customers of lending club. We observed people with current loan status are far more compared to Fully-paid and Defaulted which is lowest.

```
count = table(final_data_set$Loan_Status)
aa= as.data.frame(count)
aa
colnames(aa)=c("Loan_Status","No_of_Customers")
check=ggplot(data=aa, aes(x= reorder(Loan_Status,No_of_Customers),
y = No_of_Customers, fill=Loan_Status)) + geom_bar(stat="identity")
+geom_text(aes(label = No_of_Customers), vjust = -0.3)+ggtitle("Loan Status distribution")
+xlab("Loan Status")+ylab("No. of Customers")+theme(plot.title = element_text(hjust = 0.5))
check + scale_y_continuous(labels = scales::comma)
```



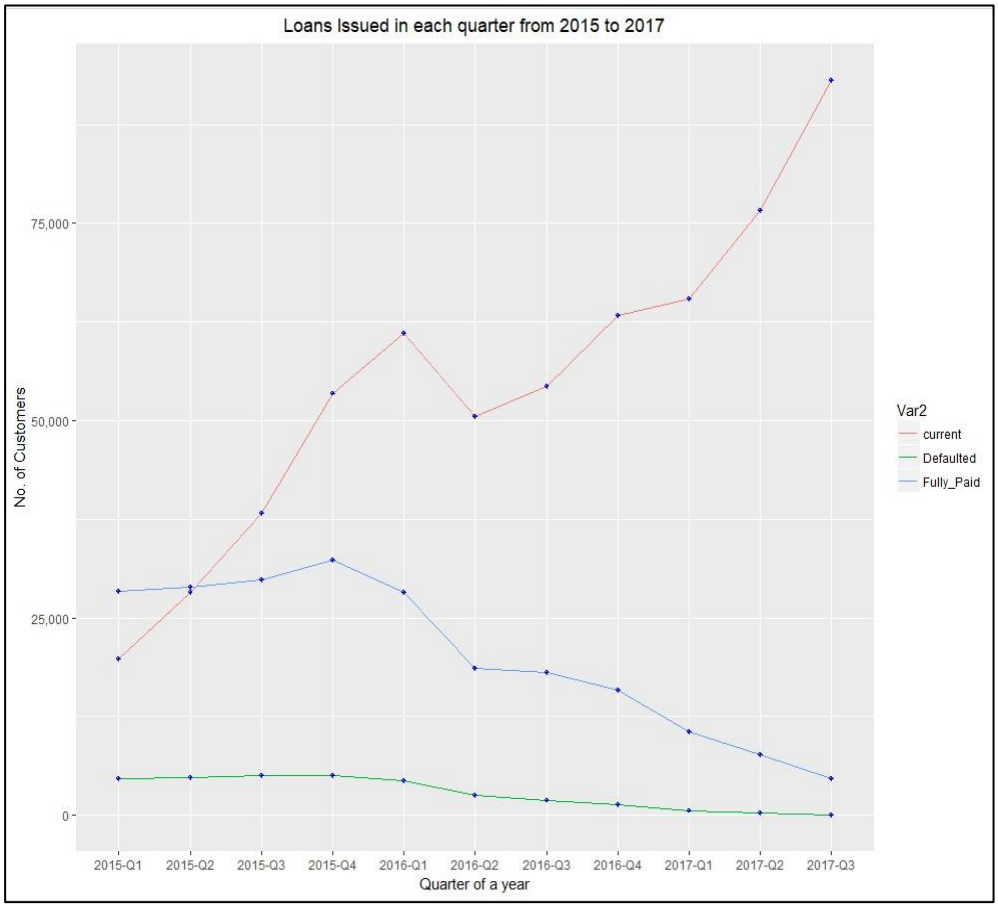
2. Grade vs No. of customers: The pie-chart below explains the Grade distribution across the loan customers, where maximum people were categorized under B and C grade with 33% each.

```
slices <- c(155613,283818,279644,105107,30976)
lbls <- c("A", "B", "C", "D", "E")
pct <- round(slices/sum(slices)*100)
lbls <- paste(lbls, pct,sep=' ') # add percents to labels
lbls <- paste(lbls,"%",sep="") # ad % to labels
pie(slices,labels = lbls, col=rainbow(length(lbls)),
    main="Grade Distribution")
```



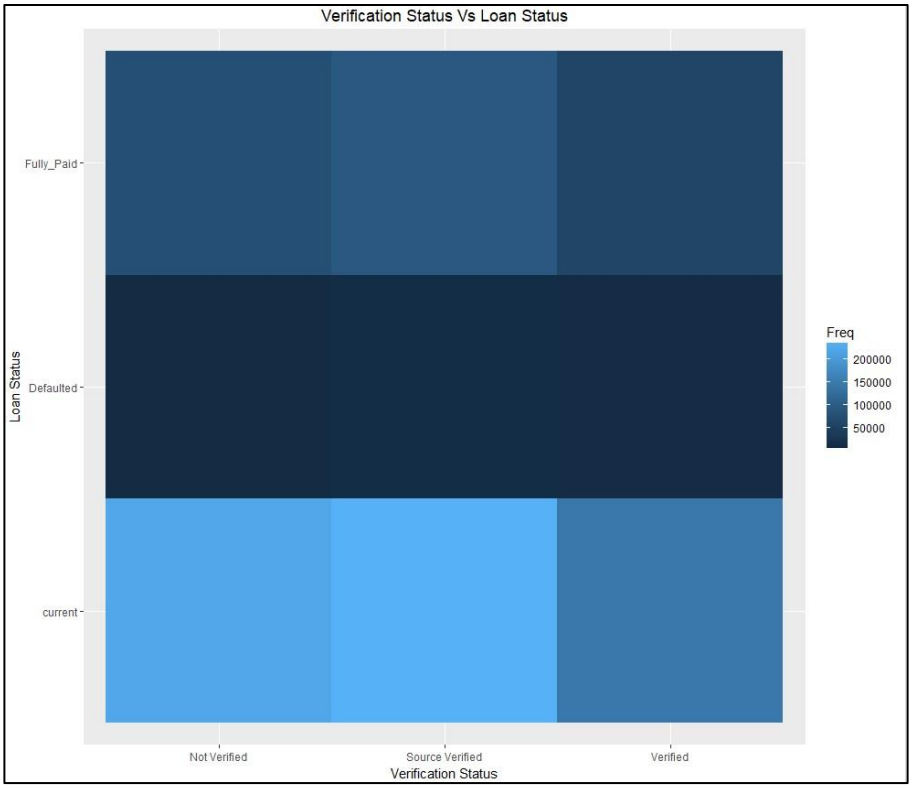
3. No. of customers vs defaulters over period: The line graph below shows the number of customers who defaulted and paid the loan among those who had taken loan in each quarter from 2015 up until 2017. From the graph, we observed that, 2015 had more number of people defaulted and line started slumping down from 2016 Q1 for default and fully paid status.

```
count = table(final_data_set$Quarter,final_data_set$Loan_Status)
count
cc= as.data.frame(count)
cc
ggplot(data=cc, aes(x=Var1, y=Freq, group=Var2)) +
  geom_line(aes(colour=Var2), size=.5) +
  geom_point(colour="blue", size=1, shape=10, fill="white")
+ggtitle("Loans Issued in each quarter from 2015 to 2017") +xlab("Quarter of a year")
+ylab("No. of Customers")+theme(plot.title = element_text(hjust = 0.5))+scale_y_continuous(labels = scales::comma)
```



4. Verification status vs loan status: The below mosaic plot shows that the loans which are source verified and fully paid are more compared to the fully paid loans which are not verified.

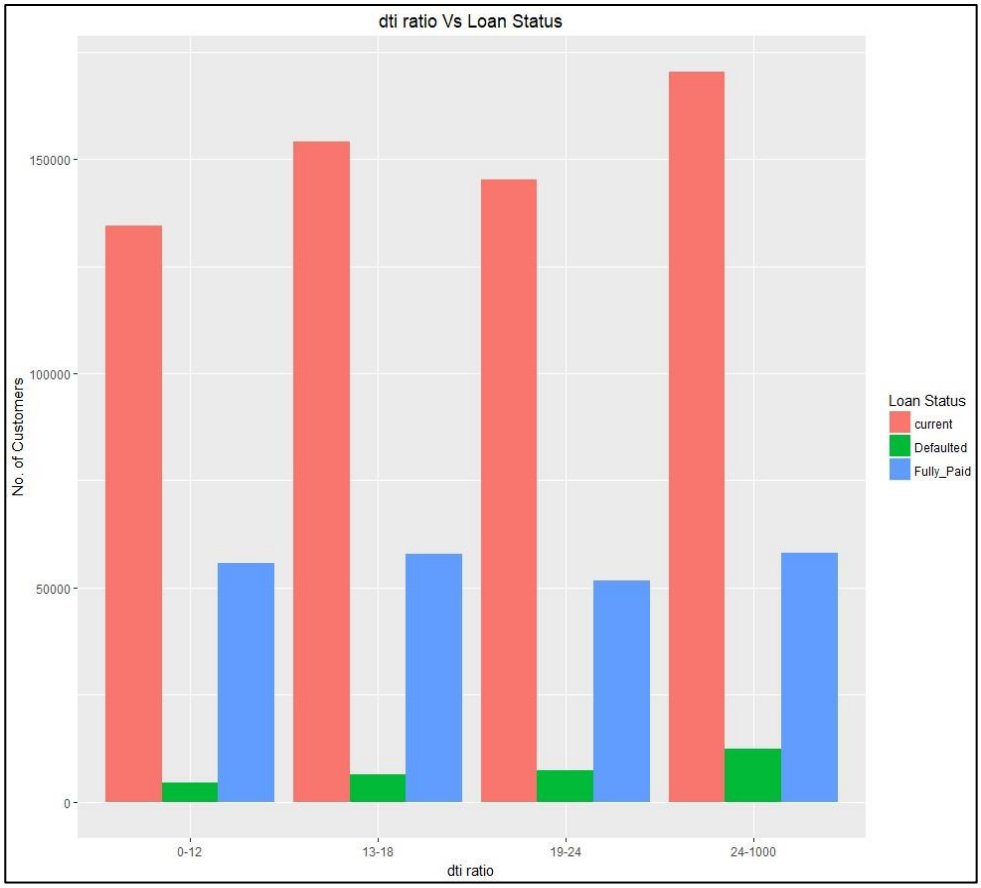
```
count = table(final_data_set$verification_status,final_data_set$Loan_Status)
dd= as.data.frame(count)
dd
ggplot(data = dd, aes(x=Var1, y=Var2, fill=Freq)) +
  geom_tile()+ggtitle("Verification Status Vs Loan Status") +xlab("Verification Status")
+ylab("Loan Status")+theme(plot.title = element_text(hjust = 0.5))
colnames(dd)=c("verification_status","Loan_Status")
grid.table(dd,rows=NULL)
```



verification_status	Loan_Status	Frequency
Not Verified	current	221598
Source Verified	current	236637
Verified	current	146064
Not Verified	Defaulted	7694
Source Verified	Defaulted	12441
Verified	Defaulted	10516
Not Verified	Fully_Paid	75828
Source Verified	Fully_Paid	91122
Verified	Fully_Paid	56342

5. Debt to income ratio vs loan status: The below bar graph shows the number of people with different loan statuses spread across the bins of dti ratio. We observed that people were almost uniformly distributed under all bins for each loan status.

```
count = table(final_data_set$dti_categ,final_data_set$Loan_Status)
ee= as.data.frame(count)
ee
ggplot(ee, aes(Var1, Freq, fill = Var2)) +
  geom_bar(stat = 'identity', position = 'dodge') +labs(title="dti ratio Vs Loan Status") +
  labs(x="dti ratio", y="No. of Customers", fill='Loan Status')+theme(plot.title = element_text(hjust = 0.5))
```



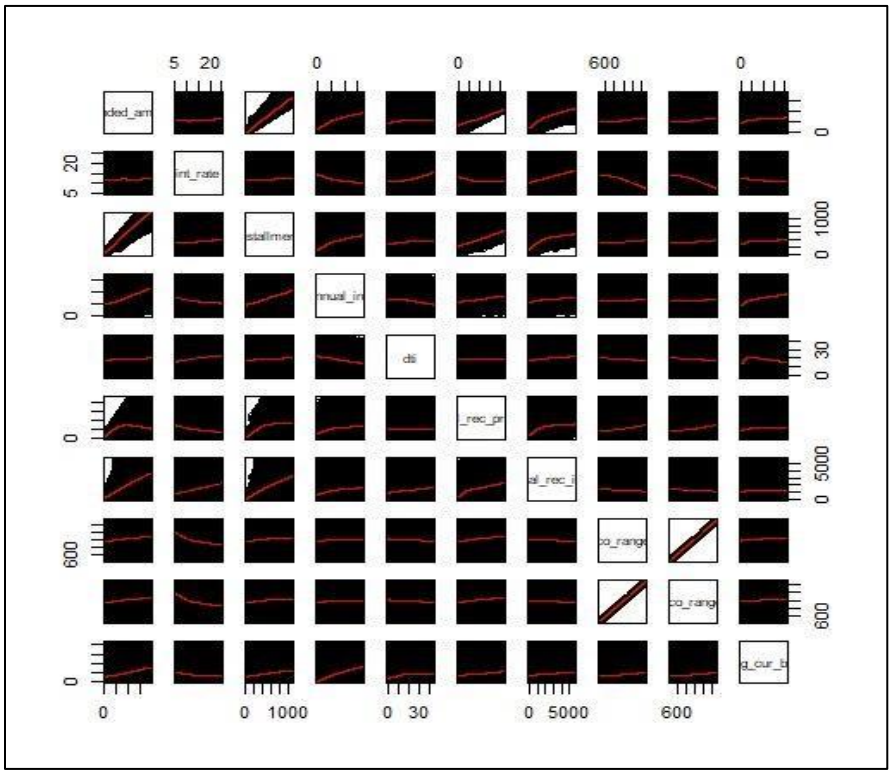
6. Purpose of loan vs No of customers: The following is the grid table showing the number of people who had taken loans from lending club for different purposes.

```
count = table(final_data_set$purpose)
ff= as.data.frame(count)
ff
colnames(ff)=c("Purpose","No of Customers")
grid.table(ff,rows=NULL)
```

Purpose	No of Customers
car	10014
credit_card	195423
debt_consolidation	487193
educational	1
home_improvement	56725
house	3157
major_purchase	19299
medical	11284
moving	6555
other	53594
renewable_energy	554
small_business	7169
vacation	7270
wedding	4

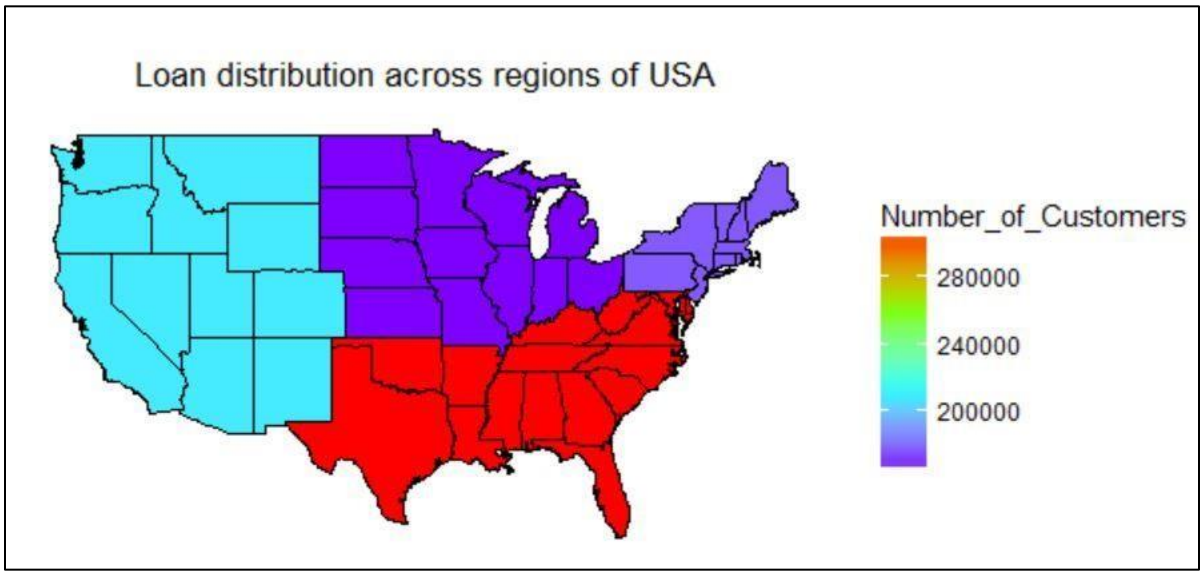
7. Correlation: The following plot shows the correlation between numerical variables in our dataset which is followed by the table displaying the correlation values.

```
pairs(Lend_num,panel=panel.smooth
```




```
> rcorr(as.matrix(Lend_num))
                    funded_amnt int_rate installment annual_inc   dti total_rec_prncp
funded_amnt                1.00     0.04        0.94       0.41  0.06            0.41
int_rate                   0.04     1.00        0.06      -0.18  0.18           -0.14
installment                0.94     0.06        1.00       0.37  0.06            0.46
annual_inc                 0.41    -0.18        0.37       1.00 -0.18            0.18
dti                        0.06     0.18        0.06      -0.18  1.00           -0.01
total_rec_prncp            0.41    -0.14        0.46       0.18 -0.01            1.00
total_rec_int              0.63     0.32        0.56       0.16  0.13            0.39
last_fico_range_high       0.11    -0.35        0.08       0.05 -0.07            0.16
last_fico_range_low        0.11    -0.35        0.08       0.05 -0.07            0.16
avg_cur_bal                0.18    -0.10        0.14       0.34  0.01            0.09
                    total_rec_int last_fico_range_high last_fico_range_low
funded_amnt                  0.63                 0.11                0.11
int_rate                     0.32                -0.35               -0.35
installment                  0.56                 0.08                0.08
annual_inc                   0.16                 0.05                0.05
dti                          0.13                -0.07               -0.07
total_rec_prncp              0.39                 0.16                0.16
total_rec_int                1.00                -0.07               -0.07
last_fico_range_high        -0.07                 1.00                1.00
last_fico_range_low         -0.07                 1.00                1.00
avg_cur_bal                  0.07                 0.07                0.07
```

8. Region vs No. of customers: The below geographical map shows the number of customers who received loans from lending club investors according to the region. We divided the country into four regions: west, Midwest, south and north east. From the map we can see the southern region received more loans than any other region



Data Dictionary

Obs. No.	Attribute Names	Description	Data Types	Source
1	addr_state	The state provided by the borrower in the loan application	Nominal	https://www.lendingclub.com/info/download-data.action
2	annual_inc	The self-reported annual income provided by the borrower during registration.	Ratio	https://www.lendingclub.com/info/download-data.action

3	application_type	Indicates whether the loan is an individual application or a joint application with two co-borrowers	Nominal	https://www.lendingclub.com/info/download-data.action
4	avg_cur_bal	Average current balance of all accounts	Ratio	https://www.lendingclub.com/info/download-data.action
5	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.	Ratio	https://www.lendingclub.com/info/download-data.action
6	dti_categ	Categorical representation of dti	Ordinal	https://www.lendingclub.com/info/download-data.action
7	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.	Ordinal	https://www.lendingclub.com/info/download-data.action
8	funded_amnt	The total amount committed to that loan at that point in time.	Ratio	https://www.lendingclub.com/info/download-data.action

9	grade	LC assigned loan grade	Ordinal	https://www.lendingclub.com/info/download-data.action
10	home_ownership	The home ownership status provided by the borrower during registration. Our values are: RENT, OWN, MORTGAGE, OTHER.	Nominal	https://www.lendingclub.com/info/download-data.action
11	id	A unique LC assigned ID for the loan listing.	Interval	https://www.lendingclub.com/info/download-data.action
12	installment	The monthly payment owed by the borrower if the loan originates.	Ratio	https://www.lendingclub.com/info/download-data.action
13	int_rate	Interest Rate on the loan	Ratio	https://www.lendingclub.com/info/download-data.action
14	issue_d	The quarter which the loan was funded	Nominal	https://www.lendingclub.com/info/download-data.action
15	last_fico_range_high	The upper boundary range the borrower's last FICO pulled belongs to.	Interval	https://www.lendingclub.com/info/download-data.action

16	last_fico_range_l ow	The lower boundary range the borrower's last FICO pulled belongs to.	Interval	https://www.lendingclub.com/info/download-data.action
17	Loan_Status	Current status of the loan	Nominal /Binary(in Future)	https://www.lendingclub.com/info/download-data.action
18	Loan_Status_Agg	Aggregated status of loan	Nominal /Binary(in Future)	https://www.lendingclub.com/info/download-data.action
19	purpose	A category provided by the borrower for the loan request.	Nominal	https://www.lendingclub.com/info/download-data.action
20	pymnt_plan	Indicates if a payment plan has been put in place for the loan	Nominal	https://www.lendingclub.com/info/download-data.action
21	Quarter	Time period of 4 months	Nominal	https://www.lendingclub.com/info/download-data.action
22	region	Geographical divisions of the country	Nominal	https://www.lendingclub.com/info/download

				-data.action
23	term	The number of payments on the loan. Values are in months and can be either 36 or 60.	Nominal	https://www.lendingclub.com/info/download-data.action
24	total_rec_int	Interest received to date	Ratio	https://www.lendingclub.com/info/download-data.action
25	total_rec_prncp	Principal received to date	Ratio	https://www.lendingclub.com/info/download-data.action
26	verification_status	Indicates if the co-borrowers' joint income was verified by LC, not verified, or if the income source was verified	Nominal	https://www.lendingclub.com/info/download-data.action

Modeling Techniques

As mentioned earlier, there are two objectives in our predictive analysis. For the first analysis, the target variable is 'Loan Status'. It is a categorical variable as the data for this variable represents a loan status in terms of payment. Due to the categorical nature of the target variable, logistic regression technique is a plausible option to use for predictive modeling.

First model for first objective is Logistic regression. Like linear regression, it is a regression analysis technique that measures the relationship between a dependent (target) variable and independent variables, except for the fact that the dependent variable in logistic regression should

be dichotomous (binary). The probabilities can be determined from logistic equation by inversing the log function. The impact of each of the predictor variables on the odds of a target variable event occurring can be determined by the coefficients of the logistic regression model. Therefore, the odds of a loan status being paid or charged off can be determined based on the predictor variables that were selected.

Second model we used for analysis is Decision Tree. On basis of split search criteria, decision tree determines the important variables and classify the observations based on the conditions at different nodes. This modeling technique is useful for providing us a visual representation of the data using a tree like model of decisions and their possible consequences. It is also useful to figure out the predictor variables that have the highest impact on the outcome of a target variable.

For the second analysis, the target variable is 'Loan amount'. This variable represents the amount of loan that is reasonable enough to give to a prospective borrower based on the borrower's attributes. Since the data under this variable is continuous in nature, linear regression is a better option. Thus, linear regression is used to determine the relationship between the target variable and other predictor variables and important predictors can be obtained from standardized estimates of predictors.

Model Assumptions:

Logistic Regression:

- It is assumed that model is capable of dealing with all the predictor variables including the dummy variables. No attempt has been made to create dummy variables separately.
- Logit function that target variable equals one of categories is linearly dependent on predictor variables.
- Interpretation of odds of target variable is done by assuming unit change in predictor variable.

Decision Tree:

- Selection of variables has been assumed to be taken care by decision tree itself while creating the model.
- By default, chi-square log worth search criteria has been assumed by Decision Tree to build the model in R.
- The data is assumed to be non-linear with target variable as usually decision tree is used for nonlinear classification.

Linear Regression Model

- This model is used when the target variable is continuous in nature.
- Under this model, we have assumed that the model fits correctly considering there is no overfitting or under fitting.
- The chosen independent variables have a linear relationship with the target variable.
- The errors have constant variance across all predictor variables(Homoscedasticity)
- The errors are normally distributed

Data Splitting and Sub Sampling

For building logistic regression and decision tree models, we have used whole dataset of 858242 observations for splitting into training, validation and testing datasets. Initially we have separated the observations having 'current' as loan status into test dataset as we shouldn't use these observations to build the model. These count to around 604299 observations. Later we divide the dataset of 253943 observations into training and validation datasets in 7:3 proportion or 70% of observations into training dataset and 30% into validation dataset.

Sample	No. of observations
Training	177761
Validation	76182

```
> summary(valid_data$funded_amnt)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1000    7000   10250   11446   15000   37100
> summary(train_data$funded_amnt)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1000    7000   10200   11424   15000   38000
```

It can be observed that the mean, median, min and max are all similar.

For linear regression, we divided the dataset into 50%, 30% and 20%. Splitting the data in this manner would help us to build a flexible model with decreased complexity.

Sample	No. of observations
Training	429121
Validation	214560
Testing	214560

```
> summary(ctr$dti)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00   12.61   18.44   18.88   24.91   43.85
> summary(cv$dti)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  -1.00   12.61   18.44   18.88   24.92   43.84
> summary(ct$dti)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00   12.59   18.46   18.87   24.91   43.85
```

It can be observed that the mean, median, min and max are all similar.

Model Building

Logistic Regression:

The target variable is Loan Status which is categorical. It has two levels: Fully Paid and Defaulted. We have included all the other variables as predictor variables and we have identified few of them as significant.

```
> Lend_reg3 = glm(Loan_Status~., binomial, data=train_data)
```

We found funded_amount, term, int_rate, installment, grade, home_ownnership, annual_inc, as some of the significant variables at 5% as shown below.

```
> summary(Lend_reg3)

Call:
glm(formula = Loan_Status ~ ., family = binomial, data = train_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-4.8397   0.0093   0.0409   0.1231   3.8698

Coefficients: (1 not defined because of singularities)
                                  Estimate Std. Error z value Pr(>|z|)
(Intercept)                     -1.899e+01  8.859e-01 -21.434  < 2e-16 ***
funded_amnt                      3.781e-05  1.615e-05   2.341  0.01924 *
term 60 months                  -2.693e-01  9.416e-02  -2.860  0.00423 **
int_rate                         5.453e-01  1.212e-02  45.010  < 2e-16 ***
installment                     -9.271e-03  5.084e-04 -18.235  < 2e-16 ***
gradeB                          -6.865e-01  6.090e-02 -11.273  < 2e-16 ***
gradeC                          -1.742e+00  8.729e-02 -19.954  < 2e-16 ***
gradeD                          -3.431e+00  1.270e-01 -27.014  < 2e-16 ***
gradeE                          -5.071e+00  1.640e-01 -30.915  < 2e-16 ***
gradeF                          -7.005e+00  2.372e-01 -29.527  < 2e-16 ***
gradeG                           1.333e+01  1.455e+03   0.009  0.99269
home_ownershipMORTGAGE          -1.932e+00  8.420e-01  -2.295  0.02174 *
home_ownershipNONE               4.564e+00  1.455e+03   0.003  0.99750
home_ownershipOWN               -1.968e+00  8.426e-01  -2.336  0.01949 *
home_ownershipRENT              -1.757e+00  8.421e-01  -2.087  0.03693 *
annual_inc                       7.464e-06  5.633e-07  13.250  < 2e-16 ***
verification_statusSource Verified -1.639e-01  3.295e-02  -4.974 6.55e-07 ***
verification_statusVerified     -1.558e-01  3.581e-02  -4.350 1.36e-05 ***
pymnt_plany                      1.605e+01  6.264e+01   0.256  0.79775
```

Equation:

Logit (Loan Status) = -18.99 + 0.00003 funded_amnt+0.54 interest rate – 0.0092 Installment – 0.06 Grade B – 1.74 Grade C – 3.4 Grade D – 5.07 Grade E – 7.00 Grade F + $7.64e^{-06}$ Annual income – 0.16 Verification - source verified – 0.15 Verification - verified – 0.04 dti + 0.0006 total_rec_prncp – 0.0002 total_rec_int + 0.02 last_fico_range_high + 1.98 Application type – joint + 0.00008 Available current balance + 0.22 dtic ateg13-18 + 0.42 dti_categ19-24 + 0.54 dti_categ24-1000

Interpretation:

- For a unit increase in interest rate, the odds of Loan getting defaulted increases by a factor $e^{0.54}$.
- For a unit increase in installment, the odds of Loan getting defaulted decreases by a factor $e^{0.06}$.
- Having the grade B, the odds of Loan getting defaulted decreases by a factor $e^{1.74}$.
- If the loan is source verified, then odds of loan getting defaulted decreases by $e^{0.16}$.
- For a unit increase in total received principle, the odds of getting defaulted increases by $e^{0.0006}$.

Similarly, interpretation can be done for remaining variables.

Decision Tree:

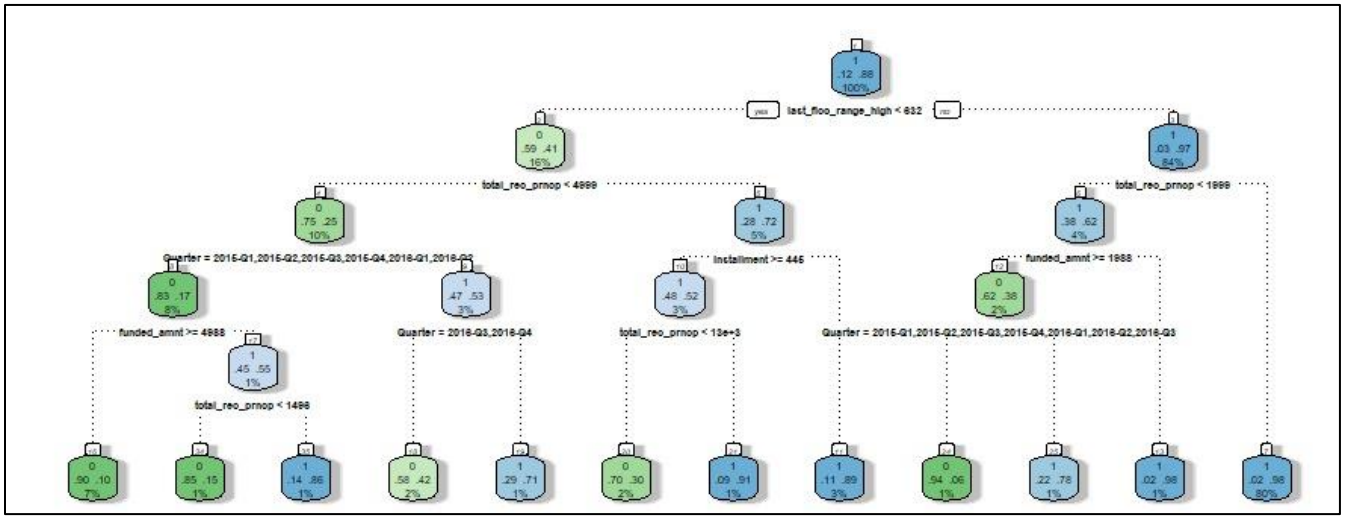
Here target variable is categorical and has two levels, Fully_paid and defaulted. As decision tree takes care of variable selection, we used all variables in model building and observed few of them as important variables. Decision tree splits the observations based on conditions at the node. Below is the plot of our decision tree.

We have observed last_fico_range, quarter, funded_amount, installment, total_rec_pncp as the most important variables in order of importance.

```
rpart_tree <- rpart(Loan_Status ~ ., train_data, method = 'class')
plot(rpart_tree)
text(rpart_tree,pretty=0)
library(rattle)
fancyRpartPlot(rpart_tree)
rpart.plot(rpart_tree)
summary(rpart_tree)
```

Partial Output:

```
> summary(rpart_tree)
Call:
rpart(formula = Loan_Status ~ ., data = train_data, method = "class")
  n= 177761

          CP nsplit rel error    xerror        xstd
1 0.23690343      0 1.0000000 1.0000000 0.006401678
2 0.19295302      1 0.7630966 0.6882923 0.005423483
3 0.02939349      2 0.5701435 0.5691648 0.004970381
4 0.02775447      5 0.4819631 0.5170116 0.004753173
5 0.01667366      7 0.4264541 0.4396439 0.004404900
6 0.01113908     11 0.3597595 0.3606450 0.004009601
7 0.01048658     12 0.3486204 0.3423285 0.003910966
8 0.01000000     13 0.3381339 0.3383203 0.003888984

Variable importance
last_fico_range_high  last_fico_range_low      total_rec_prncp          funded_amnt           installment
                  31                   30                   15                    8                     7
             Quarter        total_rec_int              purpose
                   5                    3                    1

Node number 1: 177761 observations,    complexity param=0.2369034
  predicted class=Fully_Paid  expected loss=0.1207014  P(node) =1
    class counts: 21456 156305
   probabilities: 0.121 0.879
  left son=2 (28187 obs) right son=3 (149574 obs)
  Primary splits:
      last_fico_range_high < 631.5    to the left,  improve=14766.050, (0 missing)
      last_fico_range_low  < 627.5    to the left,  improve=14766.050, (0 missing)
      total_rec_prncp      < 2999.83  to the left,  improve=10636.330, (0 missing)
      funded_amnt          < 21937.5  to the right, improve= 4048.381, (0 missing)
      installment          < 750.44   to the right, improve= 1848.039, (0 missing)
  Surrogate splits:
      last_fico_range_low < 627.5    to the left,  agree=1.000, adj=1.000, (0 split)
      total_rec_prncp     < 1999.835 to the left,  agree=0.855, adj=0.083, (0 split)
      funded_amnt         < 21987.5  to the right, agree=0.851, adj=0.061, (0 split)
      installment         < 780.81   to the right, agree=0.845, adj=0.024, (0 split)
      total_rec_int       < 5626.88  to the right, agree=0.841, adj=0.000, (0 split)
```



Interpretation:

- Node1 checks condition of last_fico_range_high, if it is greater than 632, it should go to right branch node and viceversa.
- Second level of nodes checks total_rec_prncp with 4999 and 1999 and moves into corresponding branch nodes depending on conditions.
- Finally the observations are grouped into 12 root nodes with 6 nodes having target value 1 and remaining with target value 0.

```
# predict labels for Valid data using rpart tree
rpart_pred1 <- predict(rpart_tree, train_data, type = 'class')
rpart_pred1_prob <- predict(rpart_tree, train_data, type = 'prob')
valid_data$last_fico_range_low=valid_data$X0
rpart_pred2 <- predict(rpart_tree, valid_data, type = 'class')
rpart_pred2_prob <- predict(rpart_tree, valid_data, type = 'prob')

# compare labels predicted by rpart to real ones
confusionMatrix(rpart_pred1, train_data$Loan_Status)
confusionMatrix(rpart_pred2, valid_data$Loan_Status)

install.packages('AUC')
library(AUC)
#cutoff=0.5
pred_cut_off <- ifelse(rpart_pred2_prob > 0.5, 1,0) #Setting cut-off to be at 0.5
table(valid_data$Loan_Status,as.data.frame(pred_cut_off)$Fully_Paid )
pred <- prediction(as.data.frame(pred_cut_off)$Fully_Paid,valid_data$Loan_Status)
perf <- performance(pred, "tpr", "fpr")
#Printing AUC Value
perf1 <- performance(pred, "auc")
print(perf1@y.values[[1]])
#Plotting the ROC-curve
install.packages('pROC')
library(pROC)

auc<-auc(valid_data$Loan_Status,rpart_pred2_prob[,2])
plot(roc(valid_data$Loan_Status,rpart_pred2_prob[,2]))
pred <- prediction(as.data.frame(pred_cut_off)$Fully_Paid,valid_data$Loan_Status)
roc = performance(pred, "tpr", "fpr")
plot(roc, col="orange", lwd=2)
```

```
> confusionMatrix(rpart_pred2, valid_data$Loan_Status)
Confusion Matrix and Statistics

            Reference
Prediction   Defaulted Fully_Paid
  Defaulted       6992        870
  Fully_Paid      2203      66117

               Accuracy : 0.9597
                 95% CI : (0.9582, 0.961)
    No Information Rate : 0.8793
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7973
 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.76041
            Specificity : 0.98701
         Pos Pred Value : 0.88934
         Neg Pred Value : 0.96775
             Prevalence : 0.12070
         Detection Rate : 0.09178
   Detection Prevalence : 0.10320
      Balanced Accuracy : 0.87371

       'Positive' Class : Defaulted
```

Linear Regression:

In our linear regression model we have the funded amount (funded_amnt) as the target variable, which is a continuous variable datatype. We have considered independent variables as interest rate (int_rate), installment(installment), annual income (annual_inc), dti ratio (dti_categ), average current balance (avg_cur_bal), last fico high score (last_fico_range_high), home ownership (home_ownership), verification status (verification_status), payment plan(pymnt_plan) and application type (application_type). We were successfully able to execute the linear regression model with above mentioned variables and output of the model is given below:

```
Call:
lm(formula = cv$funded_amnt ~ cv$int_rate + cv$installment +
    cv$annual_inc + cv$dti_categ + cv$avg_cur_bal + cv$last_fico_range_high +
    cv$home_ownership + cv$verification_status + cv$pymnt_plan +
    cv$application_type)

Residuals:
    Min      1Q  Median      3Q     Max
-9018.0 -1294.4  -599.7   267.3 11909.0

Coefficients:
                                      Estimate Std. Error  t value Pr(>|t|)
(Intercept)                         -4.694e+03  2.355e+02  -19.929   <2e-16 ***
cv$int_rate                          3.330e+01  1.415e+00   23.532   <2e-16 ***
cv$installment                       3.158e+01  2.798e-02 1128.697   <2e-16 ***
cv$annual_inc                        1.443e-02  1.967e-04   73.385   <2e-16 ***
cv$dti_categ13-18                    2.191e+02  1.446e+01   15.151   <2e-16 ***
cv$dti_categ19-24                    2.775e+02  1.480e+01   18.750   <2e-16 ***
cv$dti_categ24-1000                  2.025e+02  1.460e+01   13.871   <2e-16 ***
cv$avg_cur_bal                       1.365e-02  6.479e-04   21.066   <2e-16 ***
cv$last_fico_range_high              5.177e+00  1.081e-01   47.900   <2e-16 ***
cv$home_ownershipMORTGAGE            3.952e+02  2.194e+02    1.802   0.0716 .
cv$home_ownershipNONE               -2.230e+03  2.320e+03   -0.961   0.3364
cv$home_ownershipOWN                 1.826e+02  2.197e+02    0.831   0.4060
cv$home_ownershipRENT                1.206e+02  2.194e+02    0.549   0.5827
cv$verification_statusSource Verified -2.388e+02  1.179e+01  -20.255   <2e-16 ***
cv$verification_statusVerified       -4.020e+02  1.357e+01  -29.616   <2e-16 ***
cv$pymnt_plany                       -1.760e+02  2.084e+02   -0.845   0.3983
cv$application_typeJoint App          1.197e+03  3.264e+01   36.681   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2309 on 214543 degrees of freedom
Multiple R-squared:  0.8894,    Adjusted R-squared:  0.8894
F-statistic: 1.078e+05 on 16 and 214543 DF,  p-value: < 2.2e-16
```

Linear regression model with log transformation:

```
Call:
lm(formula = log(cv$funded_amnt) ~ cv$int_rate + cv$installment +
    cv$annual_inc + cv$dti_categ + cv$avg_cur_bal + cv$last_fico_range_high +
    cv$home_ownership + cv$verification_status + cv$pymnt_plan +
    cv$application_type)

Residuals:
     Min       1Q   Median       3Q      Max
-1.48197 -0.11837  0.03149  0.13911  0.75988

Coefficients:
                                        Estimate Std. Error t value Pr(>|t|)
(Intercept)                            7.736e+00  3.078e-02 251.354  < 2e-16 ***
cv$int_rate                            1.253e-03  1.849e-04   6.779 1.22e-11 ***
cv$installment                         2.861e-03  3.656e-06 782.591  < 2e-16 ***
cv$annual_inc                          6.575e-07  2.570e-08  25.581  < 2e-16 ***
cv$dti_categ13-18                      1.918e-02  1.889e-03  10.154  < 2e-16 ***
cv$dti_categ19-24                      2.102e-02  1.934e-03  10.867  < 2e-16 ***
cv$dti_categ24-1000                    1.576e-02  1.908e-03   8.263  < 2e-16 ***
cv$avg_cur_bal                         4.397e-07  8.467e-08   5.194 2.06e-07 ***
cv$last_fico_range_high                4.916e-04  1.412e-05  34.807  < 2e-16 ***
cv$home_ownershipMORTGAGE              7.586e-02  2.866e-02   2.647  0.00813 **
cv$home_ownershipNONE                  5.750e-02  3.032e-01   0.190  0.84956
cv$home_ownershipOWN                   4.157e-02  2.871e-02   1.448  0.14766
cv$home_ownershipRENT                  4.301e-02  2.867e-02   1.500  0.13353
cv$verification_statusSource Verified -3.950e-02  1.540e-03 -25.640  < 2e-16 ***
cv$verification_statusVerified        -7.607e-02  1.774e-03 -42.884  < 2e-16 ***
cv$pymnt_plany                         1.945e-02  2.723e-02   0.714  0.47498
cv$application_typeJoint App           1.846e-02  4.265e-03   4.328 1.50e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3018 on 214543 degrees of freedom
Multiple R-squared:  0.7884,    Adjusted R-squared:  0.7884
F-statistic: 4.995e+04 on 16 and 214543 DF,  p-value: < 2.2e-16
```

Equation:

Log(Funded_amnt) = 7.736 + 0.001253(int_rate) +0.002861 (installment) + 0.0000006575 (annual_inc) + 0.01918 (dti_categ13-18) + 0.02102 (dti_categ19-24) + 0.01576 (dti_categ24-1000)+ 0.0000004397 (avg_cur_bal) + 0.0004916 (last_fico_range_high) + 0.07586 (home_ownershipMORTGAGE) – 0.0395 (verification_statusSourceverified) – 0.07607 (verification_statusVerified) + 0.01846 (application_typeJoint App)

The above equation can be used to predict the funded amount to the borrower. The variable with positive sign tends to increase the funded amount with their increase. In contract with that the variable with negative sign tends to decrease the funded amount with their increase in values.

Linear Regression Interpretation:

Increase of one unit in int_rate tends to increase the funded amount by 0.001253.

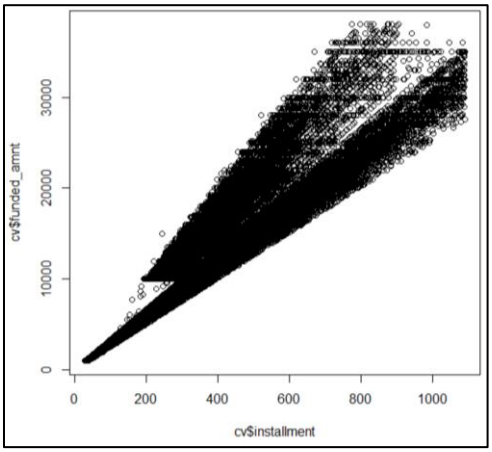
- For unit increase in interest rate the funded amount can be increased by the factor $e^{0.001253}$.
- For unit increase in installment the funded amount can be increased by the factor $e^{0.002861}$.

- For unit increase in annual income the funded amount can be increased by the factor $e^{0.0000006575}$.
- For unit increase in average current balance the funded amount can be increased by the factor $e^{0.0000004397}$.
- For unit increase in the highest fico score the funded amount can be increased by the factor $e^{0.0004916}$.
- Borrower having dti score from 13-18, versus the borrower having dti score 0-12, increases the funded amount by $e^{0.01918}$.
- Borrower having own home mortgage, versus the borrower having any home, increases the funded amount by $e^{0.07586}$.
- Borrower opted for payment plan versus the borrower not opted for payment plan, increases the funded amount by $e^{0.01945}$.
- Borrower having source verified status, versus the borrower having not verified status, decreases the funded amount by $e^{0.0395}$.
- Borrower having joint application, versus the borrower having individual application increase the funded amount by $e^{0.01846}$.

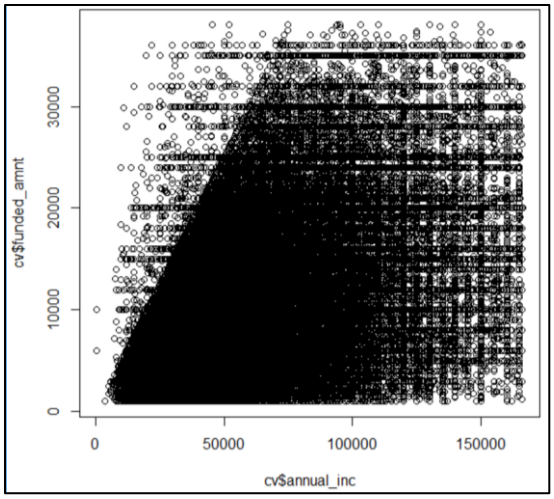
It clear that funded amount will be more with higher interest rate, high installment and high annual income. This makes sense in real world scenario and this is in accordance with our model.

Assumptions:

1. Linearity:

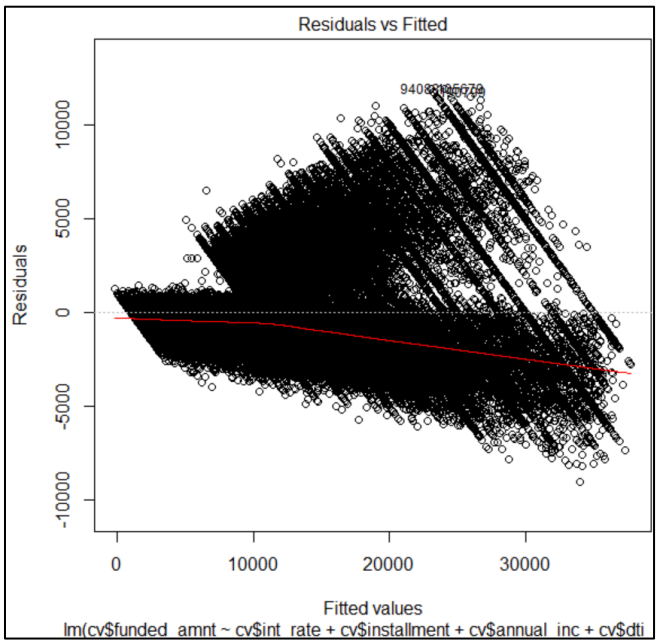


The above figure shows installment is linearly dependent on funded_amount.



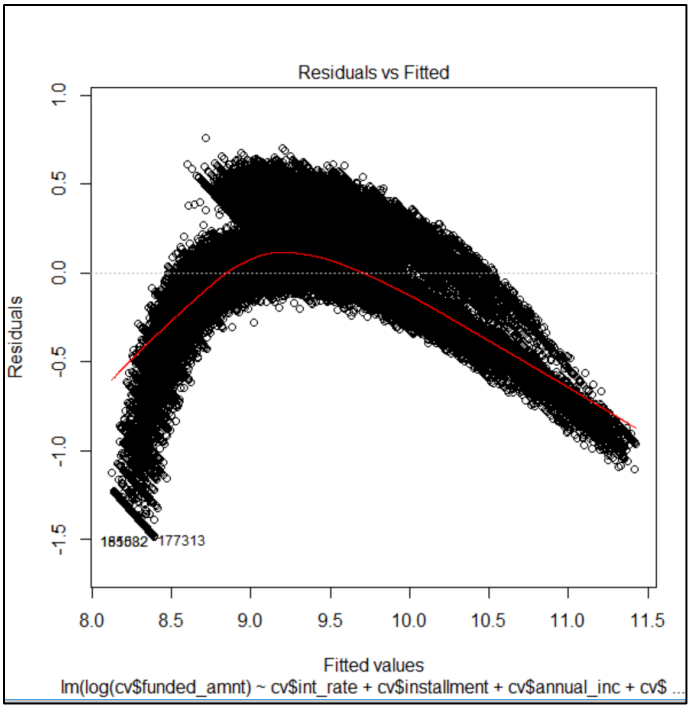
For some of the predictor variables like annual_inc , linearity assumption is violated.

2. Homoscedasticity:



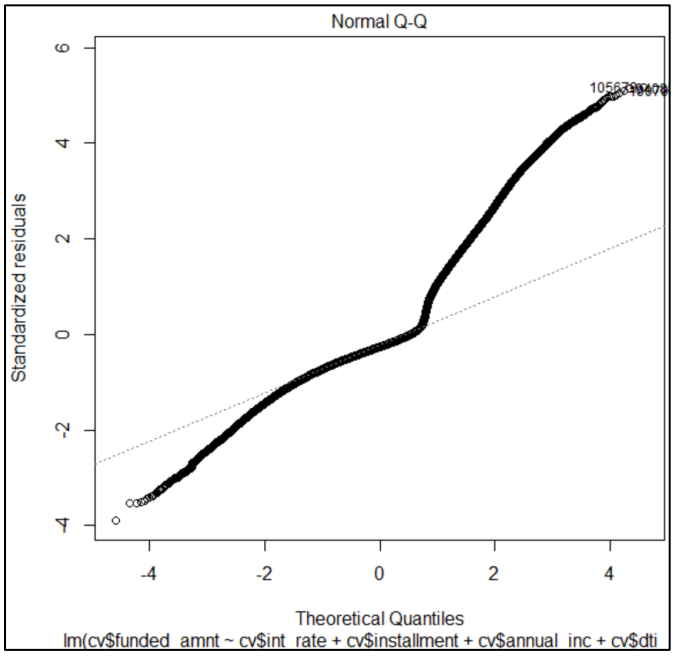
Homoscedasticity assumption violation has been observed.

After log transformation of target variable:

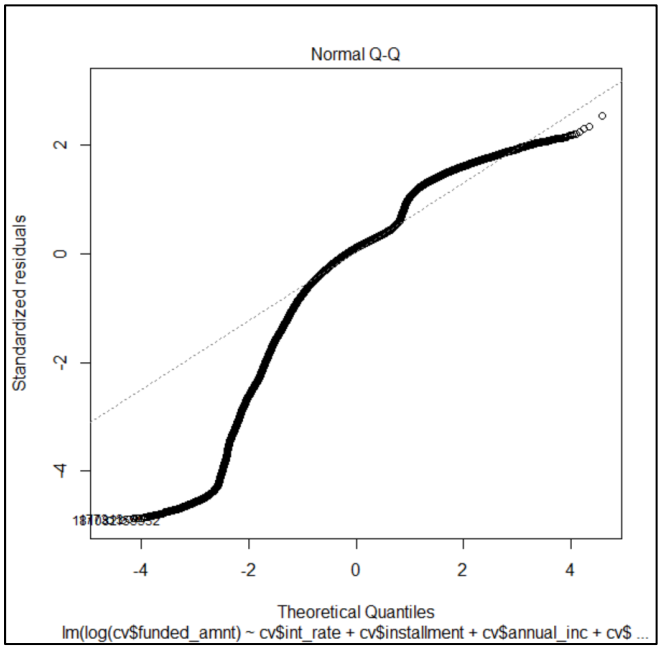


Still, the data is not able to satisfy the Homoscedasticity assumption.

3.Normality:



After Log transformation:



4. Multicollinearity:

```
> vif(cv_reg)
                             GVIF Df GVIF^(1/(2*Df))
cv$int_rate              1.269054  1        1.126523
cv$installment           1.243971  1        1.115334
cv$annual_inc            1.422075  1        1.192508
cv$dti_categ             1.094738  3        1.015200
cv$avg_cur_bal           1.558317  1        1.248326
cv$last_fico_range_high  1.173614  1        1.083335
cv$home_ownership        1.446850  4        1.047256
cv$verification_status   1.109623  2        1.026346
cv$pymnt_plan            1.000689  1        1.000345
cv$application_type      1.030394  1        1.015083
```

VIF values for all predictors is less than 10. Hence there is no multicollinearity between predictor variables.

The model is not able to satisfy both Normality and Homoscedasticity assumptions. As the results of assumptions are reasonable after log transformation, we are proceeding with linear regression using log(funded_amount) as target variable.

Model Assessment:

Logistic regression:

Confusion Matrix for cut off 0.5.

Validation Data:

```
> table(valid_data$Loan_Status,pred_cut_off)
   pred_cut_off
        0     1
  0  7484  1711
  1  1838 65147
```

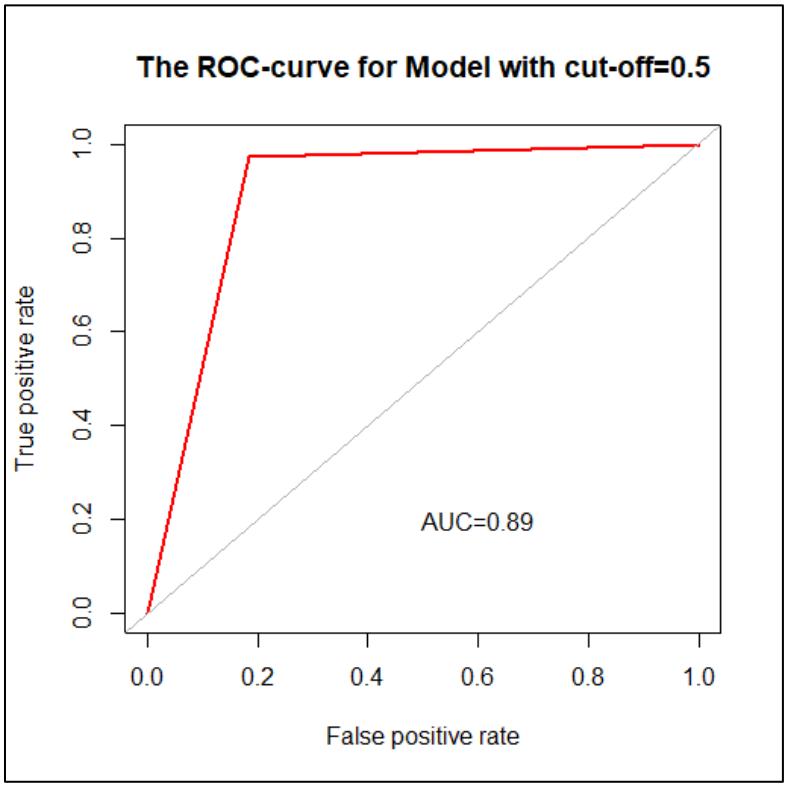
		Reference	
		0	1
Prediction	0	7484	1711
	1	1838	65147

Assessment Parameter	Value
Accuracy	95.33%
Sensitivity	97.25%
Misclassification Rate	4.67%

ROC Curve:

```
> pred <- prediction(pred_cut_off,valid_data$Loan_Status)
> perf <- performance(pred, "tpr", "fpr")
> #Printing AUC Value
> perfl <- performance(pred, "auc")
> print(perfl@y.values[[1]])
[1] 0.8932408
```

Area under curve is 0.93



Confusion Matrix for cut off 0.8:

Validation Data:

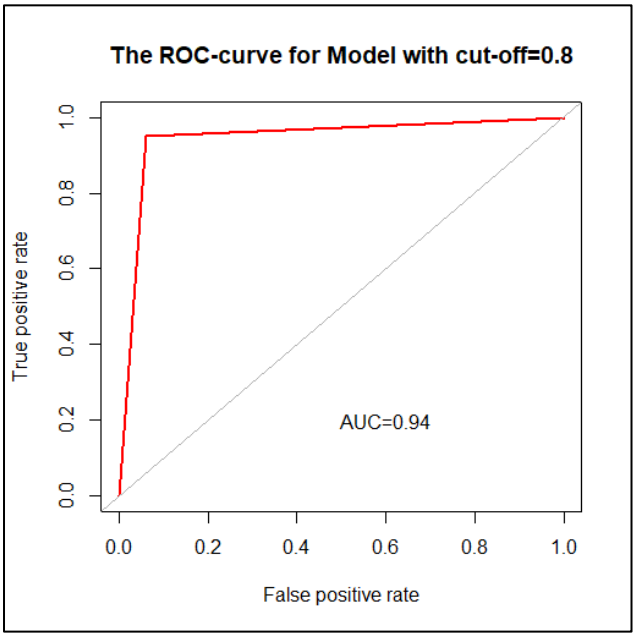
```
> table(valid_data$Loan_Status,pred_cut_off)
   pred_cut_off
        0     1
  0  8636   559
  1  3370 63615
```

		Reference	
		0	1
Prediction	0	8636	559
	1	3370	63615

Assessment Parameter	Value
Accuracy	94.84%
Misclassification Rate	5.16%
Sensitivity	94.96%

ROC Curve:

```
> pred <- prediction(pred_cut_off,valid_data$Loan_Status)
> perf <- performance(pred, "tpr", "fpr")
> perf1 <- performance(pred, "auc")
> print(perf1@y.values[[1]])
[1] 0.9444482
```



Values for cutoff 0.8 provides better results compared to 0.5 as former model has higher area under ROC curve than latter, further both cut off values has almost similar accuracy, misclassification and sensitivity values

Strengths of model:

- Logistic models can be updated easily with new data due to a better probabilistic interpretation of its output.
- The output of logistic regression is more informative than other classification techniques.

Weaknesses of model:

- The model has higher sensitivity and accuracy values, hence chance of overfitting is high
- It requires a lot of data to achieve stable, meaningful results.

- Since the model is heavily dependent on independent variables, inclusion of a wrong independent variable will cause the model to have little to no predictive value.

Decision Tree:

Training data:

```
> confusionMatrix(rpart_pred1, train_data$Loan_Status)
Confusion Matrix and Statistics

            Reference
Prediction   Defaulted Fully_Paid
  Defaulted      16198       1997
  Fully_Paid      5258     154308

               Accuracy : 0.9592
                 95% CI : (0.9583, 0.9601)
    No Information Rate : 0.8793
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7942
 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.75494
            Specificity : 0.98722
         Pos Pred Value : 0.89024
         Neg Pred Value : 0.96705
             Prevalence : 0.12070
         Detection Rate : 0.09112
   Detection Prevalence : 0.10236
      Balanced Accuracy : 0.87108

       'Positive' Class : Defaulted
```

Balanced Accuracy: 87%

Sensitivity: 75%

Validation Data:

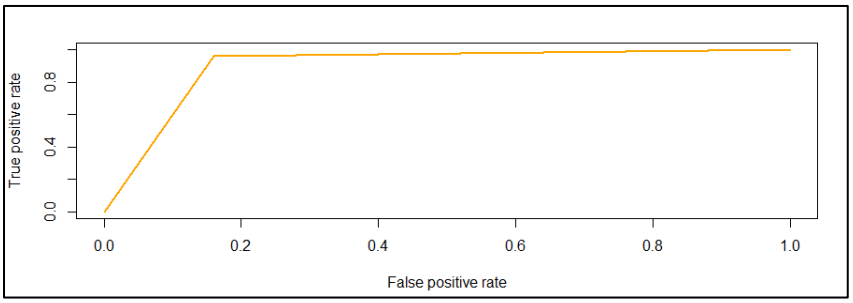
```
> confusionMatrix(rpart_pred2, valid_data$Loan_Status)
Confusion Matrix and Statistics

            Reference
Prediction   Defaulted Fully_Paid
  Defaulted       6992        870
  Fully_Paid      2203      66117

               Accuracy : 0.9597
                 95% CI : (0.9582, 0.961)
    No Information Rate : 0.8793
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7973
 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.76041
            Specificity : 0.98701
         Pos Pred Value : 0.88934
         Neg Pred Value : 0.96775
             Prevalence : 0.12070
         Detection Rate : 0.09178
   Detection Prevalence : 0.10320
      Balanced Accuracy : 0.87371

       'Positive' Class : Defaulted
```

Balanced Accuracy: 87%

Sensitivity: 76%

Area under ROC curve: 0.87

```
auc<-auc(valid_data$Loan_Status,rpart_pred2_prob[,2])
plot(roc(valid_data$Loan_Status,rpart_pred2_prob[,2]))
pred <- prediction(as.data.frame(pred_cut_off)$Fully_Paid,valid_data$Loan_Status)
roc = performance(pred, "tpr", "fpr")
plot(roc, col="orange", lwd=2)
```



Strengths of model:

- The model assigns specific values to problems and decision, reducing the ambiguity in decision making.
- The model has good accuracy and sensitivity values

Weaknesses of the model:

- Decision trees tend to have low bias which makes it difficult for the model to work with new data.
- Calculations can get very complex if the model is dealing with a large dataset with uncertain values to be accommodated by R.

Linear Regression:

We can assess the linear model based on the two values:

- Adjusted R-Square value.
- Residual standard error.
- Mean Absolute Error

Adjusted R-Square value:

We have good adjusted R-square value of 0.78. this indicates that we have good predictive model.

Residual standard error

The residual standard error for our model is 0.3015, which is relatively low. this ensures that we have good model.

Mean Absolute Error: 1667

Strength of the model:

- We have good adjusted R-square value, which indicates we can predict the loan amount correct most of the times.
- Linear regression is one of the simplest method to predict continuous variable.
- We can use log transformation to reduce the skewness of the variable.

Weaknesses of the model:

- Some of the assumptions are violating to an extent.
- We are unable to check the one of the assumption because of the package.
- It requires all the assumption to be satisfied to have an accurate result.
- Cannot handle the data set with different variance and nonlinear in nature.

Conclusions and Justifications:

1. To predict loan default, we propose to use decision tree model as it fits well with any type of data. The values of accuracy 87% and sensitivity 76% for validation data tell the efficiency of the model and complexity.
2. Logistic regression can also be used for predicting loan defaulters but it provides biased results as the values of accuracy 95% and sensitivity 95% are much higher and it appears to be overfitting the model.
3. Linear regression model using funded amount as target variable can be considered as failure model but for predicting the loan amount one can use log transformed value of funded amount as dependent variable.