

**ADVANCES IN DATA SCIENCE/ARCHITECTURE**

**Mid-Term**

**REPORT**

**Submitted By:**

**MEGHA SINGH**

**SNEHA MALSHETTI**

Table of Contents

1. Overview3

Assignment Requirements and detailed overview4

1. Objective Flow of the Project5

Flowchart of Processing of Data Analysis6

1. Data Ingestion ,Exploratory Data Analysis and Data Wrangling11

Steps for Data cleansing and EDA15

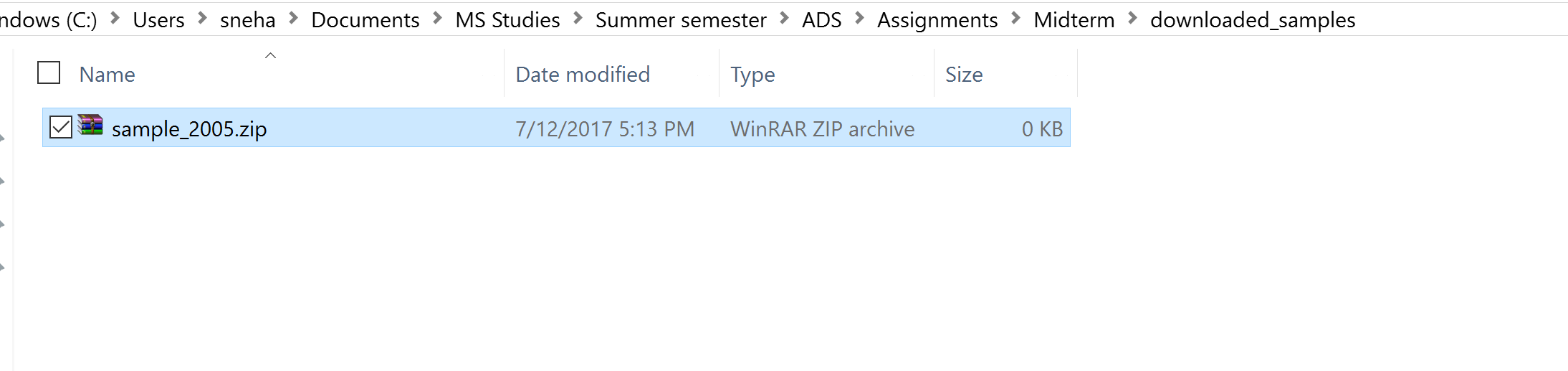
1. Citations4

**Part 2**

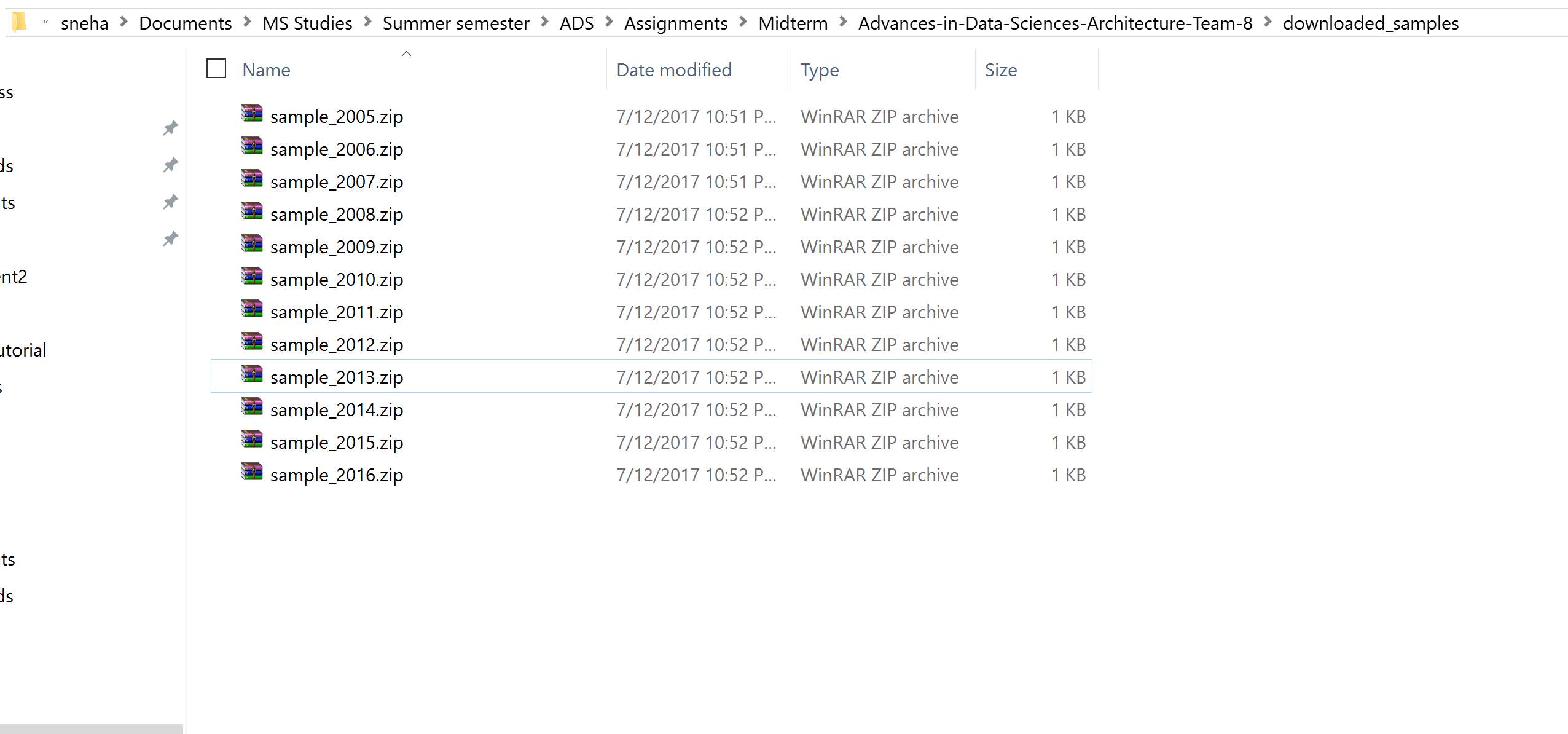
**Data Ingestion :**

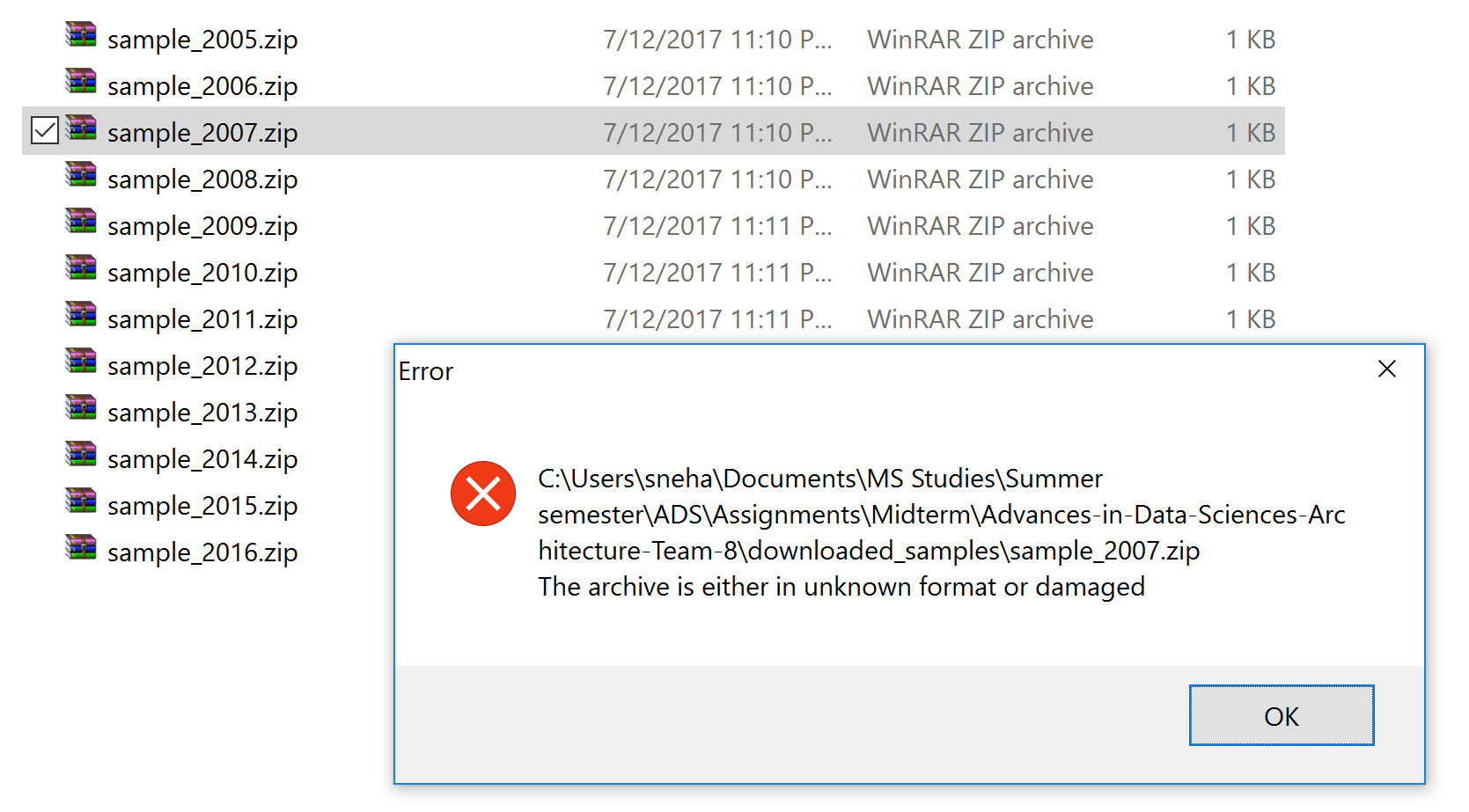
The output from the program suggests that the files have got downloaded to th current directory called “Midterm but in midterm there was a file of 0kb.





The data got downloaded in another folder with the same folder name as before but they are stored as zip files and are of 1kb each, which throws the following error when one tries to open the zipped folder.



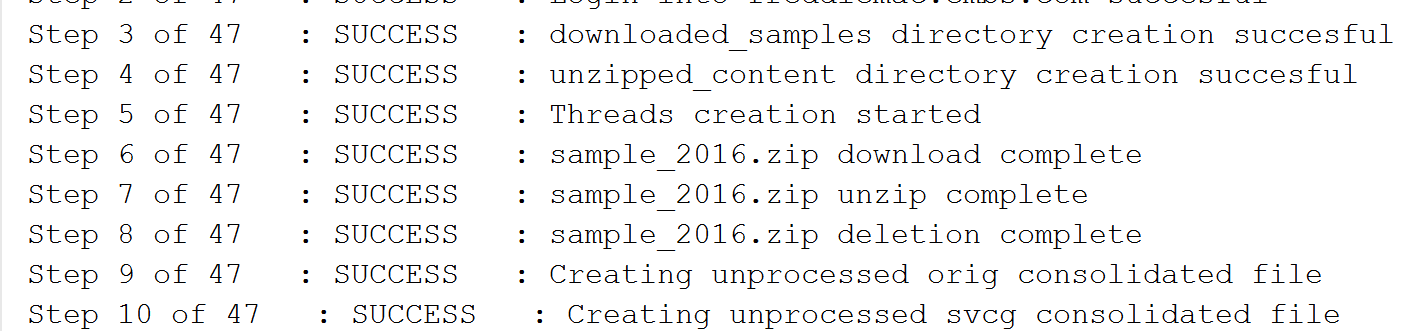


The code threw errors because the function in Urllib2 retieves the file with a wrong formatting.

The urllib.urlretrieve function works well instead.

The data download is done using **threads** parallelly to download and unzip the files and consolidate the files under two categories i.e. SVCG and Orig files.

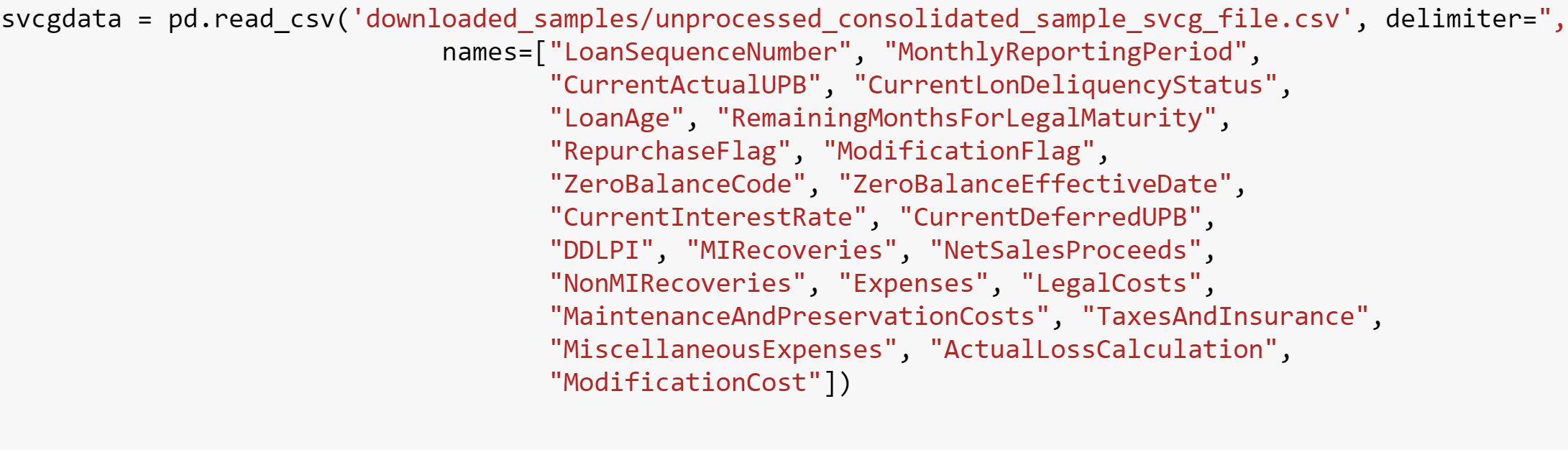
The download unzipping and consolidation is **logged** as follows :



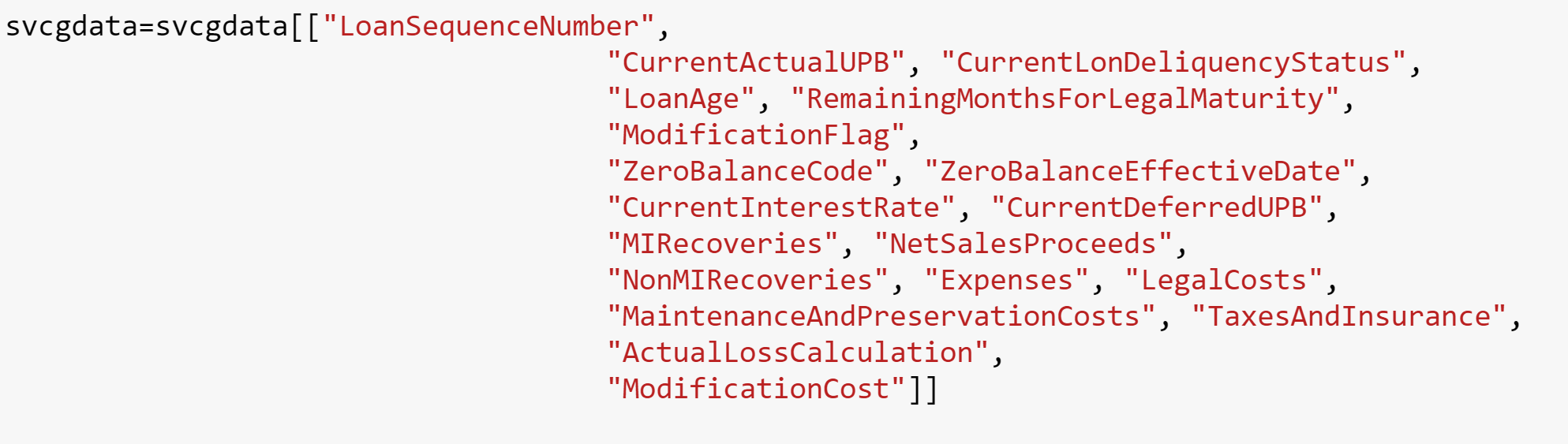
**Data Wrangling:**

The data for the different files is stored in different directories and must be pulled out of the respective directories.

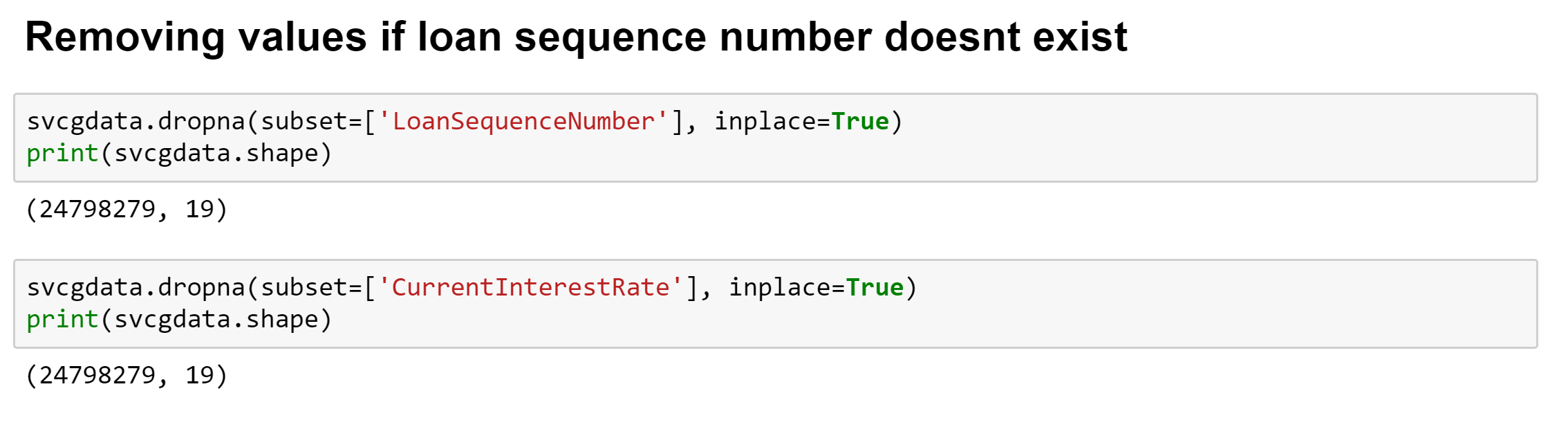
The Wrangling code to pull data and name the column looks as follows :

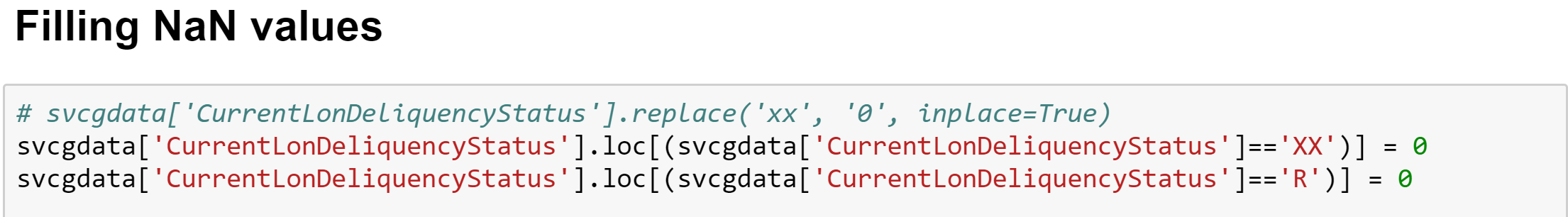


Then we select the columns required and delete the rest of the columns



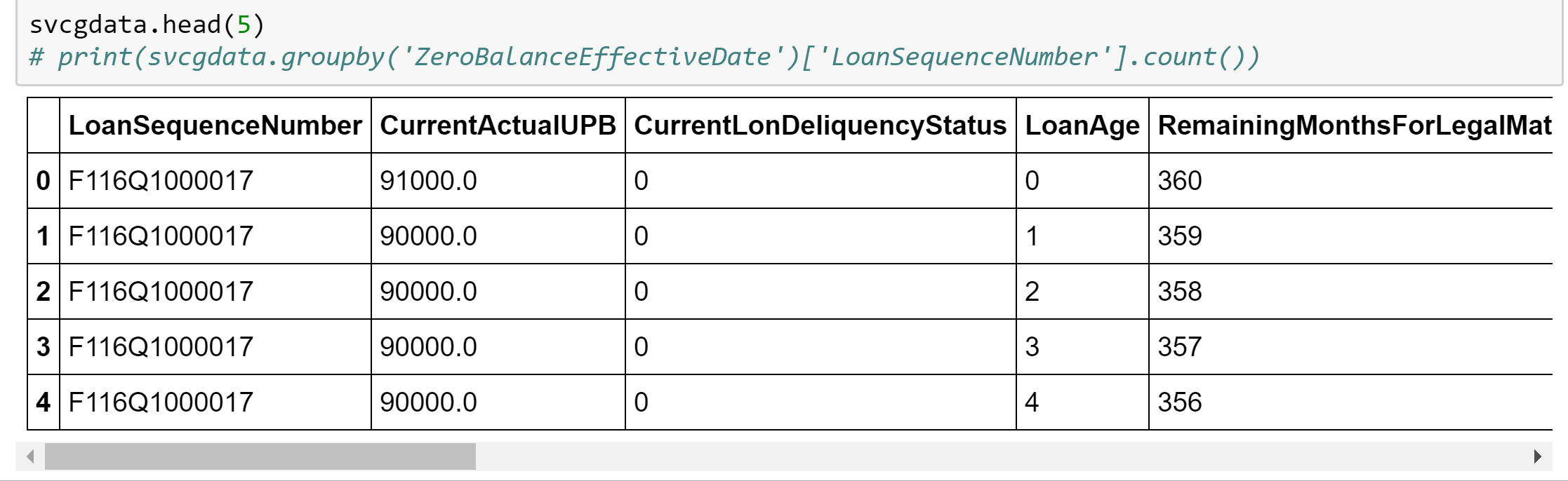
We check for any null values/missing values and remove them



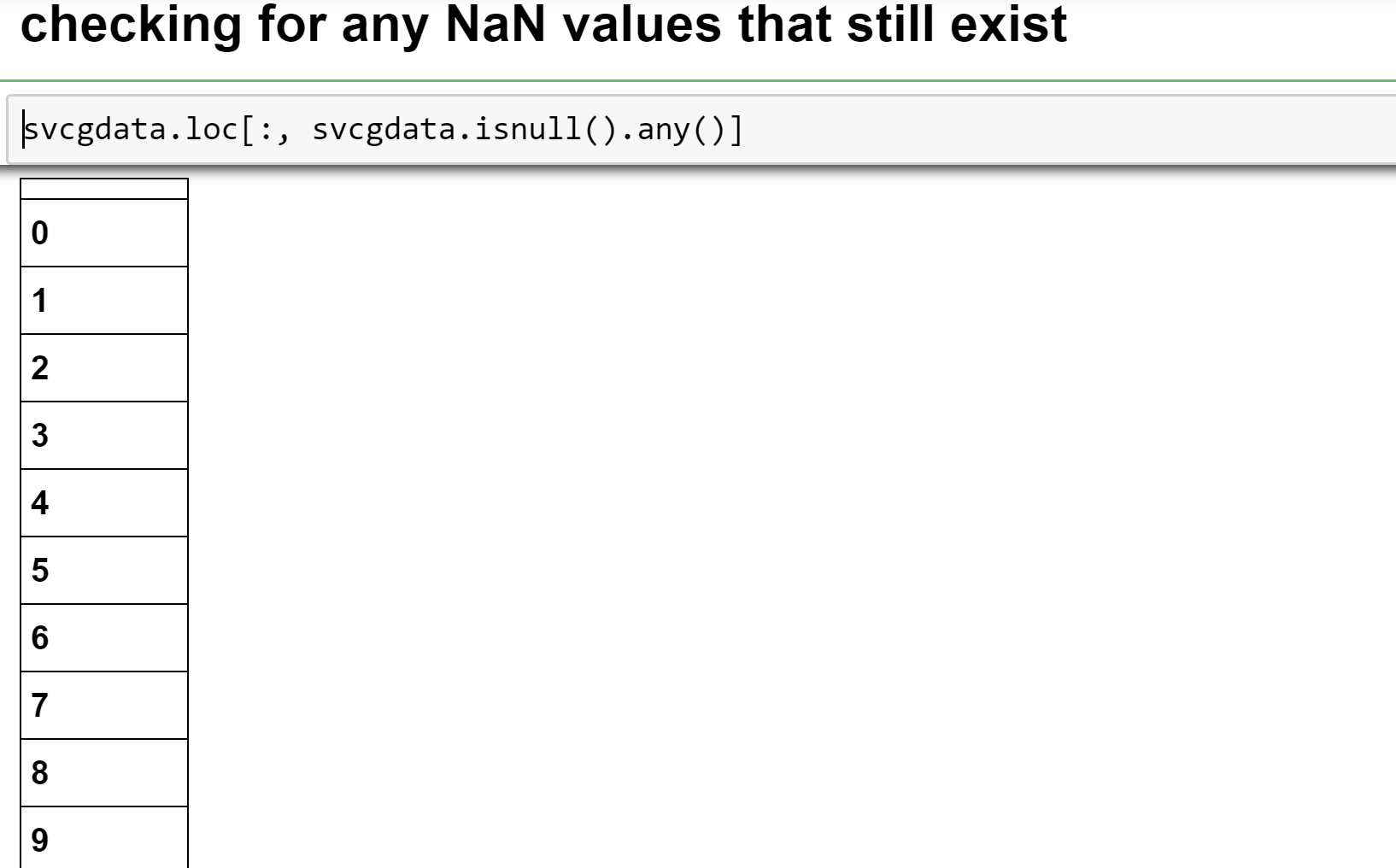








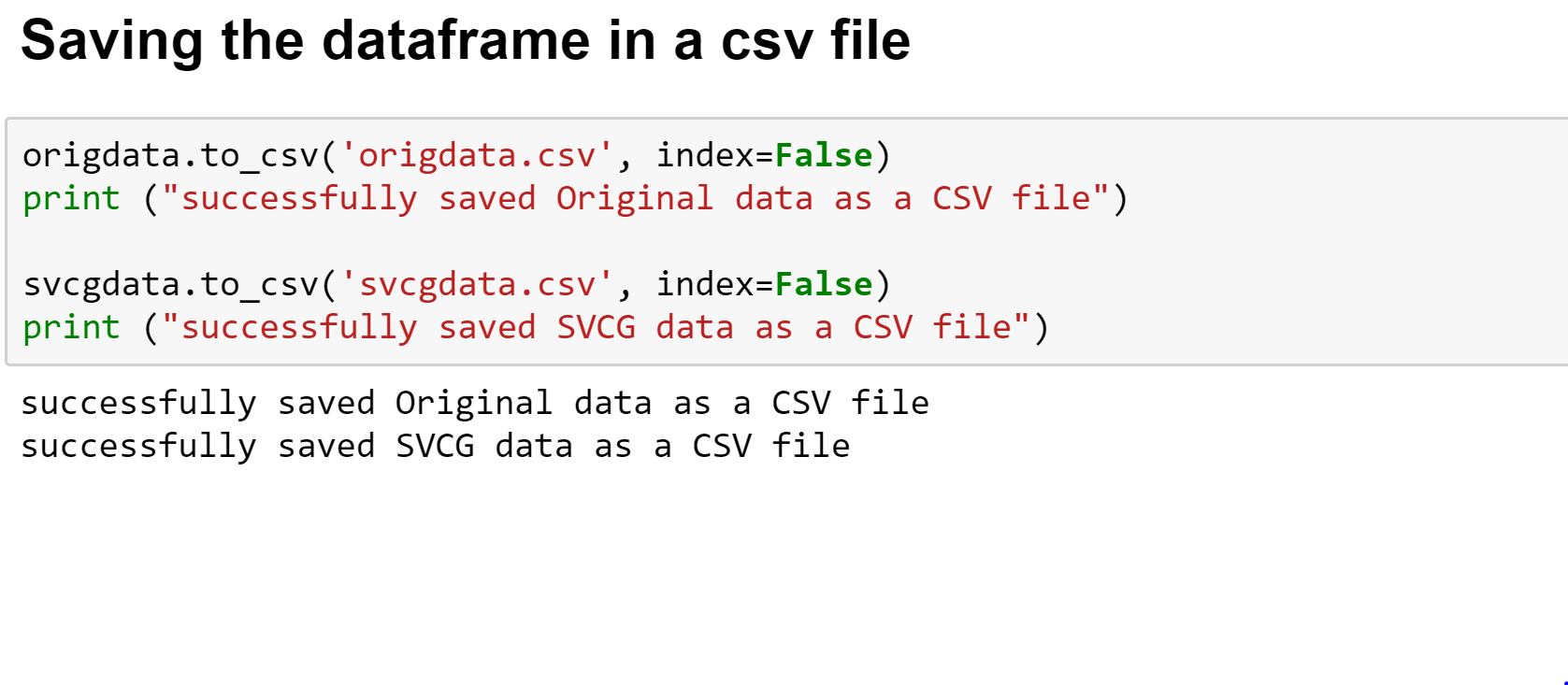
All the NaN values are removed and when we check for the values I looks as below



We try converting the datatypes into their respective expected data type as below :

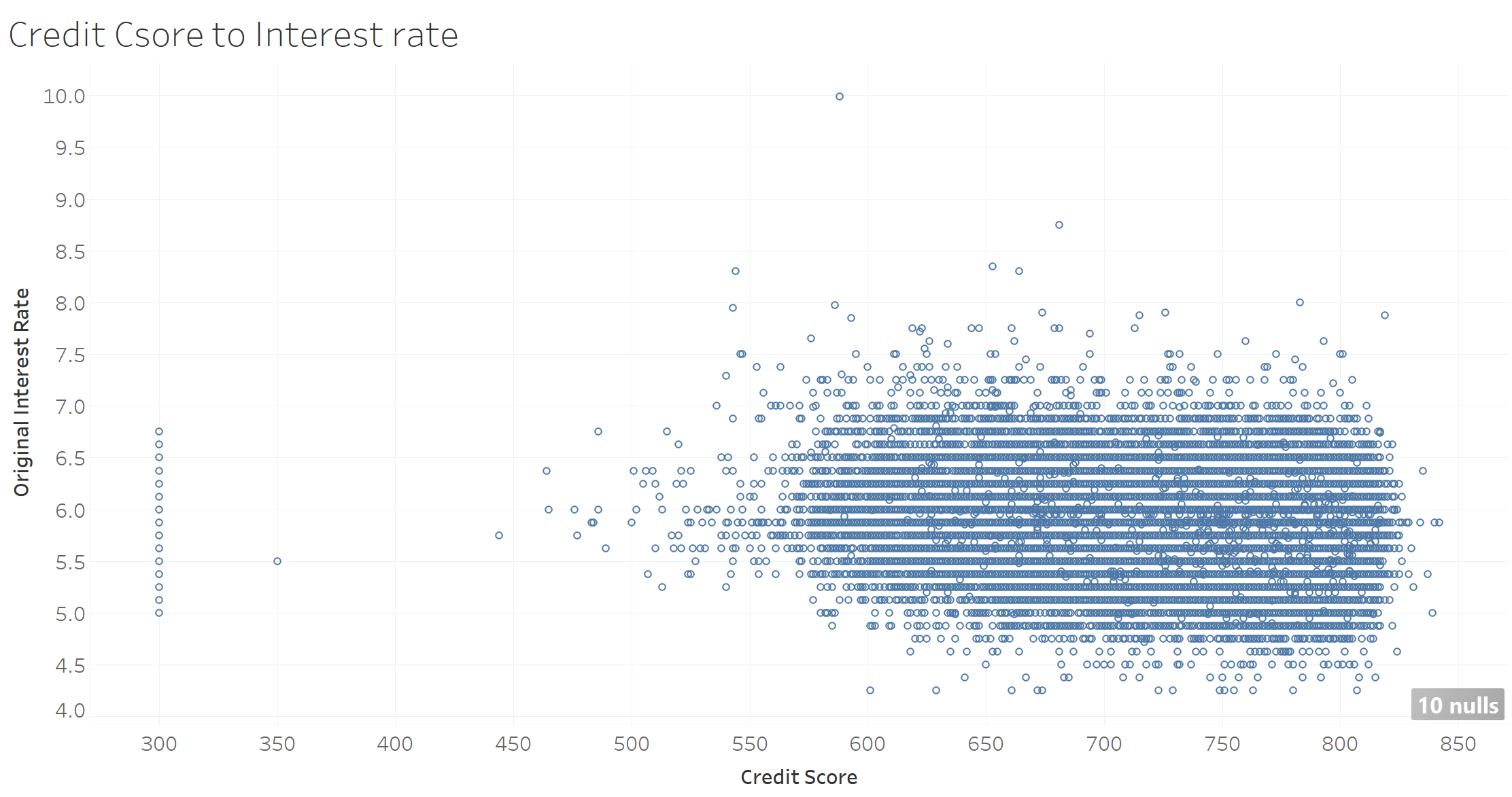


We follow the same steps for the Orig file and then save the data frames as CSV file

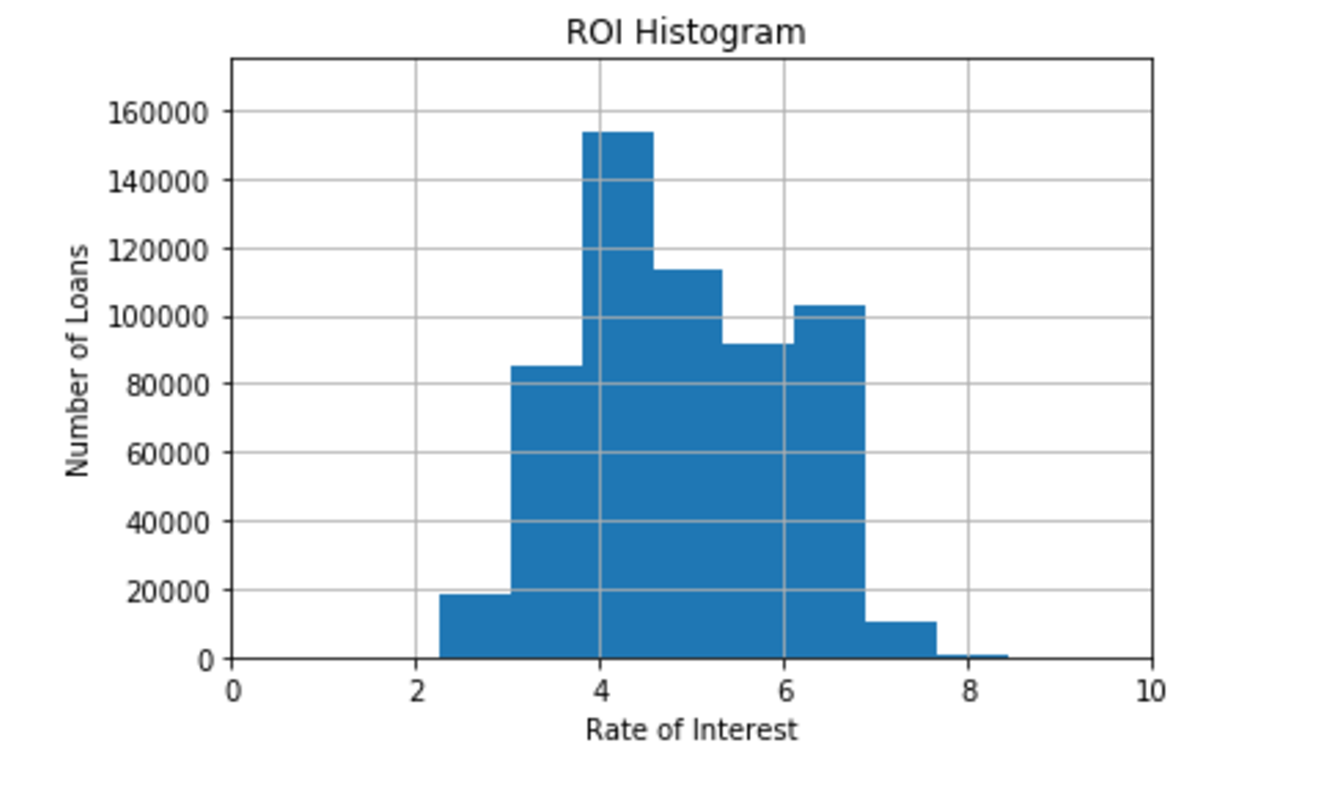


**Exploratory Data Analysis :**

A very important consideration in EDA of this data would be the Credit Score to Interest Rate graph as shown in Tableau as follows :

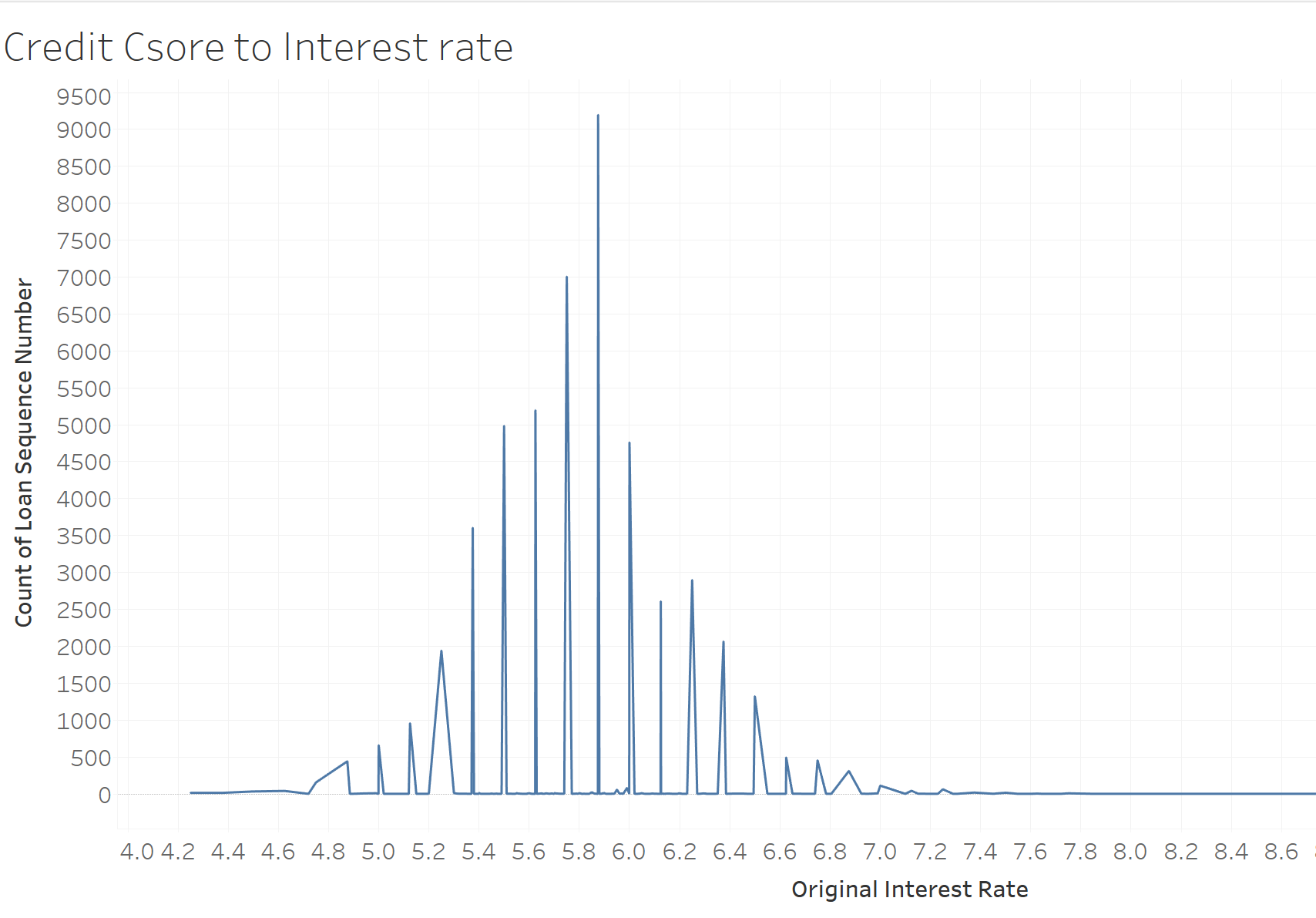


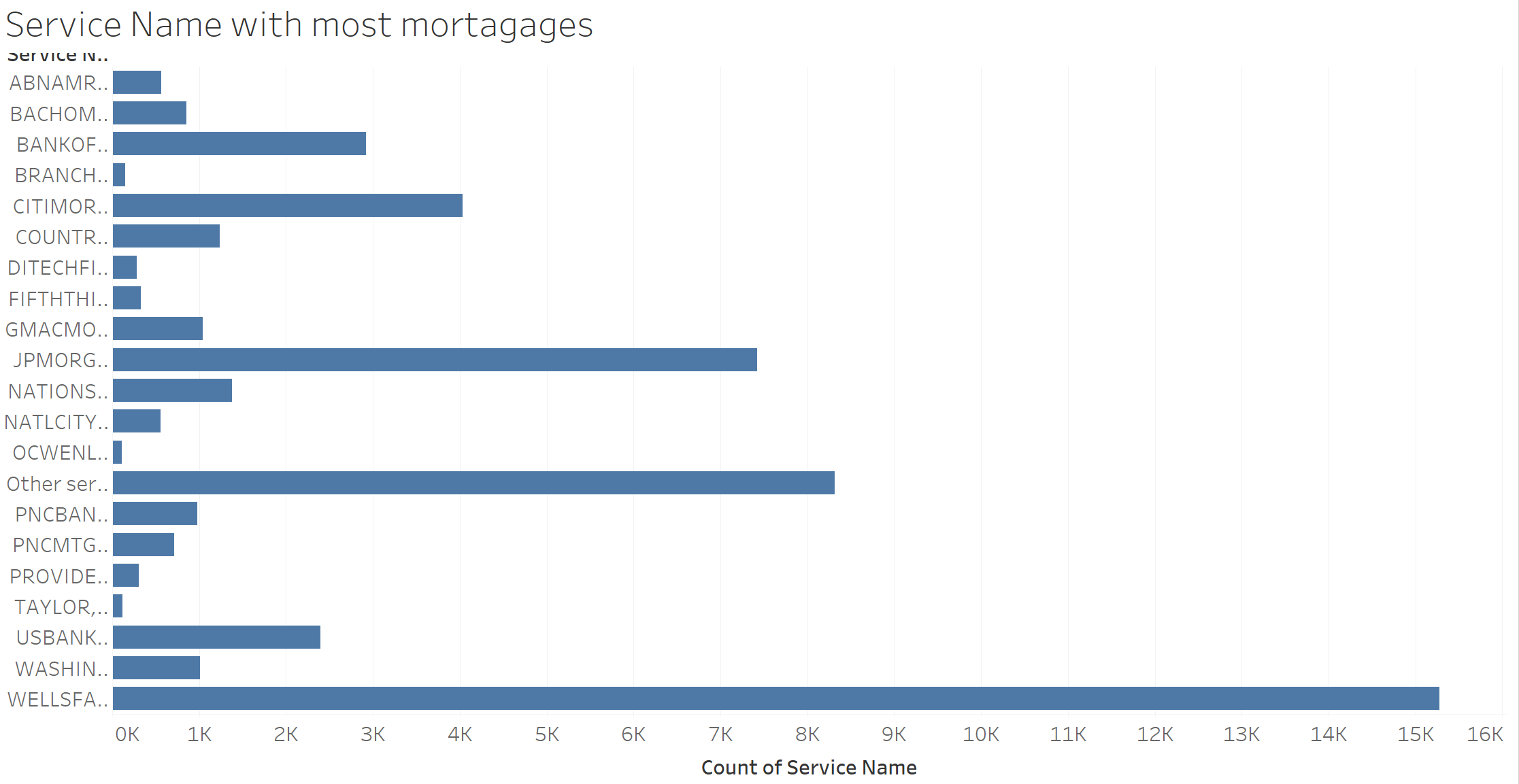
The graph above gives a scatter plot and we can reckon that the people in the categore between 550 to 830 get most loans while the interest rate is distributed between 4.5 to 7.



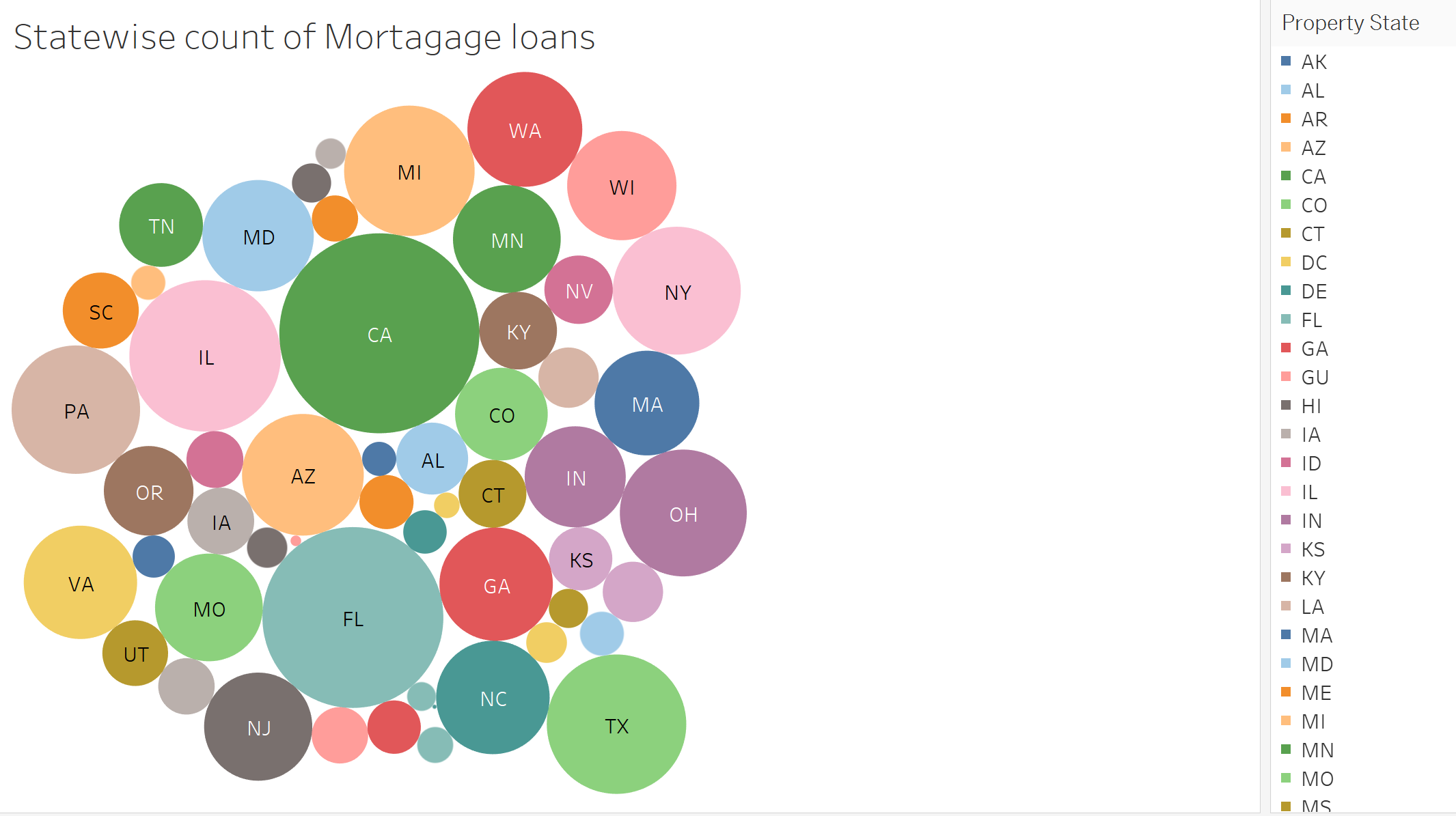
The following graph shows the number of loans against the rate of interest.

This histogram concludes that most loans were given at the rate of around 4 %

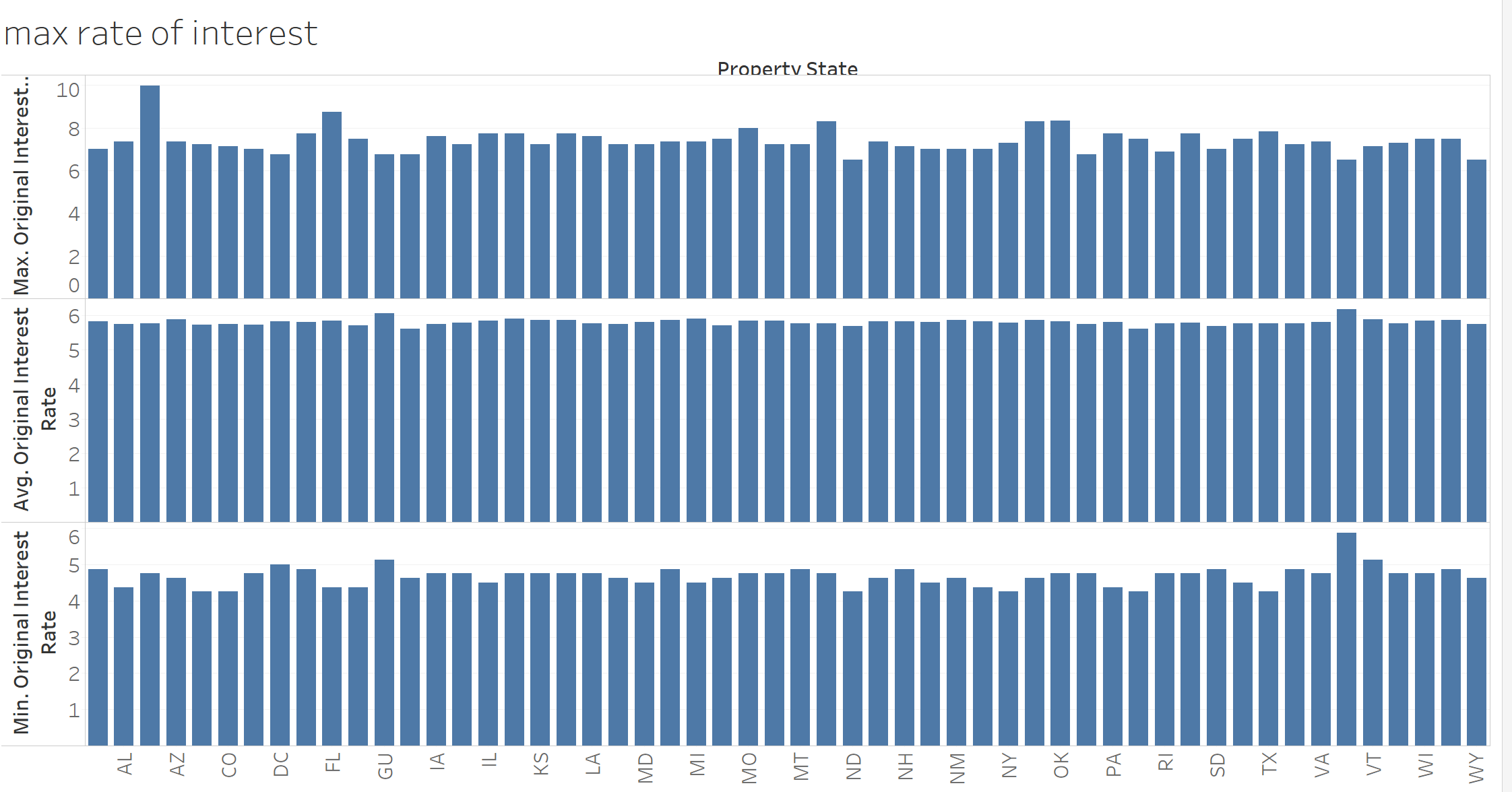




The above graph above shows the number of loans taken under the various service providers.

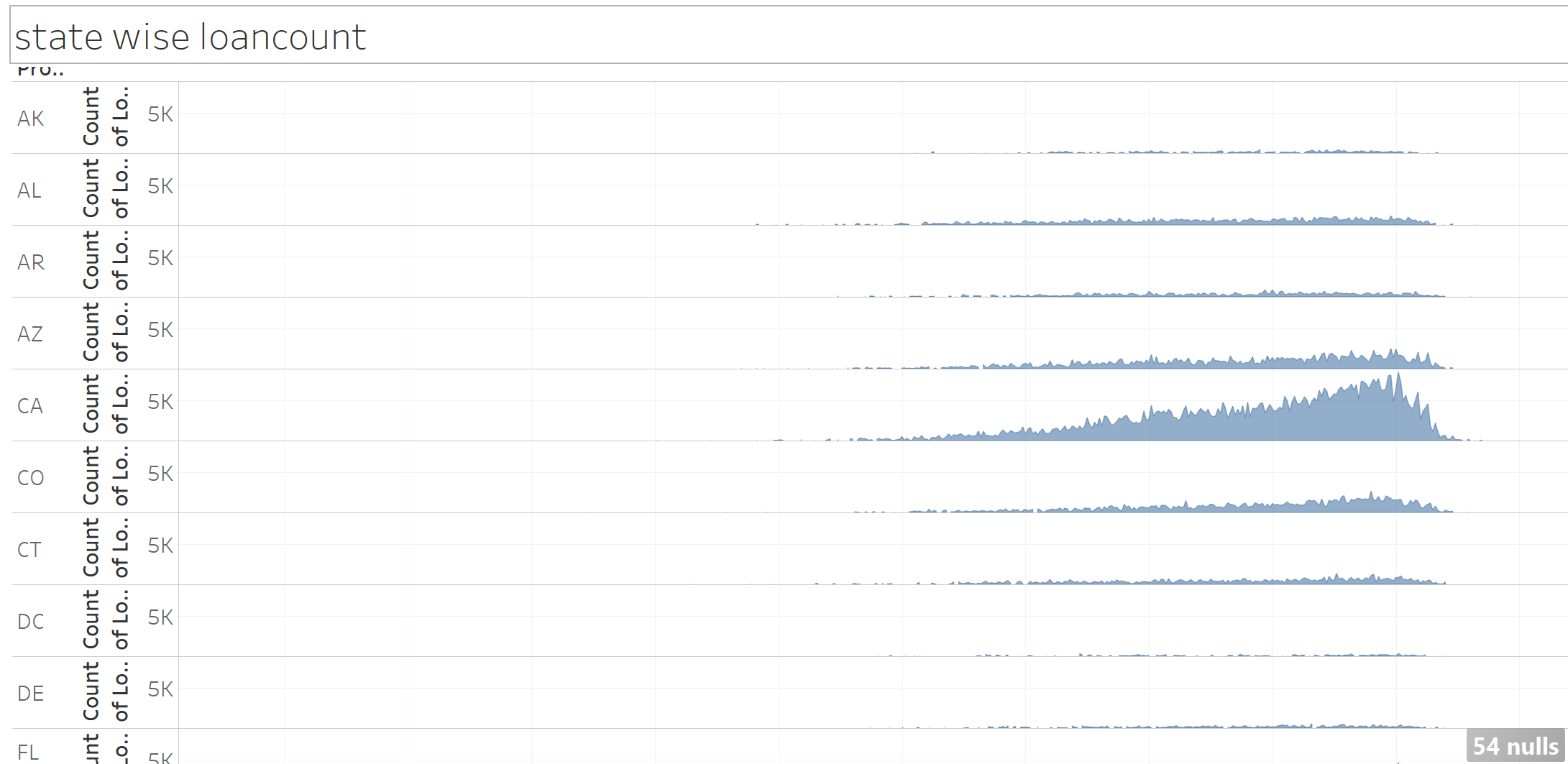


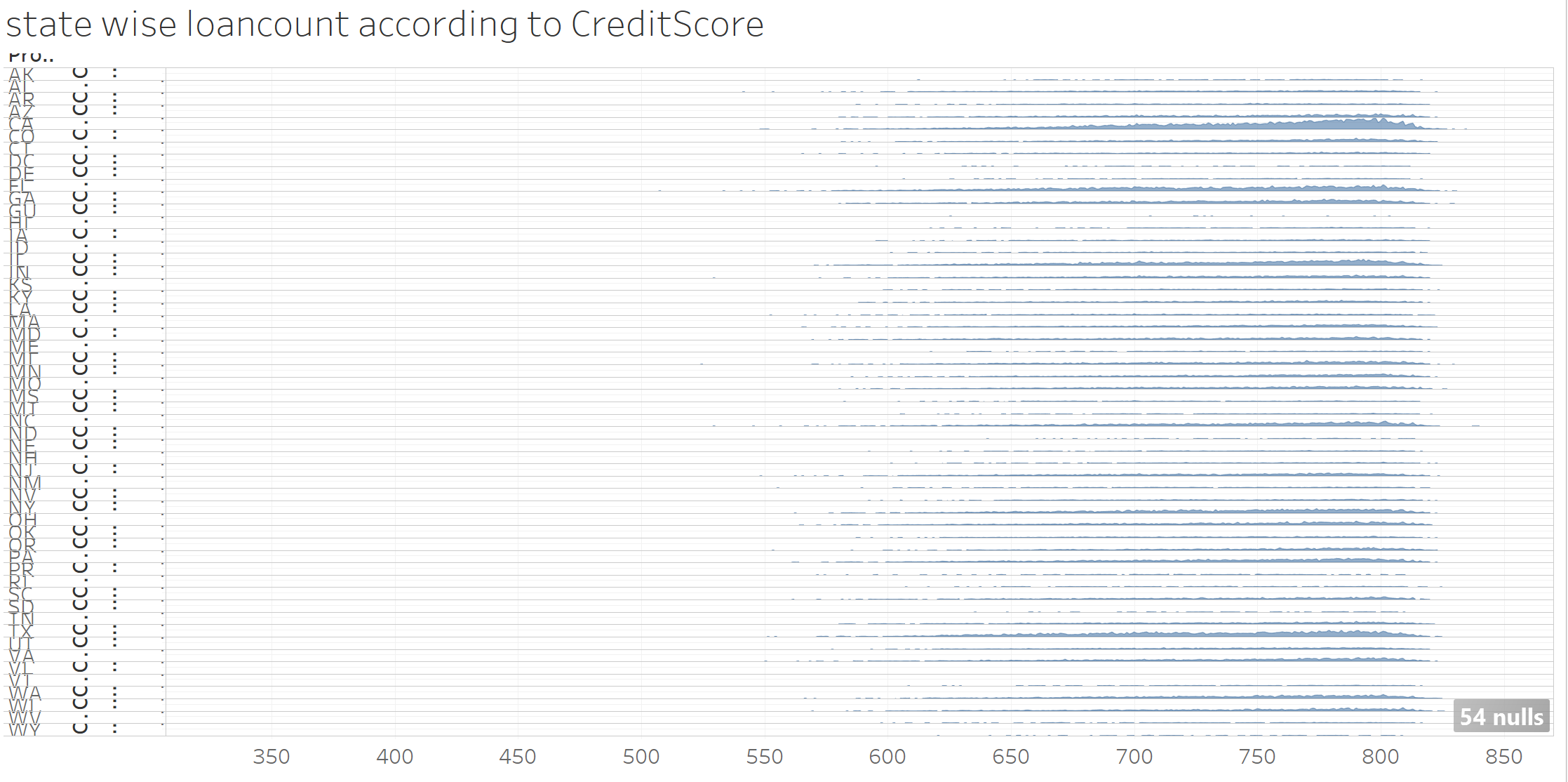
The plot size in the above graph shows the highest loan lender state. California has the highest amount of mortgage loans filed



The graph above shows the average max and minimum amount of ROI paid in the country according t the different states.

For SVCG data





The above graph shows the statewise loan count according to the credit scores of the customers. The graph shows that California had the most loans taken for the credit score of 787

**Part 2**

**1. Prediction:**

Q. Write a prediction script in a Jupyter notebook that given input (For example Q12005),

Programmatically downloads Q12005 and Q22005 origination data and pre-processes it.

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXxxx

PreProcessing

**Prior to any analysis, the data should always be inspected for:**

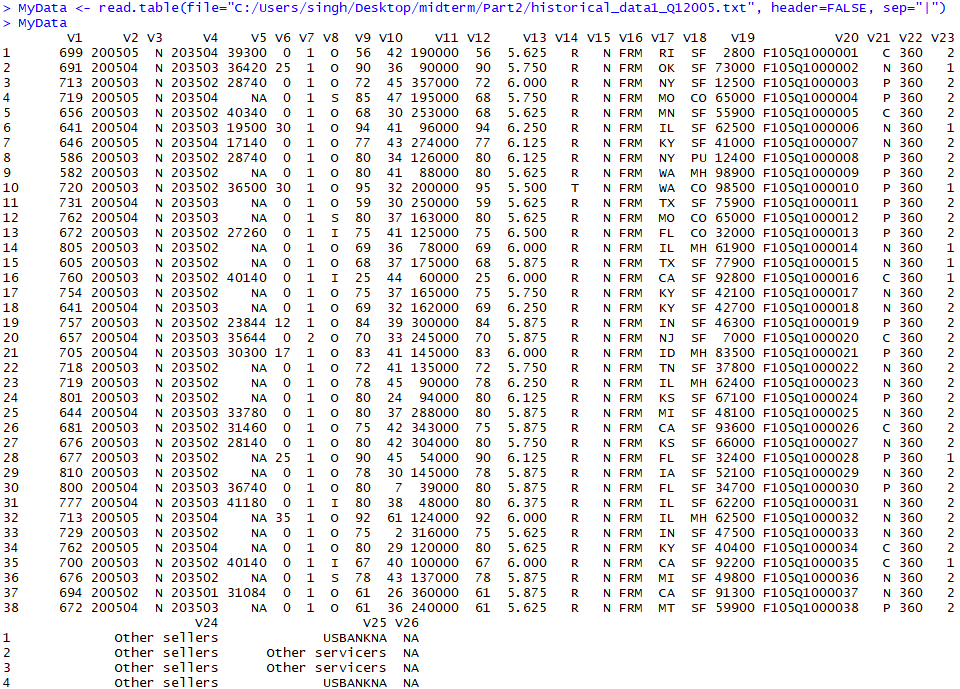
* Data-entry errors
* Missing values
* Outliers
* Unusual (e.g. asymmetric) distributions
* Changes in variability
* Clustering
* Non-linear bivariate relationships
* Unexpected patterns

Does the data look like as we expect? We can resort to:

**Numerical summaries**: − 5-number summaries − correlations − etc.

**Graphical summaries**: − boxplots − histograms − scatterplots − etc.

**Step1: Loading the data**



**Defining the data: Name the variables:**

Define Columns

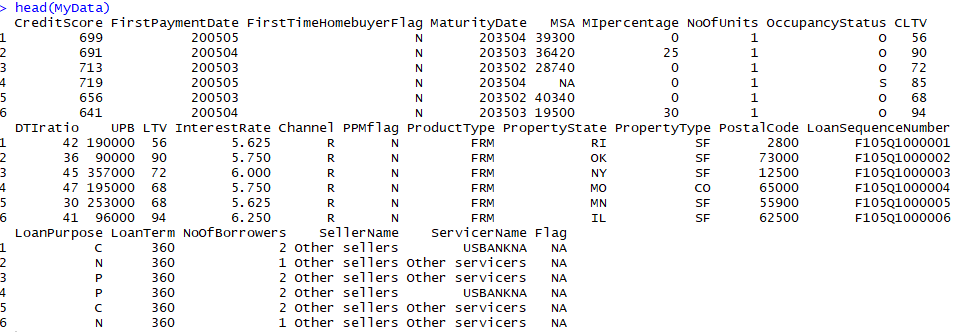
colnames(MyData)<- c("CreditScore", "FirstPaymentDate", "FirstTimeHomebuyerFlag", "MaturityDate",

"MSA", "MIpercentage", "NoOfUnits", "OccupancyStatus", "CLTV", "DTIratio",

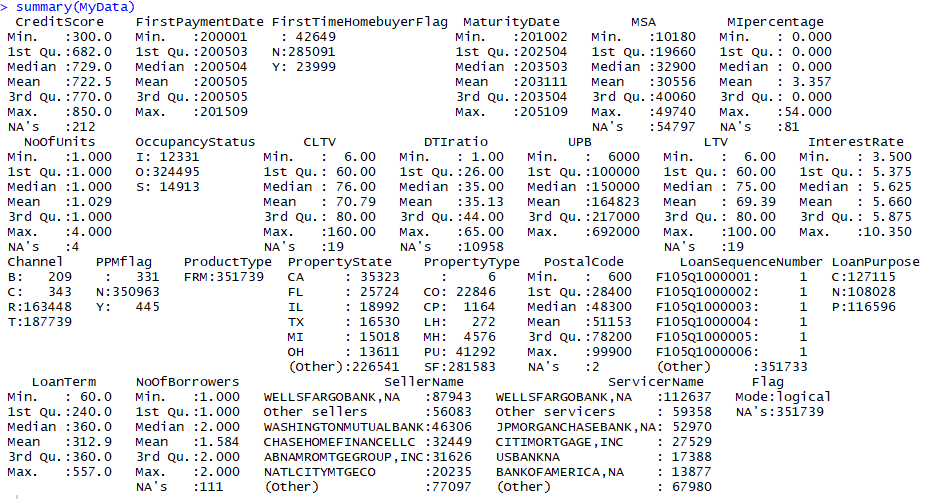
"UPB","LTV", "InterestRate", "Channel", "PPMflag", "ProductType", "PropertyState",

"PropertyType", "PostalCode", "LoanSequenceNumber", "LoanPurpose", "LoanTerm",

"NoOfBorrowers", "SellerName", "ServicerName", "Flag")



Check the summary of the dataset



From the Dataset Summary we can see there are number of NAs/ not available data in columns.

Change the data/ column value with meaningful numerical values for prediction and analysis.

**Step 2: Numerical Summaries:**

1. Filling the Missing Data:

Column1: Credit Score 301-850 ; Spaces(3) = unknown, if CS is <301 or >850

So we change it spaces(3) with 301-850 mean

Problem faced in Flag values

f

**Preprocess data Remove NA with mean or select the variables**

MyData$CreditScore[which(is.na(MyData$CreditScore))]<-0

MyData$CreditScore[MyData$CreditScore == 0] <- mean(MyData$CreditScore)

summary(MyData$CreditScore)

MyData$FirstPaymentDate[which(is.na(MyData$FirstPaymentDate))]<-0

MyData$MSA[which(is.na(MyData$MSA))]<-0

MyData$MIpercentage[which(is.na(MyData$MIpercentage))]<-0

MyData$NoOfUnits[which(is.na(MyData$NoOfUnits))]<-0

MyData$CLTV[which(is.na(MyData$CLTV))]<-0

MyData$DTIratio[which(is.na(MyData$DTIratio))]<-0

MyData$UPB[which(is.na(MyData$UPB))]<-0

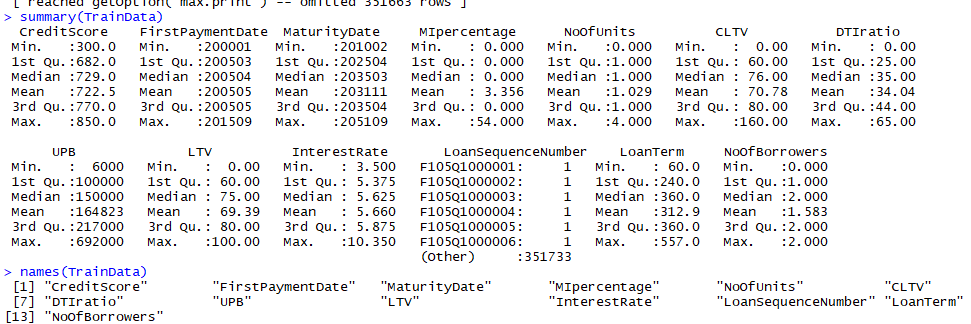
MyData$LTV[which(is.na(MyData$LTV))]<-0

MyData$InterestRate[which(is.na(MyData$InterestRate))]<-0

MyData$LoanTerm[which(is.na(MyData$LoanTerm))]<-0

MyData$NoOfBorrowers[which(is.na(MyData$NoOfBorrowers))]<-0

MyData$SellerName[which(is.na(MyData$SellerName))]<-0



Q. Builds a Regression model for the interest rate using Q12005 data as training data (col 13)

Regression analysis is used to describe the relationship between the independent variables X and how its effecting the dependent Variable Y in the dataset.

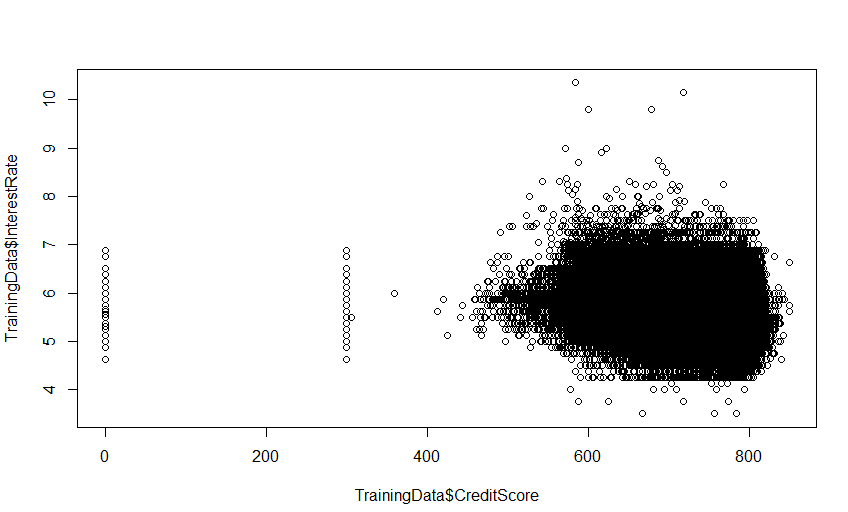
A single response variable: Y ; One or more predictor variables: X1, X2,..., Xp

p = 1: Simple Regression

p > 1: Multivariate Regression

**Credit Score** Missing Value is changed with mean as the NULL = value range between 301-850.

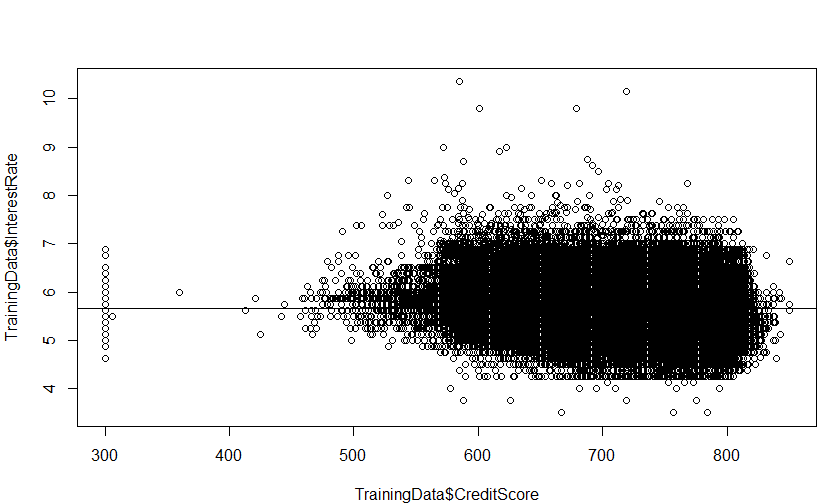
Calculate The mean of Interest Rate:



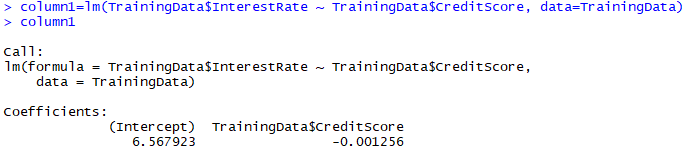
> m=mean(TrainingData$InterestRate)

> abline(h=m)

We calculated the mean of the interest rate and checked the average of Credit Score with Mean Interest Rate



Now we use linear Regression Model (lm) to fit a regression line

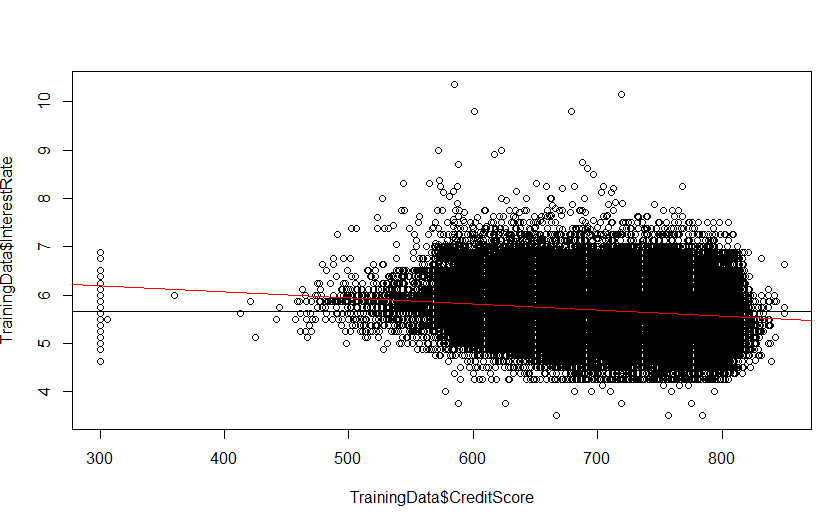


> abline(column1,col="red")

Here intercept is the value of Interest Rate = 6.567923

Whereas the other term is slope: y^/x^= -0.001256 (-ve relation)

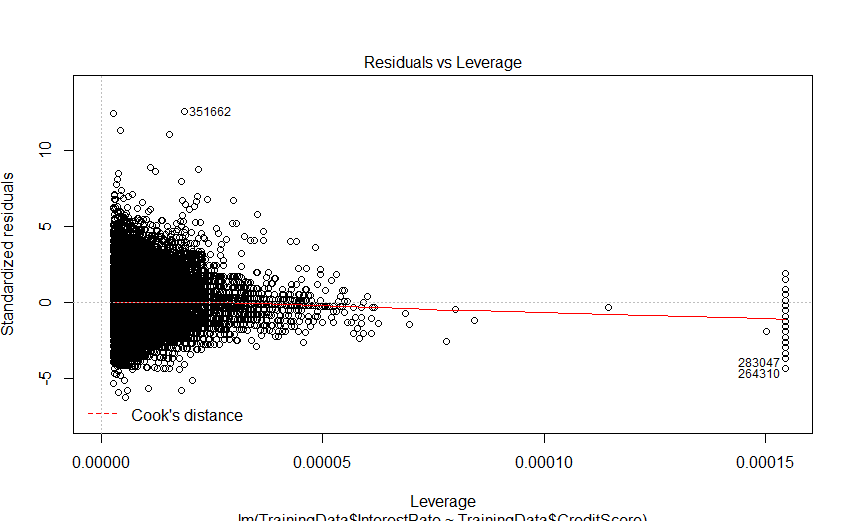
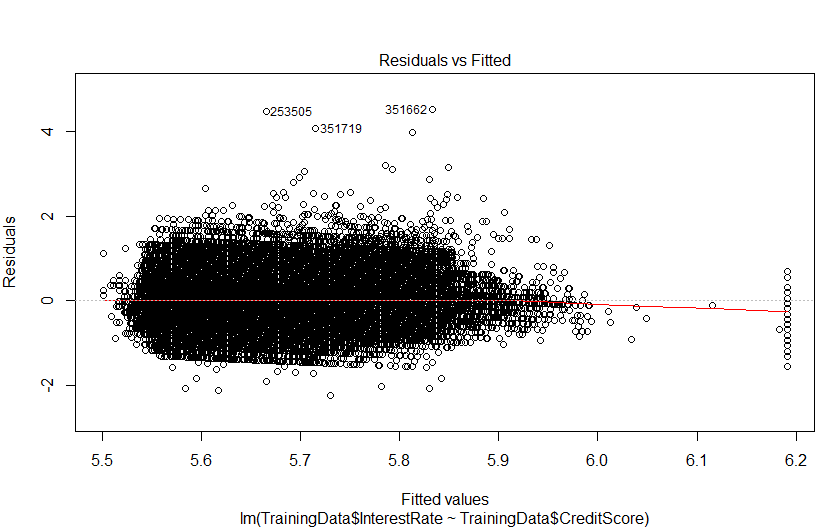
**Indicates: More the Credit Score less will be the Interest Rate**



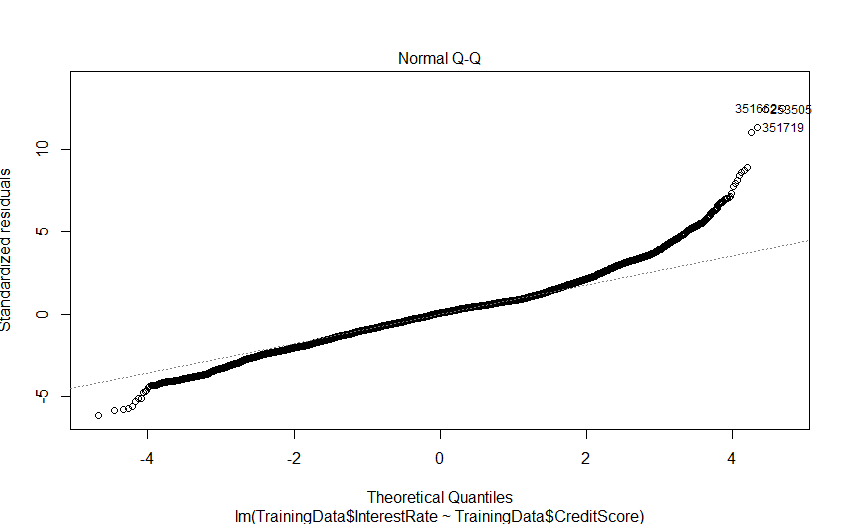
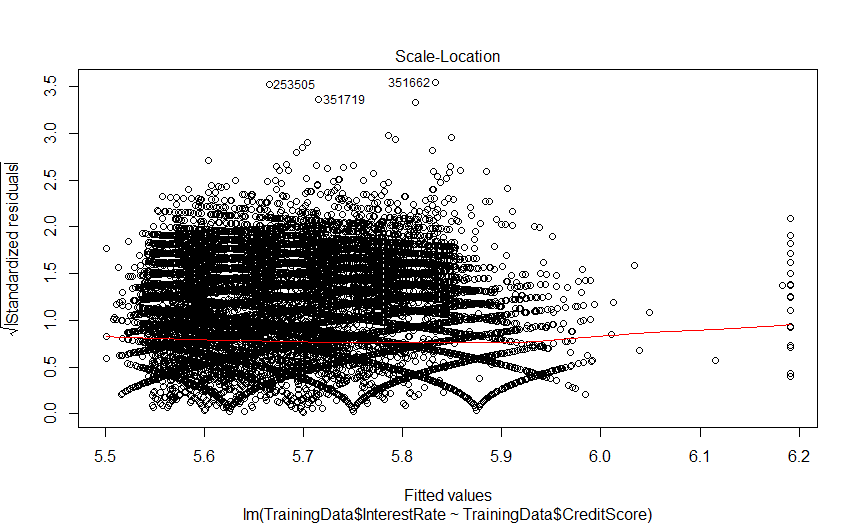
Here the variance decreases with the mean increases, higher the credit Score lower the Interest Rate on Mortgage.

So now we can look at the residuals:

> plot(column1)

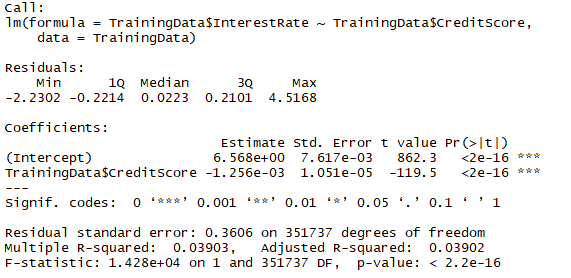
 

Plot 3: Standard residuals vs theoritical chows: wheather residuals are normaly distributed or not, but here we can see the deviations starting below the line here, which shows distribution is shaped differently than the normal distribution.(not really important for preditions)

**So we consider the plot Residuals vs Fitted value graph to check if the Credit score is best suited to determine the Interest rate Predictor.**

**Summary :**



Median of the residuals is : 0.0223, (ideally should be zero)

Mininimum : -2.23.0 & Maximum residual= 4.5168 which is closely distributed along 0 value here.

Most important here are the Coefficients and the standard Error:

Here we have the confident lever the intercept estimated is not 0

Also, the slope(-1.256e-03) is also different from 0.

Also here R-squared: 0.03903,

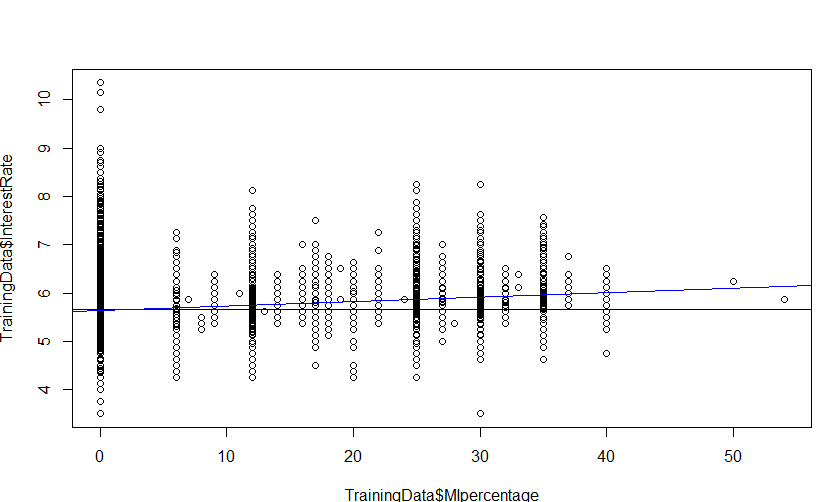
Adjusted R-squared: 0.03902 (for the degrees of freedom)

Now we use Multiple Regression to check the joint effect of multiple values on Interest Rate

For column 6: Motargage insurance and how it effects the Interest rate:



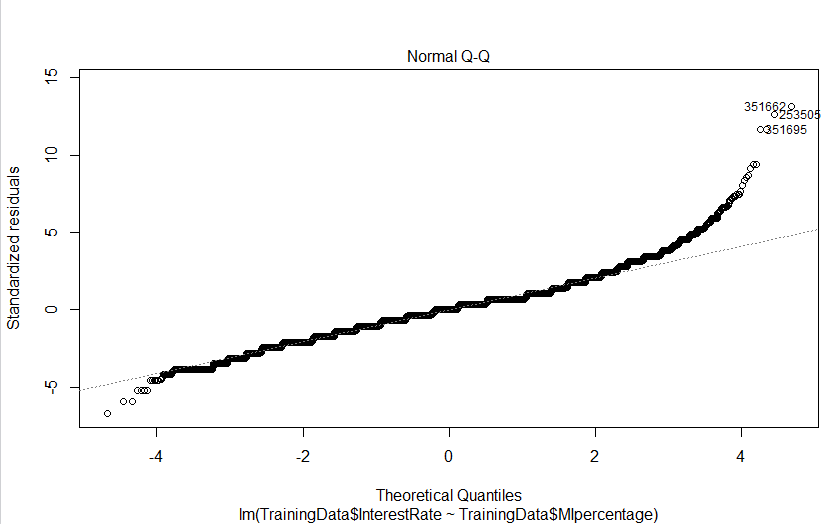
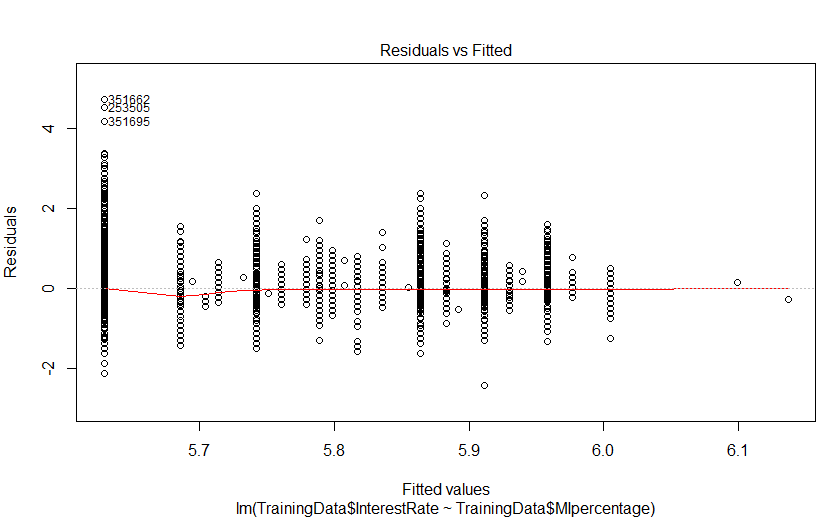
Here the Coefficient is Positive so its effect on RATE of Interest is Positive

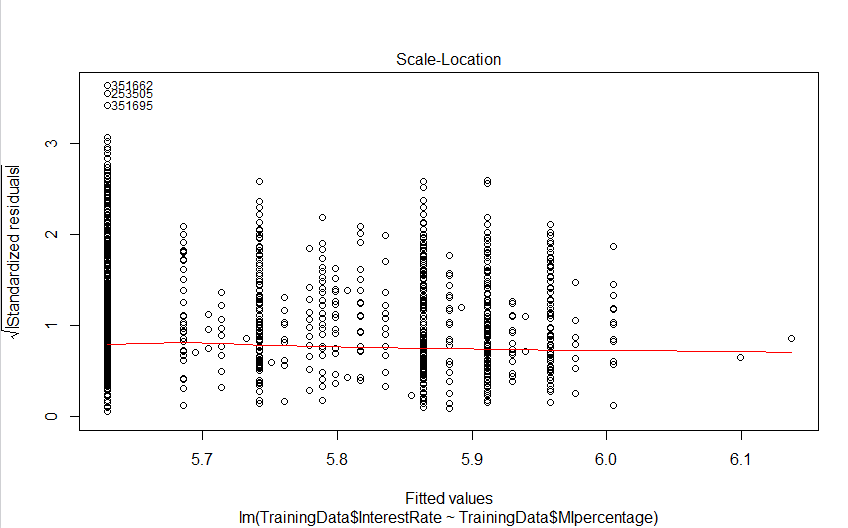
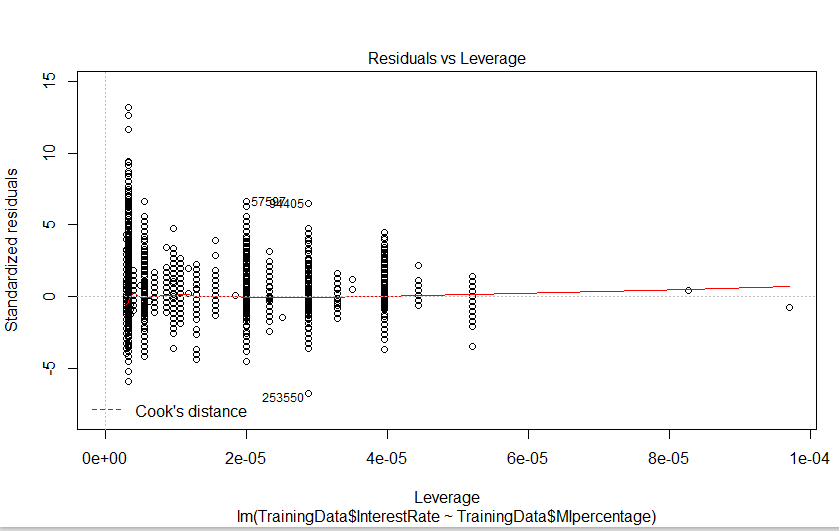


MI% is 0 then there is no influence on the Interest Rate but as it increases it increases the Interest Rate

More the MI% than more is the Interest Rate increases on the property.

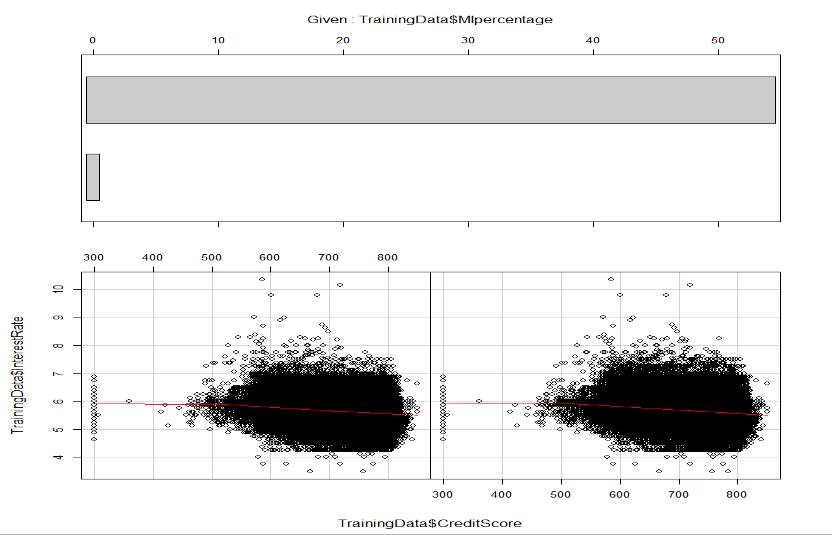
Residuals:

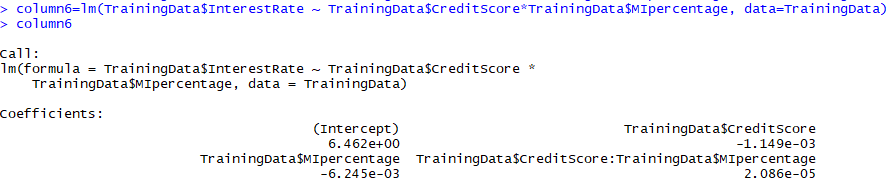


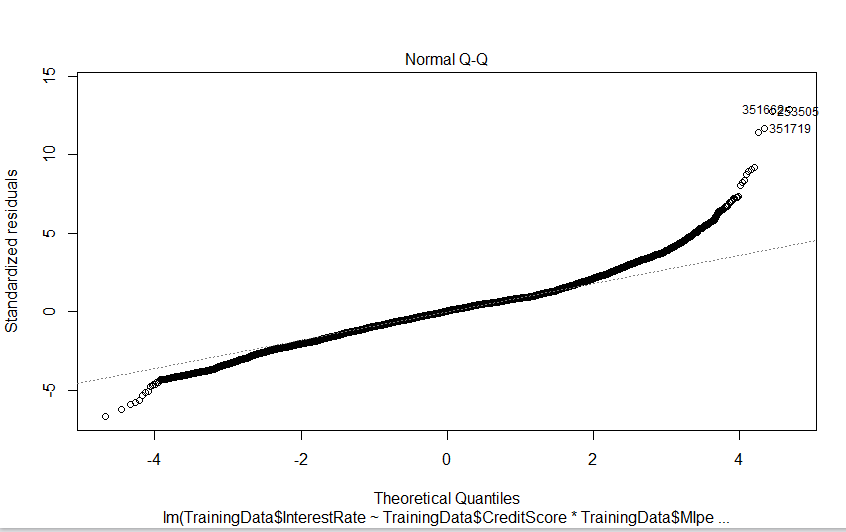
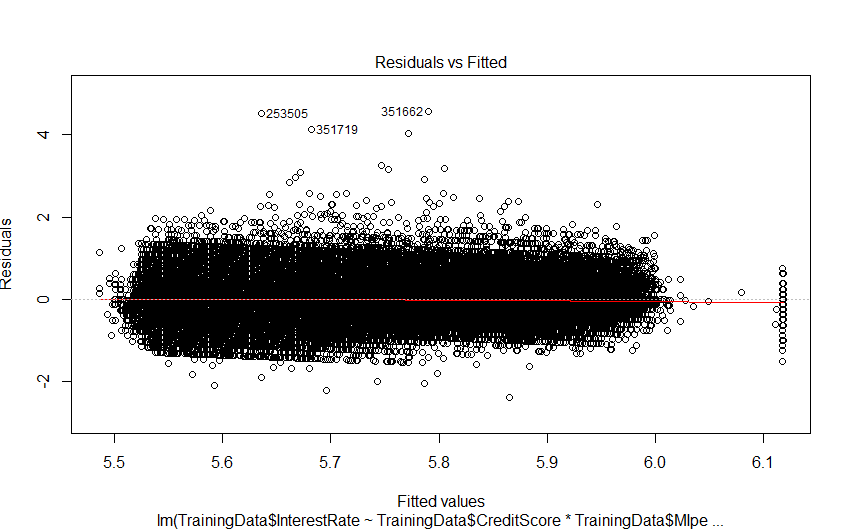
Now if we want to check how MIpercentage changes with the credit score with Interest Rate on the Motergage

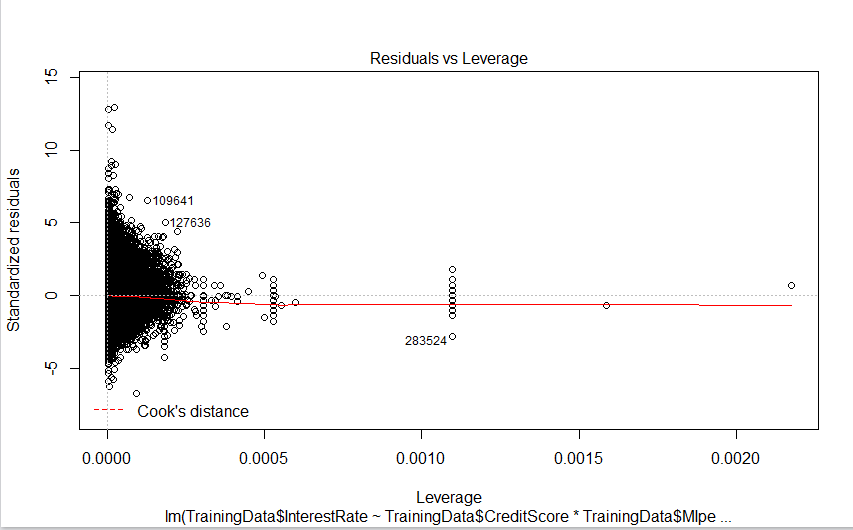
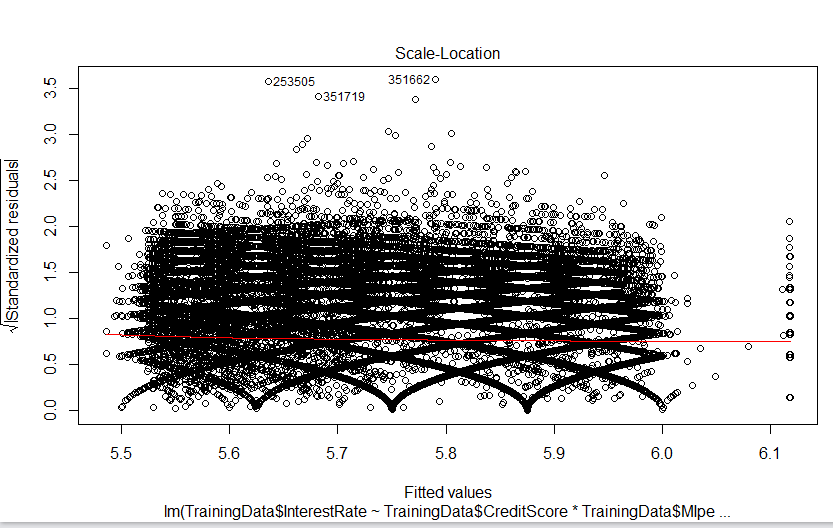
coplot(TrainingData$InterestRate~ TrainingData$CreditScore|TrainingData$MIpercentage, panel = panel.smooth)

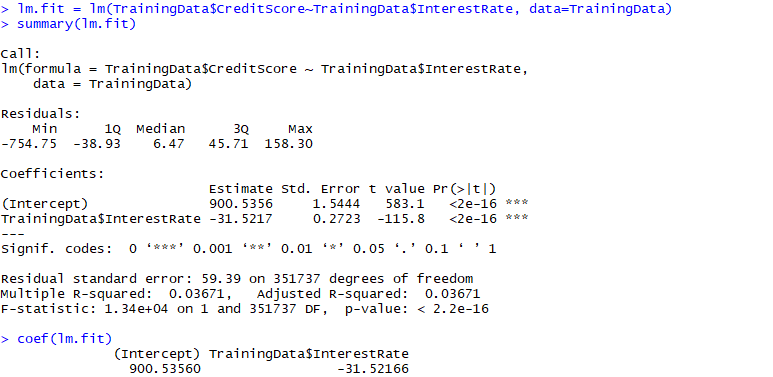


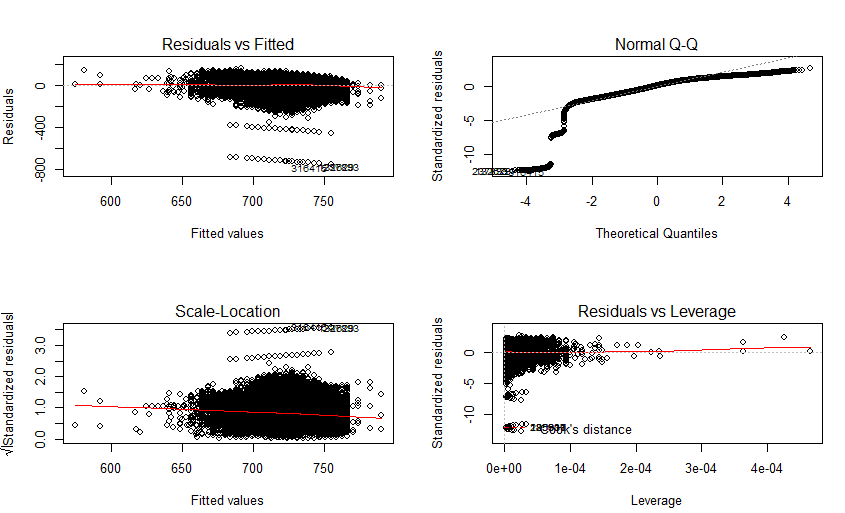


It is linear relationship with the fitted value and Residuals

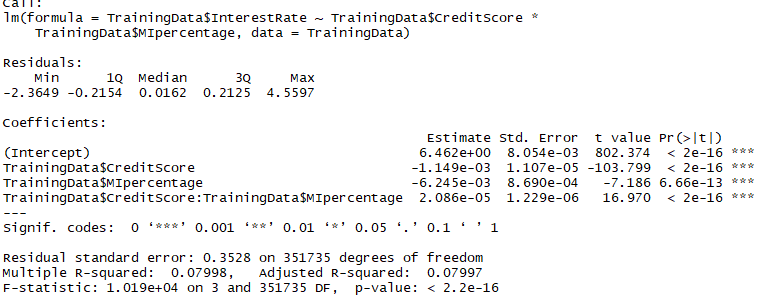








summary(column6)

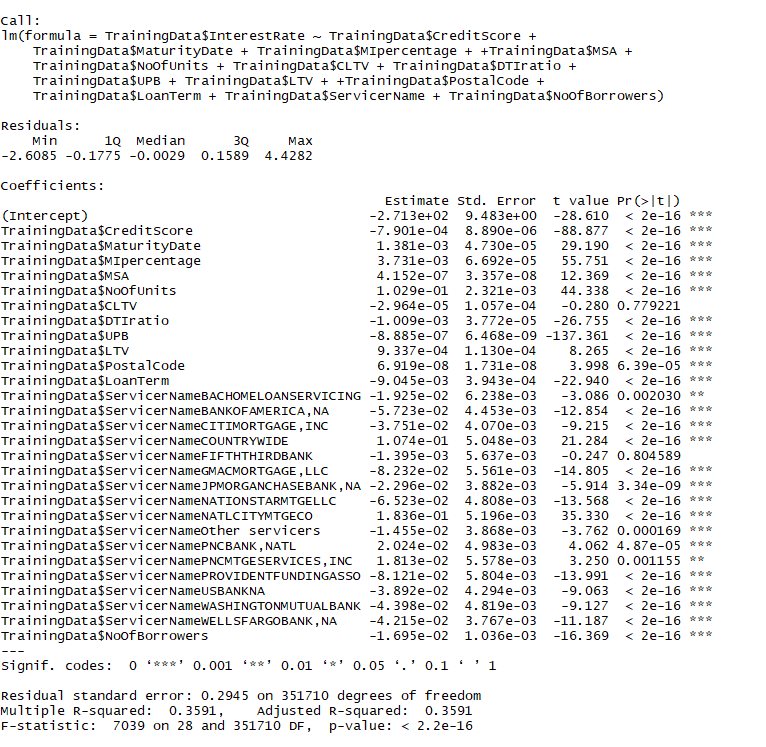


Multiple Regression for all the selected Variables:

> lm.fit=lm(TrainingData$InterestRate~TrainingData$PostalCode+TrainingData$LoanTerm+TrainingData$ServicerName+TrainingData$NoOfBorrowers)

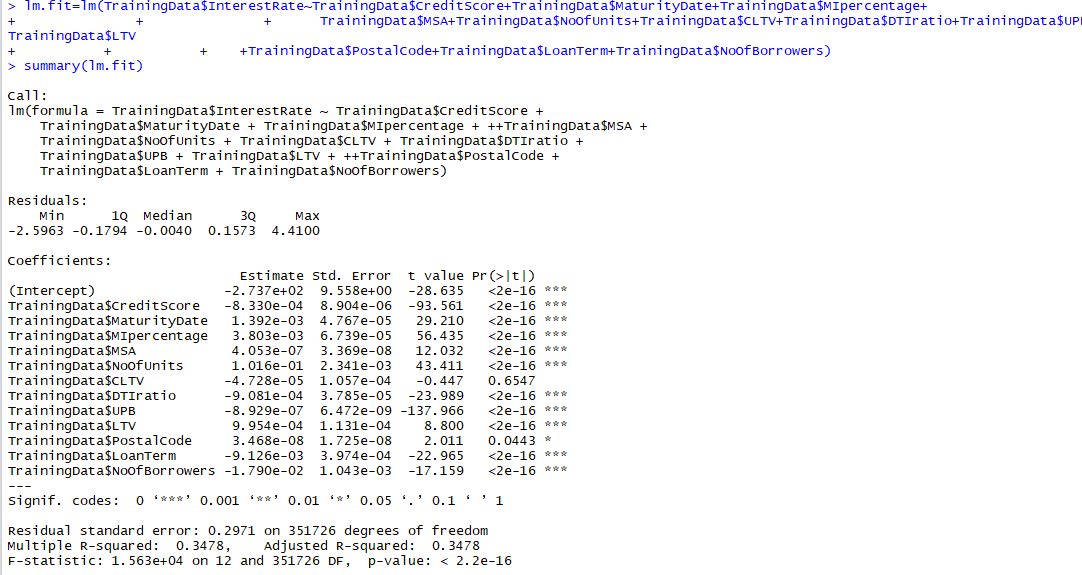
> lm.fit=lm(TrainingData$InterestRate~TrainingData$CreditScore+TrainingData$MaturityDate+TrainingData$MIpercentage + TrainingData$MSA+TrainingData$NoOfUnits+TrainingData$CLTV+TrainingData$DTIratio+TrainingData$UPB+TrainingData$LTV +TrainingData$PostalCode+TrainingData$LoanTerm+TrainingData$ServicerName+TrainingData$NoOfBorrowers)

> summary(lm.fit)



Out of these we do the variable selection using the most significant variable coefficient: and we get due to vector dataset column: state name gives multiple coefficients for Interest Rate.

So, we removed for analysis property State (though it affects the Interest Rate)



So with variable selection we have 11 columns:

TrainingData$CreditScore -8.330e-04 8.904e-06 -93.561 <2e-16 \*\*\*

TrainingData$MaturityDate 1.392e-03 4.767e-05 29.210 <2e-16 \*\*\*

TrainingData$MIpercentage 3.803e-03 6.739e-05 56.435 <2e-16 \*\*\*

TrainingData$MSA 4.053e-07 3.369e-08 12.032 <2e-16 \*\*\*

TrainingData$NoOfUnits 1.016e-01 2.341e-03 43.411 <2e-16 \*\*\*

TrainingData$CLTV -4.728e-05 1.057e-04 -0.447 0.6547

TrainingData$DTIratio -9.081e-04 3.785e-05 -23.989 <2e-16 \*\*\*

TrainingData$UPB -8.929e-07 6.472e-09 -137.966 <2e-16 \*\*\*

TrainingData$LTV 9.954e-04 1.131e-04 8.800 <2e-16 \*\*\*

TrainingData$PostalCode 3.468e-08 1.725e-08 2.011 0.0443 \*

TrainingData$LoanTerm -9.126e-03 3.974e-04 -22.965 <2e-16 \*\*\*

TrainingData$NoOfBorrowers -1.790e-02 1.043e-03 -17.159 <2e-16 \*\*\*

TrainingData$ServicerName -5.715e-02 4.449e-03 -12.847 < 2e-16 \*\*\*

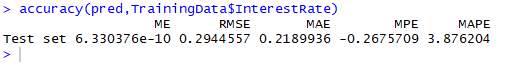
So We have Train Data Set with the selected Vaiables

Training CLTV, PostalCode has no significant coefficient so we can drop that column and rest of the columns have much significant affect Interst Rate on dataset

Here the values of

R-squared: 0.3478, Adjusted R-squared: 0.3478

Now:

* Run the model on the test set
* Get the measures of predictive accuracy
* 

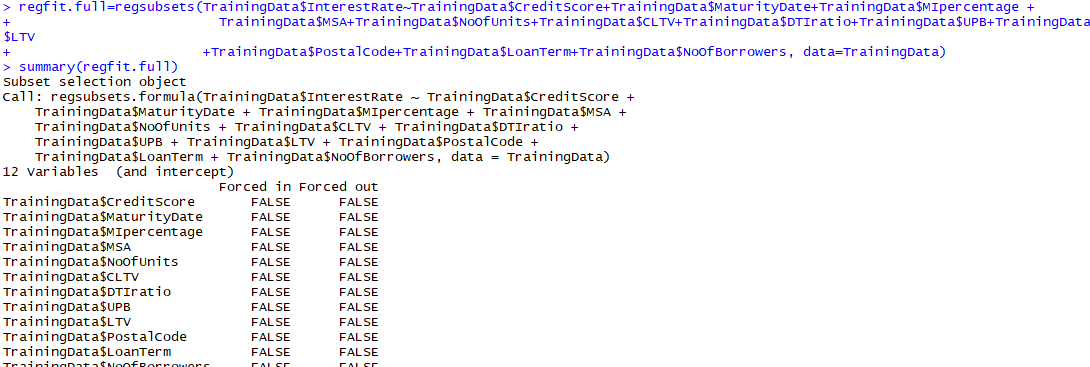
# **Estimated Multiple Regression Equation**

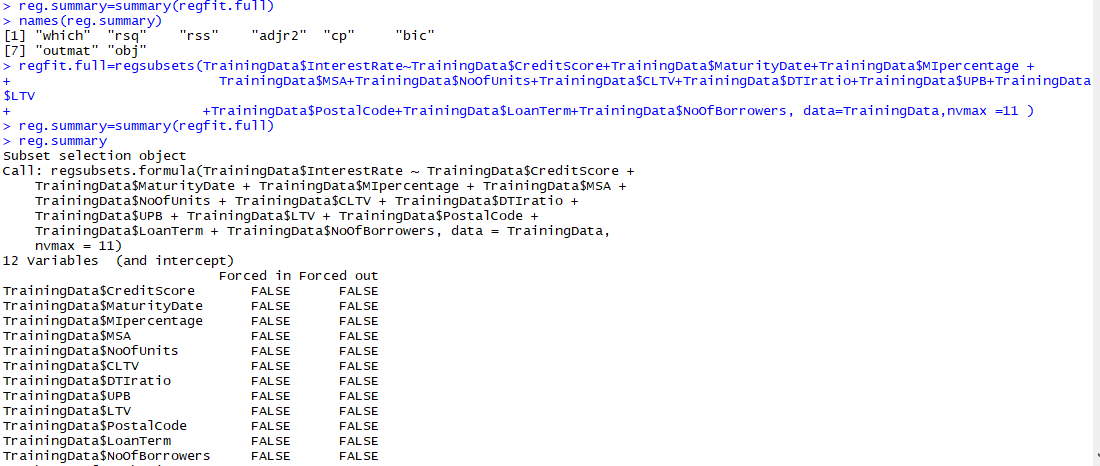
we choose the parameters *α*and *βk* (*k*= 1, 2, ..., *p*) in the [multiple linear regression model](http://www.r-tutor.com/node/100)so as to minimize the sum of squares of the error term *ϵ*, we will have the so called **estimated multiple regression equation**. It allows us to compute **fitted values**of *y*based on a set of values of *xk* (*k*= 1, 2, ..., *p*) .

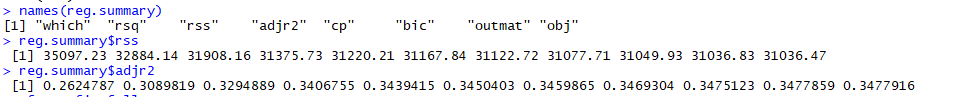
       ∑
ˆy = a +   bkxk
        k

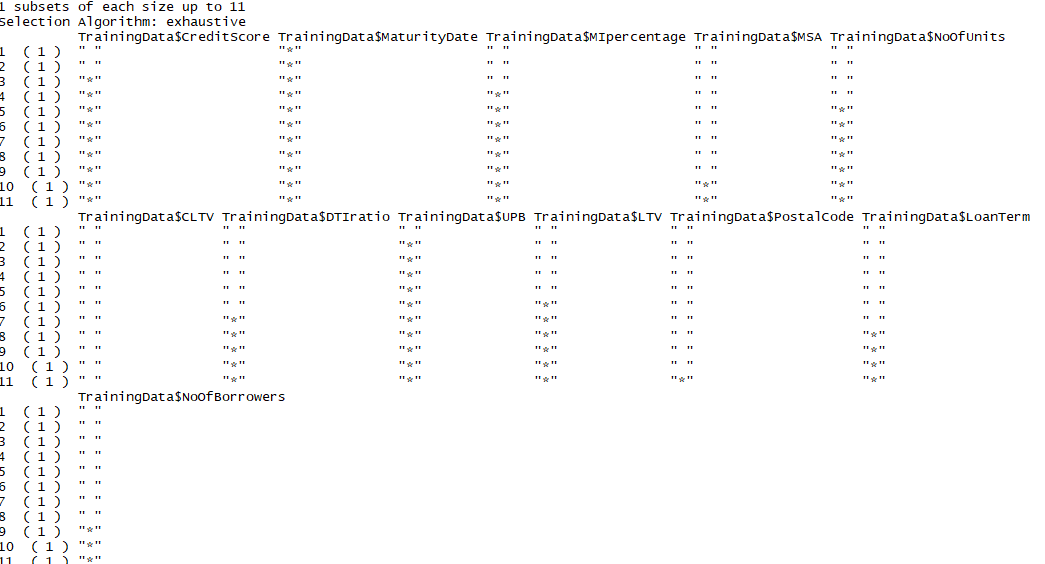

# **Confidence Interval for Linear Regression**

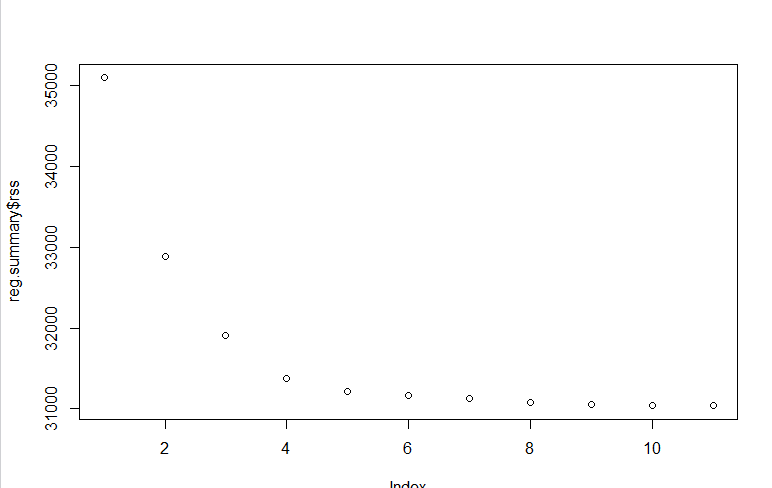
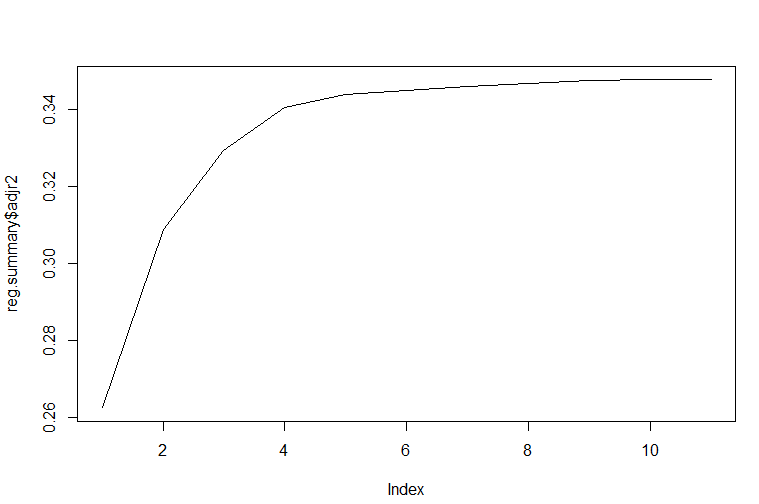
Exhaustive Search Regression









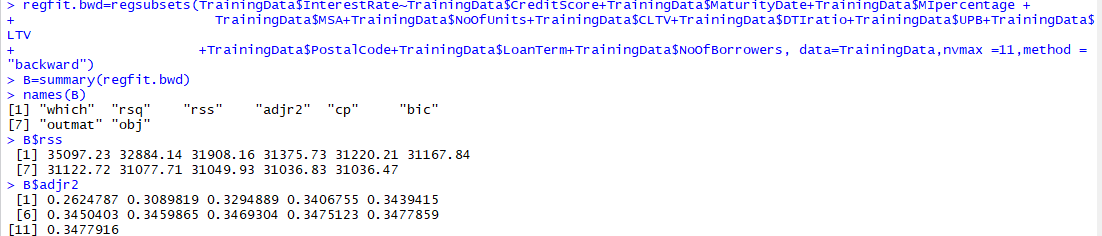
Forward Selection

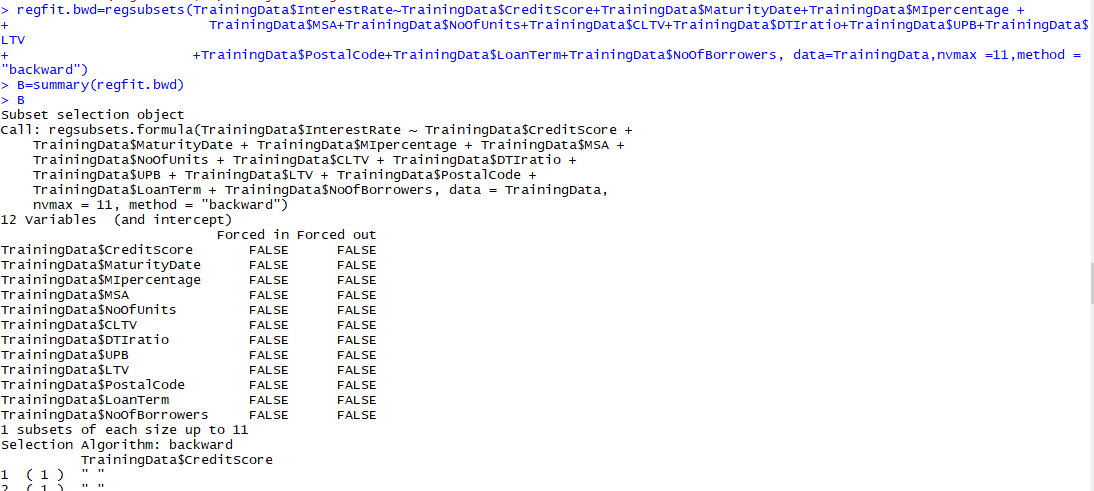
In the Forward Selection We take all the Train Data Set and Calculate the Squared Error

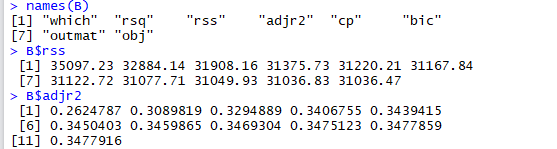


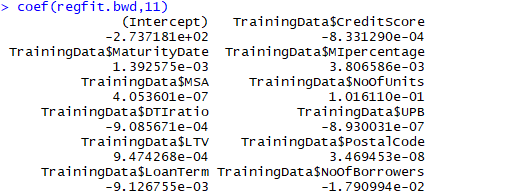


Backward Selection









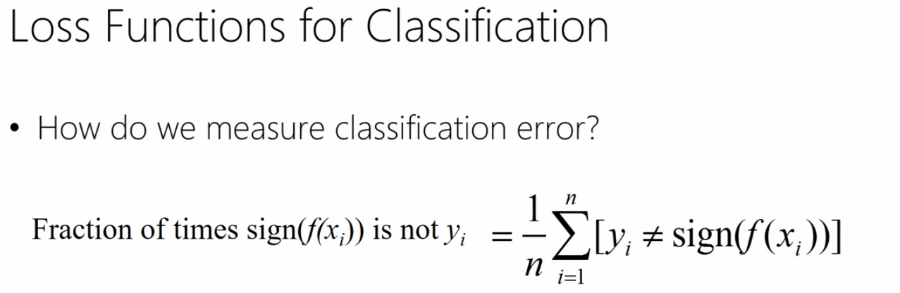
Q. Validates against the Q22005 datasets

Q. Repeat this using Random Forest & Neural Network algorithms.

Q. Choose the best model amongst the 3 types of algorithms.

Part 3:

Classification Loss Fucntion

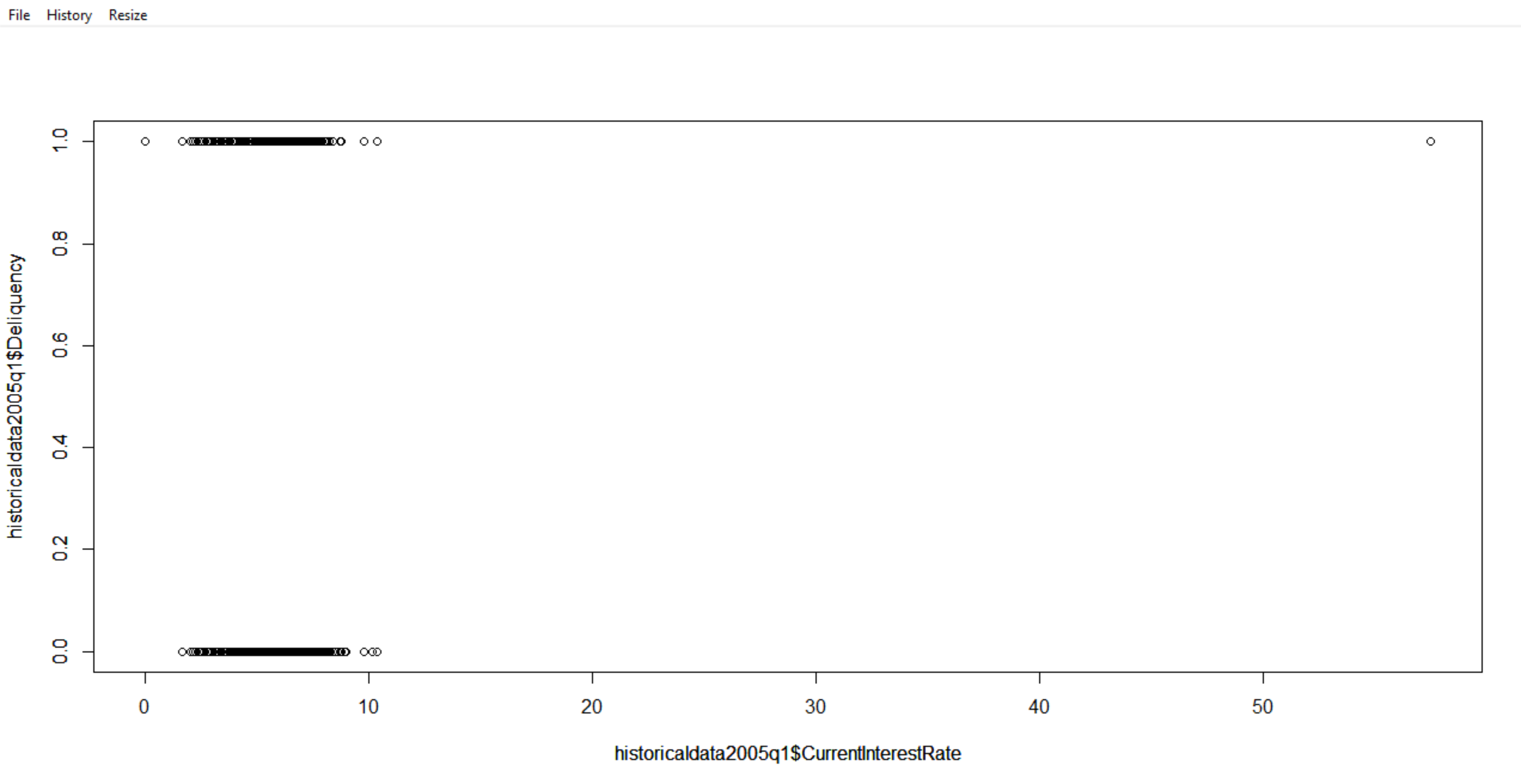


**2.Classification**

The following part was coded in R

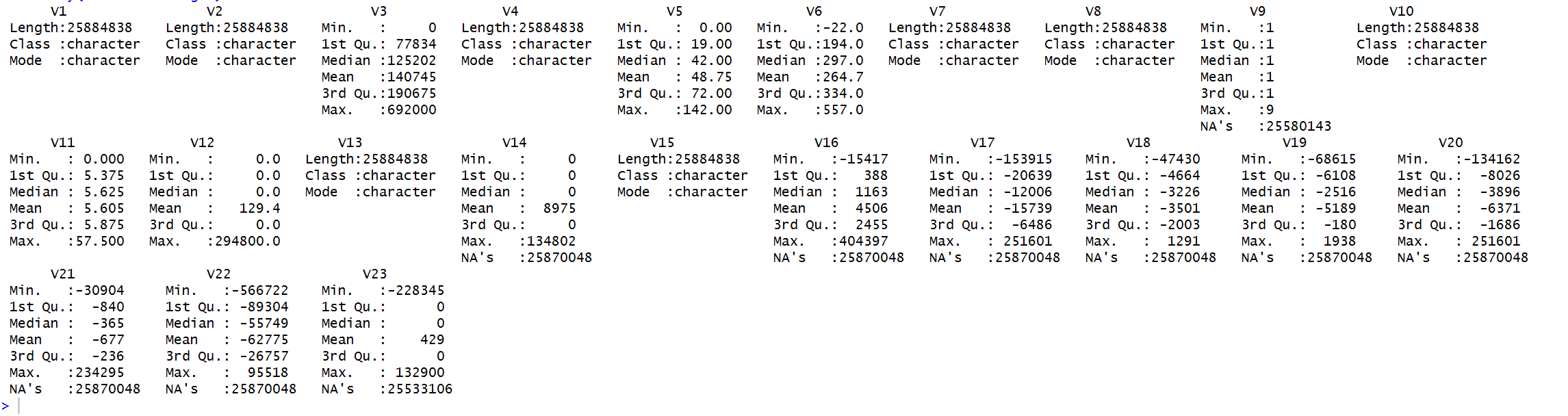
Logistic regression requires the values of CurrentDeliquencyStatus to be either ‘0’ or ‘1’.

So a customer is either delinquent or not can be predicted on the different values.



The scatter plot shows the data in the Delinquency variable.

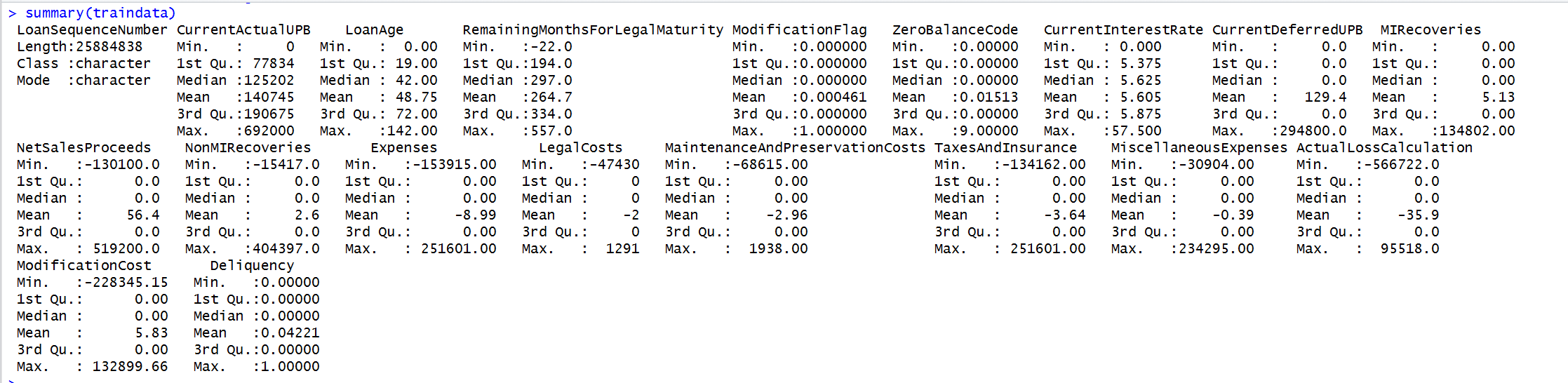
The following are the Once the columns columns used in R



After naming the data set, the data looks as follows :

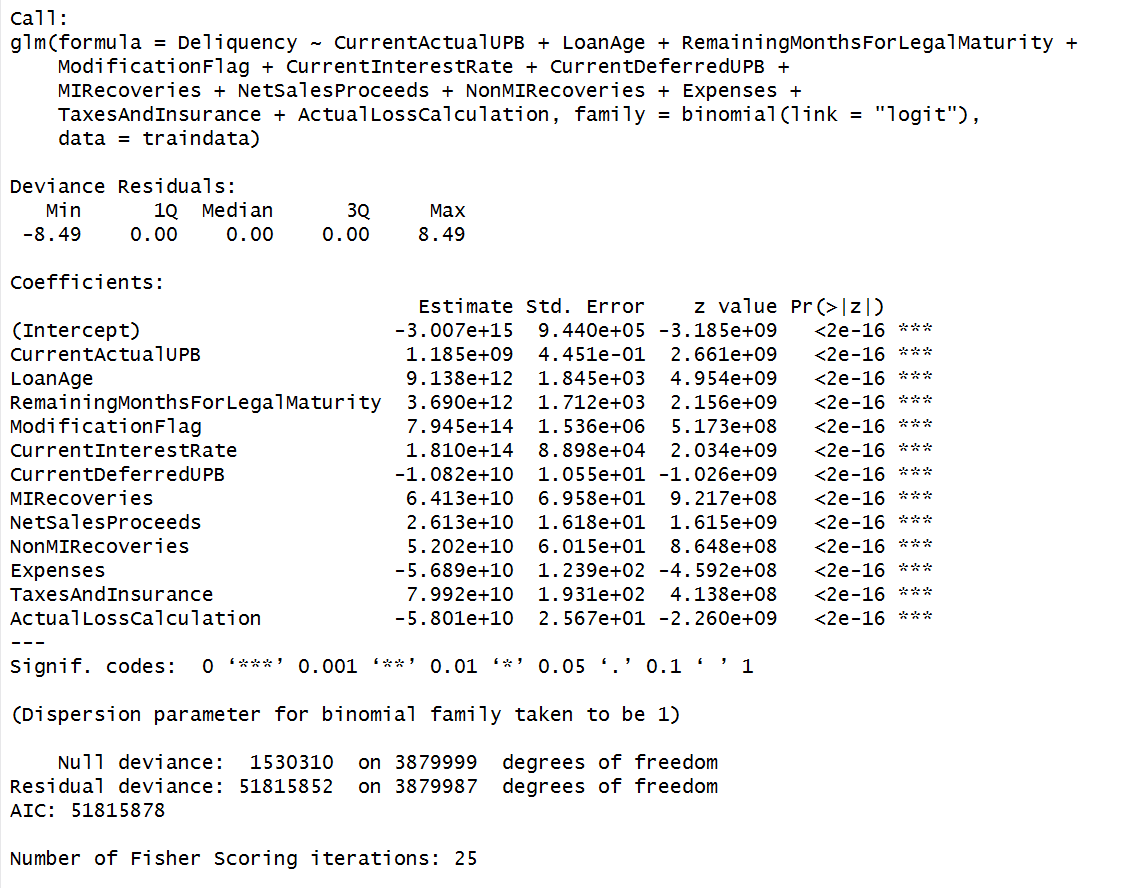


After wrangling the data summary is as follows :



Logistic regression works on the probability of either 0 or 1 and it gives the calculation of the probability of an event taking place to the probability of it not taking place.

Logistic Regression Coefficients are as follows :



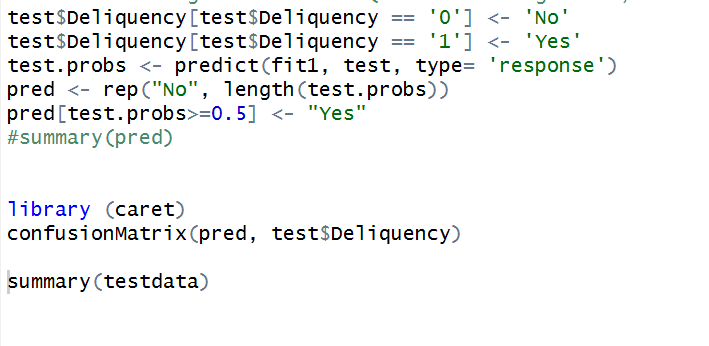
In the above stats given the estimate value tells us how the value of Delinquency varies for the different values in the respective rows .

If the value of P for a row is below 0.005, then the row is considered very significant.

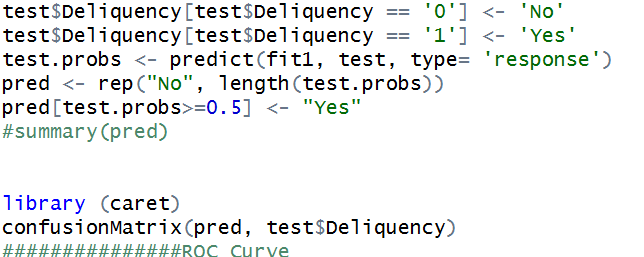
In the above stats all the rows are told to be very significant . The stars beside the P values tell us the significance of the value as well.

**Confusion Matrix:**

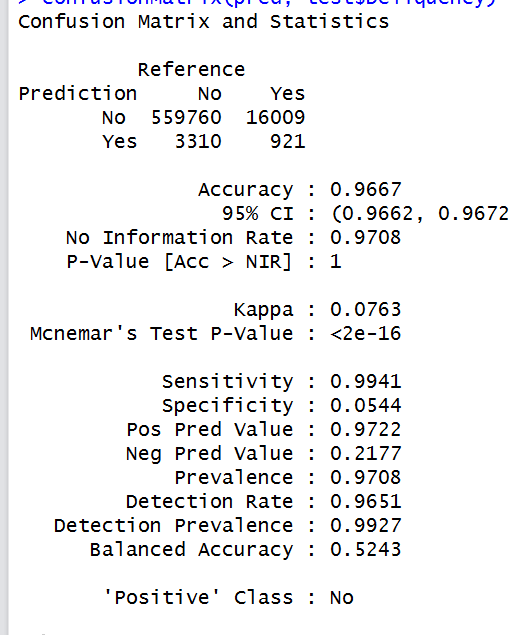
**Code :**



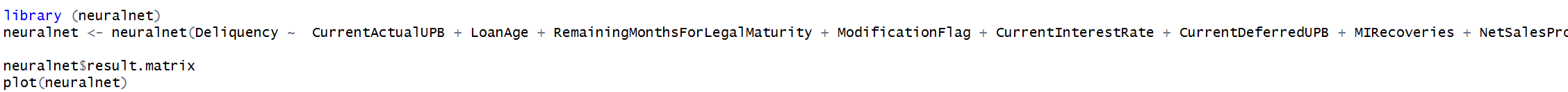
The following is the result of the Confusion Matrix:



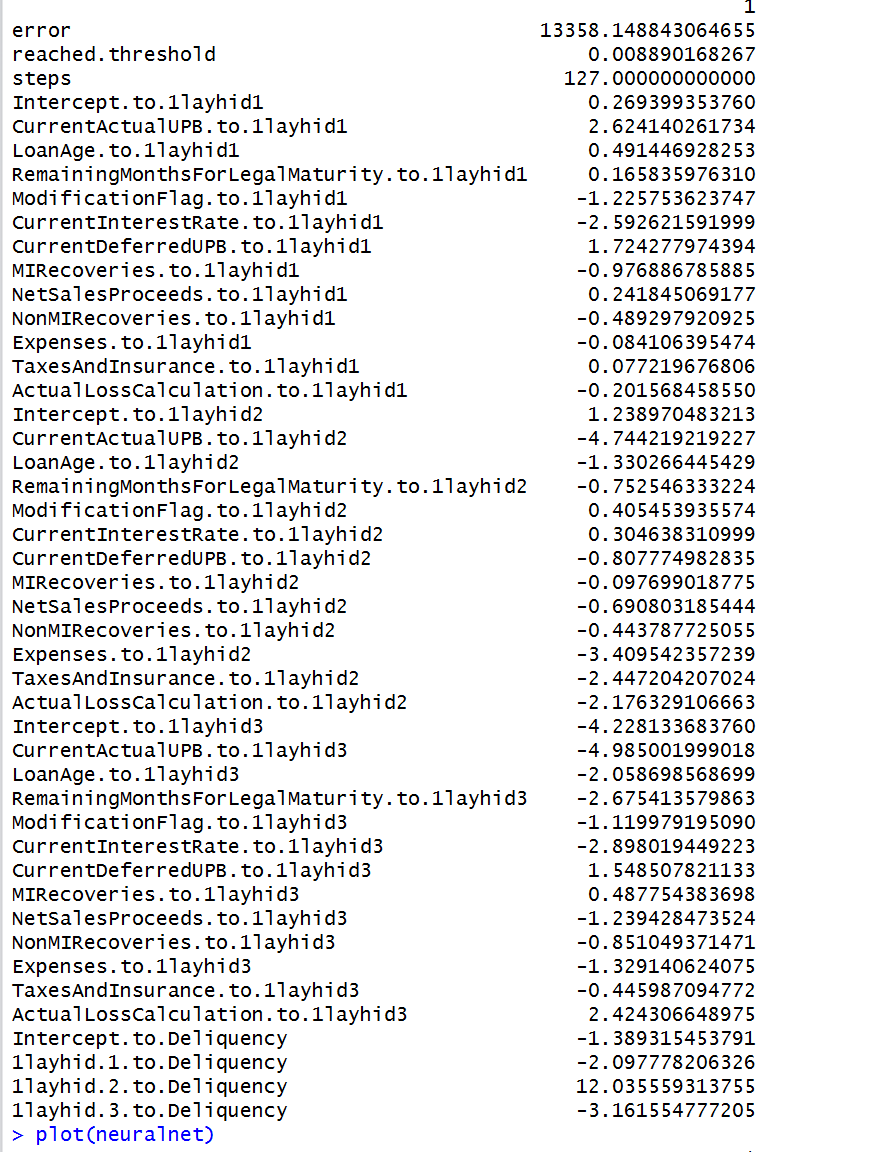
The stats for confusion matrix is as follows :



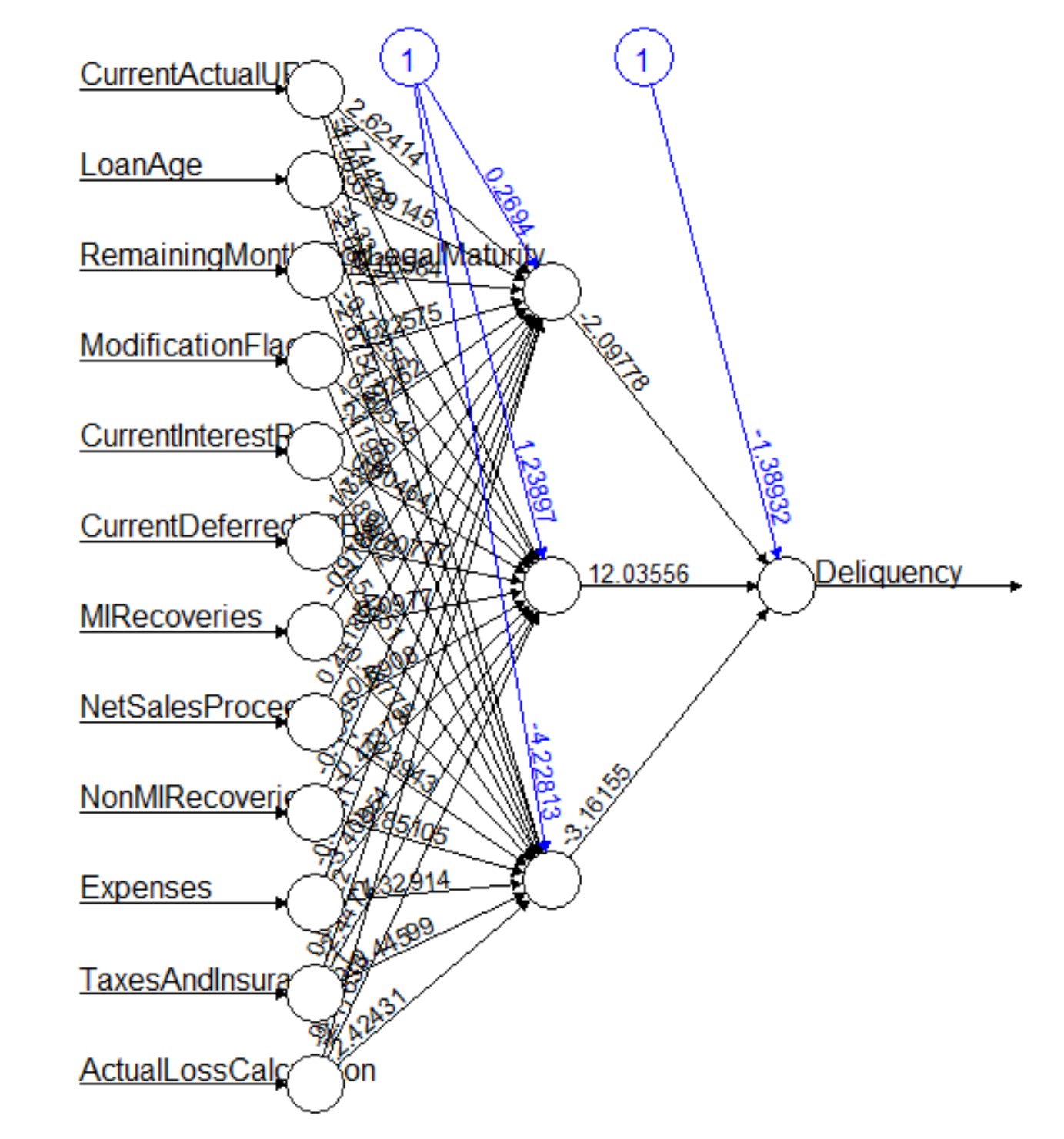
**Neural Networks Code :**



The following is the result for the Neural Networks code :



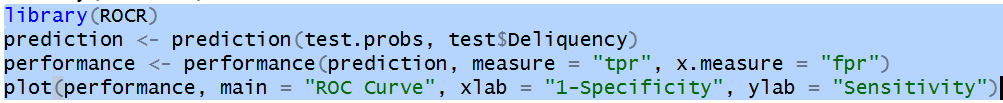
The figure below is the plot of the neural network that was created by the code above :

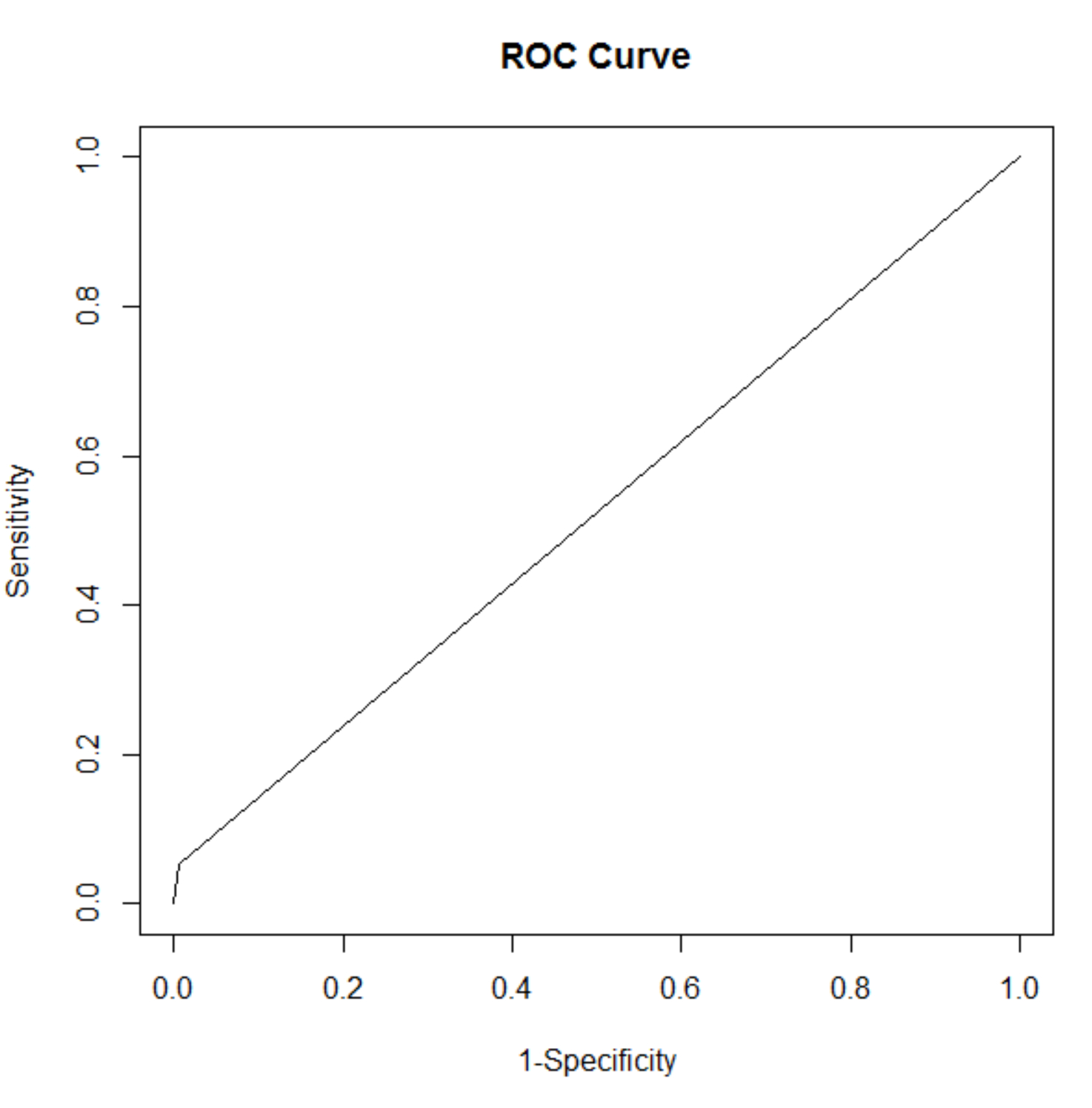


**ROC Curve**

**Code :**

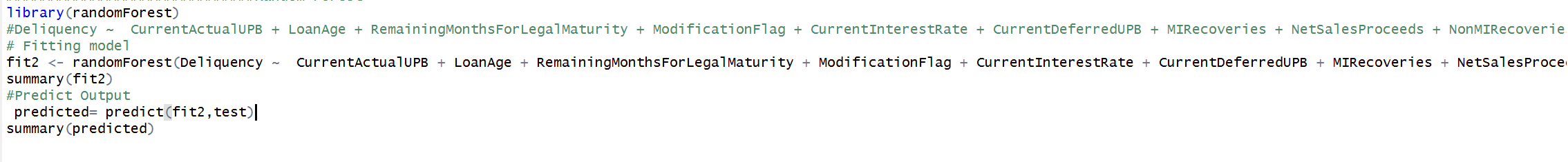
The curve comes as follows :





Random Forest:

The following is the code for the Random Forest Algorithm :



The following is the result for the Random forest code above :

