Loan Analysis

The main goal of this analysis is building a model that will predict whether customer would default on a loan. The method used to build the predictive model is Logistic Regression. The dataset was found on kaggle.

Importing the packages

#install.packages("rlang")  
library(rlang)  
library(DescTools)  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(aod)  
library(ROCR)

## Loading required package: gplots

##   
## Attaching package: 'gplots'

## The following object is masked from 'package:stats':  
##   
## lowess

library(readxl)  
#library(caret) #library not loaded  
library(MASS)

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':  
##   
## select

library(BaylorEdPsych)

##   
## Attaching package: 'BaylorEdPsych'

## The following objects are masked from 'package:DescTools':  
##   
## EtaSq, PseudoR2

library(Hmisc)

## Loading required package: lattice

## Loading required package: survival

##   
## Attaching package: 'survival'

## The following object is masked from 'package:aod':  
##   
## rats

## Loading required package: Formula

## Loading required package: ggplot2

##   
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:dplyr':  
##   
## src, summarize

## The following objects are masked from 'package:DescTools':  
##   
## %nin%, Label, Mean, Quantile

## The following objects are masked from 'package:base':  
##   
## format.pval, units

library(VIF)  
library(car)

## Loading required package: carData

##   
## Attaching package: 'car'

## The following object is masked from 'package:VIF':  
##   
## vif

## The following object is masked from 'package:dplyr':  
##   
## recode

## The following object is masked from 'package:DescTools':  
##   
## Recode

library(corrplot)

## corrplot 0.84 loaded

library(e1071)

##   
## Attaching package: 'e1071'

## The following object is masked from 'package:Hmisc':  
##   
## impute

library(dendroTools)  
library(ResourceSelection)

## ResourceSelection 0.3-2 2017-02-28

library(prediction)

##   
## Attaching package: 'prediction'

## The following object is masked from 'package:ROCR':  
##   
## prediction

LOADING THE DATASET

loan <- read.csv("D:/Fall 2018/DADM/Project/loan.csv")  
dim(loan)

## [1] 887379 74

Checking for missing data

hm\_na <- data.frame(sapply(loan, function(y) sum(length(which(is.na(y))))))  
hm\_na

## sapply.loan..function.y..sum.length.which.is.na.y.....  
## id 0  
## member\_id 0  
## loan\_amnt 0  
## funded\_amnt 0  
## funded\_amnt\_inv 0  
## term 0  
## int\_rate 0  
## installment 0  
## grade 0  
## sub\_grade 0  
## emp\_title 0  
## emp\_length 0  
## home\_ownership 0  
## annual\_inc 4  
## verification\_status 0  
## issue\_d 0  
## loan\_status 0  
## pymnt\_plan 0  
## url 0  
## desc 0  
## purpose 0  
## title 0  
## zip\_code 0  
## addr\_state 0  
## dti 0  
## delinq\_2yrs 29  
## earliest\_cr\_line 0  
## inq\_last\_6mths 29  
## mths\_since\_last\_delinq 454312  
## mths\_since\_last\_record 750326  
## open\_acc 29  
## pub\_rec 29  
## revol\_bal 0  
## revol\_util 502  
## total\_acc 29  
## initial\_list\_status 0  
## out\_prncp 0  
## out\_prncp\_inv 0  
## total\_pymnt 0  
## total\_pymnt\_inv 0  
## total\_rec\_prncp 0  
## total\_rec\_int 0  
## total\_rec\_late\_fee 0  
## recoveries 0  
## collection\_recovery\_fee 0  
## last\_pymnt\_d 0  
## last\_pymnt\_amnt 0  
## next\_pymnt\_d 0  
## last\_credit\_pull\_d 0  
## collections\_12\_mths\_ex\_med 145  
## mths\_since\_last\_major\_derog 665676  
## policy\_code 0  
## application\_type 0  
## annual\_inc\_joint 886868  
## dti\_joint 886870  
## verification\_status\_joint 0  
## acc\_now\_delinq 29  
## tot\_coll\_amt 70276  
## tot\_cur\_bal 70276  
## open\_acc\_6m 866007  
## open\_il\_6m 866007  
## open\_il\_12m 866007  
## open\_il\_24m 866007  
## mths\_since\_rcnt\_il 866569  
## total\_bal\_il 866007  
## il\_util 868762  
## open\_rv\_12m 866007  
## open\_rv\_24m 866007  
## max\_bal\_bc 866007  
## all\_util 866007  
## total\_rev\_hi\_lim 70276  
## inq\_fi 866007  
## total\_cu\_tl 866007  
## inq\_last\_12m 866007

If the number of NA’s exeeds 60% of all observations in a column then we eliminate the column.

loan <- loan[, colSums(is.na(loan))<= 0.6\*nrow(loan)]

hm\_na1 <- data.frame(sapply(loan, function(y) sum(length(which(is.na(y))))))  
hm\_na1

## sapply.loan..function.y..sum.length.which.is.na.y.....  
## id 0  
## member\_id 0  
## loan\_amnt 0  
## funded\_amnt 0  
## funded\_amnt\_inv 0  
## term 0  
## int\_rate 0  
## installment 0  
## grade 0  
## sub\_grade 0  
## emp\_title 0  
## emp\_length 0  
## home\_ownership 0  
## annual\_inc 4  
## verification\_status 0  
## issue\_d 0  
## loan\_status 0  
## pymnt\_plan 0  
## url 0  
## desc 0  
## purpose 0  
## title 0  
## zip\_code 0  
## addr\_state 0  
## dti 0  
## delinq\_2yrs 29  
## earliest\_cr\_line 0  
## inq\_last\_6mths 29  
## mths\_since\_last\_delinq 454312  
## open\_acc 29  
## pub\_rec 29  
## revol\_bal 0  
## revol\_util 502  
## total\_acc 29  
## initial\_list\_status 0  
## out\_prncp 0  
## out\_prncp\_inv 0  
## total\_pymnt 0  
## total\_pymnt\_inv 0  
## total\_rec\_prncp 0  
## total\_rec\_int 0  
## total\_rec\_late\_fee 0  
## recoveries 0  
## collection\_recovery\_fee 0  
## last\_pymnt\_d 0  
## last\_pymnt\_amnt 0  
## next\_pymnt\_d 0  
## last\_credit\_pull\_d 0  
## collections\_12\_mths\_ex\_med 145  
## policy\_code 0  
## application\_type 0  
## verification\_status\_joint 0  
## acc\_now\_delinq 29  
## tot\_coll\_amt 70276  
## tot\_cur\_bal 70276  
## total\_rev\_hi\_lim 70276

As a matter of fact since our database is pretty large, we decided to avoid filling rows that consist of empty (NA) values, and deleted all of them. It shouldn’t have an impact on the result of our final model.

loan <- na.omit(loan)

hm\_na2 <- data.frame(sapply(loan, function(y) sum(length(which(is.na(y))))))  
hm\_na2

## sapply.loan..function.y..sum.length.which.is.na.y.....  
## id 0  
## member\_id 0  
## loan\_amnt 0  
## funded\_amnt 0  
## funded\_amnt\_inv 0  
## term 0  
## int\_rate 0  
## installment 0  
## grade 0  
## sub\_grade 0  
## emp\_title 0  
## emp\_length 0  
## home\_ownership 0  
## annual\_inc 0  
## verification\_status 0  
## issue\_d 0  
## loan\_status 0  
## pymnt\_plan 0  
## url 0  
## desc 0  
## purpose 0  
## title 0  
## zip\_code 0  
## addr\_state 0  
## dti 0  
## delinq\_2yrs 0  
## earliest\_cr\_line 0  
## inq\_last\_6mths 0  
## mths\_since\_last\_delinq 0  
## open\_acc 0  
## pub\_rec 0  
## revol\_bal 0  
## revol\_util 0  
## total\_acc 0  
## initial\_list\_status 0  
## out\_prncp 0  
## out\_prncp\_inv 0  
## total\_pymnt 0  
## total\_pymnt\_inv 0  
## total\_rec\_prncp 0  
## total\_rec\_int 0  
## total\_rec\_late\_fee 0  
## recoveries 0  
## collection\_recovery\_fee 0  
## last\_pymnt\_d 0  
## last\_pymnt\_amnt 0  
## next\_pymnt\_d 0  
## last\_credit\_pull\_d 0  
## collections\_12\_mths\_ex\_med 0  
## policy\_code 0  
## application\_type 0  
## verification\_status\_joint 0  
## acc\_now\_delinq 0  
## tot\_coll\_amt 0  
## tot\_cur\_bal 0  
## total\_rev\_hi\_lim 0

We have removed all the NA values in the dataset. Now we can visualize the data to gain some meaningful insight

dim(loan)

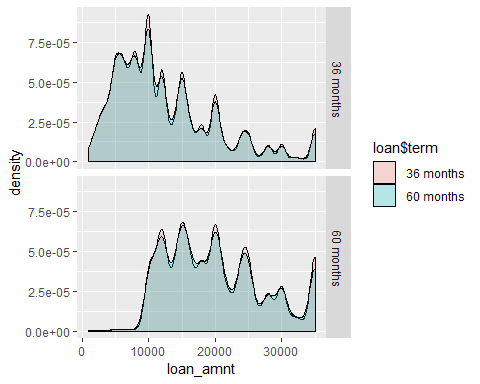
## [1] 407770 56

str(loan)

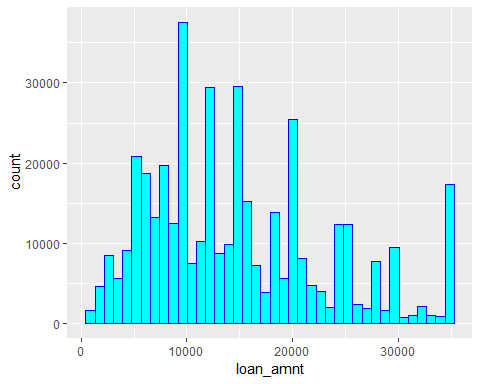
## 'data.frame': 407770 obs. of 56 variables:  
## $ id : int 10159498 10139658 10159548 10129477 10159611 10179520 10149526 10224583 10129506 10109602 ...  
## $ member\_id : int 1319523 11991209 12011167 11981093 12011228 12031088 12001069 12086734 11981122 11961187 ...  
## $ loan\_amnt : num 12000 12000 15000 14000 10000 3000 27600 11100 20800 4500 ...  
## $ funded\_amnt : num 12000 12000 15000 14000 10000 3000 27600 11100 20800 4500 ...  
## $ funded\_amnt\_inv : num 12000 12000 15000 14000 10000 3000 27600 11100 20800 4500 ...  
## $ term : Factor w/ 2 levels " 36 months"," 60 months": 1 1 1 1 1 1 2 1 1 1 ...  
## $ int\_rate : num 6.62 13.53 8.9 12.85 9.67 ...  
## $ installment : num 368 407 476 471 321 ...  
## $ grade : Factor w/ 7 levels "A","B","C","D",..: 1 2 1 2 2 2 4 3 2 4 ...  
## $ sub\_grade : Factor w/ 35 levels "A1","A2","A3",..: 2 10 5 9 6 9 20 13 10 19 ...  
## $ emp\_title : Factor w/ 299273 levels "","'Property Manager",..: 157840 184341 9559 18151 216716 23455 259477 266824 185439 236229 ...  
## $ emp\_length : Factor w/ 12 levels "< 1 year","1 year",..: 3 3 4 6 9 3 8 3 3 3 ...  
## $ home\_ownership : Factor w/ 6 levels "ANY","MORTGAGE",..: 2 6 2 6 2 6 2 2 6 6 ...  
## $ annual\_inc : num 105000 40000 63000 88000 102000 25000 73000 90000 81500 105000 ...  
## $ verification\_status : Factor w/ 3 levels "Not Verified",..: 1 2 1 1 1 3 2 1 3 1 ...  
## $ issue\_d : Factor w/ 103 levels "Apr-2008","Apr-2009",..: 24 24 24 24 24 24 24 24 24 24 ...  
## $ loan\_status : Factor w/ 10 levels "Charged Off",..: 2 6 2 2 2 6 2 2 6 2 ...  
## $ pymnt\_plan : Factor w/ 2 levels "n","y": 1 1 1 1 1 1 1 1 1 1 ...  
## $ url : Factor w/ 887379 levels "https://www.lendingclub.com/browse/loanDetail.action?loan\_id=1000007",..: 8177 6586 8180 5799 8189 9846 7395 10467 5804 4138 ...  
## $ desc : Factor w/ 124471 levels "","- Pay off Dell Financial: $ 1300.00 - Pay off IRS for 2005: $ 1400.00 - Pay off Mac Comp : $ 1700.00 - Pay o"| \_\_truncated\_\_,..: 1 1 116290 1 1 1 116176 1 116225 1 ...  
## $ purpose : Factor w/ 14 levels "car","credit\_card",..: 3 3 3 3 3 3 3 10 3 10 ...  
## $ title : Factor w/ 63146 levels "","'08 & '09 Roth IRA Investments",..: 60999 18724 46987 18724 9727 17787 12945 45551 52918 5092 ...  
## $ zip\_code : Factor w/ 935 levels "007xx","008xx",..: 52 817 321 269 20 309 757 93 90 705 ...  
## $ addr\_state : Factor w/ 51 levels "AK","AL","AR",..: 7 33 10 28 20 10 6 35 35 44 ...  
## $ dti : num 14.1 16.9 16.5 10 15.6 ...  
## $ delinq\_2yrs : num 0 0 0 1 2 0 1 1 0 1 ...  
## $ earliest\_cr\_line : Factor w/ 698 levels "","Apr-1955",..: 445 627 449 383 618 501 384 396 393 441 ...  
## $ inq\_last\_6mths : num 1 0 0 0 0 0 1 0 2 0 ...  
## $ mths\_since\_last\_delinq : num 43 53 34 16 11 58 7 16 64 10 ...  
## $ open\_acc : num 12 7 8 6 9 5 10 9 29 5 ...  
## $ pub\_rec : num 0 2 0 1 0 2 0 0 0 0 ...  
## $ revol\_bal : num 13168 5572 11431 3686 9912 ...  
## $ revol\_util : num 21.6 68.8 74.2 81.9 44.4 54.2 82.8 66.2 54.5 93.1 ...  
## $ total\_acc : num 22 32 29 14 22 26 24 12 41 10 ...  
## $ initial\_list\_status : Factor w/ 2 levels "f","w": 2 2 2 1 1 1 1 1 1 1 ...  
## $ out\_prncp : num 4267 0 5449 5274 3659 ...  
## $ out\_prncp\_inv : num 4267 0 5449 5274 3659 ...  
## $ total\_pymnt : num 8843 13360 11431 11297 7707 ...  
## $ total\_pymnt\_inv : num 8843 13360 11431 11297 7707 ...  
## $ total\_rec\_prncp : num 7733 12000 9551 8726 6341 ...  
## $ total\_rec\_int : num 1109 1360 1880 2571 1366 ...  
## $ total\_rec\_late\_fee : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ recoveries : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ collection\_recovery\_fee : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ last\_pymnt\_d : Factor w/ 99 levels "","Apr-2008",..: 43 99 43 43 43 50 43 43 75 43 ...  
## $ last\_pymnt\_amnt : num 368 119 476 471 321 ...  
## $ next\_pymnt\_d : Factor w/ 101 levels "","Apr-2008",..: 35 1 35 35 35 1 35 35 1 35 ...  
## $ last\_credit\_pull\_d : Factor w/ 104 levels "","Apr-2009",..: 43 43 43 43 43 43 43 43 43 43 ...  
## $ collections\_12\_mths\_ex\_med: num 0 0 0 0 0 0 0 0 0 0 ...  
## $ policy\_code : num 1 1 1 1 1 1 1 1 1 1 ...  
## $ application\_type : Factor w/ 2 levels "INDIVIDUAL","JOINT": 1 1 1 1 1 1 1 1 1 1 ...  
## $ verification\_status\_joint : Factor w/ 4 levels "","Not Verified",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ acc\_now\_delinq : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ tot\_coll\_amt : num 0 15386 1514 0 0 ...  
## $ tot\_cur\_bal : num 267646 13605 272492 17672 39143 ...  
## $ total\_rev\_hi\_lim : num 61100 8100 15400 4500 22300 5300 32600 10000 43100 20500 ...  
## - attr(\*, "na.action")= 'omit' Named int 1 2 3 4 5 6 7 8 9 10 ...  
## ..- attr(\*, "names")= chr "1" "2" "3" "4" ...

DATA VISUALIZATION

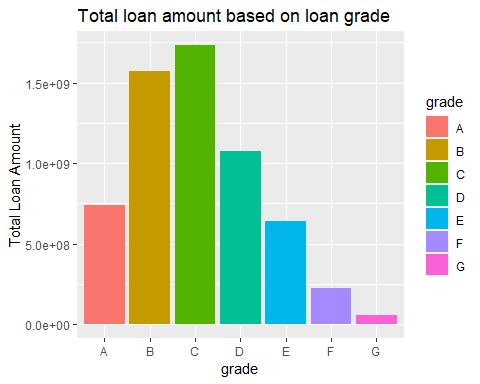
ggplot(data=loan,aes(loan\_amnt, fill=loan$term))+  
 geom\_density(alpha=0.25) +   
 facet\_grid(loan$term~ .)



ggplot(data=loan, aes(loan\_amnt))+geom\_histogram(bins=40,color="blue",fill="cyan")

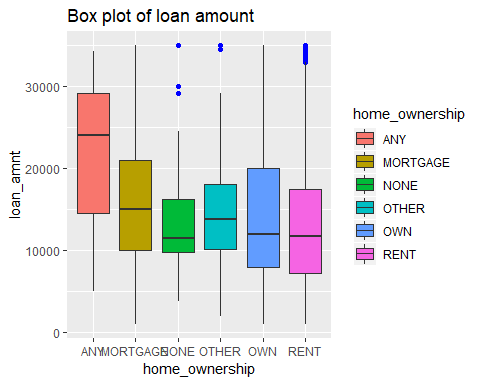


ggplot(loan, aes(x=grade, y=loan\_amnt, fill=grade)) +  
 stat\_summary(fun.y="sum", geom="bar") +  
 labs(y ="Total Loan Amount",title="Total loan amount based on loan grade")



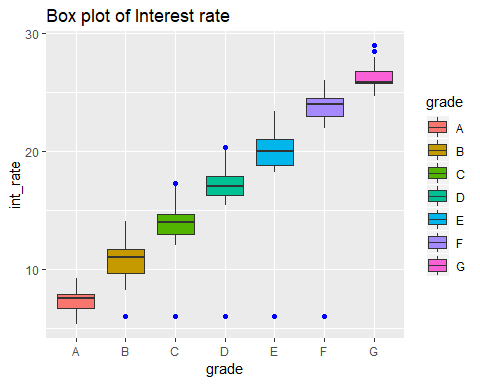
examining the distribution of loan amount according to ownership of house

ggplot(data=loan, aes(home\_ownership,loan\_amnt,fill=home\_ownership))+geom\_boxplot(outlier.color = "blue")+labs(title="Box plot of loan amount")



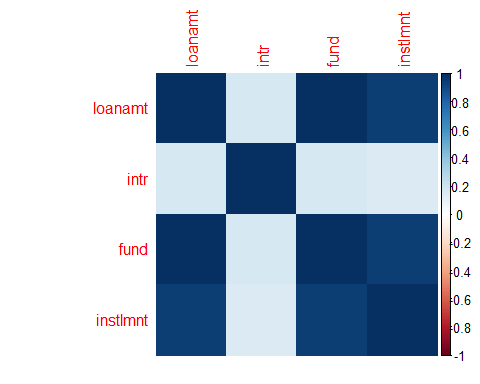
examining distribution of interest rates for different grades

ggplot(data=loan, aes(grade,int\_rate,fill=grade))+geom\_boxplot(outlier.color = "blue")+labs(title="Box plot of Interest rate")

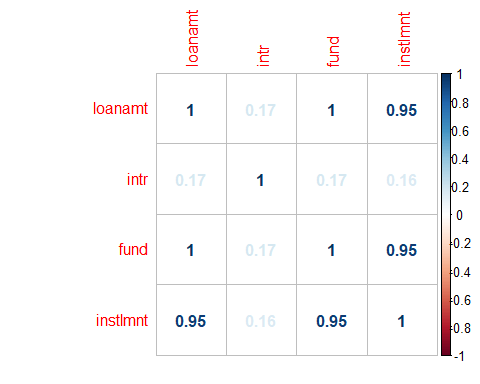


To examine the correlation among variables: loan amout, funded amount and installment amount

data2 <- data.frame(loanamt =loan$loan\_amnt, intr =loan$int\_rate,fund= loan$funded\_amnt, instlmnt = loan$installment)  
M<- cor(data2)  
corrplot(M,method = "color")



corrplot(M, method="number")



DATA TRANSFORMATION

In order to avoid some bias assesments we will filter our database just to ‘loan\_status’ values - Fully Paid & Charge Off.

## Filter database to observations where loan was 'Fully\_paid', or 'Charged\_off'  
unique(loan$loan\_status)

## [1] Current Fully Paid Charged Off   
## [4] Late (31-120 days) In Grace Period Late (16-30 days)   
## [7] Default Issued   
## 10 Levels: Charged Off Current ... Late (31-120 days)

loan <- loan %>% dplyr::select(everything()) %>% filter(loan\_status %in% c("Fully Paid", "Charged Off"))

## Warning: `lang()` is soft-deprecated as of rlang 0.2.0.  
## Please use `call2()` instead  
## This warning is displayed once per session.

## Warning: `new\_overscope()` is soft-deprecated as of rlang 0.2.0.  
## Please use `new\_data\_mask()` instead  
## This warning is displayed once per session.

dim(loan)

## [1] 89866 56

Next step is like previously mentioned, filtering just to ‘Individual’ applicants.

unique(loan$application\_type)

## [1] INDIVIDUAL  
## Levels: INDIVIDUAL JOINT

loan <- loan %>% dplyr::select(everything()) %>%filter(application\_type %in% c("INDIVIDUAL"))

dim(loan)

## [1] 89866 56

Now we decided to remove unnecessary columns that refers to Active loan accounts, Joint Applicants, and some based on intuition like policy code or employee title (too much unique values to classify).

names(loan)

## [1] "id" "member\_id"   
## [3] "loan\_amnt" "funded\_amnt"   
## [5] "funded\_amnt\_inv" "term"   
## [7] "int\_rate" "installment"   
## [9] "grade" "sub\_grade"   
## [11] "emp\_title" "emp\_length"   
## [13] "home\_ownership" "annual\_inc"   
## [15] "verification\_status" "issue\_d"   
## [17] "loan\_status" "pymnt\_plan"   
## [19] "url" "desc"   
## [21] "purpose" "title"   
## [23] "zip\_code" "addr\_state"   
## [25] "dti" "delinq\_2yrs"   
## [27] "earliest\_cr\_line" "inq\_last\_6mths"   
## [29] "mths\_since\_last\_delinq" "open\_acc"   
## [31] "pub\_rec" "revol\_bal"   
## [33] "revol\_util" "total\_acc"   
## [35] "initial\_list\_status" "out\_prncp"   
## [37] "out\_prncp\_inv" "total\_pymnt"   
## [39] "total\_pymnt\_inv" "total\_rec\_prncp"   
## [41] "total\_rec\_int" "total\_rec\_late\_fee"   
## [43] "recoveries" "collection\_recovery\_fee"   
## [45] "last\_pymnt\_d" "last\_pymnt\_amnt"   
## [47] "next\_pymnt\_d" "last\_credit\_pull\_d"   
## [49] "collections\_12\_mths\_ex\_med" "policy\_code"   
## [51] "application\_type" "verification\_status\_joint"   
## [53] "acc\_now\_delinq" "tot\_coll\_amt"   
## [55] "tot\_cur\_bal" "total\_rev\_hi\_lim"

loan <- loan[ , -which(names(loan) %in% c("id","member\_id","verification\_status\_joint" , "funded\_amnt", "funded\_amnt\_inv", "sub\_grade", "issue\_d"  
 , "url", "desc", "title", "zip\_code", "emp\_title" , "addr\_state", "initial\_list\_status"  
 ,"out\_prncp", "out\_prncp\_inv", "total\_pymnt", "total\_pymnt\_inv", "total\_rec\_prncp"  
 ,"total\_rec\_int", "total\_rec\_late\_fee", "collection\_recovery\_fee", "last\_pymnt\_d"  
 ,"last\_pymnt\_amnt", "next\_pymnt\_d", "last\_credit\_pull\_d", "pymnt\_plan", "policy\_code","application\_type"))]

dim(loan)

## [1] 89866 27

Now we have filtered down to 27 columns

## Binarization of Term  
unique(loan$term)

## [1] 36 months 60 months  
## Levels: 36 months 60 months

loan$term <- as.integer(gsub("months", "", loan$term))  
loan$term[loan$term == 36] <- 1  
loan$term[loan$term != 1] <- 0

## Categorization of grade  
unique(loan$grade)

## [1] B C A E D G F  
## Levels: A B C D E F G

loan$grade <- as.character(loan$grade)  
loan$grade[loan$grade == "A"] <- 7   
loan$grade[loan$grade == "B"] <- 6  
loan$grade[loan$grade == "C"] <- 5  
loan$grade[loan$grade == "D"] <- 4  
loan$grade[loan$grade == "E"] <- 3  
loan$grade[loan$grade == "F"] <- 2  
loan$grade[loan$grade == "G"] <- 1  
loan$grade <- as.integer(loan$grade)

Getting rid of unnecessary characters within emp\_length variable.

unique(loan$emp\_length)

## [1] 10+ years 4 years 5 years 1 year 8 years 3 years < 1 year   
## [8] 9 years 6 years n/a 2 years 7 years   
## 12 Levels: < 1 year 1 year 10+ years 2 years 3 years 4 years ... n/a

loan$emp\_length <- gsub("<", "", loan$emp\_length)  
loan$emp\_length <- gsub("years", "", loan$emp\_length)  
loan$emp\_length <- gsub("year", "", loan$emp\_length)  
loan$emp\_length <- gsub("n/a", "", loan$emp\_length)  
loan$emp\_length <- gsub(" ", "", loan$emp\_length)  
loan$emp\_length <- gsub("\\+", "", loan$emp\_length)  
loan$emp\_length <- ifelse(loan$emp\_length =="", 10, loan$emp\_length)  
loan$emp\_length <- as.integer(loan$emp\_length)  
unique(loan$emp\_length)

## [1] 10 4 5 1 8 3 9 6 2 7

In order to use home\_ownership as one of our potential predictors, we decided to binarize it. Assumed that people who are possesing it’s property or mostly posses, having it bought by mortage, will represent one group. People who rents a flat/house, none, any and other values, represent second group of society.

unique(loan$home\_ownership)

## [1] RENT OWN MORTGAGE NONE OTHER ANY   
## Levels: ANY MORTGAGE NONE OTHER OWN RENT

loan$home\_ownership <- as.character(loan$home\_ownership)  
loan$home\_ownership[loan$home\_ownership=="OWN" | loan$home\_ownership=="MORTGAGE" ] <- 1   
loan$home\_ownership[loan$home\_ownership!=1] <- 0  
loan$home\_ownership <- as.numeric(loan$home\_ownership)

Binarization of verification\_status variable.

unique(loan$verification\_status)

## [1] Source Verified Verified Not Verified   
## Levels: Not Verified Source Verified Verified

loan$verification\_status <- ifelse(loan$verification\_status == "Not Verified", 0, 1)

Purpose variable was binarize based on Lending Club offer and intuition. As one of these values reffers to personal needs and the other parts to financial issues. We decided to binarize this variable as shown below.

## Binarization of purpose  
unique(loan$purpose)

## [1] debt\_consolidation home\_improvement credit\_card   
## [4] other moving car   
## [7] major\_purchase small\_business medical   
## [10] vacation house renewable\_energy   
## [13] wedding   
## 14 Levels: car credit\_card debt\_consolidation ... wedding

loan$purpose <- as.character(loan$purpose)  
loan$purpose[loan$purpose == "home\_improvement" | loan$purpose == "other" | loan$purpose == "moving" | loan$purpose == "vacation" |   
 loan$purpose == "major\_purchase"| loan$purpose == "small\_business"| loan$purpose == "car" | loan$purpose == "medical"|  
 loan$purpose == "house" | loan$purpose == "renewable\_energy" | loan$purpose == "wedding"] <- 1  
loan$purpose[loan$purpose != 1] <- 0  
loan$purpose <- as.numeric(loan$purpose)

To generalize variable Earliest\_Cr\_line we decided to remove part which reflects specific month when credit line was opened and just stick to the year.

## Earliest\_Cr\_line  
head(loan$earliest\_cr\_line)

## [1] Oct-1998 May-1991 Jun-1998 Nov-1978 Dec-2001 Feb-2004  
## 698 Levels: Apr-1955 Apr-1958 Apr-1960 Apr-1961 Apr-1962 ... Sep-2012

loan$earliest\_cr\_line <- as.character(loan$earliest\_cr\_line)  
substrRight <- function(x, n){  
 substr(x, nchar(x)-n+1, nchar(x))  
}  
loan$earliest\_cr\_line <- substrRight(loan$earliest\_cr\_line, 4)  
loan$earliest\_cr\_line <- as.numeric(loan$earliest\_cr\_line)

The final part of transformation is binarization of variable that will be our dependent variable in a model. Loan\_status will answer whether applicant should default on a loan, or not.

## Binarization of dependent variable loan\_status  
loan$loan\_status <- as.character(loan$loan\_status)  
loan$loan\_status[loan$loan\_status == "Fully Paid"] <- 1  
loan$loan\_status[loan$loan\_status != 1] <- 0  
loan$loan\_status <- as.numeric(loan$loan\_status)

Checking the transformed data

str(loan)

## 'data.frame': 89866 obs. of 27 variables:  
## $ loan\_amnt : num 12000 3000 20800 7200 20000 10000 14000 13000 12000 12000 ...  
## $ term : num 1 1 1 1 1 0 1 1 1 1 ...  
## $ int\_rate : num 13.5 12.8 13.5 11 14 ...  
## $ installment : num 407 101 706 236 683 ...  
## $ grade : int 6 6 6 6 5 5 7 6 5 6 ...  
## $ emp\_length : int 10 10 10 4 10 5 10 10 5 10 ...  
## $ home\_ownership : num 0 0 0 1 1 0 1 1 1 1 ...  
## $ annual\_inc : num 40000 25000 81500 70000 80000 ...  
## $ verification\_status : num 1 1 1 1 1 1 1 0 0 1 ...  
## $ loan\_status : num 1 1 1 1 1 0 1 1 1 1 ...  
## $ purpose : num 0 0 0 0 0 0 0 1 0 0 ...  
## $ dti : num 16.9 24.7 16.7 19.2 16.7 ...  
## $ delinq\_2yrs : num 0 0 0 0 1 0 0 1 1 1 ...  
## $ earliest\_cr\_line : num 1998 1991 1998 1978 2001 ...  
## $ inq\_last\_6mths : num 0 0 2 0 1 0 4 2 1 2 ...  
## $ mths\_since\_last\_delinq : num 53 58 64 59 8 63 30 11 17 20 ...  
## $ open\_acc : num 7 5 29 14 10 13 20 12 11 13 ...  
## $ pub\_rec : num 2 2 0 1 1 0 0 0 0 0 ...  
## $ revol\_bal : num 5572 2875 23473 3479 12948 ...  
## $ revol\_util : num 68.8 54.2 54.5 35.5 45 46.9 13.9 34.8 83.2 40.2 ...  
## $ total\_acc : num 32 26 41 49 26 22 52 36 31 38 ...  
## $ recoveries : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ collections\_12\_mths\_ex\_med: num 0 0 0 0 0 0 0 0 0 0 ...  
## $ acc\_now\_delinq : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ tot\_coll\_amt : num 15386 154 0 0 0 ...  
## $ tot\_cur\_bal : num 13605 19530 23473 45598 16852 ...  
## $ total\_rev\_hi\_lim : num 8100 5300 43100 9800 28800 11500 73900 51600 15000 11800 ...

BUILDING LOGISTIC REGRESSION MODEL

In this section, we will build the logistic regression model that will help us predict of final assesment of further applicants.

One of the assumptions required to build model of logistic regression is not possesing strong correlated variables. To check, if we have ones we decided to build correlation matrix and remove one of pair strongly correlated variables.

## Looking for correlation  
Corr\_ <- cor(loan)  
corrplot(Corr\_, method = "square", type = "upper")



Easily we can notice that we have some strongly correlated variables. we have to remove grade, installment and total\_rev\_hi\_lim.

## Remove of very strong correlated variables  
#loan <- loan[ , -which(names(loan) %in% c("grade", "installment", "total\_rev\_hi\_lim"))]

summary(loan)

## loan\_amnt term int\_rate installment   
## Min. : 1000 Min. :0.0000 Min. : 5.32 Min. : 30.42   
## 1st Qu.: 8000 1st Qu.:1.0000 1st Qu.:11.99 1st Qu.: 262.78   
## Median :12000 Median :1.0000 Median :14.33 Median : 382.94   
## Mean :14051 Mean :0.7672 Mean :14.69 Mean : 438.51   
## 3rd Qu.:19200 3rd Qu.:1.0000 3rd Qu.:17.57 3rd Qu.: 568.64   
## Max. :35000 Max. :1.0000 Max. :28.99 Max. :1424.57   
## grade emp\_length home\_ownership annual\_inc   
## Min. :1.000 Min. : 1.000 Min. :0.000 Min. : 3000   
## 1st Qu.:4.000 1st Qu.: 3.000 1st Qu.:0.000 1st Qu.: 50000   
## Median :5.000 Median : 7.000 Median :1.000 Median : 67000   
## Mean :4.974 Mean : 6.481 Mean :0.629 Mean : 77462   
## 3rd Qu.:6.000 3rd Qu.:10.000 3rd Qu.:1.000 3rd Qu.: 92000   
## Max. :7.000 Max. :10.000 Max. :1.000 Max. :7141778   
## verification\_status loan\_status purpose dti   
## Min. :0.0000 Min. :0.0000 Min. :0.0000 Min. : 0.00   
## 1st Qu.:0.0000 1st Qu.:1.0000 1st Qu.:0.0000 1st Qu.:11.22   
## Median :1.0000 Median :1.0000 Median :0.0000 Median :16.57   
## Mean :0.6939 Mean :0.8168 Mean :0.1799 Mean :17.07   
## 3rd Qu.:1.0000 3rd Qu.:1.0000 3rd Qu.:0.0000 3rd Qu.:22.53   
## Max. :1.0000 Max. :1.0000 Max. :1.0000 Max. :39.99   
## delinq\_2yrs earliest\_cr\_line inq\_last\_6mths   
## Min. : 0.0000 Min. :1954 Min. :0.0000   
## 1st Qu.: 0.0000 1st Qu.:1993 1st Qu.:0.0000   
## Median : 0.0000 Median :1998 Median :1.0000   
## Mean : 0.5897 Mean :1997 Mean :0.9013   
## 3rd Qu.: 1.0000 3rd Qu.:2001 3rd Qu.:1.0000   
## Max. :29.0000 Max. :2012 Max. :7.0000   
## mths\_since\_last\_delinq open\_acc pub\_rec revol\_bal   
## Min. : 0.00 Min. : 1.00 Min. : 0.0000 Min. : 0   
## 1st Qu.: 16.00 1st Qu.: 8.00 1st Qu.: 0.0000 1st Qu.: 5667   
## Median : 32.00 Median :11.00 Median : 0.0000 Median : 10135   
## Mean : 34.82 Mean :11.63 Mean : 0.1858 Mean : 14099   
## 3rd Qu.: 51.00 3rd Qu.:14.00 3rd Qu.: 0.0000 3rd Qu.: 17476   
## Max. :152.00 Max. :55.00 Max. :15.0000 Max. :975800   
## revol\_util total\_acc recoveries   
## Min. : 0.00 Min. : 3.00 Min. : 0.0   
## 1st Qu.: 37.50 1st Qu.: 20.00 1st Qu.: 0.0   
## Median : 55.30 Median : 27.00 Median : 0.0   
## Mean : 54.58 Mean : 28.27 Mean : 177.2   
## 3rd Qu.: 72.50 3rd Qu.: 35.00 3rd Qu.: 0.0   
## Max. :892.30 Max. :110.00 Max. :31900.5   
## collections\_12\_mths\_ex\_med acc\_now\_delinq tot\_coll\_amt   
## Min. :0.00000 Min. :0.000000 Min. : 0   
## 1st Qu.:0.00000 1st Qu.:0.000000 1st Qu.: 0   
## Median :0.00000 Median :0.000000 Median : 0   
## Mean :0.01373 Mean :0.008757 Mean : 368   
## 3rd Qu.:0.00000 3rd Qu.:0.000000 3rd Qu.: 0   
## Max. :6.00000 Max. :5.000000 Max. :9152545   
## tot\_cur\_bal total\_rev\_hi\_lim  
## Min. : 0 Min. : 100   
## 1st Qu.: 31162 1st Qu.: 12100   
## Median : 94374 Median : 20100   
## Mean : 147562 Mean : 26621   
## 3rd Qu.: 221752 3rd Qu.: 33000   
## Max. :8000078 Max. :988000

FIRST MODEL

after all these transformation we can finally build first logistic regression model.

## First model  
model\_1 <- glm(loan\_status ~., data = loan, family = binomial)

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(model\_1)

##   
## Call:  
## glm(formula = loan\_status ~ ., family = binomial, data = loan)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -3.6455 0.2761 0.3928 0.4991 1.3926   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -7.329e-01 3.714e+00 -0.197 0.843550   
## loan\_amnt -3.562e-06 1.167e-05 -0.305 0.760099   
## term 4.014e-01 7.169e-02 5.599 2.16e-08 \*\*\*  
## int\_rate 1.684e-03 9.689e-03 0.174 0.862018   
## installment -3.478e-04 3.576e-04 -0.973 0.330676   
## grade 2.084e-01 3.011e-02 6.920 4.51e-12 \*\*\*  
## emp\_length 7.833e-03 3.432e-03 2.282 0.022478 \*   
## home\_ownership 1.885e-01 2.782e-02 6.774 1.25e-11 \*\*\*  
## annual\_inc 6.167e-06 4.810e-07 12.821 < 2e-16 \*\*\*  
## verification\_status -1.392e-01 2.901e-02 -4.798 1.60e-06 \*\*\*  
## purpose -1.368e-01 3.196e-02 -4.279 1.87e-05 \*\*\*  
## dti -3.072e-02 1.673e-03 -18.356 < 2e-16 \*\*\*  
## delinq\_2yrs -5.054e-02 1.228e-02 -4.117 3.84e-05 \*\*\*  
## earliest\_cr\_line 8.906e-04 1.838e-03 0.485 0.627932   
## inq\_last\_6mths -5.077e-02 1.108e-02 -4.581 4.63e-06 \*\*\*  
## mths\_since\_last\_delinq 1.941e-03 6.510e-04 2.982 0.002861 \*\*   
## open\_acc -2.501e-02 3.309e-03 -7.559 4.06e-14 \*\*\*  
## pub\_rec -7.153e-02 2.156e-02 -3.318 0.000907 \*\*\*  
## revol\_bal -6.694e-06 2.078e-06 -3.222 0.001274 \*\*   
## revol\_util -1.875e-03 7.110e-04 -2.638 0.008351 \*\*   
## total\_acc 1.093e-02 1.427e-03 7.658 1.88e-14 \*\*\*  
## recoveries -1.554e+00 5.606e+00 -0.277 0.781558   
## collections\_12\_mths\_ex\_med -1.997e-01 7.844e-02 -2.546 0.010906 \*   
## acc\_now\_delinq 6.043e-02 1.123e-01 0.538 0.590432   
## tot\_coll\_amt 4.643e-06 6.070e-06 0.765 0.444349   
## tot\_cur\_bal 5.006e-07 1.222e-07 4.097 4.18e-05 \*\*\*  
## total\_rev\_hi\_lim 6.973e-06 1.448e-06 4.815 1.47e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 85580 on 89865 degrees of freedom  
## Residual deviance: 52354 on 89839 degrees of freedom  
## AIC: 52408  
##   
## Number of Fisher Scoring iterations: 25

We can notice that built model is pretty good as it is first one I build after all transformations, becuase most of used variables are relevant. Nevertheless, there are still many ways to improve it. One of the metrics that determines quality of econometric model is AIC criteria. That is why I decided to use stepAIC algorithm to gain model with the lowest possible AIC score which gives us best possible model where main assesment criteria is AIC score of course.

## Secon model adjusted by AIC criteria  
model\_2 <- stepAIC(model\_1, direction = 'both')

## Start: AIC=52407.84  
## loan\_status ~ loan\_amnt + term + int\_rate + installment + grade +   
## emp\_length + home\_ownership + annual\_inc + verification\_status +   
## purpose + dti + delinq\_2yrs + earliest\_cr\_line + inq\_last\_6mths +   
## mths\_since\_last\_delinq + open\_acc + pub\_rec + revol\_bal +   
## revol\_util + total\_acc + recoveries + collections\_12\_mths\_ex\_med +   
## acc\_now\_delinq + tot\_coll\_amt + tot\_cur\_bal + total\_rev\_hi\_lim

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Df Deviance AIC  
## - int\_rate 1 52354 52406  
## - loan\_amnt 1 52354 52406  
## - earliest\_cr\_line 1 52354 52406  
## - acc\_now\_delinq 1 52354 52406  
## - tot\_coll\_amt 1 52355 52407  
## - installment 1 52355 52407  
## <none> 52354 52408  
## - emp\_length 1 52359 52411  
## - collections\_12\_mths\_ex\_med 1 52360 52412  
## - revol\_util 1 52361 52413  
## - mths\_since\_last\_delinq 1 52363 52415  
## - revol\_bal 1 52364 52416  
## - pub\_rec 1 52364 52416  
## - delinq\_2yrs 1 52370 52422  
## - tot\_cur\_bal 1 52371 52423  
## - purpose 1 52372 52424  
## - inq\_last\_6mths 1 52375 52427  
## - verification\_status 1 52377 52429  
## - total\_rev\_hi\_lim 1 52379 52431  
## - term 1 52385 52437  
## - home\_ownership 1 52400 52452  
## - grade 1 52402 52454  
## - open\_acc 1 52410 52462  
## - total\_acc 1 52414 52466  
## - annual\_inc 1 52541 52593  
## - dti 1 52691 52743  
## - recoveries 1 79382 79434

## Warning: glm.fit: algorithm did not converge  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

##   
## Step: AIC=52405.87  
## loan\_status ~ loan\_amnt + term + installment + grade + emp\_length +   
## home\_ownership + annual\_inc + verification\_status + purpose +   
## dti + delinq\_2yrs + earliest\_cr\_line + inq\_last\_6mths + mths\_since\_last\_delinq +   
## open\_acc + pub\_rec + revol\_bal + revol\_util + total\_acc +   
## recoveries + collections\_12\_mths\_ex\_med + acc\_now\_delinq +   
## tot\_coll\_amt + tot\_cur\_bal + total\_rev\_hi\_lim

## Warning: glm.fit: algorithm did not converge  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Df Deviance AIC  
## - loan\_amnt 1 52354 52404  
## - earliest\_cr\_line 1 52354 52404  
## - acc\_now\_delinq 1 52354 52404  
## - tot\_coll\_amt 1 52355 52405  
## - installment 1 52355 52405  
## <none> 52354 52406  
## + int\_rate 1 52354 52408  
## - emp\_length 1 52359 52409  
## - collections\_12\_mths\_ex\_med 1 52360 52410  
## - revol\_util 1 52361 52411  
## - mths\_since\_last\_delinq 1 52363 52413  
## - revol\_bal 1 52364 52414  
## - pub\_rec 1 52365 52415  
## - delinq\_2yrs 1 52370 52420  
## - tot\_cur\_bal 1 52371 52421  
## - purpose 1 52372 52422  
## - inq\_last\_6mths 1 52375 52425  
## - verification\_status 1 52377 52427  
## - total\_rev\_hi\_lim 1 52379 52429  
## - term 1 52386 52436  
## - home\_ownership 1 52400 52450  
## - open\_acc 1 52410 52460  
## - total\_acc 1 52414 52464  
## - annual\_inc 1 52542 52592  
## - grade 1 52584 52634  
## - dti 1 52691 52741  
## - recoveries 1 79690 79740

## Warning: glm.fit: algorithm did not converge  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

##   
## Step: AIC=52404  
## loan\_status ~ term + installment + grade + emp\_length + home\_ownership +   
## annual\_inc + verification\_status + purpose + dti + delinq\_2yrs +   
## earliest\_cr\_line + inq\_last\_6mths + mths\_since\_last\_delinq +   
## open\_acc + pub\_rec + revol\_bal + revol\_util + total\_acc +   
## recoveries + collections\_12\_mths\_ex\_med + acc\_now\_delinq +   
## tot\_coll\_amt + tot\_cur\_bal + total\_rev\_hi\_lim

## Warning: glm.fit: algorithm did not converge  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Df Deviance AIC  
## - earliest\_cr\_line 1 52354 52402  
## - acc\_now\_delinq 1 52354 52402  
## - tot\_coll\_amt 1 52355 52403  
## <none> 52354 52404  
## + loan\_amnt 1 52354 52406  
## + int\_rate 1 52354 52406  
## - emp\_length 1 52359 52407  
## - collections\_12\_mths\_ex\_med 1 52360 52408  
## - revol\_util 1 52361 52409  
## - mths\_since\_last\_delinq 1 52363 52411  
## - revol\_bal 1 52364 52412  
## - pub\_rec 1 52365 52413  
## - delinq\_2yrs 1 52370 52418  
## - tot\_cur\_bal 1 52371 52419  
## - purpose 1 52372 52420  
## - inq\_last\_6mths 1 52375 52423  
## - verification\_status 1 52377 52425  
## - total\_rev\_hi\_lim 1 52379 52427  
## - home\_ownership 1 52400 52448  
## - open\_acc 1 52410 52458  
## - installment 1 52411 52459  
## - total\_acc 1 52414 52462  
## - term 1 52534 52582  
## - annual\_inc 1 52542 52590  
## - grade 1 52657 52705  
## - dti 1 52692 52740  
## - recoveries 1 79696 79744

## Warning: glm.fit: algorithm did not converge  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

##   
## Step: AIC=52402.23  
## loan\_status ~ term + installment + grade + emp\_length + home\_ownership +   
## annual\_inc + verification\_status + purpose + dti + delinq\_2yrs +   
## inq\_last\_6mths + mths\_since\_last\_delinq + open\_acc + pub\_rec +   
## revol\_bal + revol\_util + total\_acc + recoveries + collections\_12\_mths\_ex\_med +   
## acc\_now\_delinq + tot\_coll\_amt + tot\_cur\_bal + total\_rev\_hi\_lim

## Warning: glm.fit: algorithm did not converge  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Df Deviance AIC  
## - acc\_now\_delinq 1 52355 52401  
## - tot\_coll\_amt 1 52355 52401  
## <none> 52354 52402  
## + earliest\_cr\_line 1 52354 52404  
## + loan\_amnt 1 52354 52404  
## + int\_rate 1 52354 52404  
## - emp\_length 1 52359 52405  
## - collections\_12\_mths\_ex\_med 1 52360 52406  
## - revol\_util 1 52361 52407  
## - mths\_since\_last\_delinq 1 52363 52409  
## - revol\_bal 1 52364 52410  
## - pub\_rec 1 52365 52411  
## - delinq\_2yrs 1 52371 52417  
## - tot\_cur\_bal 1 52371 52417  
## - purpose 1 52372 52418  
## - inq\_last\_6mths 1 52375 52421  
## - verification\_status 1 52378 52424  
## - total\_rev\_hi\_lim 1 52379 52425  
## - home\_ownership 1 52400 52446  
## - open\_acc 1 52410 52456  
## - installment 1 52412 52458  
## - total\_acc 1 52415 52461  
## - term 1 52534 52580  
## - annual\_inc 1 52542 52588  
## - grade 1 52657 52703  
## - dti 1 52692 52738  
## - recoveries 1 79700 79746

## Warning: glm.fit: algorithm did not converge  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

##   
## Step: AIC=52400.52  
## loan\_status ~ term + installment + grade + emp\_length + home\_ownership +   
## annual\_inc + verification\_status + purpose + dti + delinq\_2yrs +   
## inq\_last\_6mths + mths\_since\_last\_delinq + open\_acc + pub\_rec +   
## revol\_bal + revol\_util + total\_acc + recoveries + collections\_12\_mths\_ex\_med +   
## tot\_coll\_amt + tot\_cur\_bal + total\_rev\_hi\_lim

## Warning: glm.fit: algorithm did not converge  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Df Deviance AIC  
## - tot\_coll\_amt 1 52355 52399  
## <none> 52355 52401  
## + acc\_now\_delinq 1 52354 52402  
## + earliest\_cr\_line 1 52354 52402  
## + loan\_amnt 1 52354 52402  
## + int\_rate 1 52354 52402  
## - emp\_length 1 52360 52404  
## - collections\_12\_mths\_ex\_med 1 52361 52405  
## - revol\_util 1 52362 52406  
## - mths\_since\_last\_delinq 1 52363 52407  
## - revol\_bal 1 52365 52409  
## - pub\_rec 1 52365 52409  
## - delinq\_2yrs 1 52371 52415  
## - tot\_cur\_bal 1 52372 52416  
## - purpose 1 52373 52417  
## - inq\_last\_6mths 1 52375 52419  
## - verification\_status 1 52378 52422  
## - total\_rev\_hi\_lim 1 52379 52423  
## - home\_ownership 1 52400 52444  
## - open\_acc 1 52411 52455  
## - installment 1 52412 52456  
## - total\_acc 1 52415 52459  
## - term 1 52535 52579  
## - annual\_inc 1 52542 52586  
## - grade 1 52657 52701  
## - dti 1 52692 52736  
## - recoveries 1 79700 79744

## Warning: glm.fit: algorithm did not converge  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

##   
## Step: AIC=52399.25  
## loan\_status ~ term + installment + grade + emp\_length + home\_ownership +   
## annual\_inc + verification\_status + purpose + dti + delinq\_2yrs +   
## inq\_last\_6mths + mths\_since\_last\_delinq + open\_acc + pub\_rec +   
## revol\_bal + revol\_util + total\_acc + recoveries + collections\_12\_mths\_ex\_med +   
## tot\_cur\_bal + total\_rev\_hi\_lim

## Warning: glm.fit: algorithm did not converge  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Df Deviance AIC  
## <none> 52355 52399  
## + tot\_coll\_amt 1 52355 52401  
## + acc\_now\_delinq 1 52355 52401  
## + earliest\_cr\_line 1 52355 52401  
## + loan\_amnt 1 52355 52401  
## + int\_rate 1 52355 52401  
## - emp\_length 1 52360 52402  
## - collections\_12\_mths\_ex\_med 1 52361 52403  
## - revol\_util 1 52363 52405  
## - mths\_since\_last\_delinq 1 52364 52406  
## - revol\_bal 1 52365 52407  
## - pub\_rec 1 52366 52408  
## - delinq\_2yrs 1 52372 52414  
## - tot\_cur\_bal 1 52373 52415  
## - purpose 1 52373 52415  
## - inq\_last\_6mths 1 52376 52418  
## - verification\_status 1 52379 52421  
## - total\_rev\_hi\_lim 1 52380 52422  
## - home\_ownership 1 52401 52443  
## - open\_acc 1 52412 52454  
## - installment 1 52413 52455  
## - total\_acc 1 52416 52458  
## - term 1 52536 52578  
## - annual\_inc 1 52543 52585  
## - grade 1 52658 52700  
## - dti 1 52693 52735  
## - recoveries 1 79701 79743

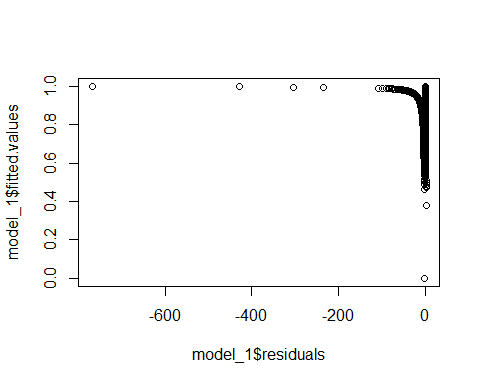
summary(model\_2)

##   
## Call:  
## glm(formula = loan\_status ~ term + installment + grade + emp\_length +   
## home\_ownership + annual\_inc + verification\_status + purpose +   
## dti + delinq\_2yrs + inq\_last\_6mths + mths\_since\_last\_delinq +   
## open\_acc + pub\_rec + revol\_bal + revol\_util + total\_acc +   
## recoveries + collections\_12\_mths\_ex\_med + tot\_cur\_bal + total\_rev\_hi\_lim,   
## family = binomial, data = loan)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -3.6410 0.2761 0.3930 0.4993 1.3887   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.102e+00 9.181e-02 12.005 < 2e-16 \*\*\*  
## term 4.211e-01 3.112e-02 13.534 < 2e-16 \*\*\*  
## installment -4.553e-04 5.974e-05 -7.622 2.51e-14 \*\*\*  
## grade 2.006e-01 1.147e-02 17.484 < 2e-16 \*\*\*  
## emp\_length 7.602e-03 3.386e-03 2.245 0.024751 \*   
## home\_ownership 1.874e-01 2.772e-02 6.761 1.37e-11 \*\*\*  
## annual\_inc 6.148e-06 4.789e-07 12.838 < 2e-16 \*\*\*  
## verification\_status -1.390e-01 2.895e-02 -4.803 1.57e-06 \*\*\*  
## purpose -1.364e-01 3.190e-02 -4.277 1.89e-05 \*\*\*  
## dti -3.073e-02 1.672e-03 -18.382 < 2e-16 \*\*\*  
## delinq\_2yrs -5.077e-02 1.226e-02 -4.140 3.47e-05 \*\*\*  
## inq\_last\_6mths -5.044e-02 1.097e-02 -4.598 4.27e-06 \*\*\*  
## mths\_since\_last\_delinq 1.915e-03 6.478e-04 2.957 0.003111 \*\*   
## open\_acc -2.486e-02 3.293e-03 -7.551 4.31e-14 \*\*\*  
## pub\_rec -7.190e-02 2.148e-02 -3.347 0.000815 \*\*\*  
## revol\_bal -6.627e-06 2.070e-06 -3.202 0.001364 \*\*   
## revol\_util -1.922e-03 7.040e-04 -2.731 0.006320 \*\*   
## total\_acc 1.084e-02 1.404e-03 7.725 1.12e-14 \*\*\*  
## recoveries -1.554e+00 5.613e+00 -0.277 0.781938   
## collections\_12\_mths\_ex\_med -1.957e-01 7.835e-02 -2.498 0.012497 \*   
## tot\_cur\_bal 5.028e-07 1.218e-07 4.129 3.64e-05 \*\*\*  
## total\_rev\_hi\_lim 6.847e-06 1.433e-06 4.779 1.76e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 85580 on 89865 degrees of freedom  
## Residual deviance: 52355 on 89844 degrees of freedom  
## AIC: 52399  
##   
## Number of Fisher Scoring iterations: 25

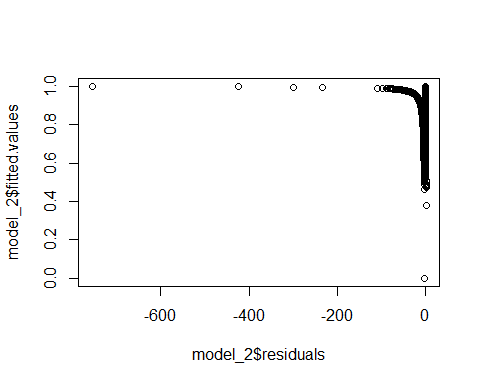
compareCoefs(model\_1, model\_2, se = FALSE)

## Calls:  
## 1: glm(formula = loan\_status ~ ., family = binomial, data = loan)  
## 2: glm(formula = loan\_status ~ term + installment + grade + emp\_length   
## + home\_ownership + annual\_inc + verification\_status + purpose + dti +   
## delinq\_2yrs + inq\_last\_6mths + mths\_since\_last\_delinq + open\_acc +   
## pub\_rec + revol\_bal + revol\_util + total\_acc + recoveries +   
## collections\_12\_mths\_ex\_med + tot\_cur\_bal + total\_rev\_hi\_lim, family =   
## binomial, data = loan)  
##   
## Model 1 Model 2  
## (Intercept) -0.733 1.102  
## loan\_amnt -3.56e-06   
## term 0.401 0.421  
## int\_rate 0.00168   
## installment -0.000348 -0.000455  
## grade 0.208 0.201  
## emp\_length 0.00783 0.00760  
## home\_ownership 0.188 0.187  
## annual\_inc 6.17e-06 6.15e-06  
## verification\_status -0.139 -0.139  
## purpose -0.137 -0.136  
## dti -0.0307 -0.0307  
## delinq\_2yrs -0.0505 -0.0508  
## earliest\_cr\_line 0.000891   
## inq\_last\_6mths -0.0508 -0.0504  
## mths\_since\_last\_delinq 0.00194 0.00192  
## open\_acc -0.0250 -0.0249  
## pub\_rec -0.0715 -0.0719  
## revol\_bal -6.69e-06 -6.63e-06  
## revol\_util -0.00188 -0.00192  
## total\_acc 0.0109 0.0108  
## recoveries -1.55 -1.55  
## collections\_12\_mths\_ex\_med -0.200 -0.196  
## acc\_now\_delinq 0.0604   
## tot\_coll\_amt 4.64e-06   
## tot\_cur\_bal 5.01e-07 5.03e-07  
## total\_rev\_hi\_lim 6.97e-06 6.85e-06

plot(model\_1$fitted.values~model\_1$residuals)



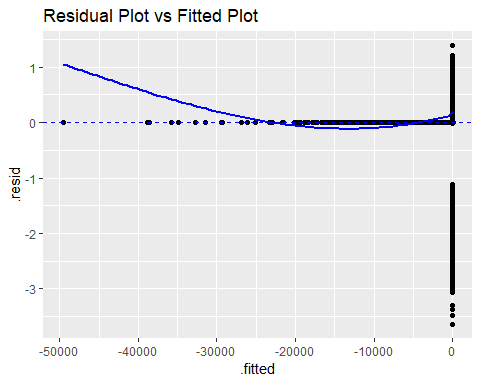
plot(model\_2$fitted.values~model\_2$residuals)



mod = fortify(model\_1)

ggplot(model\_1, aes(.fitted, .resid)) + geom\_point() + geom\_hline(yintercept=0, color="blue", linetype="dashed") + ggtitle("Residual Plot vs Fitted Plot") + geom\_smooth(color = "blue", se = F)

## `geom\_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'



library(lmtest)

## Loading required package: zoo

##   
## Attaching package: 'zoo'

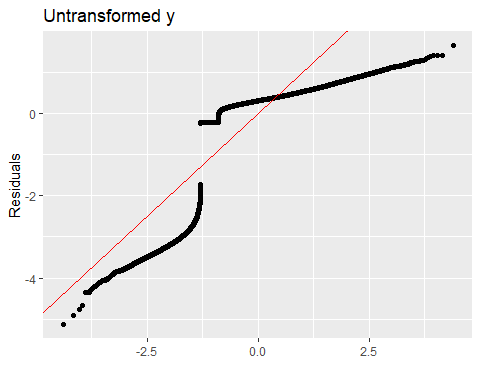
## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

bptest(model\_1, data = loan)

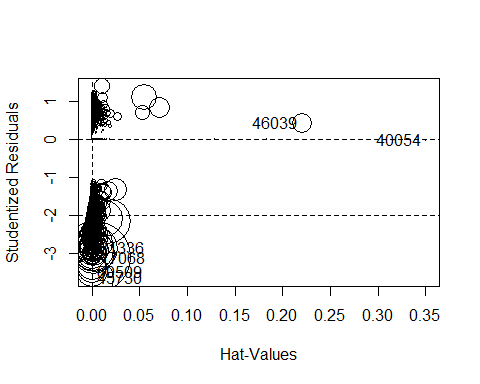
##   
## studentized Breusch-Pagan test  
##   
## data: model\_1  
## BP = 9860.9, df = 26, p-value < 2.2e-16

The test results show that p-value is less than the significance level of 0.05. Hence we reject the null hypothesis that the variance of the residuals is constant and conclude that heteroskedasticiy is present.

p1 = qplot(sample = scale(.resid), data = model\_1) + geom\_abline(intercept = 0, slope = 1, color = "red") + labs(title = "Untransformed y", y = "Residuals")   
p1



influencePlot(model\_1)



## StudRes Hat CookD  
## 17068 -3.12075719 0.0033738378 1.353384e-02  
## 40054 -0.01559239 0.3505238155 2.946363e-06  
## 45730 -3.66299788 0.0001664991 4.736391e-03  
## 46039 0.41947991 0.2204420045 1.071134e-03  
## 60509 -3.49263568 0.0001705204 2.707380e-03  
## 81336 -2.87529356 0.0027058058 5.721061e-03

library(faraway)

##   
## Attaching package: 'faraway'

## The following objects are masked from 'package:car':  
##   
## logit, vif

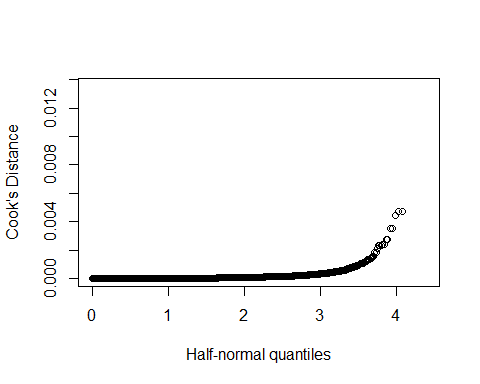
## The following object is masked from 'package:VIF':  
##   
## vif

## The following objects are masked from 'package:survival':  
##   
## rats, solder

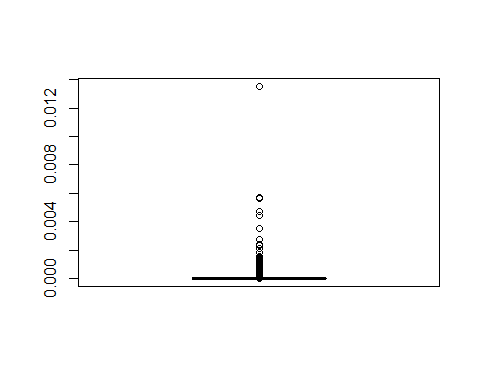
## The following object is masked from 'package:lattice':  
##   
## melanoma

## The following objects are masked from 'package:aod':  
##   
## rats, salmonella

cook <- cooks.distance(model\_1)   
halfnorm(cook, 3, labs = islands, ylab = "Cook's Distance")



boxplot(cook)

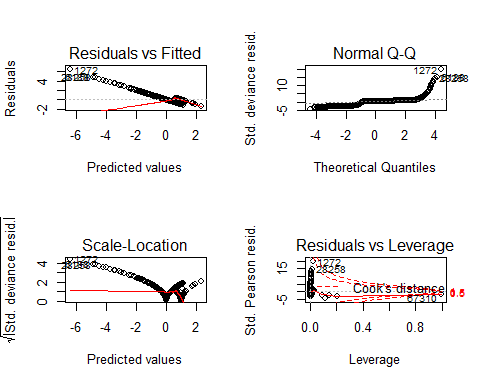


mlam = glm(loan\_status~. , data = loan, subset = (cook < 0.003))  
compareCoefs(model\_1, mlam)

## Calls:  
## 1: glm(formula = loan\_status ~ ., family = binomial, data = loan)  
## 2: glm(formula = loan\_status ~ ., data = loan, subset = (cook < 0.003))  
##   
## Model 1 Model 2  
## (Intercept) -0.733 0.611  
## SE 3.714 0.351  
##   
## loan\_amnt -3.56e-06 5.19e-06  
## SE 1.17e-05 1.21e-06  
##   
## term 0.40137 0.04775  
## SE 0.07169 0.00763  
##   
## int\_rate 0.001684 -0.008666  
## SE 0.009689 0.000926  
##   
## installment -3.48e-04 -1.12e-04  
## SE 3.58e-04 3.69e-05  
##   
## grade 0.208398 -0.000591  
## SE 0.030115 0.002933  
##   
## emp\_length 0.007833 0.001493  
## SE 0.003432 0.000329  
##   
## home\_ownership 0.18847 0.02025  
## SE 0.02782 0.00266  
##   
## annual\_inc 6.17e-06 1.44e-07  
## SE 4.81e-07 2.12e-08  
##   
## verification\_status -0.13917 -0.01425  
## SE 0.02901 0.00263  
##   
## purpose -0.13678 -0.01254  
## SE 0.03196 0.00306  
##   
## dti -0.030718 -0.003664  
## SE 0.001673 0.000159  
##   
## delinq\_2yrs -0.05054 -0.00496  
## SE 0.01228 0.00128  
##   
## earliest\_cr\_line 0.000891 0.000171  
## SE 0.001838 0.000174  
##   
## inq\_last\_6mths -0.05077 -0.00459  
## SE 0.01108 0.00109  
##   
## mths\_since\_last\_delinq 1.94e-03 1.49e-04  
## SE 6.51e-04 6.29e-05  
##   
## open\_acc -0.02501 -0.00252  
## SE 0.00331 0.00032  
##   
## pub\_rec -0.071533 0.000736  
## SE 0.021559 0.002217  
##   
## revol\_bal -6.69e-06 4.28e-08  
## SE 2.08e-06 1.53e-07  
##   
## revol\_util -1.88e-03 -2.57e-04  
## SE 7.11e-04 6.43e-05  
##   
## total\_acc 0.010930 0.001649  
## SE 0.001427 0.000133  
##   
## recoveries -1.55e+00 -2.27e-04  
## SE 5.61e+00 1.47e-06  
##   
## collections\_12\_mths\_ex\_med -0.19968 -0.00417  
## SE 0.07844 0.00869  
##   
## acc\_now\_delinq 0.06043 0.00835  
## SE 0.11229 0.01123  
##   
## tot\_coll\_amt 4.64e-06 1.54e-08  
## SE 6.07e-06 3.62e-08  
##   
## tot\_cur\_bal 5.01e-07 5.28e-08  
## SE 1.22e-07 9.36e-09  
##   
## total\_rev\_hi\_lim 6.97e-06 2.08e-07  
## SE 1.45e-06 1.06e-07  
##

oldpar = par(mfrow = c(2, 2))   
plot(mlam, main = "")

## Warning in sqrt(crit \* p \* (1 - hh)/hh): NaNs produced  
  
## Warning in sqrt(crit \* p \* (1 - hh)/hh): NaNs produced



#install.packages("moments")  
library(moments)

##   
## Attaching package: 'moments'

## The following objects are masked from 'package:e1071':  
##   
## kurtosis, moment, skewness

skewness(loan)

## loan\_amnt term   
## 0.8140249 -1.2644909   
## int\_rate installment   
## 0.3327261 1.0199656   
## grade emp\_length   
## -0.5794318 -0.3443233   
## home\_ownership annual\_inc   
## -0.5339474 38.9941747   
## verification\_status loan\_status   
## -0.8412773 -1.6382589   
## purpose dti   
## 1.6663324 0.2535531   
## delinq\_2yrs earliest\_cr\_line   
## 3.9405709 -1.1429184   
## inq\_last\_6mths mths\_since\_last\_delinq   
## 1.3645965 0.4319950   
## open\_acc pub\_rec   
## 1.1719153 4.6714325   
## revol\_bal revol\_util   
## 10.0524398 0.3589820   
## total\_acc recoveries   
## 0.7985078 8.1297612   
## collections\_12\_mths\_ex\_med acc\_now\_delinq   
## 11.7164447 14.4606693   
## tot\_coll\_amt tot\_cur\_bal   
## 297.2561767 3.9841140   
## total\_rev\_hi\_lim   
## 5.9191485

str(loan)

## 'data.frame': 89866 obs. of 27 variables:  
## $ loan\_amnt : num 12000 3000 20800 7200 20000 10000 14000 13000 12000 12000 ...  
## $ term : num 1 1 1 1 1 0 1 1 1 1 ...  
## $ int\_rate : num 13.5 12.8 13.5 11 14 ...  
## $ installment : num 407 101 706 236 683 ...  
## $ grade : int 6 6 6 6 5 5 7 6 5 6 ...  
## $ emp\_length : int 10 10 10 4 10 5 10 10 5 10 ...  
## $ home\_ownership : num 0 0 0 1 1 0 1 1 1 1 ...  
## $ annual\_inc : num 40000 25000 81500 70000 80000 ...  
## $ verification\_status : num 1 1 1 1 1 1 1 0 0 1 ...  
## $ loan\_status : num 1 1 1 1 1 0 1 1 1 1 ...  
## $ purpose : num 0 0 0 0 0 0 0 1 0 0 ...  
## $ dti : num 16.9 24.7 16.7 19.2 16.7 ...  
## $ delinq\_2yrs : num 0 0 0 0 1 0 0 1 1 1 ...  
## $ earliest\_cr\_line : num 1998 1991 1998 1978 2001 ...  
## $ inq\_last\_6mths : num 0 0 2 0 1 0 4 2 1 2 ...  
## $ mths\_since\_last\_delinq : num 53 58 64 59 8 63 30 11 17 20 ...  
## $ open\_acc : num 7 5 29 14 10 13 20 12 11 13 ...  
## $ pub\_rec : num 2 2 0 1 1 0 0 0 0 0 ...  
## $ revol\_bal : num 5572 2875 23473 3479 12948 ...  
## $ revol\_util : num 68.8 54.2 54.5 35.5 45 46.9 13.9 34.8 83.2 40.2 ...  
## $ total\_acc : num 32 26 41 49 26 22 52 36 31 38 ...  
## $ recoveries : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ collections\_12\_mths\_ex\_med: num 0 0 0 0 0 0 0 0 0 0 ...  
## $ acc\_now\_delinq : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ tot\_coll\_amt : num 15386 154 0 0 0 ...  
## $ tot\_cur\_bal : num 13605 19530 23473 45598 16852 ...  
## $ total\_rev\_hi\_lim : num 8100 5300 43100 9800 28800 11500 73900 51600 15000 11800 ...

PREDICTION

To make a prediction, at first we need to split our database on train and test datasets. To check if model built on one set of data will predict correctly data that were not used to built one. We split data with propotion 20% test, 80% train.

## Preparing train/test dataset  
perc <- floor((nrow(loan)/4)\*3)   
loan <- loan[sample(nrow(loan)), ]   
data.train <- loan[1:perc, ]   
data.test <- loan[(perc+1):nrow(loan), ]  
data.train <- na.omit(data.train)  
data.test <- na.omit(data.test)

Now we build model with the same variables as we’ve got in model 2, but using train data.

##New training model  
model\_train <- glm(loan\_status ~ loan\_amnt + term + int\_rate + emp\_length + home\_ownership + annual\_inc +  
 verification\_status + purpose + dti + delinq\_2yrs + inq\_last\_6mths + mths\_since\_last\_delinq +   
 open\_acc + pub\_rec + revol\_bal + revol\_util + total\_acc + recoveries + collections\_12\_mths\_ex\_med+  
 tot\_cur\_bal, data = data.train, family = binomial)

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(model\_train)

##   
## Call:  
## glm(formula = loan\_status ~ loan\_amnt + term + int\_rate + emp\_length +   
## home\_ownership + annual\_inc + verification\_status + purpose +   
## dti + delinq\_2yrs + inq\_last\_6mths + mths\_since\_last\_delinq +   
## open\_acc + pub\_rec + revol\_bal + revol\_util + total\_acc +   
## recoveries + collections\_12\_mths\_ex\_med + tot\_cur\_bal, family = binomial,   
## data = data.train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -3.6324 0.2772 0.3962 0.5032 1.7398   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 3.178e+00 1.056e-01 30.085 < 2e-16 \*\*\*  
## loan\_amnt -1.307e-05 2.239e-06 -5.839 5.25e-09 \*\*\*  
## term 3.697e-01 3.679e-02 10.049 < 2e-16 \*\*\*  
## int\_rate -6.318e-02 4.014e-03 -15.740 < 2e-16 \*\*\*  
## emp\_length 7.127e-03 3.893e-03 1.831 0.067142 .   
## home\_ownership 2.013e-01 3.182e-02 6.325 2.53e-10 \*\*\*  
## annual\_inc 6.208e-06 5.546e-07 11.193 < 2e-16 \*\*\*  
## verification\_status -1.173e-01 3.325e-02 -3.528 0.000419 \*\*\*  
## purpose -1.458e-01 3.667e-02 -3.977 6.98e-05 \*\*\*  
## dti -3.160e-02 1.926e-03 -16.409 < 2e-16 \*\*\*  
## delinq\_2yrs -5.457e-02 1.414e-02 -3.859 0.000114 \*\*\*  
## inq\_last\_6mths -4.609e-02 1.278e-02 -3.607 0.000310 \*\*\*  
## mths\_since\_last\_delinq 1.879e-03 7.450e-04 2.522 0.011673 \*   
## open\_acc -2.030e-02 3.700e-03 -5.487 4.08e-08 \*\*\*  
## pub\_rec -8.930e-02 2.446e-02 -3.651 0.000261 \*\*\*  
## revol\_bal 3.028e-06 1.200e-06 2.522 0.011666 \*   
## revol\_util -4.413e-03 6.587e-04 -6.700 2.08e-11 \*\*\*  
## total\_acc 1.057e-02 1.616e-03 6.543 6.04e-11 \*\*\*  
## recoveries -1.584e+00 6.515e+00 -0.243 0.807890   
## collections\_12\_mths\_ex\_med -2.317e-01 9.372e-02 -2.472 0.013422 \*   
## tot\_cur\_bal 4.425e-07 1.397e-07 3.167 0.001540 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 64498 on 67398 degrees of freedom  
## Residual deviance: 39547 on 67378 degrees of freedom  
## AIC: 39589  
##   
## Number of Fisher Scoring iterations: 25

round\_df(data.frame(PseudoR2(model\_train)), 4)

## PseudoR2.model\_train.  
## McFadden 0.3869  
## Adj.McFadden 0.3862  
## Cox.Snell 0.3094  
## Nagelkerke 0.5023  
## McKelvey.Zavoina 1.0000  
## Effron 0.4440  
## Count 0.9021  
## Adj.Count 0.4698  
## AIC 39588.7495  
## Corrected.AIC 39588.7632

Model is slightly worse as it has more irrelevant variables comparing to model 2, but still passes McFadden Pseudo-R2 and McKelvey & Zavoin Pseudo-R2 criteria.

Now we can make prediction on our test data.

#install.packages("caret")  
#library("caret")  
## Prediction on data.test  
results <- predict.glm(model\_train, newdata = data.test, type = "response")  
data.test$prob <- results  
data.test$prob <- ifelse(data.test$prob <= 0.5 , 0, 1)  
data.test$prob <- as.factor(data.test$prob)  
data.test$loan\_status <- as.factor(data.test$loan\_status)  
table(data.test$prob, data.test$loan\_status)

##   
## 0 1  
## 0 1885 1  
## 1 2125 18456

ACCURACY OF THE FINAL MODEL

miscerror <- mean(data.test$prob != data.test$loan\_status, na.rm = T)  
print(paste('accuracy', 1-miscerror))

## [1] "accuracy 0.905372323852762"

ROC Curve

library(ROCR)  
#p <- predict(model\_train, newdata = data.test, type = "response")  
#pr <- prediction(p, data.test$loan\_status )

#plot(roc(data.test$loan\_status, p, direction="<"),  
 #col="yellow", lwd=3, main="The turtle finds its way")  
  
#glm\_simple\_roc <- simple\_roc(data.test$loan\_status=="TRUE", glm\_link\_scores)  
#with(glm\_simple\_roc, points(1 - FPR, TPR, col=1 + labels))

t.test(data.train, conf.level = 0.9)

##   
## One Sample t-test  
##   
## data: data.train  
## t = 304.8, df = 1819800, p-value < 2.2e-16  
## alternative hypothesis: true mean is not equal to 0  
## 90 percent confidence interval:  
## 10427.99 10541.15  
## sample estimates:  
## mean of x   
## 10484.57

anova(model\_1, test="Chisq")

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred  
  
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Analysis of Deviance Table  
##   
## Model: binomial, link: logit  
##   
## Response: loan\_status  
##   
## Terms added sequentially (first to last)  
##   
##   
## Df Deviance Resid. Df Resid. Dev Pr(>Chi)   
## NULL 89865 85580   
## loan\_amnt 1 393.1 89864 85187 < 2.2e-16 \*\*\*  
## term 1 1166.3 89863 84021 < 2.2e-16 \*\*\*  
## int\_rate 1 2894.2 89862 81126 < 2.2e-16 \*\*\*  
## installment 1 4.2 89861 81122 0.04104 \*   
## grade 1 32.6 89860 81090 1.105e-08 \*\*\*  
## emp\_length 1 44.2 89859 81045 3.012e-11 \*\*\*  
## home\_ownership 1 159.3 89858 80886 < 2.2e-16 \*\*\*  
## annual\_inc 1 696.0 89857 80190 < 2.2e-16 \*\*\*  
## verification\_status 1 47.4 89856 80143 5.727e-12 \*\*\*  
## purpose 1 4.8 89855 80138 0.02874 \*   
## dti 1 431.6 89854 79706 < 2.2e-16 \*\*\*  
## delinq\_2yrs 1 55.3 89853 79651 1.021e-13 \*\*\*  
## earliest\_cr\_line 1 0.7 89852 79650 0.41322   
## inq\_last\_6mths 1 17.0 89851 79633 3.668e-05 \*\*\*  
## mths\_since\_last\_delinq 1 5.4 89850 79628 0.02050 \*   
## open\_acc 1 0.6 89849 79627 0.43690   
## pub\_rec 1 0.4 89848 79627 0.52764   
## revol\_bal 1 2.8 89847 79624 0.09319 .   
## revol\_util 1 66.1 89846 79558 4.191e-16 \*\*\*  
## total\_acc 1 130.1 89845 79428 < 2.2e-16 \*\*\*  
## recoveries 1 27026.5 89844 52401 < 2.2e-16 \*\*\*  
## collections\_12\_mths\_ex\_med 1 6.1 89843 52395 0.01382 \*   
## acc\_now\_delinq 1 0.6 89842 52395 0.44917   
## tot\_coll\_amt 1 0.6 89841 52394 0.45640   
## tot\_cur\_bal 1 15.3 89840 52379 9.189e-05 \*\*\*  
## total\_rev\_hi\_lim 1 25.1 89839 52354 5.485e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

anova(model\_1, model\_2)

## Analysis of Deviance Table  
##   
## Model 1: loan\_status ~ loan\_amnt + term + int\_rate + installment + grade +   
## emp\_length + home\_ownership + annual\_inc + verification\_status +   
## purpose + dti + delinq\_2yrs + earliest\_cr\_line + inq\_last\_6mths +   
## mths\_since\_last\_delinq + open\_acc + pub\_rec + revol\_bal +   
## revol\_util + total\_acc + recoveries + collections\_12\_mths\_ex\_med +   
## acc\_now\_delinq + tot\_coll\_amt + tot\_cur\_bal + total\_rev\_hi\_lim  
## Model 2: loan\_status ~ term + installment + grade + emp\_length + home\_ownership +   
## annual\_inc + verification\_status + purpose + dti + delinq\_2yrs +   
## inq\_last\_6mths + mths\_since\_last\_delinq + open\_acc + pub\_rec +   
## revol\_bal + revol\_util + total\_acc + recoveries + collections\_12\_mths\_ex\_med +   
## tot\_cur\_bal + total\_rev\_hi\_lim  
## Resid. Df Resid. Dev Df Deviance  
## 1 89839 52354   
## 2 89844 52355 -5 -1.4029

#install.packages("pscl")  
library(pscl)

## Classes and Methods for R developed in the  
## Political Science Computational Laboratory  
## Department of Political Science  
## Stanford University  
## Simon Jackman  
## hurdle and zeroinfl functions by Achim Zeileis

pR2(model\_1)

## llh llhNull G2 McFadden r2ML   
## -2.617692e+04 -4.279004e+04 3.322624e+04 3.882474e-01 3.090798e-01   
## r2CU   
## 5.032640e-01

pR2(model\_2)

## llh llhNull G2 McFadden r2ML   
## -2.617762e+04 -4.279004e+04 3.322484e+04 3.882310e-01 3.090690e-01   
## r2CU   
## 5.032464e-01