Assignment-4

Anil Akyildirim, John K. Hancock, John Suh, Emmanuel Hayble-Gomes, Chunjie Nan

04/05/2020

Contents

Introduction

In this homework assignment, we will explore, analyze and model a data set containing approximately 8000 records representing a customer at an auto insurance company. Each record has two response variables. The first response variable, TARGET_FLAG, is a 1 or a 0. A "1" means that the person was in a car crash. A zero means that the person was not in a car crash. The second response variable is TARGET_AMT. This value is zero if the person did not crash their car. But if they did crash their car, this number will be a value greater than zero.

Our objective is to build multiple linear regression and binary logistic regression models on the training data to predict the probability that a person will crash their car and also the amount of money it will cost if the person does crash their car. You can only use the variables given to you (or variables that you derive from the variables provided). Below is a short description of the variables of interest in the data set:

About the Data

** Index: Identification Variable (do not use)

** TARGET_FLAG: Was Car in a crash? 1=YES 0=NO

** TARGET_AMT: If car was in a crash, what was the cost

** AGE: Age of Driver

** BLUEBOOK: Value of Vehicle

** CAR_AGE: Vehicle Age

** CAR_TYPE: Type of Car

** CAR_USE: Vehicle Use

** CLM_FREQ: # Claims (Past 5 Years)

** EDUCATION: Max Education Level

** HOMEKIDS: # Children at Home

** HOME_VAL: Home Value

** INCOME: Income

** JOB: Job Category

** KIDSDRIV: # Driving Children

** MSTATUS: Marital Status

** MVR_PTS: Motor Vehicle Record Points

** OLDCLAIM: Total Claims (Past 5 Years)

** PARENT1: Single Parent

** RED_CAR: A Red Car

** REVOKED: License Revoked (Past 7 Years)

** SEX: Gender

** TIF: Time in Force

** TRAVTIME: Distance to Work

** URBANICITY: Home/Work Area

** YOJ: Years on Job

Data Exploration

```
# Load Libraries
library(ggplot2)
library(ggcorrplot)
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 3.6.2
```

```
## corrplot 0.84 loaded
```

```
library(psych)
```

```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```r
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
```

```r
library(tidyr)
library(caret)
```

```
## Loading required package: lattice
```

```r
library(MASS)
```

```
##
## Attaching package: 'MASS'

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

```r
library(pROC)
```

```
## Warning: package 'pROC' was built under R version 3.6.3

## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```r
library(glmnet)
```

```
## Loading required package: Matrix

##
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack

## Loaded glmnet 3.0-1
```

```r
library(mltest)
library(stringr)
library(ggpubr)
```

```
## Warning: package 'ggpubr' was built under R version 3.6.3

## Loading required package: magrittr

##
## Attaching package: 'magrittr'

## The following object is masked from 'package:tidyr':
##
##     extract
```

```r
library(geoR)
```

```
## Warning: package 'geoR' was built under R version 3.6.3

## --------------------------------------------------------------
##  Analysis of Geostatistical Data
##  For an Introduction to geoR go to http://www.leg.ufpr.br/geoR
##  geoR version 1.8-1 (built on 2020-02-08) is now loaded
## --------------------------------------------------------------
```

```r
library(mice)
```

```
## Warning: package 'mice' was built under R version 3.6.3

##
## Attaching package: 'mice'

## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```r
library(knitr)
library(kableExtra)
```

```
##
## Attaching package: 'kableExtra'

## The following object is masked from 'package:dplyr':
##
##     group_rows
```

```r
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

```r
library(MuMIn)
```

```
## Warning: package 'MuMIn' was built under R version 3.6.3
```

```r
# Load the Datasets

insurance_train <- read.csv("https://raw.githubusercontent.com/anilak1978/data621/master/insurance_trai

insurance_eva <- read.csv("https://raw.githubusercontent.com/anilak1978/data621/master/insurance-evalua
```

We have loaded both train and evaluation data sets into R. Let's take a look at the first few observations in the training and evaluation data set.

```r
head(insurance_train)
```

```
##   INDEX TARGET_FLAG TARGET_AMT KIDSDRIV AGE HOMEKIDS YOJ   INCOME PARENT1
## 1     1           0          0        0  60        0  11  $67,349      No
## 2     2           0          0        0  43        0  11  $91,449      No
## 3     4           0          0        0  35        1  10  $16,039      No
## 4     5           0          0        0  51        0  14                No
## 5     6           0          0        0  50        0  NA $114,986      No
## 6     7           1       2946        0  34        1  12 $125,301     Yes
##   HOME_VAL MSTATUS SEX     EDUCATION           JOB TRAVTIME    CAR_USE
## 1       $0    z_No   M           PhD  Professional       14    Private
## 2 $257,252    z_No   M z_High School z_Blue Collar       22 Commercial
## 3 $124,191     Yes z_F z_High School      Clerical        5    Private
## 4 $306,251     Yes   M  <High School z_Blue Collar       32    Private
## 5 $243,925     Yes z_F           PhD        Doctor       36    Private
## 6       $0    z_No z_F     Bachelors z_Blue Collar       46 Commercial
##   BLUEBOOK TIF   CAR_TYPE RED_CAR OLDCLAIM CLM_FREQ REVOKED MVR_PTS
## 1  $14,230  11    Minivan     yes   $4,461        2      No       3
## 2  $14,940   1    Minivan     yes       $0        0      No       0
## 3   $4,010   4      z_SUV      no  $38,690        2      No       3
## 4  $15,440   7    Minivan     yes       $0        0      No       0
## 5  $18,000   1      z_SUV      no  $19,217        2     Yes       3
## 6  $17,430   1 Sports Car      no       $0        0      No       0
##   CAR_AGE          URBANICITY
## 1      18 Highly Urban/ Urban
## 2       1 Highly Urban/ Urban
## 3      10 Highly Urban/ Urban
## 4       6 Highly Urban/ Urban
## 5      17 Highly Urban/ Urban
## 6       7 Highly Urban/ Urban
```

```r
head(insurance_eva)
```

```
##   INDEX TARGET_FLAG TARGET_AMT KIDSDRIV AGE HOMEKIDS YOJ  INCOME PARENT1
## 1     3          NA         NA        0  48        0  11 $52,881      No
## 2     9          NA         NA        1  40        1  11 $50,815     Yes
## 3    10          NA         NA        0  44        2  12 $43,486     Yes
## 4    18          NA         NA        0  35        2  NA $21,204     Yes
## 5    21          NA         NA        0  59        0  12 $87,460      No
## 6    30          NA         NA        0  46        0  14              No
##   HOME_VAL MSTATUS SEX     EDUCATION           JOB TRAVTIME    CAR_USE
## 1       $0    z_No   M     Bachelors       Manager       26    Private
## 2       $0    z_No   M z_High School       Manager       21    Private
## 3       $0    z_No z_F z_High School z_Blue Collar       30 Commercial
## 4       $0    z_No   M z_High School      Clerical       74    Private
## 5       $0    z_No   M z_High School       Manager       45    Private
## 6 $207,519     Yes   M     Bachelors  Professional        7 Commercial
##   BLUEBOOK TIF    CAR_TYPE RED_CAR OLDCLAIM CLM_FREQ REVOKED MVR_PTS
## 1  $21,970   1         Van     yes       $0        0      No       2
## 2  $18,930   6     Minivan      no   $3,295        1      No       2
## 3   $5,900  10       z_SUV      no       $0        0      No       0
## 4   $9,230   6      Pickup      no       $0        0     Yes       0
## 5  $15,420   1     Minivan     yes  $44,857        2      No       4
## 6  $25,660   1 Panel Truck      no   $2,119        1      No       2
##   CAR_AGE            URBANICITY
## 1      10   Highly Urban/ Urban
## 2       1   Highly Urban/ Urban
## 3      10 z_Highly Rural/ Rural
## 4       4 z_Highly Rural/ Rural
## 5       1   Highly Urban/ Urban
## 6      12   Highly Urban/ Urban
```

We have some issues with the data values with $. on some columns. We also columns that have "z_" and "<" values.

Let's fix the "$" in both training and evaluation datasets.

```r
currency_fix <- function(x) {
  num <- str_replace_all(x, "\\$","")
  num <- as.numeric(str_replace_all(num, "\\,",""))
  num
}
```

```r
#train data
insurance_train$INCOME <- currency_fix(insurance_train$INCOME)
insurance_train$HOME_VAL <- currency_fix(insurance_train$HOME_VAL)
insurance_train$BLUEBOOK <- currency_fix(insurance_train$BLUEBOOK)
insurance_train$OLDCLAIM <- currency_fix(insurance_train$OLDCLAIM)

# test data
insurance_eva$INCOME <- currency_fix(insurance_eva$INCOME)
insurance_eva$HOME_VAL <- currency_fix(insurance_eva$HOME_VAL)
insurance_eva$BLUEBOOK <- currency_fix(insurance_eva$BLUEBOOK)
insurance_eva$OLDCLAIM <- currency_fix(insurance_eva$OLDCLAIM)
```

Now lets fix the "z_" and "<" in both train and evaluation data sets.

```
# train data
insurance_train[sapply(insurance_train, is.factor)] <- lapply(insurance_train[sapply(insurance_train, is
                                                  function(x) str_replace(x,"z_|<",""))

insurance_train[sapply(insurance_train, is.character)] <- lapply(insurance_train[sapply(insurance_train

# test data
insurance_eva[sapply(insurance_eva, is.factor)] <- lapply(insurance_eva[sapply(insurance_eva, is.factor)
                                                  function(x) str_replace(x,"z_|<",""))

insurance_eva[sapply(insurance_eva, is.character)] <- lapply(insurance_eva[sapply(insurance_eva, is.cha
```

We fixed the strange characters in both train and evaluation data sets. Let's look at the structure of our training data sets.

```
str(insurance_train)
```

```
## 'data.frame':    8161 obs. of  26 variables:
##  $ INDEX      : int  1 2 4 5 6 7 8 11 12 13 ...
##  $ TARGET_FLAG: int  0 0 0 0 0 1 0 1 1 0 ...
##  $ TARGET_AMT : num  0 0 0 0 0 ...
##  $ KIDSDRIV   : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ AGE        : int  60 43 35 51 50 34 54 37 34 50 ...
##  $ HOMEKIDS   : int  0 0 1 0 0 1 0 2 0 0 ...
##  $ YOJ        : int  11 11 10 14 NA 12 NA NA 10 7 ...
##  $ INCOME     : num  67349 91449 16039 NA 114986 ...
##  $ PARENT1    : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 2 1 1 1 1 ...
##  $ HOME_VAL   : num  0 257252 124191 306251 243925 ...
##  $ MSTATUS    : Factor w/ 2 levels "No","Yes": 1 1 2 2 2 1 2 2 1 1 ...
##  $ SEX        : Factor w/ 2 levels "F","M": 2 2 1 2 1 1 1 2 1 2 ...
##  $ EDUCATION  : Factor w/ 4 levels "Bachelors","High School",..: 4 2 2 2 4 1 2 1 1 1 ...
##  $ JOB        : Factor w/ 9 levels "","Blue Collar",..: 8 2 3 2 4 2 2 2 3 8 ...
##  $ TRAVTIME   : int  14 22 5 32 36 46 33 44 34 48 ...
##  $ CAR_USE    : Factor w/ 2 levels "Commercial","Private": 2 1 2 2 2 1 2 1 2 1 ...
##  $ BLUEBOOK   : num  14230 14940 4010 15440 18000 ...
##  $ TIF        : int  11 1 4 7 1 1 1 1 1 7 ...
##  $ CAR_TYPE   : Factor w/ 6 levels "Minivan","Panel Truck",..: 1 1 5 1 5 4 5 6 5 6 ...
##  $ RED_CAR    : Factor w/ 2 levels "no","yes": 2 2 1 2 1 1 1 2 1 1 ...
##  $ OLDCLAIM   : num  4461 0 38690 0 19217 ...
##  $ CLM_FREQ   : int  2 0 2 0 2 0 0 1 0 0 ...
##  $ REVOKED    : Factor w/ 2 levels "No","Yes": 1 1 1 1 2 1 1 2 1 1 ...
##  $ MVR_PTS    : int  3 0 3 0 3 0 0 10 0 1 ...
##  $ CAR_AGE    : int  18 1 10 6 17 7 1 7 1 17 ...
##  $ URBANICITY : Factor w/ 2 levels "Highly Rural/ Rural",..: 2 2 2 2 2 2 2 2 2 1 ...
```

We have two response variables `TARGET_FLAG`and `TARGET_AMT` contains numerical and binary values. We want to make sure the binary TARGET_FLAG response variable is a factor for our Data Exploration.

```
# train data
insurance_train$TARGET_FLAG=as.factor(insurance_train$TARGET_FLAG)
```

```r
# test data
insurance_eva$TARGET_FLAG=as.factor(insurance_eva$TARGET_FLAG)
```

We can also ignore the INDEX variable as it doesnt have any impact to analysis.

KIDSDRIV, HOMEKIDS, CLM_FREQ, MVR_PTS, AGE, YOJ, TRAVTIME, TIF, CAR_AGE are discrete variables. PARENT1, MSTATUS, SEX, CAR_USE, RED_CAR, REVOKED, URBANCITY are binary categorical variables. JOB, CAR_TYPE, EDUCATION are other categorical variables. INCOME, HOME_VAL, BLUEBOOK and OLDCLAIM are continous numerical variables.

Let's review some of the basic descriptive statistics.

```r
# look at descriptive statistics
metastats <- data.frame(describe(insurance_train))
metastats <- tibble::rownames_to_column(metastats, "STATS")
metastats["pct_missing"] <- round(metastats["n"]/8161, 3)
head(metastats)
```

```
##          STATS vars    n         mean           sd median      trimmed
## 1        INDEX    1 8161 5151.8676633 2978.8939616   5133 5.151931e+03
## 2 TARGET_FLAG*    2 8161    1.2638157    0.4407276      1 1.204779e+00
## 3   TARGET_AMT    3 8161 1504.3246481 4704.0269298      0 5.937121e+02
## 4     KIDSDRIV    4 8161    0.1710575    0.5115341      0 2.527186e-02
## 5          AGE    5 8155   44.7903127    8.6275895     45 4.483065e+01
## 6     HOMEKIDS    6 8161    0.7212351    1.1163233      0 4.971665e-01
##         mad min      max    range         skew     kurtosis           se
## 1 3841.4166   1  10302.0  10301.0  0.002003877  -1.20342129 32.974889978
## 2    0.0000   1      2.0      1.0  1.071661372  -0.85164621  0.004878637
## 3    0.0000   0 107586.1 107586.1  8.706303371 112.28843858 52.071262844
## 4    0.0000   0      4.0      4.0  3.351837433  11.78019156  0.005662431
## 5    8.8956  16     81.0     65.0 -0.028988948  -0.06170196  0.095538295
## 6    0.0000   0      5.0      5.0  1.341127092   0.64899146  0.012357149
##   pct_missing
## 1       1.000
## 2       1.000
## 3       1.000
## 4       1.000
## 5       0.999
## 6       1.000
```

```r
summary(insurance_train)
```

```
##      INDEX        TARGET_FLAG   TARGET_AMT        KIDSDRIV
##  Min.   :    1   0:6008      Min.   :     0   Min.   :0.0000
##  1st Qu.: 2559   1:2153      1st Qu.:     0   1st Qu.:0.0000
##  Median : 5133               Median :     0   Median :0.0000
##  Mean   : 5152               Mean   :  1504   Mean   :0.1711
##  3rd Qu.: 7745               3rd Qu.:  1036   3rd Qu.:0.0000
##  Max.   :10302               Max.   :107586   Max.   :4.0000
##
##       AGE           HOMEKIDS           YOJ           INCOME
##  Min.   :16.00   Min.   :0.0000   Min.   : 0.0   Min.   :     0
##  1st Qu.:39.00   1st Qu.:0.0000   1st Qu.: 9.0   1st Qu.: 28097
```

```
##  Median :45.00   Median :0.0000   Median :11.0   Median : 54028
##  Mean   :44.79   Mean   :0.7212   Mean   :10.5   Mean   : 61898
##  3rd Qu.:51.00   3rd Qu.:1.0000   3rd Qu.:13.0   3rd Qu.: 85986
##  Max.   :81.00   Max.   :5.0000   Max.   :23.0   Max.   :367030
##  NA's   :6                        NA's   :454    NA's   :445
##  PARENT1       HOME_VAL      MSTATUS    SEX           EDUCATION
##  No :7084   Min.   :     0   No :3267   F:4375   Bachelors   :2242
##  Yes:1077   1st Qu.:     0   Yes:4894   M:3786   High School:3533
##             Median :161160                       Masters    :1658
##             Mean   :154867                       PhD        : 728
##             3rd Qu.:238724
##             Max.   :885282
##             NA's   :464
##            JOB          TRAVTIME           CAR_USE        BLUEBOOK
##  Blue Collar :1825   Min.   :  5.00   Commercial:3029   Min.   : 1500
##  Clerical    :1271   1st Qu.: 22.00   Private   :5132   1st Qu.: 9280
##  Professional:1117   Median : 33.00                     Median :14440
##  Manager     : 988   Mean   : 33.49                     Mean   :15710
##  Lawyer      : 835   3rd Qu.: 44.00                     3rd Qu.:20850
##  Student     : 712   Max.   :142.00                     Max.   :69740
##  (Other)     :1413
##       TIF                  CAR_TYPE    RED_CAR       OLDCLAIM
##  Min.   : 1.000   Minivan    :2145   no :5783   Min.   :    0
##  1st Qu.: 1.000   Panel Truck: 676   yes:2378   1st Qu.:    0
##  Median : 4.000   Pickup     :1389              Median :    0
##  Mean   : 5.351   Sports Car : 907              Mean   : 4037
##  3rd Qu.: 7.000   SUV        :2294              3rd Qu.: 4636
##  Max.   :25.000   Van        : 750              Max.   :57037
##
##     CLM_FREQ      REVOKED       MVR_PTS          CAR_AGE
##  Min.   :0.0000   No :7161   Min.   : 0.000   Min.   :-3.000
##  1st Qu.:0.0000   Yes:1000   1st Qu.: 0.000   1st Qu.: 1.000
##  Median :0.0000              Median : 1.000   Median : 8.000
##  Mean   :0.7986              Mean   : 1.696   Mean   : 8.328
##  3rd Qu.:2.0000              3rd Qu.: 3.000   3rd Qu.:12.000
##  Max.   :5.0000              Max.   :13.000   Max.   :28.000
##                                               NA's   :510
##                URBANICITY
##  Highly Rural/ Rural:1669
##  Highly Urban/ Urban:6492
##
##
##
##
##
```

Let's look to see if there are any missing values.

```r
colSums(is.na(insurance_train))
```

```
##       INDEX TARGET_FLAG  TARGET_AMT    KIDSDRIV         AGE    HOMEKIDS
##           0           0           0           0           6           0
##         YOJ      INCOME     PARENT1    HOME_VAL     MSTATUS         SEX
```

```
##         454         445           0         464           0           0
##   EDUCATION         JOB    TRAVTIME     CAR_USE    BLUEBOOK         TIF
##           0           0           0           0           0           0
##    CAR_TYPE     RED_CAR    OLDCLAIM    CLM_FREQ     REVOKED     MVR_PTS
##           0           0           0           0           0           0
##     CAR_AGE  URBANICITY
##         510           0
```

```
colSums(is.na(insurance_eva))
```

```
##       INDEX TARGET_FLAG  TARGET_AMT    KIDSDRIV         AGE    HOMEKIDS
##           0        2141        2141           0           1           0
##         YOJ      INCOME     PARENT1    HOME_VAL     MSTATUS         SEX
##          94         125           0         111           0           0
##   EDUCATION         JOB    TRAVTIME     CAR_USE    BLUEBOOK         TIF
##           0           0           0           0           0           0
##    CAR_TYPE     RED_CAR    OLDCLAIM    CLM_FREQ     REVOKED     MVR_PTS
##           0           0           0           0           0           0
##     CAR_AGE  URBANICITY
##         129           0
```

```
# Percentage of missing values
missing_values <- metastats %>%
  filter(pct_missing < 1) %>%
  dplyr::select(STATS, pct_missing) %>%
  arrange(pct_missing)
missing_values
```

```
##      STATS pct_missing
## 1  CAR_AGE       0.938
## 2 HOME_VAL       0.943
## 3      YOJ       0.944
## 4   INCOME       0.945
## 5      AGE       0.999
```

We have some missing values. We will fix them at the Data Preperation section.

As part of data exploration, we would like to find out dsitribution of categorical, descrete and continous variables. We will also see the outliers and analyze the skewness of the variables. We will further look at the correlation between variables to see if there are multicollinearity among the independent variables.

Let's start looking at the distribution of each descriptive, categorical and continous variables individually.

```
# Distribution for KIDSDRIV
s1 <- ggplot(insurance_train, aes(KIDSDRIV))+
  geom_bar(aes(fill=TARGET_FLAG), width = 0.5) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6))

# Distribution for HOMEKIDS
s2 <- ggplot(insurance_train, aes(HOMEKIDS))+
  geom_bar(aes(fill=TARGET_FLAG), width = 0.5) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6))
```

```
# Distribution for PARENT1
s3 <- ggplot(insurance_train, aes(PARENT1))+
  geom_bar(aes(fill=TARGET_FLAG), width = 0.5) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6))

# Distribution for MSTATUS
s4 <- ggplot(insurance_train, aes(MSTATUS))+
  geom_bar(aes(fill=TARGET_FLAG), width = 0.5) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6))

# Distribution for SEX
s5 <- ggplot(insurance_train, aes(SEX))+
  geom_bar(aes(fill=TARGET_FLAG), width = 0.5) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6))

# Distribution for EDUCATION
s6 <- ggplot(insurance_train, aes(EDUCATION))+
  geom_bar(aes(fill=TARGET_FLAG), width = 0.5) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6))

# Distribution for JOB
s7 <- ggplot(insurance_train, aes(JOB))+
  geom_bar(aes(fill=TARGET_FLAG), width = 0.5) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6))

# Distribution for CAR_USE
s8 <- ggplot(insurance_train, aes(CAR_USE))+
  geom_bar(aes(fill=TARGET_FLAG), width = 0.5) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6))

# Distribution for CAR_TYPE
s9 <- ggplot(insurance_train, aes(CAR_TYPE))+
  geom_bar(aes(fill=TARGET_FLAG), width = 0.5) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6))

# Distribution for RED_CAR
s10 <- ggplot(insurance_train, aes(RED_CAR))+
  geom_bar(aes(fill=TARGET_FLAG), width = 0.5) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6))

# Distribution for REVOKED
s11 <- ggplot(insurance_train, aes(REVOKED))+
  geom_bar(aes(fill=TARGET_FLAG), width = 0.5) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6))

# Distribution for URBAN CITY
s12 <- ggplot(insurance_train, aes(URBANICITY))+
  geom_bar(aes(fill=TARGET_FLAG), width = 0.5) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6))

# Distribution for CLM_FREQ
s13 <- ggplot(insurance_train, aes(CLM_FREQ))+
  geom_bar(aes(fill=TARGET_FLAG), width = 0.5) +
```

```r
  theme(axis.text.x = element_text(angle=65, vjust=0.6))


grid.arrange(s1, s2, s3, s4, s5, s6, s7, s8, s9, s10, s11, s12, s13, nrow=7)
```

When we look at the distribution of Kids Driving, we see that most of them are not in a car crash. Distribution of Kids being home, we see that most of them are not in a car crash. Single Parent distribution, we see most of the non single parent families are not in a car crash. Distribution of Marriage Status displaying is that, most married families are not in a car crash.

Additionaly looking at distribution of the categorical variables, we can see that KIDSDRIV and PARENT1 shows us that if we dont have any kids, it is more likely for us to have a car crash. Being male or female doesnt really matter in terms of car crashes. We also see that high school students, blue collor employees, SUV owners, people that had their license revoked get into more car crash.

```
#Distribution AGE
a1 <- ggplot(insurance_train, aes(AGE)) + scale_fill_brewer(palette = "Spectral")+
  geom_histogram(aes(fill=TARGET_FLAG),
                 bins=5,
                 col="black")
#Distribution YOJ
a2 <- ggplot(insurance_train, aes(YOJ)) + scale_fill_brewer(palette = "Spectral")+
  geom_histogram(aes(fill=TARGET_FLAG),
                 bins=5,
                 col="black")

#Distribution TRAVTIME
a3 <- ggplot(insurance_train, aes(TRAVTIME)) + scale_fill_brewer(palette = "Spectral")+
  geom_histogram(aes(fill=TARGET_FLAG),
                 bins=5,
                 col="black")

#Distribution TIF
a4 <- ggplot(insurance_train, aes(TIF)) + scale_fill_brewer(palette = "Spectral")+
  geom_histogram(aes(fill=TARGET_FLAG),
                 bins=5,
                 col="black")

#Distribution CAR_AGE
a5 <- ggplot(insurance_train, aes(CAR_AGE)) + scale_fill_brewer(palette = "Spectral")+
  geom_histogram(aes(fill=TARGET_FLAG),
                 bins=5,
                 col="black")

#Distribution INCOME
a6 <- ggplot(insurance_train, aes(INCOME)) + scale_fill_brewer(palette = "Spectral")+
  geom_histogram(aes(fill=TARGET_FLAG),
                 bins=5,
                 col="black")

#Distribution BLUEBOOK
a7 <- ggplot(insurance_train, aes(BLUEBOOK)) + scale_fill_brewer(palette = "Spectral")+
  geom_histogram(aes(fill=TARGET_FLAG),
                 bins=5,
                 col="black")

#Distribution OLDCLAIM
a8 <- ggplot(insurance_train, aes(OLDCLAIM)) + scale_fill_brewer(palette = "Spectral")+
  geom_histogram(aes(fill=TARGET_FLAG),
                 bins=5,
                 col="black")
```

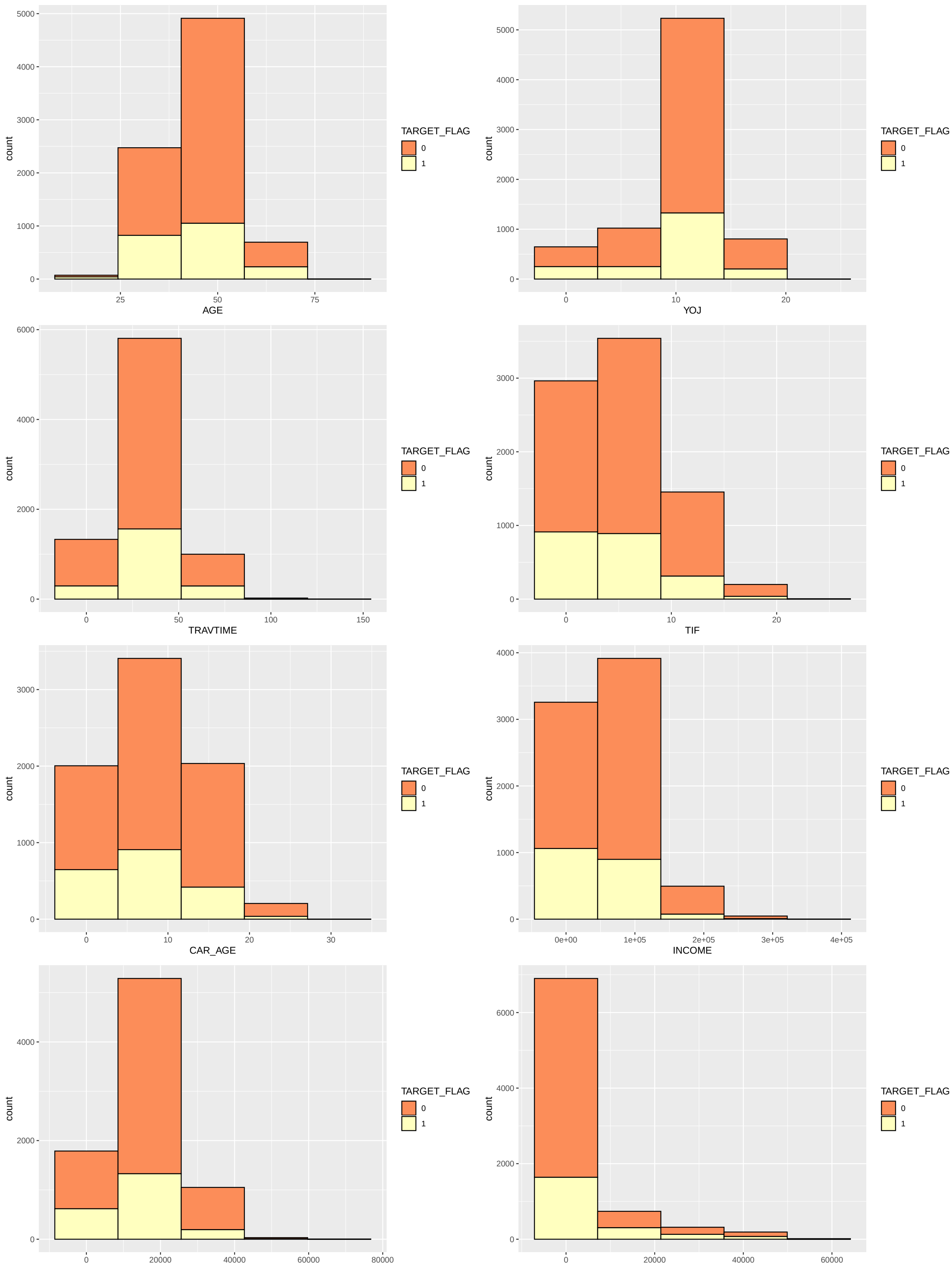
```r
grid.arrange(a1, a2, a3, a4, a5, a6, a7, a8, nrow=4)
```

```
## Warning: Removed 6 rows containing non-finite values (stat_bin).

## Warning: Removed 454 rows containing non-finite values (stat_bin).

## Warning: Removed 510 rows containing non-finite values (stat_bin).

## Warning: Removed 445 rows containing non-finite values (stat_bin).
```

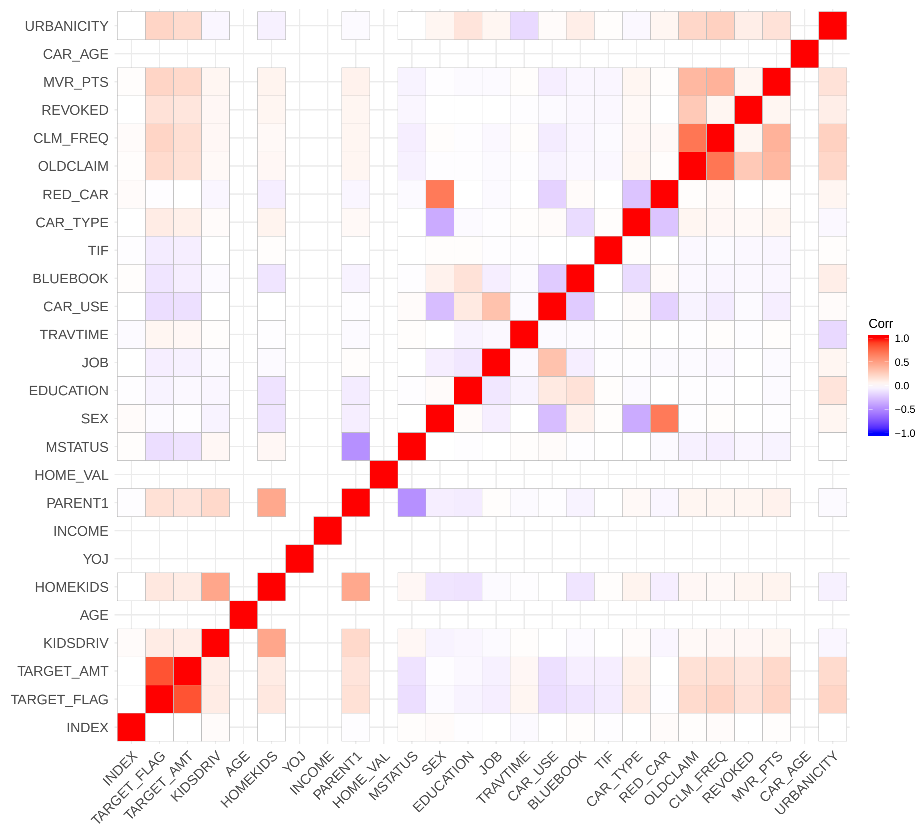


We can see the distribution and skeweness from above plots. In terms of distribution, we see only AGE and YOJ is normally distributed and the rest of the variables had some sort of skeweness. When creating our models, with some of them, we will transform the data, handle the skeweness in order to create a more accurate model.

Let's look at the correlation.

```
insurance_train_num <- data.frame(lapply(insurance_train, function(x) as.numeric(as.factor(x))))

corr <- cor(insurance_train_num)
options(repr.plot.width = 14, repr.plot.height = 8)
ggcorrplot(corr)
```



Based on the correlation matrix, we see that MVR_PTS, CLM_FREQ and OLDCLAIM have the most correlation with the response variables. There are little to no multicollinearity among the independent variables.

Let's further look to see if there are any outliers.

```
b1 <- ggplot(insurance_train, aes(TARGET_FLAG, AGE))+
  geom_boxplot(varwidth=T, fill="plum")

b2 <- ggplot(insurance_train, aes(TARGET_FLAG, BLUEBOOK))+
  geom_boxplot(varwidth=T, fill="plum")

b3 <- ggplot(insurance_train, aes(TARGET_FLAG, CAR_AGE))+
  geom_boxplot(varwidth=T, fill="plum")
```

```r
b4 <- ggplot(insurance_train, aes(TARGET_FLAG, HOME_VAL))+
  geom_boxplot(varwidth=T, fill="plum")

b5 <- ggplot(insurance_train, aes(TARGET_FLAG, INCOME))+
  geom_boxplot(varwidth=T, fill="plum")

b6 <- ggplot(insurance_train, aes(TARGET_FLAG, MVR_PTS))+
  geom_boxplot(varwidth=T, fill="plum")

b7 <- ggplot(insurance_train, aes(TARGET_FLAG, OLDCLAIM))+
  geom_boxplot(varwidth=T, fill="plum")

b8 <- ggplot(insurance_train, aes(TARGET_FLAG, TIF))+
  geom_boxplot(varwidth=T, fill="plum")

b9 <- ggplot(insurance_train, aes(TARGET_FLAG, TRAVTIME))+
  geom_boxplot(varwidth=T, fill="plum")

b10 <- ggplot(insurance_train, aes(TARGET_FLAG, YOJ))+
  geom_boxplot(varwidth=T, fill="plum")

grid.arrange(b1, b2, b3, b4, b5, b6, b7, b8, b9, b10, nrow=5)
```

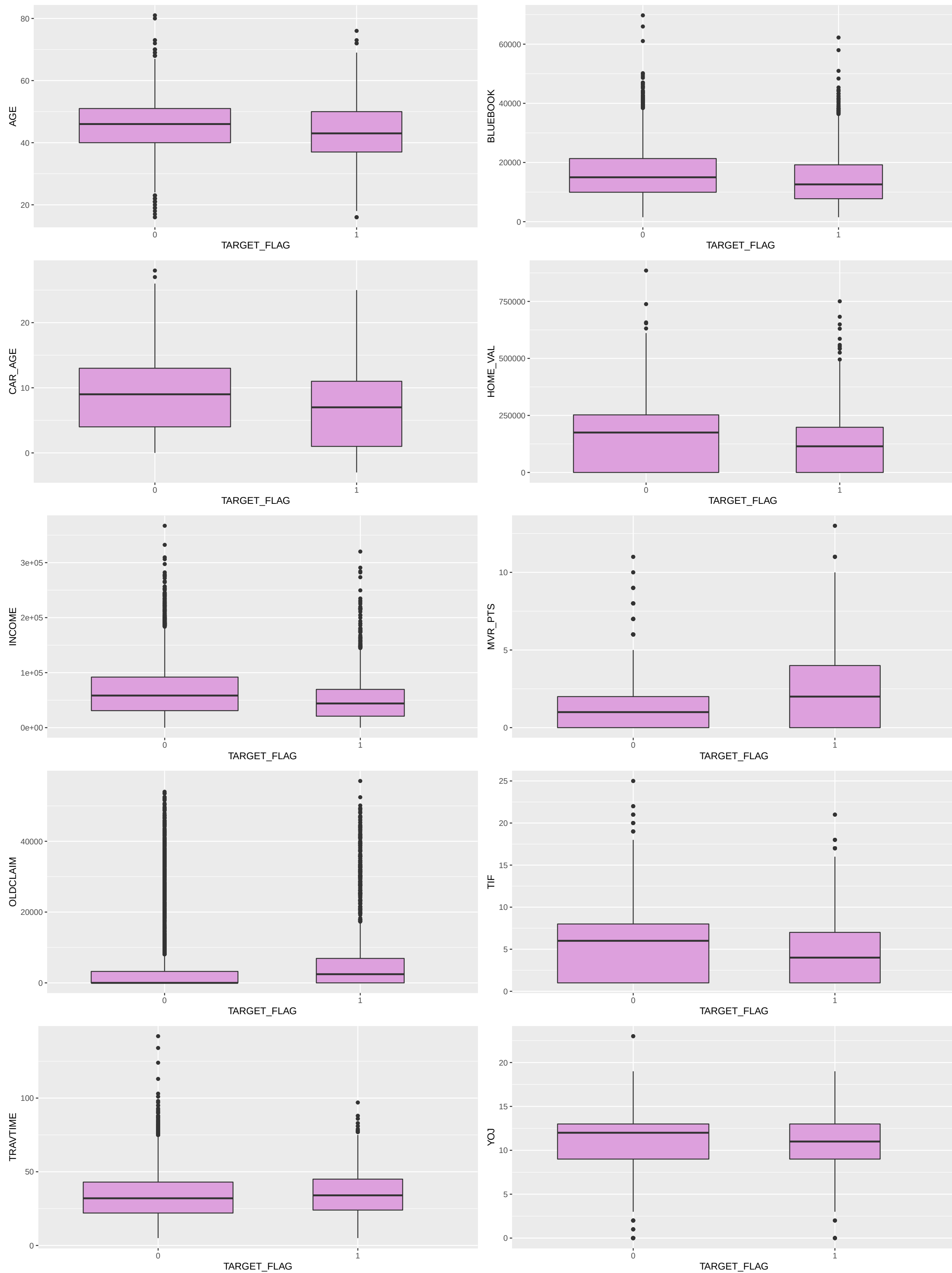
```
## Warning: Removed 6 rows containing non-finite values (stat_boxplot).

## Warning: Removed 510 rows containing non-finite values (stat_boxplot).

## Warning: Removed 464 rows containing non-finite values (stat_boxplot).

## Warning: Removed 445 rows containing non-finite values (stat_boxplot).

## Warning: Removed 454 rows containing non-finite values (stat_boxplot).
```



Based on the above; we can see that BLUEBOOK, INCOME, OLDCLAIM have high number of outliers.

Data Preperation

In the data preperation phase, we will mostly handle the missing values in both training and evaluation data set. We will handle the missing values by using mice package. Here are some references for this package; (https://datascienceplus.com/imputing-missing-data-with-r-mice-package/ , https://www.analyticsvidhya.com/blog/2016/03/tutorial-powerful-packages-imputing-missing-values/, https://cran.r-project.org/web/packages/mice/mice.pdf)

Based on my search It is commonly used package for creating multiple imputations, instead of one single one such as replacing nan with mean. We will apply mice package imputation for both testing and evaluation data sets.

```
# multiple imputations to train data
init <- mice(insurance_train)
```

```
##
##  iter imp variable
##   1   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
```

```
meth <- init$method
predM <- init$predictorMatrix
predM[, c("TARGET_FLAG", "TARGET_AMT")] <- 0
insurance_train_clean <- mice(insurance_train, method = 'rf', predictorMatrix=predM)
```

```
##
##  iter imp variable
##   1   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
```

```
##   1   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
```

```
insurance_train_cleaned <- complete(insurance_train_clean)
print(paste0("Missing value: ", sum(is.na(insurance_train_cleaned))))
```

```
## [1] "Missing value: 0"
```

We should also apply the same to the evluation data set as well.

```
# multiple imputations to test data
insurance_eva$AGE <- ifelse(is.na(insurance_eva$AGE), mean(insurance_eva$AGE),insurance_eva$AGE)
init <- mice(insurance_eva)
```

```
##
##  iter imp variable
##   1   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
```

```
##   4   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE

## Warning: Number of logged events: 2
```

```r
meth <- init$method
predM <- init$predictorMatrix
insurance_eva_clean <- mice(insurance_eva, method = 'rf', predictorMatrix=predM)
```

```
##
##  iter imp variable
##   1   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   1   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   2   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   3   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   4   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   1  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   2  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   3  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   4  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE
##   5   5  AGE  YOJ  INCOME  HOME_VAL  CAR_AGE

## Warning: Number of logged events: 2
```

```r
insurance_eva_cleaned <- complete(insurance_eva_clean)
insurance_eva_cleaned <- data.frame(lapply(insurance_eva_cleaned, function(x) as.numeric(as.factor(x))))
print(paste0("Missing value: ", sum(is.na(insurance_eva_cleaned))))
```

```
## [1] "Missing value: 4282"
```

Before we start building our models, we have to create train and test data sets for both logistic and multiple linear regression. We will split the insurance_train_cleaned data set 80/20 into training and testing datasets.

```
#split data into test and train for both models.
set.seed(101)
train_logistic <- createDataPartition(y = insurance_train_cleaned$TARGET_FLAG, p = 0.80, list = FALSE)
train_multiple <- createDataPartition(y = insurance_train_cleaned$TARGET_AMT, p = 0.80, list = FALSE) #
insurance_train_logistic <- insurance_train_cleaned[train_logistic,]
insurance_test_logistic  <- insurance_train_cleaned[-train_logistic,]
insurance_train_multiple <- insurance_train_cleaned[train_multiple,]
insurance_test_multiple <- insurance_train_cleaned[-train_multiple,]
```

Let's look at how we broke out the test and train datasets.

```
str(insurance_train_logistic)
```

```
## 'data.frame':    6530 obs. of  26 variables:
##  $ INDEX      : int  1 2 4 5 6 8 11 12 14 15 ...
##  $ TARGET_FLAG: Factor w/ 2 levels "0","1": 1 1 1 1 1 1 2 2 2 1 ...
##  $ TARGET_AMT : num  0 0 0 0 0 ...
##  $ KIDSDRIV   : int  0 0 0 0 0 0 1 0 0 0 ...
##  $ AGE        : int  60 43 35 51 50 54 37 34 53 43 ...
##  $ HOMEKIDS   : int  0 0 1 0 0 0 2 0 0 0 ...
##  $ YOJ        : int  11 11 10 14 10 11 9 10 14 5 ...
##  $ INCOME     : num  67349 91449 16039 94160 114986 ...
##  $ PARENT1    : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ HOME_VAL   : num  0 257252 124191 306251 243925 ...
##  $ MSTATUS    : Factor w/ 2 levels "No","Yes": 1 1 2 2 2 2 2 1 1 2 ...
##  $ SEX        : Factor w/ 2 levels "F","M": 2 2 1 2 1 1 2 1 1 1 ...
##  $ EDUCATION  : Factor w/ 4 levels "Bachelors","High School",..: 4 2 2 2 4 2 1 1 3 3 ...
##  $ JOB        : Factor w/ 9 levels "","Blue Collar",..: 8 2 3 2 4 2 2 3 6 8 ...
##  $ TRAVTIME   : int  14 22 5 32 36 33 44 34 15 36 ...
##  $ CAR_USE    : Factor w/ 2 levels "Commercial","Private": 2 1 2 2 2 2 1 2 2 2 ...
##  $ BLUEBOOK   : num  14230 14940 4010 15440 18000 ...
##  $ TIF        : int  11 1 4 7 1 1 1 1 1 7 ...
##  $ CAR_TYPE   : Factor w/ 6 levels "Minivan","Panel Truck",..: 1 1 5 1 5 5 6 5 4 1 ...
##  $ RED_CAR    : Factor w/ 2 levels "no","yes": 2 2 1 2 1 1 2 1 1 1 ...
##  $ OLDCLAIM   : num  4461 0 38690 0 19217 ...
##  $ CLM_FREQ   : int  2 0 2 0 2 0 1 0 0 0 ...
##  $ REVOKED    : Factor w/ 2 levels "No","Yes": 1 1 1 1 2 1 2 1 1 1 ...
##  $ MVR_PTS    : int  3 0 3 0 3 0 10 0 0 0 ...
##  $ CAR_AGE    : int  18 1 10 6 17 1 7 1 11 1 ...
##  $ URBANICITY : Factor w/ 2 levels "Highly Rural/ Rural",..: 2 2 2 2 2 2 2 2 2 1 ...
```

Training model now has 6530 observations and test data set has 1631 observations.

Build Models

In our first model, we will create Multiple Linear Regression Model and use the TARGET_AMT as the response variable and use all the explanatory variables. In this model, we will use the imputed data training data set.

```r
# create model 1 multiple regression
insurance_numeric <- data.frame(lapply(insurance_train_multiple, function(x) as.numeric(as.factor(x))))
insurance_numeric <- dplyr::select(insurance_numeric, -"TARGET_FLAG") #change data types to numeric
model_1 <- lm(TARGET_AMT ~ ., insurance_numeric)
summary(model_1)
```

```
##
## Call:
## lm(formula = TARGET_AMT ~ ., data = insurance_numeric)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -733.73 -234.96 -110.05   55.03 1492.68
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.184e+02  6.243e+01  -1.897 0.057860 .
## INDEX       -1.531e-03  2.535e-03  -0.604 0.546009
## KIDSDRIV     3.727e+01  1.068e+01   3.490 0.000486 ***
## AGE          1.069e-01  6.558e-01   0.163 0.870550
## HOMEKIDS     8.455e+00  6.161e+00   1.372 0.170023
## YOJ         -1.973e+00  1.314e+00  -1.502 0.133272
## INCOME      -2.154e-02  4.760e-03  -4.526 6.13e-06 ***
## PARENT1      6.253e+01  1.922e+01   3.253 0.001149 **
## HOME_VAL    -1.284e-02  5.431e-03  -2.363 0.018143 *
## MSTATUS     -6.580e+01  1.375e+01  -4.784 1.75e-06 ***
## SEX         -9.240e+00  1.385e+01  -0.667 0.504708
## EDUCATION    6.925e+00  5.945e+00   1.165 0.244154
## JOB         -5.716e+00  2.014e+00  -2.838 0.004560 **
## TRAVTIME     1.591e+00  3.047e-01   5.223 1.82e-07 ***
## CAR_USE     -1.182e+02  1.143e+01 -10.341  < 2e-16 ***
## BLUEBOOK    -2.290e-02  8.010e-03  -2.859 0.004258 **
## TIF         -6.325e+00  1.154e+00  -5.481 4.40e-08 ***
## CAR_TYPE     1.272e+01  2.981e+00   4.266 2.02e-05 ***
## RED_CAR     -8.890e+00  1.413e+01  -0.629 0.529308
## OLDCLAIM    -1.709e-02  1.020e-02  -1.675 0.094041 .
## CLM_FREQ     2.405e+01  5.943e+00   4.047 5.26e-05 ***
## REVOKED      1.256e+02  1.554e+01   8.087 7.25e-16 ***
## MVR_PTS      2.214e+01  2.489e+00   8.894  < 2e-16 ***
## CAR_AGE     -3.138e+00  9.845e-01  -3.187 0.001443 **
## URBANICITY   2.201e+02  1.309e+01  16.820  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 385.2 on 6504 degrees of freedom
## Multiple R-squared:  0.1601, Adjusted R-squared:  0.157
## F-statistic: 51.66 on 24 and 6504 DF,  p-value: < 2.2e-16
```

We have a low p value and our Adjusted R-squared is 0.15. We can only explain 15% of the data with this model. This is definately not a good model.

In our second model we will use the same model, let's try to use the training data set without the transformation (non imputed data), use TARGET_AMT as the response variable and use all the explanatory variables.

```r
# create model 2 multiple regression
insurance_numeric_2 <- data.frame(lapply(insurance_train, function(x) as.numeric(as.factor(x))))
insurance_numeric_2 <- dplyr::select(insurance_numeric_2, -"TARGET_FLAG") #remove TARGET_FLAG
model_2 <- lm(TARGET_AMT ~ ., insurance_numeric_2)
summary(model_2)
```

```
##
## Call:
## lm(formula = TARGET_AMT ~ ., data = insurance_numeric_2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -918.61 -286.96 -133.91   60.91 1944.56
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.817e+02  7.603e+01  -2.390 0.016888 *
## INDEX        1.071e-03  2.488e-03   0.430 0.666887
## KIDSDRIV     4.190e+01  1.318e+01   3.179 0.001483 **
## AGE         -1.157e-01  8.110e-01  -0.143 0.886564
## HOMEKIDS     1.367e+01  7.531e+00   1.815 0.069582 .
## YOJ         -1.079e+00  1.624e+00  -0.665 0.506354
## INCOME      -2.361e-02  4.807e-03  -4.911 9.26e-07 ***
## PARENT1      7.657e+01  2.335e+01   3.279 0.001047 **
## HOME_VAL    -1.078e-02  5.504e-03  -1.958 0.050270 .
## MSTATUS     -7.145e+01  1.684e+01  -4.242 2.25e-05 ***
## SEX         -6.425e-01  1.691e+01  -0.038 0.969693
## EDUCATION    1.094e+01  7.324e+00   1.493 0.135464
## JOB         -6.443e+00  2.485e+00  -2.593 0.009549 **
## TRAVTIME     2.170e+00  3.771e-01   5.754 9.13e-09 ***
## CAR_USE     -1.391e+02  1.398e+01  -9.950  < 2e-16 ***
## BLUEBOOK    -2.931e-02  9.303e-03  -3.151 0.001635 **
## TIF         -7.319e+00  1.415e+00  -5.171 2.39e-07 ***
## CAR_TYPE     1.927e+01  3.653e+00   5.276 1.37e-07 ***
## RED_CAR     -1.727e+01  1.731e+01  -0.998 0.318494
## OLDCLAIM    -5.038e-03  1.039e-02  -0.485 0.627666
## CLM_FREQ     2.329e+01  7.354e+00   3.166 0.001551 **
## REVOKED      1.304e+02  1.913e+01   6.817 1.01e-11 ***
## MVR_PTS      2.592e+01  3.033e+00   8.547  < 2e-16 ***
## CAR_AGE     -4.053e+00  1.218e+00  -3.326 0.000885 ***
## URBANICITY   2.689e+02  1.593e+01  16.876  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 469.8 on 6423 degrees of freedom
##   (1713 observations deleted due to missingness)
## Multiple R-squared:  0.1575, Adjusted R-squared:  0.1544
## F-statistic: 50.04 on 24 and 6423 DF,  p-value: < 2.2e-16
```

Again our R-squared is really low 0.15 and we can only explain 15% of the data with this model.

In our third model, we will create a model, using logistic regression, use TARGET_FLAG as the response variable and use all the explanatory varibles.

```r
# create model 3 binary logistic regression
logit_data <- data.frame(lapply(insurance_train_logistic, function(x) as.numeric(as.factor(x)))) %>%
  mutate(TARGET_FLAG = as.factor(TARGET_FLAG)) %>%
  dplyr::select(-"TARGET_AMT")

model_3 <- glm(TARGET_FLAG ~ ., family = "binomial", logit_data)
summary(model_3)
```

```
##
## Call:
## glm(formula = TARGET_FLAG ~ ., family = "binomial", data = logit_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5081  -0.7283  -0.4068   0.6428   3.0313
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -4.848e+00  4.595e-01 -10.550  < 2e-16 ***
## INDEX       -8.701e-06  1.714e-05  -0.508 0.611604
## KIDSDRIV     3.406e-01  6.725e-02   5.064 4.11e-07 ***
## AGE         -2.671e-03  4.375e-03  -0.611 0.541483
## HOMEKIDS     4.260e-02  4.124e-02   1.033 0.301640
## YOJ         -9.020e-03  8.628e-03  -1.046 0.295788
## INCOME      -1.829e-04  3.119e-05  -5.865 4.49e-09 ***
## PARENT1      4.075e-01  1.228e-01   3.318 0.000908 ***
## HOME_VAL    -1.148e-04  3.655e-05  -3.142 0.001681 **
## MSTATUS     -4.643e-01  9.334e-02  -4.974 6.56e-07 ***
## SEX         -6.903e-02  9.404e-02  -0.734 0.462933
## EDUCATION    4.360e-02  4.139e-02   1.053 0.292233
## JOB         -5.772e-02  1.332e-02  -4.334 1.46e-05 ***
## TRAVTIME     1.558e-02  2.113e-03   7.374 1.66e-13 ***
## CAR_USE     -8.548e-01  7.579e-02 -11.279  < 2e-16 ***
## BLUEBOOK    -3.622e-04  5.456e-05  -6.638 3.17e-11 ***
## TIF         -4.619e-02  8.066e-03  -5.726 1.03e-08 ***
## CAR_TYPE     1.358e-01  2.053e-02   6.612 3.80e-11 ***
## RED_CAR     -7.628e-02  9.631e-02  -0.792 0.428359
## OLDCLAIM    -2.842e-05  6.146e-05  -0.463 0.643707
## CLM_FREQ     1.712e-01  3.598e-02   4.757 1.96e-06 ***
## REVOKED      7.766e-01  9.491e-02   8.182 2.79e-16 ***
## MVR_PTS      1.171e-01  1.527e-02   7.673 1.68e-14 ***
## CAR_AGE     -1.711e-02  6.650e-03  -2.573 0.010085 *
## URBANICITY   2.373e+00  1.291e-01  18.382  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 7536.3  on 6529  degrees of freedom
## Residual deviance: 5897.8  on 6505  degrees of freedom
## AIC: 5947.8
##
## Number of Fisher Scoring iterations: 5
```

All predictors are significant (we can of course ignore index) except KIDSDRIV, TRAVTIME, CLM_FREQ. We will further look at the accuracy, roc and auc at our model selection section.

In our 4th model, we will create a logistic regression model, using TARGET_FLAG as the response variable and all the explanatory variables on non imputed data.

```r
# model 4 binary logistic model
logit_data_2 <- data.frame(lapply(insurance_train, function(x) as.numeric(as.factor(x)))) %>%
  mutate(TARGET_FLAG = as.factor(TARGET_FLAG)) %>%
  dplyr::select(-"TARGET_AMT")

model_4 <- glm(TARGET_FLAG ~ ., family = "binomial", logit_data_2)
summary(model_4)
```

```
##
## Call:
## glm(formula = TARGET_FLAG ~ ., family = "binomial", data = logit_data_2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5330  -0.7200  -0.4184   0.6445   3.1596
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -4.689e+00  4.478e-01 -10.472  < 2e-16 ***
## INDEX        1.001e-05  1.381e-05   0.725 0.468314
## KIDSDRIV     3.275e-01  6.814e-02   4.807 1.54e-06 ***
## AGE         -3.867e-03  4.410e-03  -0.877 0.380551
## HOMEKIDS     4.158e-02  4.097e-02   1.015 0.310168
## YOJ         -6.396e-03  8.787e-03  -0.728 0.466685
## INCOME      -1.399e-04  2.571e-05  -5.441 5.30e-08 ***
## PARENT1      4.405e-01  1.212e-01   3.635 0.000278 ***
## HOME_VAL    -9.974e-05  2.991e-05  -3.335 0.000853 ***
## MSTATUS     -4.372e-01  9.329e-02  -4.686 2.78e-06 ***
## SEX         -6.416e-03  9.385e-02  -0.068 0.945496
## EDUCATION    4.147e-02  4.161e-02   0.997 0.318960
## JOB         -4.760e-02  1.344e-02  -3.542 0.000397 ***
## TRAVTIME     1.618e-02  2.114e-03   7.655 1.93e-14 ***
## CAR_USE     -8.714e-01  7.562e-02 -11.523  < 2e-16 ***
## BLUEBOOK    -3.080e-04  5.176e-05  -5.951 2.66e-09 ***
## TIF         -5.214e-02  8.170e-03  -6.382 1.75e-10 ***
## CAR_TYPE     1.385e-01  2.067e-02   6.704 2.03e-11 ***
## RED_CAR     -1.321e-01  9.603e-02  -1.375 0.169065
## OLDCLAIM    -4.778e-05  5.120e-05  -0.933 0.350668
## CLM_FREQ     1.803e-01  3.592e-02   5.019 5.19e-07 ***
## REVOKED      7.664e-01  9.577e-02   8.003 1.21e-15 ***
## MVR_PTS      1.174e-01  1.529e-02   7.676 1.64e-14 ***
## CAR_AGE     -2.431e-02  6.758e-03  -3.597 0.000322 ***
## URBANICITY   2.223e+00  1.231e-01  18.054  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 7445.1  on 6447  degrees of freedom
## Residual deviance: 5857.2  on 6423  degrees of freedom
##   (1713 observations deleted due to missingness)
## AIC: 5907.2
##
## Number of Fisher Scoring iterations: 5
```

In model 4 , PARENT1, HOME_VAL JOB and URBANCITY predictors are significant in predicting TARGET_FLAG.

In our 5th model, we will create a stepwise transformed logistic regression model, leveraging Model which uses TARGET_FLAG as the response variable, and all the explanatory variables on cleaned trained and transformed data.

```r
#build model 5 binary logistic model
model_5 <- stepAIC(model_3, direction = "both", trace = FALSE)
summary(model_5)
```

```
##
## Call:
## glm(formula = TARGET_FLAG ~ KIDSDRIV + INCOME + PARENT1 + HOME_VAL +
##     MSTATUS + JOB + TRAVTIME + CAR_USE + BLUEBOOK + TIF + CAR_TYPE +
##     RED_CAR + CLM_FREQ + REVOKED + MVR_PTS + CAR_AGE + URBANICITY,
##     family = "binomial", data = logit_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5370  -0.7271  -0.4065   0.6459   3.0423
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.101e+00  4.099e-01 -12.445  < 2e-16 ***
## KIDSDRIV     3.690e-01  6.026e-02   6.124 9.14e-10 ***
## INCOME      -1.892e-04  3.004e-05  -6.297 3.03e-10 ***
## PARENT1      4.916e-01  1.050e-01   4.681 2.86e-06 ***
## HOME_VAL    -1.239e-04  3.629e-05  -3.413 0.000643 ***
## MSTATUS     -4.360e-01  8.844e-02  -4.931 8.20e-07 ***
## JOB         -5.794e-02  1.286e-02  -4.507 6.58e-06 ***
## TRAVTIME     1.545e-02  2.108e-03   7.331 2.28e-13 ***
## CAR_USE     -8.393e-01  7.248e-02 -11.580  < 2e-16 ***
## BLUEBOOK    -3.643e-04  5.420e-05  -6.721 1.80e-11 ***
## TIF         -4.592e-02  8.053e-03  -5.703 1.18e-08 ***
## CAR_TYPE     1.403e-01  1.975e-02   7.105 1.21e-12 ***
## RED_CAR     -1.222e-01  7.551e-02  -1.618 0.105664
## CLM_FREQ     1.594e-01  2.812e-02   5.668 1.44e-08 ***
## REVOKED      7.661e-01  8.905e-02   8.603  < 2e-16 ***
## MVR_PTS      1.175e-01  1.510e-02   7.777 7.40e-15 ***
## CAR_AGE     -1.529e-02  6.323e-03  -2.418 0.015599 *
## URBANICITY   2.366e+00  1.287e-01  18.379  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 7536.3  on 6529  degrees of freedom
## Residual deviance: 5903.3  on 6512  degrees of freedom
## AIC: 5939.3
##
## Number of Fisher Scoring iterations: 5
```

Addition to the 5 models we created, we can handle skweness of certain variables with boxcox transformation and create updated models. Our 6th model will boxcox transformation and use all variables as explanatory variables and response variable TARGET_FLAG.

```r
# build model 6 binary logistic model
insurance_transformed <- preProcess(logit_data, c("BoxCox"))
insurance_transformed_1 <- predict(insurance_transformed, logit_data)
model_6 <- glm(TARGET_FLAG ~ ., family = "binomial", insurance_transformed_1)
summary(model_6)
```

```
##
## Call:
## glm(formula = TARGET_FLAG ~ ., family = "binomial", data = insurance_transformed_1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3154  -0.7219  -0.4037   0.6719   2.9971
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.829e+00  4.582e-01  -8.358  < 2e-16 ***
## INDEX       -5.818e-05  1.818e-04  -0.320  0.74893
## KIDSDRIV     1.270e+00  2.745e-01   4.625 3.74e-06 ***
## AGE         -5.336e-04  4.541e-03  -0.117  0.90647
## HOMEKIDS     4.471e-01  2.494e-01   1.793  0.07302 .
## YOJ          1.823e-03  2.472e-03   0.737  0.46093
## INCOME      -7.960e-03  1.073e-03  -7.416 1.21e-13 ***
## PARENT1      3.012e-01  1.338e-01   2.251  0.02441 *
## HOME_VAL    -2.054e-02  5.340e-03  -3.847  0.00012 ***
## MSTATUS     -4.614e-01  1.017e-01  -4.539 5.65e-06 ***
## SEX         -4.408e-02  9.458e-02  -0.466  0.64116
## EDUCATION    4.981e-02  6.537e-02   0.762  0.44611
## JOB         -1.530e-01  2.864e-02  -5.342 9.19e-08 ***
## TRAVTIME     4.261e-02  5.610e-03   7.595 3.07e-14 ***
## CAR_USE     -8.022e-01  7.698e-02 -10.420  < 2e-16 ***
## BLUEBOOK    -5.744e-03  8.020e-04  -7.162 7.93e-13 ***
## TIF         -1.566e-01  2.666e-02  -5.874 4.25e-09 ***
## CAR_TYPE     1.994e-01  2.860e-02   6.970 3.17e-12 ***
## RED_CAR     -6.858e-02  9.640e-02  -0.711  0.47681
## OLDCLAIM    -1.882e-02  3.599e-02  -0.523  0.60093
## CLM_FREQ     1.167e+00  4.676e-01   2.496  0.01257 *
## REVOKED      7.656e-01  9.136e-02   8.381  < 2e-16 ***
## MVR_PTS      4.071e-01  6.991e-02   5.824 5.75e-09 ***
## CAR_AGE     -5.952e-02  1.898e-02  -3.136  0.00171 **
## URBANICITY   2.367e+00  1.296e-01  18.258  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 7536.3  on 6529  degrees of freedom
## Residual deviance: 5876.5  on 6505  degrees of freedom
## AIC: 5926.5
##
## Number of Fisher Scoring iterations: 5
```

Since two multiple linear regression model we created have low R-squared values. We will create two more with boxcox transformation of explanatory variables.

```
#build model 7 multiple regression
insurance_transformed_2 <- preProcess(insurance_numeric, c("BoxCox"))
insurance_transformed_3<- predict(insurance_transformed_2, insurance_numeric)
model_7 <- lm(TARGET_AMT ~ ., insurance_transformed_3)
summary(model_7)
```

```
##
## Call:
## lm(formula = TARGET_AMT ~ ., data = insurance_transformed_3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.6700 -0.5493 -0.2185  0.5870  2.2336
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.612e-02  1.196e-01   0.636 0.524476
## INDEX       -2.564e-06  5.187e-05  -0.049 0.960570
## KIDSDRIV     3.427e-01  8.264e-02   4.147 3.40e-05 ***
## AGE         -2.279e-04  1.311e-03  -0.174 0.861998
## HOMEKIDS     9.609e-02  6.981e-02   1.377 0.168703
## YOJ         -4.210e-04  7.134e-04  -0.590 0.555116
## INCOME      -2.024e-03  3.121e-04  -6.484 9.59e-11 ***
## PARENT1      1.349e-01  3.953e-02   3.412 0.000648 ***
## HOME_VAL    -6.199e-03  1.573e-03  -3.940 8.22e-05 ***
## MSTATUS     -1.287e-01  2.840e-02  -4.532 5.94e-06 ***
## SEX         -2.517e-02  2.683e-02  -0.938 0.348101
## EDUCATION    1.324e-03  1.809e-02   0.073 0.941688
## JOB         -3.913e-02  8.328e-03  -4.698 2.68e-06 ***
## TRAVTIME     1.077e-02  1.546e-03   6.965 3.60e-12 ***
## CAR_USE     -2.508e-01  2.237e-02 -11.207  < 2e-16 ***
## BLUEBOOK    -1.420e-03  2.324e-04  -6.110 1.05e-09 ***
## TIF         -5.504e-02  7.604e-03  -7.239 5.06e-13 ***
## CAR_TYPE     4.771e-02  7.920e-03   6.024 1.79e-09 ***
## RED_CAR     -1.174e-02  2.726e-02  -0.430 0.666870
## OLDCLAIM    -1.477e-02  1.151e-02  -1.283 0.199587
## CLM_FREQ     4.454e-01  1.430e-01   3.115 0.001850 **
## REVOKED      2.671e-01  2.889e-02   9.246  < 2e-16 ***
## MVR_PTS      1.417e-01  2.048e-02   6.920 4.94e-12 ***
## CAR_AGE     -2.486e-02  5.481e-03  -4.535 5.86e-06 ***
## URBANICITY   5.254e-01  2.540e-02  20.685  < 2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7433 on 6504 degrees of freedom
## Multiple R-squared:  0.2211, Adjusted R-squared:  0.2183
## F-statistic: 76.94 on 24 and 6504 DF,  p-value: < 2.2e-16
```

We improved the R-squared however, it is still not good for a decent model. We will apply log transformation for the response variable TARGET_AMT, square root transformation for income variable, quarter root transformation for HOME_VAL variable in order to fix the skeweness.

```
# build model 8 multiple regression model
boxcoxfit(insurance_train_multiple$TARGET_AMT[insurance_train_multiple$TARGET_FLAG==1]) # highly #right
```

```
## Fitted parameters:
##    lambda      beta   sigmasq
## 0.0194616 8.9659038 0.8983304
##
## Convergence code returned by optim: 0
```

```
insurance_train_multiple$TARGET_AMT <- log(insurance_train_multiple$TARGET_AMT) # log transformation
boxcoxfit(insurance_train_multiple$INCOME[insurance_train_multiple$INCOME >0])
```

```
## Fitted parameters:
##       lambda         beta      sigmasq
##    0.4328855  264.6842407 7196.3616414
##
## Convergence code returned by optim: 0
```

```
insurance_train_multiple$INCOME <- insurance_train_multiple$INCOME ^0.5 #square root transformation
boxcoxfit(insurance_train_multiple$HOME_VAL[insurance_train_multiple$HOME_VAL > 0])
```

```
## Fitted parameters:
##     lambda       beta    sigmasq
##  0.2068219 55.8487226 29.9715440
##
## Convergence code returned by optim: 0
```

```
insurance_train_multiple$HOME_VAL <- insurance_train_multiple$HOME_VAL^0.25 # quarter root transformati
boxcoxfit(insurance_train_multiple$BLUEBOOK)
```

```
## Fitted parameters:
##       lambda         beta      sigmasq
##    0.4495451  162.4292616 1779.6320893
##
## Convergence code returned by optim: 0
```

```
insurance_train_multiple$BLUEBOOK <- insurance_train_multiple$BLUEBOOK^0.5 # square root transformation
boxcoxfit(insurance_train_multiple$OLDCLAIM[insurance_train_multiple$OLDCLAIM>0])
```

```
## Fitted parameters:
##      lambda        beta     sigmasq
## -0.04682587  7.16712225  0.42539611
##
## Convergence code returned by optim: 0
```

```r
insurance_train_multiple$OLD_CLAIM <- log(insurance_train_multiple$OLDCLAIM + 1)  #log(x+1) transformat

insurance_numeric_3 <- data.frame(lapply(insurance_train_multiple, function(x) as.numeric(as.factor(x)))
```

```r
#build multiple regression model continued
model_8 <- lm(TARGET_AMT ~. , data=insurance_numeric_3)
summary(model_8)
```

```
##
## Call:
## lm(formula = TARGET_AMT ~ ., data = insurance_numeric_3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -811.86  -14.83    0.91   14.67  805.92
##
## Coefficients: (1 not defined because of singularities)
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -8.091e+02  3.812e+01 -21.224  < 2e-16 ***
## INDEX       -1.199e-03  1.526e-03  -0.786  0.43212
## TARGET_FLAG  7.898e+02  7.379e+00 107.040  < 2e-16 ***
## KIDSDRIV    -4.345e+00  6.439e+00  -0.675  0.49977
## AGE          2.771e-01  3.947e-01   0.702  0.48270
## HOMEKIDS     1.770e+00  3.708e+00   0.477  0.63312
## YOJ          1.796e-01  7.910e-01   0.227  0.82043
## INCOME      -1.225e-03  2.871e-03  -0.427  0.66952
## PARENT1      2.679e+00  1.158e+01   0.231  0.81710
## HOME_VAL    -7.985e-04  3.270e-03  -0.244  0.80712
## MSTATUS     -5.074e+00  8.296e+00  -0.612  0.54084
## SEX          3.191e+00  8.335e+00   0.383  0.70187
## EDUCATION    5.562e+00  3.578e+00   1.554  0.12011
## JOB          5.699e-01  1.214e+00   0.470  0.63867
## TRAVTIME     7.421e-03  1.840e-01   0.040  0.96783
## CAR_USE     -8.092e+00  6.952e+00  -1.164  0.24446
## BLUEBOOK     1.333e-02  4.832e-03   2.758  0.00584 **
## TIF          2.095e-01  6.971e-01   0.301  0.76380
## CAR_TYPE    -1.115e+00  1.799e+00  -0.620  0.53551
## RED_CAR     -4.004e+00  8.504e+00  -0.471  0.63776
## OLDCLAIM    -5.659e-04  6.142e-03  -0.092  0.92659
## CLM_FREQ    -3.380e+00  3.585e+00  -0.943  0.34593
## REVOKED      9.595e+00  9.411e+00   1.019  0.30801
## MVR_PTS      4.381e+00  1.507e+00   2.907  0.00366 **
## CAR_AGE     -2.051e-01  5.931e-01  -0.346  0.72951
## URBANICITY  -4.108e-01  8.140e+00  -0.050  0.95976
## OLD_CLAIM           NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 231.8 on 6503 degrees of freedom
## Multiple R-squared:  0.6959, Adjusted R-squared:  0.6947
## F-statistic: 595.2 on 25 and 6503 DF,  p-value: < 2.2e-16
```

With the last multiple linear regression, we were able to improve the first three models we created. With this model, we are able to explain 70% of the variablity in the data.

Select Models

First, let's start with the binary logistic models and compare the fits of them.

```r
model_3_out <- cbind(AIC=AIC(model_3), AICc=AICc(model_3), BIC = BIC(model_3), loglik=logLik(model_3))
model_4_out <- cbind(AIC=AIC(model_4), AICc=AICc(model_4), BIC = BIC(model_4), loglik=logLik(model_4))
model_5_out <- cbind(AIC=AIC(model_5), AICc=AICc(model_5), BIC = BIC(model_5), loglik=logLik(model_5))
model_6_out <- cbind(AIC=AIC(model_6), AICc=AICc(model_6), BIC = BIC(model_6), loglik=logLik(model_6))


model_comp <- rbind(model_3_out, model_4_out, model_5_out, model_6_out)
rownames(model_comp) <- c("model_3","model_4","model_5","model_6")

model_comp
```

```
##              AIC     AICc      BIC    loglik
## model_3 5947.798 5947.998 6117.402 -2948.899
## model_4 5907.187 5907.389 6076.475 -2928.593
## model_5 5939.275 5939.380 6061.390 -2951.638
## model_6 5926.536 5926.736 6096.140 -2938.268
```

Based on these we can look at model 3,5 and 6 which we used imputed train data set.

```r
# convert the insurance test data set to logit data
logit_data_test <- data.frame(lapply(insurance_test_logistic, function(x) as.numeric(as.factor(x)))) %>
  mutate(TARGET_FLAG = as.factor(TARGET_FLAG)) %>%
  dplyr::select(-"TARGET_AMT")
```

```r
# models 3,5,6 prediction probs using test dataset.
m3_pred <- predict(model_3, logit_data_test, type="response")
m5_pred <- predict(model_5, logit_data_test, type="response")
m6_pred <- predict(model_6, logit_data_test, type="response")

#AUC
paste("Model 3:",round(as.numeric(roc(logit_data_test$TARGET_FLAG, m3_pred)["auc"]),3))
```

```
## Setting levels: control = 1, case = 2
```

```
## Setting direction: controls < cases
```

```
## [1] "Model 3: 0.785"
```

```r
paste("Model 5:",round(as.numeric(roc(logit_data_test$TARGET_FLAG, m5_pred)["auc"]),3))
```

```
## Setting levels: control = 1, case = 2
## Setting direction: controls < cases
```

```
## [1] "Model 5: 0.783"
```

```r
paste("Model 6 mod:",round(as.numeric(roc(logit_data_test$TARGET_FLAG, m6_pred)["auc"]),3))
```

```
## Setting levels: control = 1, case = 2
## Setting direction: controls < cases
```

```
## [1] "Model 6 mod: 0.643"
```

Model 3 and 5 has higher accuracy score. Let's build metrics table using predictions with the test data set.

```r
# comparing all binary logistic models using various measures
m3 <- confusionMatrix(as.factor(as.integer(fitted(model_3) > .5)), as.factor(model_3$y), positive = "1"
m5 <- confusionMatrix(as.factor(as.integer(fitted(model_5) > .5)), as.factor(model_5$y), positive = "1"
m6 <- confusionMatrix(as.factor(as.integer(fitted(model_6) > .5)), as.factor(model_6$y), positive = "1"
```

```r
roc3 <- roc(logit_data_test$TARGET_FLAG,  predict(model_3, logit_data_test, interval = "prediction"))
```

```
## Setting levels: control = 1, case = 2
```

```
## Setting direction: controls < cases
```

```r
roc5 <- roc(logit_data_test$TARGET_FLAG,  predict(model_5, logit_data_test, interval = "prediction"))
```

```
## Setting levels: control = 1, case = 2
## Setting direction: controls < cases
```

```r
roc6 <- roc(logit_data_test$TARGET_FLAG,  predict(model_6, logit_data_test, interval = "prediction"))
```

```
## Setting levels: control = 1, case = 2
## Setting direction: controls < cases
```

```r
metrics_3 <- c(m3$overall[1], "Class. Error Rate" = 1 - as.numeric(m3$overall[1]), m3$byClass[c(1, 2, 5
metrics_5 <- c(m5$overall[1], "Class. Error Rate" = 1 - as.numeric(m5$overall[1]), m5$byClass[c(1, 2, 5
metrics_6 <- c(m6$overall[1], "Class. Error Rate" = 1 - as.numeric(m6$overall[1]), m6$byClass[c(1, 2, 5
```

```r
kable(cbind(metrics_3, metrics_5, metrics_6), col.names = c("Model 3", "Model 5", "Model 5"))  %>%
  kable_styling(full_width = T)
```

	Model 3	Model 5	Model 5
Accuracy	0.7866769	0.7871363	0.7849923
Class. Error Rate	0.2133231	0.2128637	0.2150077
Sensitivity	0.4080093	0.4062681	0.4051074
Specificity	0.9224048	0.9236530	0.9211566
Precision	0.6533457	0.6560450	0.6480966
F1	0.5023223	0.5017921	0.4985714
AUC	0.7848189	0.7831439	0.6437097

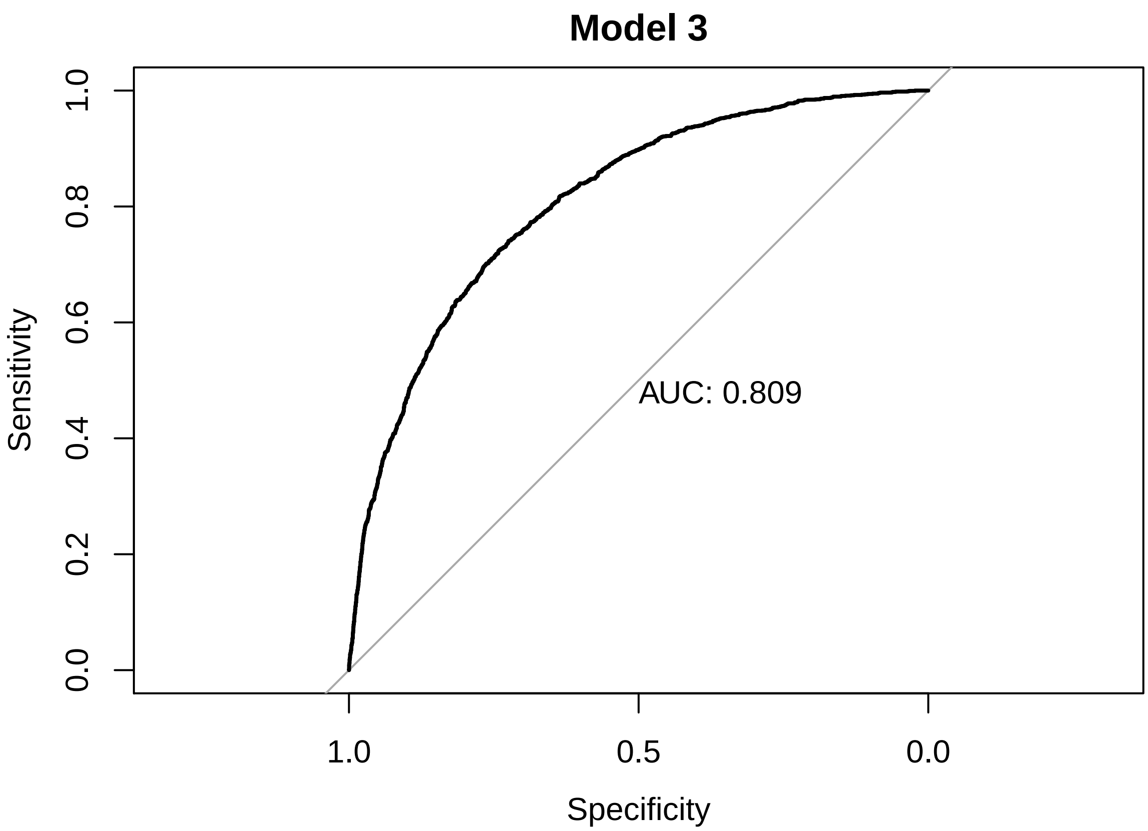
Based on the accuracy and auc which is based on True Positive Rate and False Positive Rate, we can select either Model 3, 4 and or 5. Considering Accuracy is slightly higher on Model 3, we might want to use that for our predictions.

Let's also plot the ROC curve for each binary logistic model.

```
# plotting roc curve of model 3
plot(roc(logit_data$TARGET_FLAG,  predict(model_3, logit_data, interval = "prediction")), print.auc = T
```

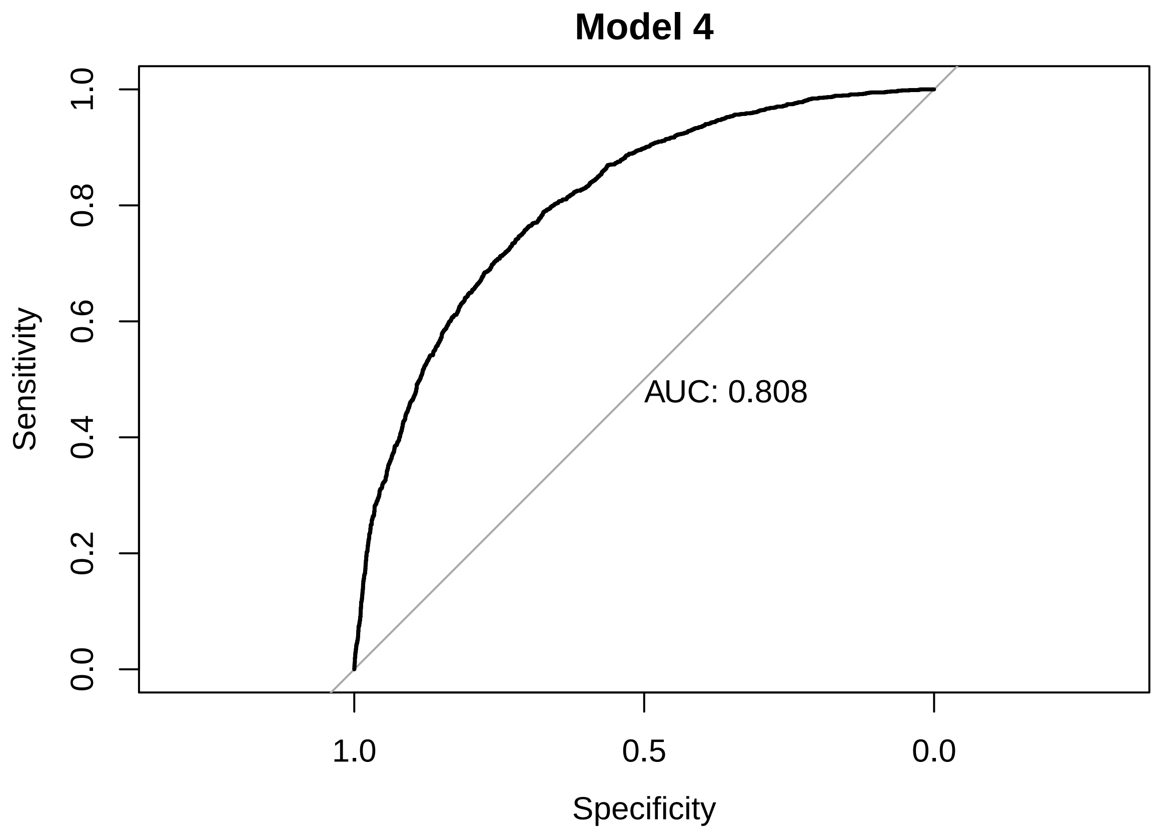
```
## Setting levels: control = 1, case = 2
```

```
## Setting direction: controls < cases
```



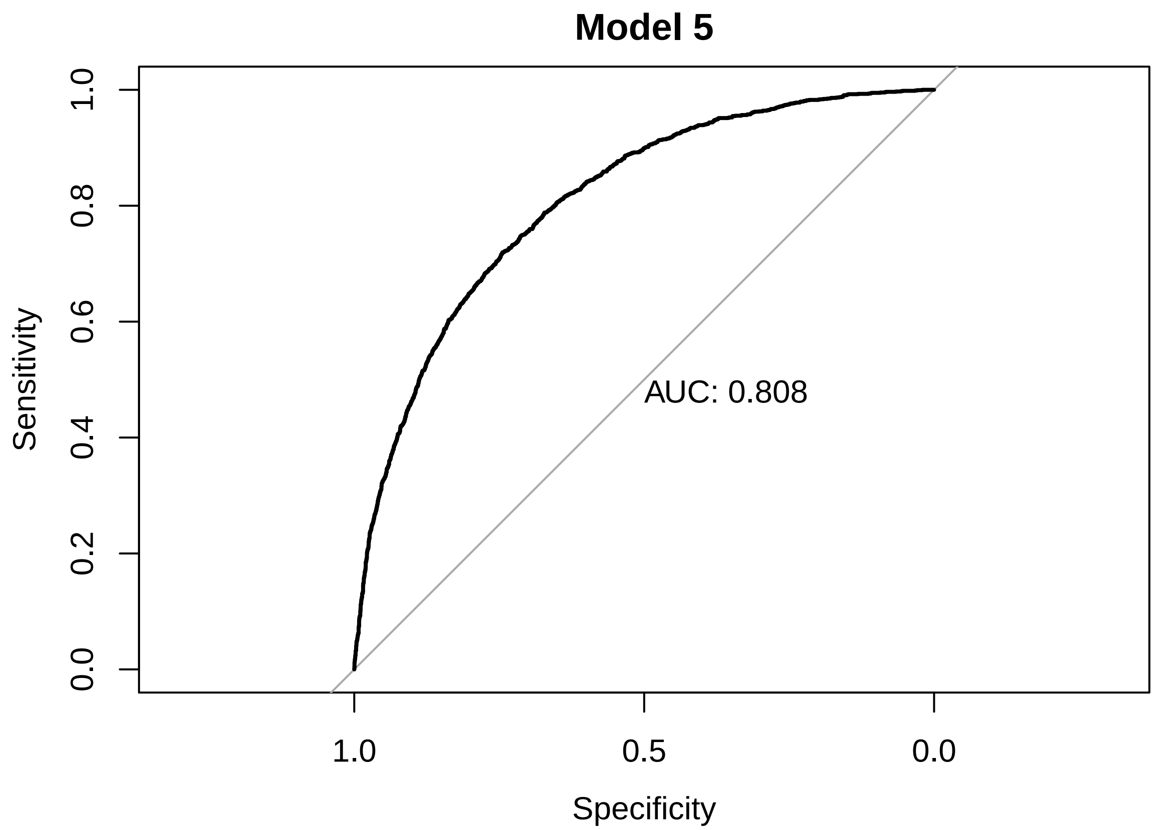

```
# plotting roc curve of model 4
plot(roc(logit_data$TARGET_FLAG,  predict(model_4, logit_data, interval = "prediction")), print.auc = T
```

```
## Setting levels: control = 1, case = 2
## Setting direction: controls < cases
```



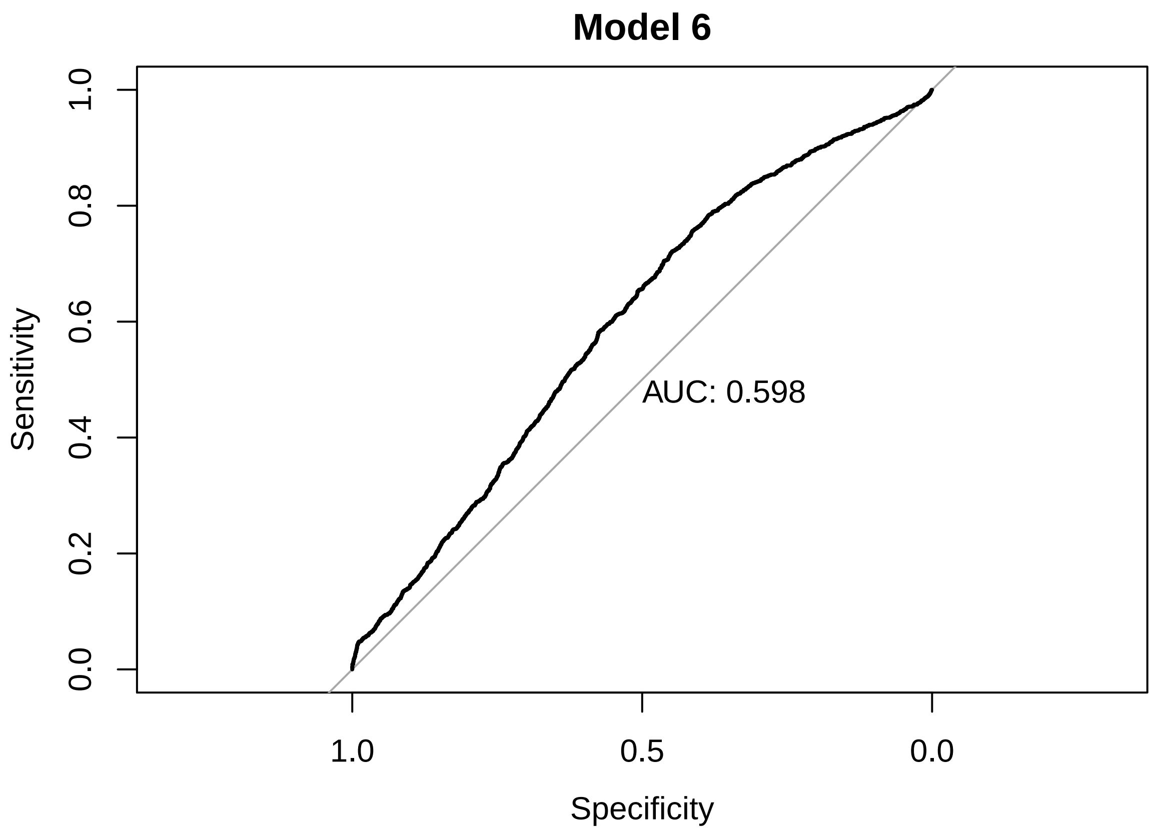

```
# plotting roc curve of model 5
plot(roc(logit_data$TARGET_FLAG,  predict(model_5, logit_data, interval = "prediction")), print.auc = TI
```

```
## Setting levels: control = 1, case = 2
## Setting direction: controls < cases
```




```
# plotting roc curve of model 6
plot(roc(logit_data$TARGET_FLAG,  predict(model_6, logit_data, interval = "prediction")), print.auc = TI
```

```
## Setting levels: control = 1, case = 2
## Setting direction: controls < cases
```




```r
# comparing all multiple regression models
a1 <- mean((summary(model_1))$residuals^2)
a2 <- mean((summary(model_2))$residuals^2)
a3 <- mean((summary(model_7))$residuals^2)
a4 <- mean((summary(model_8))$residuals^2)
a5 <- rbind(a1, a2, a3, a4)

b1 <- summary(model_1)$r.squared
b2 <- summary(model_2)$r.squared
b3 <- summary(model_7)$r.squared
b4 <- summary(model_8)$r.squared
b5 <- rbind(b1, b2, b3, b4)

c1 <- summary(model_1)$fstatistic
c2 <- summary(model_2)$fstatistic
c3 <- summary(model_7)$fstatistic
c4 <- summary(model_8)$fstatistic
c5 <- rbind(c1, c2, c3, c4)


mlr_metrics <- data.frame(cbind(a5, b5, c5), row.names = c("Model 1", "Model 2", "Model 7", "Model 8"))
colnames(mlr_metrics) <- c("MSE", "R-Squared", "value", "numdf", "dendf")
kable(mlr_metrics) %>%
  kable_styling(full_width = T) %>%
  add_header_above(c(" ", " " = 2, "F-Statistic" = 3))
```

			F-Statistic		
	MSE	R-Squared	value	numdf	dendf
Model 1	1.477939e+05	0.1601017	51.65812	24	6504
Model 2	2.198972e+05	0.1575256	50.04046	24	6423
Model 7	5.503632e-01	0.2211352	76.94228	24	6504
Model 8	5.351186e+04	0.6958974	595.24928	25	6503

When comparing the two Multiple Linear Regression Models we created, we see that the R-squared for both models are low. Both of our models are not right for the data. 15% of these models fits with the data. The last multiple linear regression model we created has 0.69 R-squared value, which makes it the right model to select for multiple linear regression model.

Prediction

We created 8 models and based on the statistic metrics for each model, we can select model 3 and model 8 to make predictions.

```
mypred <- predict(model_3, insurance_eva_cleaned, type='response')
insurance_eva_cleaned$TARGET_FLAG <- ifelse(mypred  >= 0.276, 1, 0)
write.csv(insurance_eva_cleaned, "evaluation_TARGET_FLAG.csv")
```

```
# since we selected model 8 we have to apply the same log transformation to the response variables in e
insurance_eva_cleaned$TARGET_AMT <- log(insurance_eva_cleaned$TARGET_AMT) # log transformation
insurance_eva_cleaned$INCOME <- insurance_eva_cleaned$INCOME ^0.5 #square root transformation
insurance_eva_cleaned$HOME_VAL <- insurance_eva_cleaned$HOME_VAL^0.25 # quarter root transformation
insurance_eva_cleaned$BLUEBOOK <- insurance_eva_cleaned$BLUEBOOK^0.5 # square root transformation
insurance_eva_cleaned$OLD_CLAIM <- log(insurance_eva_cleaned$OLDCLAIM + 1)  #log(x+1) transformation

mypred_2 <- exp(predict(model_8, insurance_eva_cleaned))
```

```
## Warning in predict.lm(model_8, insurance_eva_cleaned): prediction from a
## rank-deficient fit may be misleading
```

```
insurance_eva_cleaned$TARGET_AMT <- mypred_2
write.csv(insurance_eva_cleaned, "evaluation_TARGET_AMT.csv")
```

Conclusion

Based on the evaluation of Binary Logistic Models, we can select Model 3 which has the highest Accuracy about 79%. The same model also shows that the Area Under the Curve(AUC) is about 81%. For the Multiple Linear Regression Model, we can select Model 8 with since R-squared is 0.69. The prediction for TARGET_AMT and TARGET_FLAG can be found as csv file in our github repo.