DATA_621_HW4

Chi Pong, Euclid Zhang, Jie Zou, Joseph Connolly, LeTicia Cancel

3/14/2022

```
train_df <- read.csv("insurance_training_data.csv")
test_df <- read.csv("insurance-evaluation-data.csv")
```

DATA EXPLORATION

Data Summary

First, let's correct the formats/values of the data

```
train_df$INDEX <- NULL
train_df$INCOME <- as.numeric(gsub('[$,]', '', train_df$INCOME))
train_df$HOME_VAL <- as.numeric(gsub('[$,]', '', train_df$HOME_VAL))
train_df$BLUEBOOK <- as.numeric(gsub('[$,]', '', train_df$BLUEBOOK))
train_df$OLDCLAIM <- as.numeric(gsub('[$,]', '', train_df$OLDCLAIM))
train_df$PARENT1 <- gsub("z_", "", train_df$PARENT1)
train_df$MSTATUS <- gsub("z_", "", train_df$MSTATUS)
train_df$SEX <- gsub("z_", "", train_df$SEX)
train_df$EDUCATION <- gsub("z_", "", train_df$EDUCATION)
train_df$EDUCATION <- gsub("<", "Less Than", train_df$EDUCATION)
train_df$JOB <- gsub("z_", "", train_df$JOB)
train_df$CAR_TYPE <- gsub("z_", "", train_df$CAR_TYPE)
train_df$URBANICITY <- ifelse(train_df$URBANICITY=="Highly Urban/ Urban","Urban","Rural")

train_df$JOB[train_df$JOB == ""] <- NA

train_df[c("TARGET_FLAG","PARENT1","MSTATUS", "SEX", "EDUCATION", "JOB","CAR_TYPE",
           "RED_CAR", "URBANICITY", "CAR_USE","REVOKED")] <-
                lapply(train_df[c("TARGET_FLAG","PARENT1","MSTATUS", "SEX",
                                  "EDUCATION", "JOB","CAR_TYPE", "RED_CAR",
                                  "URBANICITY", "CAR_USE","REVOKED")], factor)
```

Below is the summary of the cleaned up data.

```
summary(train_df)
```

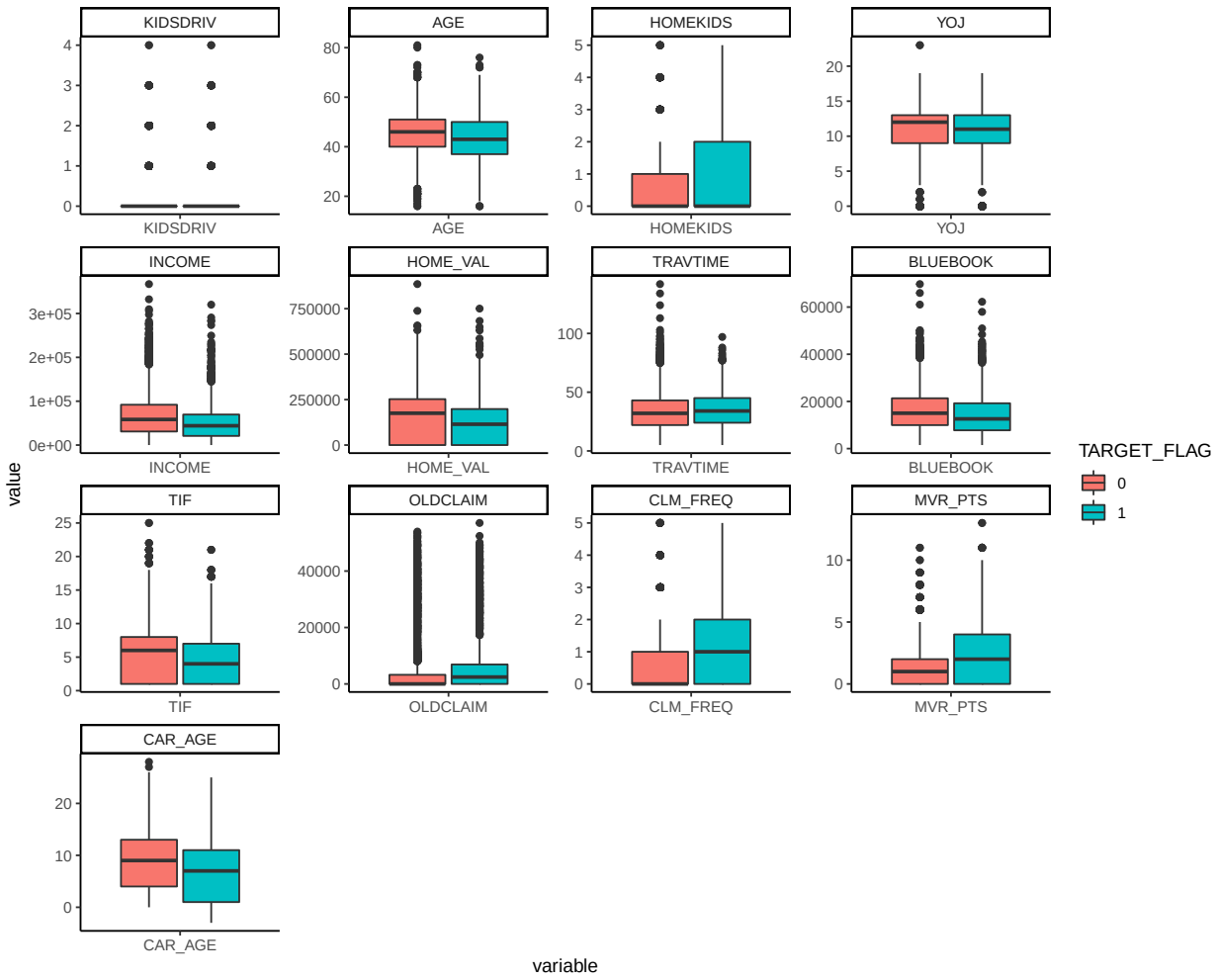
```
##  TARGET_FLAG   TARGET_AMT        KIDSDRIV           AGE           HOMEKIDS
##  0:6008      Min.   :     0   Min.   :0.0000   Min.   :16.00   Min.   :0.0000
##  1:2153      1st Qu.:     0   1st Qu.:0.0000   1st Qu.:39.00   1st Qu.:0.0000
##              Median :     0   Median :0.0000   Median :45.00   Median :0.0000
```

```
##              Mean   :  1504   Mean   :0.1711   Mean   :44.79   Mean   :0.7212
##              3rd Qu.:  1036   3rd Qu.:0.0000   3rd Qu.:51.00   3rd Qu.:1.0000
##              Max.   :107586   Max.   :4.0000   Max.   :81.00   Max.   :5.0000
##                                                NA's   :6
##       YOJ           INCOME       PARENT1       HOME_VAL      MSTATUS
##  Min.   : 0.0   Min.   :     0   No :7084   Min.   :     0   No :3267
##  1st Qu.: 9.0   1st Qu.: 28097   Yes:1077   1st Qu.:     0   Yes:4894
##  Median :11.0   Median : 54028              Median :161160
##  Mean   :10.5   Mean   : 61898              Mean   :154867
##  3rd Qu.:13.0   3rd Qu.: 85986              3rd Qu.:238724
##  Max.   :23.0   Max.   :367030              Max.   :885282
##  NA's   :454    NA's   :445                 NA's   :464
##  SEX                     EDUCATION              JOB          TRAVTIME
##  F:4375   Bachelors           :2242   Blue Collar :1825   Min.   :  5.00
##  M:3786   High School         :2330   Clerical    :1271   1st Qu.: 22.00
##           Less ThanHigh School:1203   Professional:1117   Median : 33.00
##           Masters             :1658   Manager     : 988   Mean   : 33.49
##           PhD                 : 728   Lawyer      : 835   3rd Qu.: 44.00
##                                       (Other)     :1599   Max.   :142.00
##                                       NA's        : 526
##        CAR_USE        BLUEBOOK          TIF               CAR_TYPE
##  Commercial:3029   Min.   : 1500   Min.   : 1.000   Minivan    :2145
##  Private   :5132   1st Qu.: 9280   1st Qu.: 1.000   Panel Truck: 676
##                    Median :14440   Median : 4.000   Pickup     :1389
##                    Mean   :15710   Mean   : 5.351   Sports Car : 907
##                    3rd Qu.:20850   3rd Qu.: 7.000   SUV        :2294
##                    Max.   :69740   Max.   :25.000   Van        : 750
##
##  RED_CAR       OLDCLAIM        CLM_FREQ      REVOKED       MVR_PTS
##  no :5783   Min.   :    0   Min.   :0.0000   No :7161   Min.   : 0.000
##  yes:2378   1st Qu.:    0   1st Qu.:0.0000   Yes:1000   1st Qu.: 0.000
##             Median :    0   Median :0.0000              Median : 1.000
##             Mean   : 4037   Mean   :0.7986              Mean   : 1.696
##             3rd Qu.: 4636   3rd Qu.:2.0000              3rd Qu.: 3.000
##             Max.   :57037   Max.   :5.0000              Max.   :13.000
##
##     CAR_AGE       URBANICITY
##  Min.   :-3.000   Rural:1669
##  1st Qu.: 1.000   Urban:6492
##  Median : 8.000
##  Mean   : 8.328
##  3rd Qu.:12.000
##  Max.   :28.000
##  NA's   :510
```

YOJ, **INCOME**, **HOME_VAL**, **CAR_AGE** have a lot of missing values, we will perform multiple imputations to fill in the missing values.
CAR_AGE also has an incorrect value of -3. We will also replace it by imputation.

Box Plots

```
data.m <- melt(train_df[c("TARGET_FLAG","KIDSDRIV","AGE","HOMEKIDS","YOJ","INCOME",
                          "HOME_VAL", "TRAVTIME","BLUEBOOK","TIF","OLDCLAIM","CLM_FREQ",
                          "MVR_PTS","CAR_AGE")], id.vars = 'TARGET_FLAG')
ggplot(data.m, aes(x = variable, y = value, fill = TARGET_FLAG)) + geom_boxplot() +
  facet_wrap(~ variable, scales = 'free') + theme_classic()
```

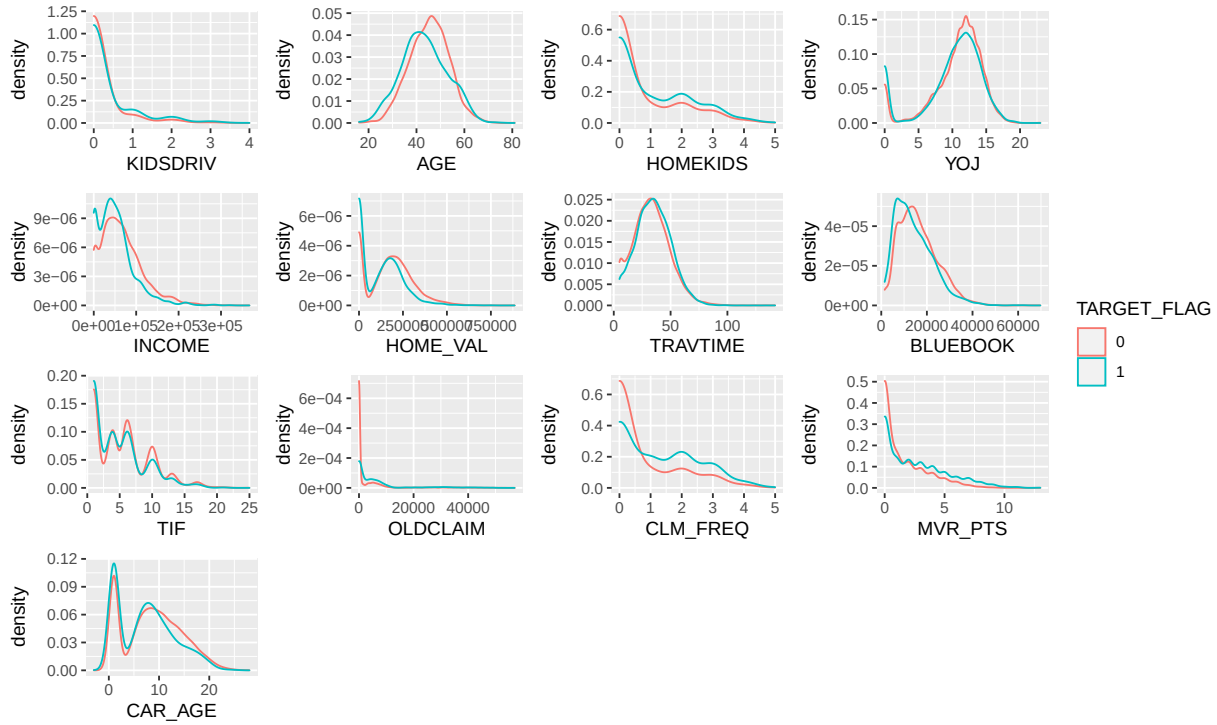


The box plots show that a lot of numric variables are right skewed, we will transform the variables to reduce outliers.

Distribution plots

```
plot_KIDSDRIV <- ggplot(train_df, aes(x=KIDSDRIV, color=TARGET_FLAG)) + geom_density(na.rm =TRUE, bw=0.3
plot_AGE <- ggplot(train_df, aes(x=AGE, color=TARGET_FLAG)) + geom_density(na.rm =TRUE)
plot_HOMEKIDS <- ggplot(train_df, aes(x=HOMEKIDS, color=TARGET_FLAG)) + geom_density(na.rm =TRUE, bw=0.4
plot_YOJ <- ggplot(train_df, aes(x=YOJ, color=TARGET_FLAG)) + geom_density(na.rm =TRUE)
plot_INCOME <- ggplot(train_df, aes(x=INCOME, color=TARGET_FLAG)) + geom_density(na.rm =TRUE)
plot_HOME_VAL <- ggplot(train_df, aes(x=HOME_VAL, color=TARGET_FLAG)) + geom_density(na.rm =TRUE)
plot_TRAVTIME <- ggplot(train_df, aes(x=TRAVTIME, color=TARGET_FLAG)) + geom_density(na.rm =TRUE)
plot_BLUEBOOK <- ggplot(train_df, aes(x=BLUEBOOK, color=TARGET_FLAG)) + geom_density(na.rm =TRUE)
```

```
plot_TIF <- ggplot(train_df, aes(x=TIF, color=TARGET_FLAG)) + geom_density(na.rm =TRUE)
plot_OLDCLAIM <- ggplot(train_df, aes(x=OLDCLAIM, color=TARGET_FLAG)) + geom_density(na.rm =TRUE)
plot_CLM_FREQ <- ggplot(train_df, aes(x=CLM_FREQ, color=TARGET_FLAG)) + geom_density(na.rm =TRUE, bw=0.4
plot_MVR_PTS <- ggplot(train_df, aes(x=MVR_PTS, color=TARGET_FLAG)) + geom_density(na.rm =TRUE, bw=0.4)
plots_CAR_AGE <- ggplot(train_df, aes(x=CAR_AGE, color=TARGET_FLAG)) + geom_density(na.rm =TRUE)

plot_KIDSDRIV+plot_AGE+plot_HOMEKIDS+plot_YOJ+plot_INCOME+plot_HOME_VAL+
  plot_TRAVTIME+plot_BLUEBOOK+plot_TIF+plot_OLDCLAIM+plot_CLM_FREQ+
  plot_MVR_PTS+plots_CAR_AGE+
  plot_layout(ncol = 4, guides = "collect")
```

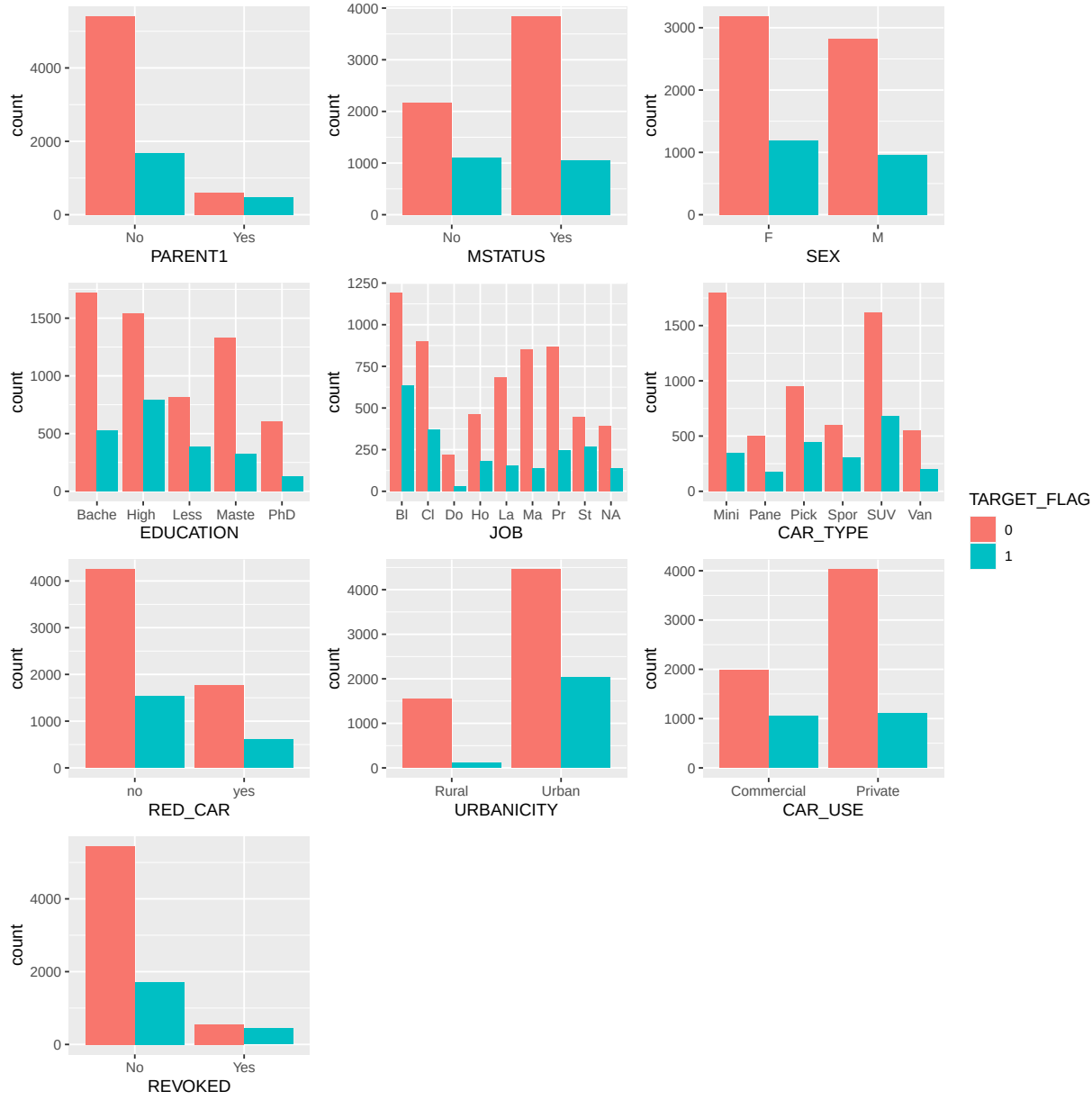


Most of the distributions are similar for target = 0 and target = 1. **OLDCLAIM** and **CLM_FREQ** are good candidates predicting whether there is a crash.

We can also look at the categorical variables:

```
plot_PARENT1 <- ggplot(train_df,aes(x=PARENT1,fill=TARGET_FLAG))+geom_bar(position = position_dodge())
plot_MSTATUS <- ggplot(train_df,aes(x=MSTATUS,fill=TARGET_FLAG))+geom_bar(position = position_dodge())
plot_SEX <- ggplot(train_df,aes(x=SEX,fill=TARGET_FLAG))+geom_bar(position = position_dodge())
plot_EDUCATION <- ggplot(train_df,aes(x=substring(train_df$EDUCATION,1,5),fill=TARGET_FLAG))+
                  geom_bar(position = position_dodge())+xlab("EDUCATION")
plot_JOB <- ggplot(train_df,aes(x=substring(train_df$JOB,1,2),fill=TARGET_FLAG))+
                  geom_bar(position = position_dodge())+xlab("JOB")
plot_CAR_TYPE <- ggplot(train_df,aes(x=substring(train_df$CAR_TYPE,1,4),fill=TARGET_FLAG))+
                  geom_bar(position = position_dodge())+xlab("CAR_TYPE")
plot_RED_CAR <- ggplot(train_df,aes(x=RED_CAR,fill=TARGET_FLAG))+geom_bar(position = position_dodge())
plot_URBANICITY <- ggplot(train_df,aes(x=URBANICITY,fill=TARGET_FLAG))+geom_bar(position = position_dod
plot_CAR_USE <- ggplot(train_df,aes(x=CAR_USE,fill=TARGET_FLAG))+geom_bar(position = position_dodge())
plot_REVOKED <- ggplot(train_df,aes(x=REVOKED,fill=TARGET_FLAG))+geom_bar(position = position_dodge())
```

```
plot_PARENT1+plot_MSTATUS+plot_SEX+plot_EDUCATION+plot_JOB+plot_CAR_TYPE+plot_RED_CAR+
  plot_URBANICITY+plot_CAR_USE+plot_REVOKED+plot_layout(ncol = 3, guides = "collect")
```



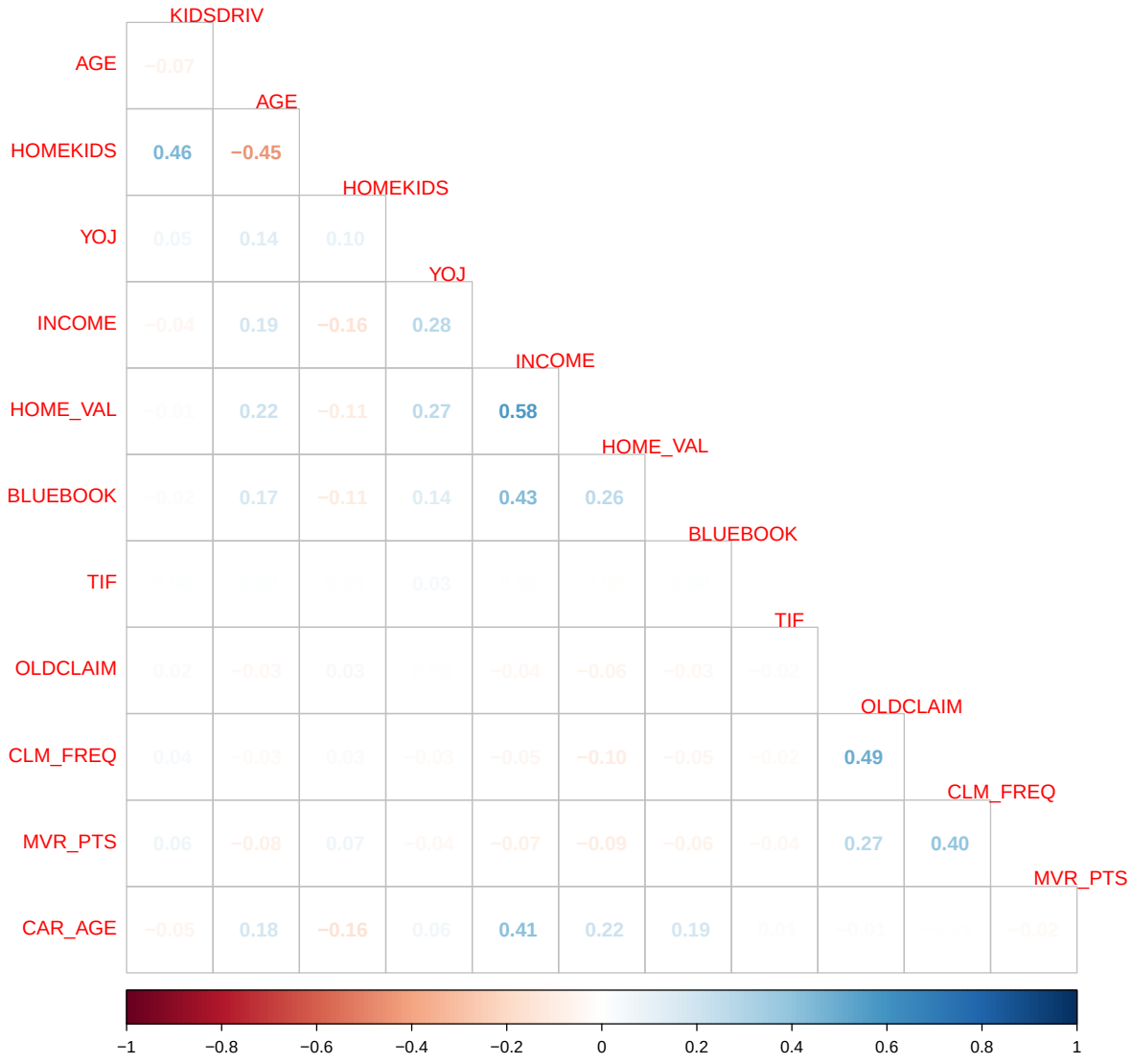
PARENT1, **MSTATUS**, **URBANICITY**, **CAR_USE** AND **REVOKED** seem to have notable difference in the distributions between target = 0 and target = 1

Correlations

Now let's look at the correlations between the variables

```
corrplot::corrplot(cor(train_df[c("KIDSDRIV","AGE","HOMEKIDS","YOJ","INCOME",
                                  "HOME_VAL","BLUEBOOK","TIF","OLDCLAIM", "CLM_FREQ",
```

```
                                    "MVR_PTS","CAR_AGE")], use = "na.or.complete"),
                  method = 'number', type = 'lower', diag = FALSE, tl.srt = 0.1)
```



None of the variables have very strong correlations. There is no serious problem of multi-collinearity.

DATA PREPARATION

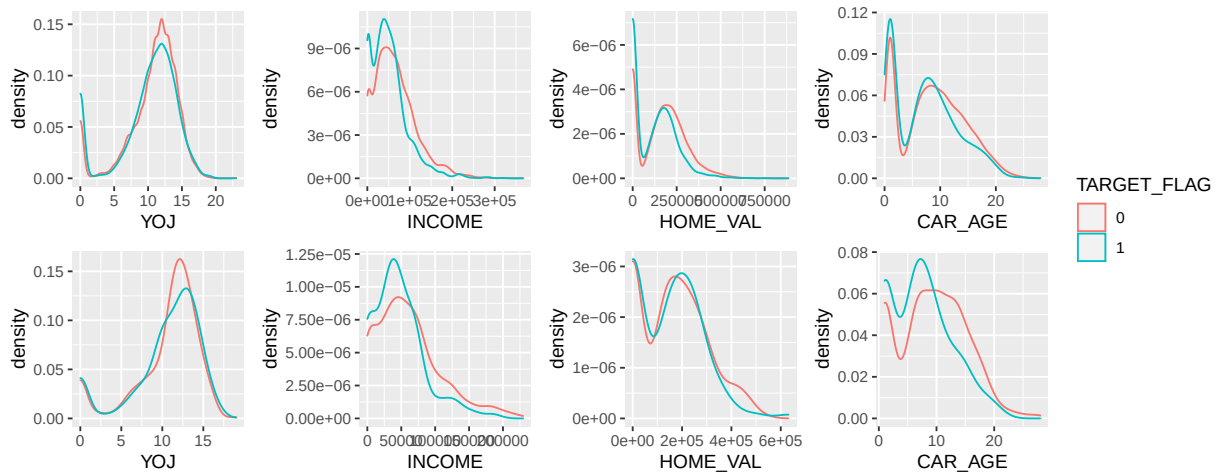
Data Imputation

```
#save the indicators of missing values. It will be used to verify the distributions
#of the imputed values
```

```
YOJ_NA <- is.na(train_df$YOJ)
INCOME_NA <- is.na(train_df$INCOME)
HOME_VAL_NA <- is.na(train_df$HOME_VAL)
CAR_AGE_NA <- is.na(train_df$CAR_AGE)

#remove incorrect CAR_AGE value for imputation
train_df$CAR_AGE[train_df$CAR_AGE < 0] <- NA

#temporary exclude TARGET_FLAG and TARGET_AMT in our imputation
TARGET_FLAG <- train_df$TARGET_FLAG
TARGET_AMT <- train_df$TARGET_AMT
train_df$TARGET_FLAG <- NULL
train_df$TARGET_AMT <- NULL

#save the imputation models to impute the test data set later
mickey <- parlmice(train_df, maxit = 5, m = 1, printFlag = FALSE, seed = 2022, cluster.seed = 2022)

#save the imputation result
train_df <- complete(mickey,1)

#Add TARGET_FLAG and TARGET_AMT back to our dataframe
train_df$TARGET_FLAG <- TARGET_FLAG
train_df$TARGET_AMT <- TARGET_AMT
TARGET_FLAG <- NULL
TARGET_AMT <- NULL

#write.csv(train_df,"train_df.csv", row.names = FALSE)

# train_df <- read.csv("train_df.csv", stringsAsFactors = TRUE)
# train_df$TARGET_FLAG <- as.factor(train_df$TARGET_FLAG)
```

We can compare the imputed data values and the original data values.
The plots on the top row below show the distributions of the values from the original data.
The plots on the bottom row below show the distributions of the imputed values.

```
plot_YOJ <- ggplot(train_df[!YOJ_NA,], aes(x=YOJ, color=TARGET_FLAG)) + geom_density(na.rm =TRUE)
plot_INCOME <- ggplot(train_df[!INCOME_NA,], aes(x=INCOME, color=TARGET_FLAG)) + geom_density(na.rm =TRU
plot_HOME_VAL <- ggplot(train_df[!HOME_VAL_NA,], aes(x=HOME_VAL, color=TARGET_FLAG)) + geom_density(na.
plot_CAR_AGE <- ggplot(train_df[!CAR_AGE_NA,], aes(x=CAR_AGE, color=TARGET_FLAG)) + geom_density(na.rm =

plot_YOJ2 <- ggplot(train_df[YOJ_NA,], aes(x=YOJ, color=TARGET_FLAG)) + geom_density(na.rm =TRUE)
plot_INCOME2 <- ggplot(train_df[INCOME_NA,], aes(x=INCOME, color=TARGET_FLAG)) + geom_density(na.rm =TRU
plot_HOME_VAL2 <- ggplot(train_df[HOME_VAL_NA,], aes(x=HOME_VAL, color=TARGET_FLAG)) + geom_density(na.
plot_CAR_AGE2 <- ggplot(train_df[CAR_AGE_NA,], aes(x=CAR_AGE, color=TARGET_FLAG)) + geom_density(na.rm =

plot_YOJ+plot_INCOME+plot_HOME_VAL+plot_CAR_AGE+
  plot_YOJ2+plot_INCOME2+plot_HOME_VAL2+plot_CAR_AGE2+
  plot_layout(ncol = 4, guides = "collect")
```



The distributions look similar and so the imputed values are plausible.

Data Transformation

YOJ: The density plot shows the variable is zero-inflated. The coefficient for YOJ=0 and the coefficient for YOJ>0 may be significantly different. Therefore, we would add a dummy variable indicating whether the variable is 0. **HOME_VAL**: The variable is also zero-inflated, we would add a dummy variable indicating whether the person has a house. **INCOME**: We would add a dummy variable indicating whether the person has a job. Practically, it is a key factor in insurance pricing. **OLDCLAIM**: We would add a dummy variable indicating whether the person had an old claim. The coefficient for OLDCLAIM=0 and the coefficient for OLDCLAIM>0 may be significantly different.

INCOME, **HOME_VAL**, **BLUEBOOK**, **OLDCLAIM**: We will log transform all monetary variables as they are right-skewed.

```
train_df$YOJ_Y <- as.factor(ifelse(train_df$YOJ == 0,0,1))
train_df$INCOME_Y <- as.factor(ifelse(train_df$INCOME == 0,0,1))
train_df$HOME_VAL_Y <- as.factor(ifelse(train_df$HOME_VAL == 0,0,1))
train_df$OLDCLAIM_Y <- as.factor(ifelse(train_df$OLDCLAIM == 0,0,1))

train_df$INCOME_LOG <- log(train_df$INCOME+1)
train_df$HOME_VAL_LOG <- log(train_df$HOME_VAL+1)
train_df$BLUEBOOK_LOG <- log(train_df$BLUEBOOK)
train_df$OLDCLAIM_LOG <- log(train_df$OLDCLAIM+1)

train_df$INCOME <- NULL
train_df$HOME_VAL <- NULL
train_df$BLUEBOOK <- NULL
train_df$OLDCLAIM <- NULL

logistic_train_df <- train_df
```

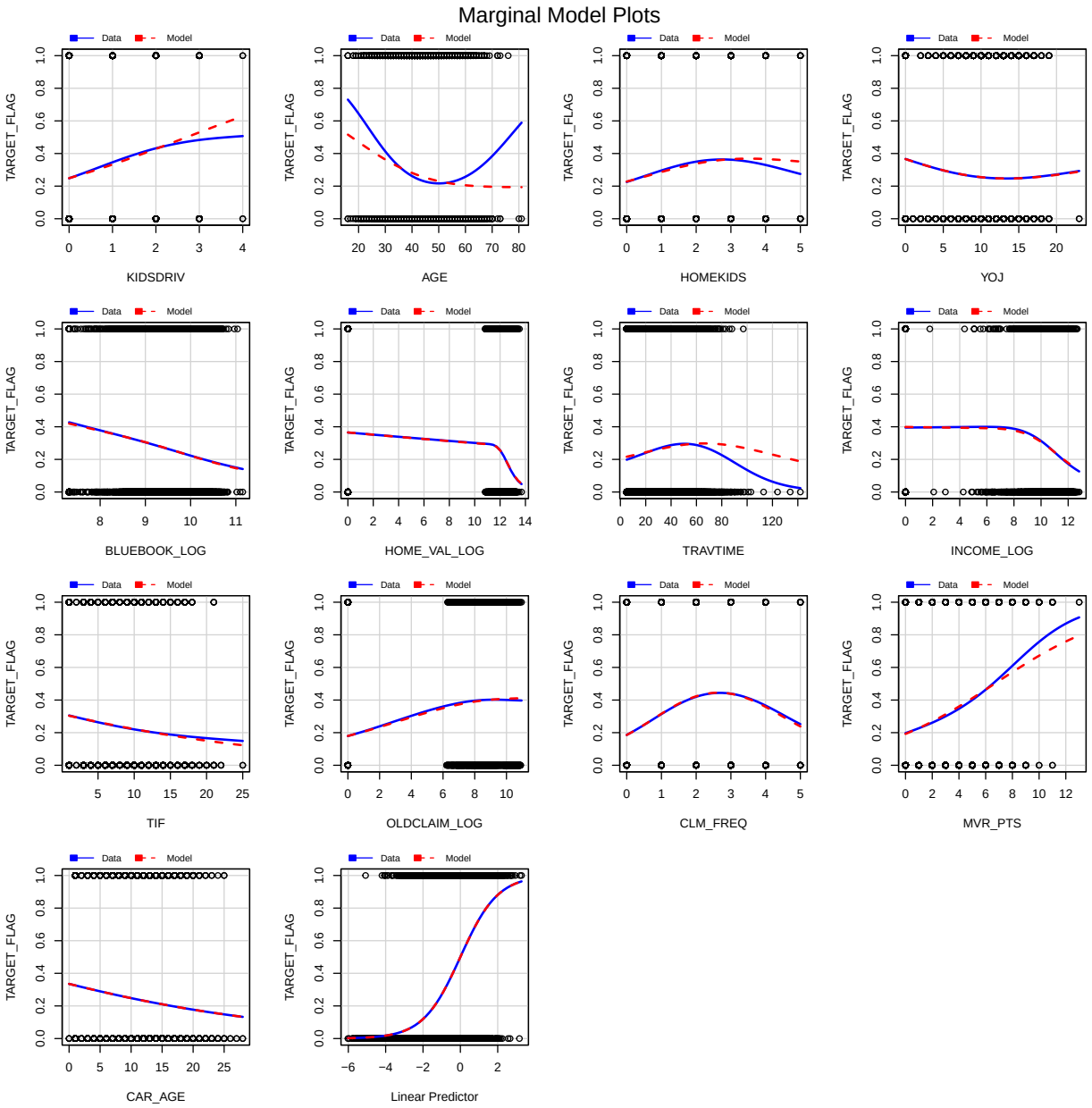
Logistic Models

Preliminary model

First, let build a test model to see if any additional transformations are needed to fit our logistic models

```
test_model <- glm(TARGET_FLAG~.-TARGET_AMT, family = binomial, logistic_train_df)
```

```
marginalModelPlots(test_model,~KIDSDRIV + AGE + HOMEKIDS + YOJ  + BLUEBOOK_LOG +
                     HOME_VAL_LOG + TRAVTIME+ INCOME_LOG + TIF + OLDCLAIM_LOG +
                      CLM_FREQ + MVR_PTS + CAR_AGE, layout =c(4,4))
```



Additonal Transformations

Additional transformation are needed for **KIDSDRIV**, **AGE**, **HOMEKIDS**, **TRAVTIME**, and **MVR_PTS**

From the density plots above, the see that **AGE** is approximately normal for both target = 0 and target = 1. From the text book *A Modern Approach To Regression With R*, if the variance of the variable is different for the two response value, then a squared term should be added.

```
data.frame(Variance_of_AGE_TARGET0=c(var(logistic_train_df$AGE[logistic_train_df$TARGET_FLAG==0])),
           Variance_of_AGE_TARGET1=c(var(logistic_train_df$AGE[logistic_train_df$TARGET_FLAG==1])))
```

```
##   Variance_of_AGE_TARGET0 Variance_of_AGE_TARGET1
## 1                67.26788                91.63286
```

```
var.test(AGE ~ TARGET_FLAG, logistic_train_df, alternative = "two.sided")
```

```
##
##  F test to compare two variances
##
## data:  AGE by TARGET_FLAG
## F = 0.7341, num df = 6007, denom df = 2152, p-value < 2.2e-16
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.6843142 0.7865982
## sample estimates:
## ratio of variances
##          0.7341022
```

The variance is apparently different. We will add a squared term and check if that fits the model.

```
logistic_train_df$AGE_Squared <- logistic_train_df$AGE^2
```

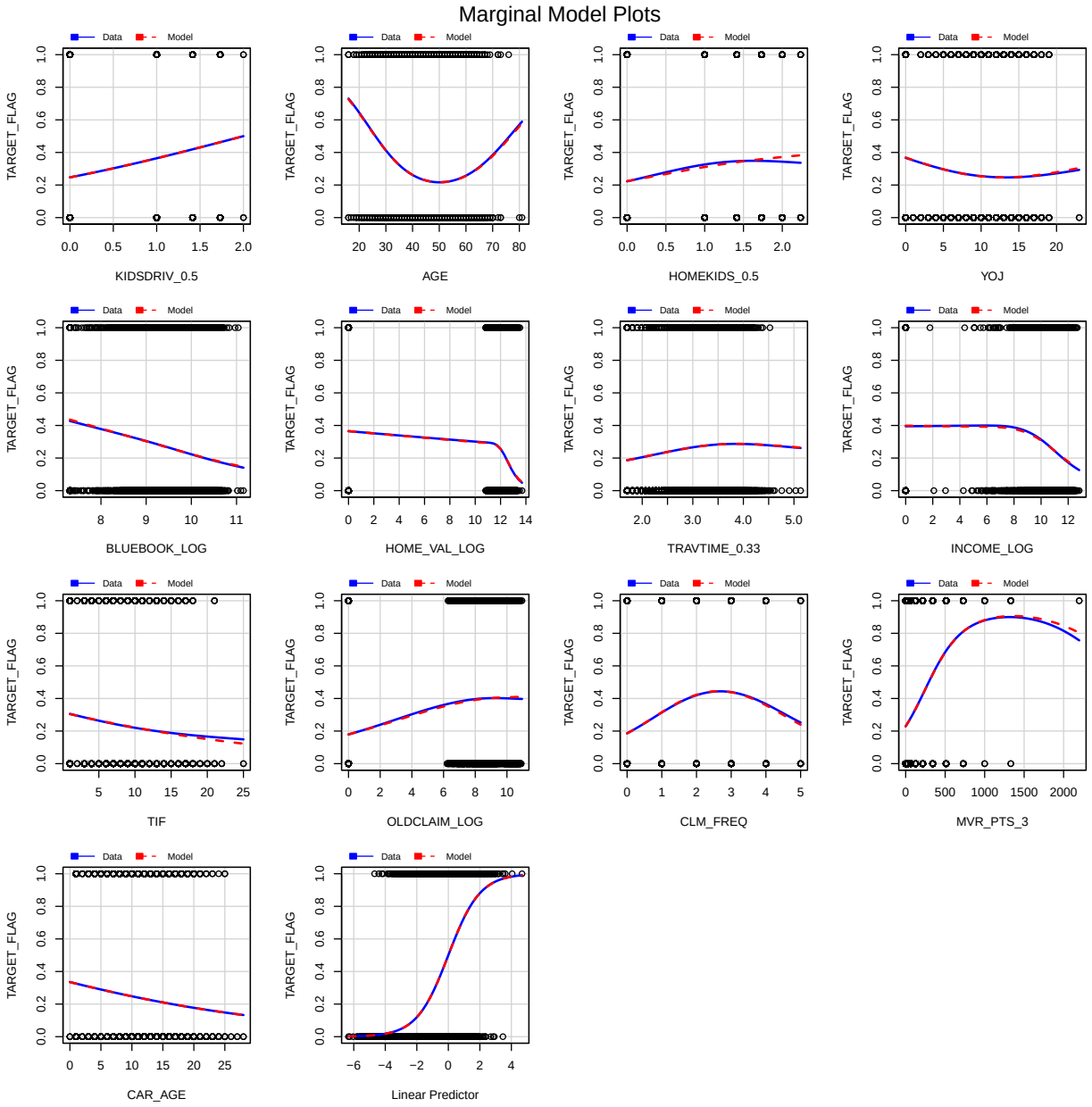
For **KIDSDRIV**, **HOMEKIDS**, **TRAVTIME**, and **MVR_PTS**, power transformations are used and the powers are determined by trial and error

```
logistic_train_df$KIDSDRIV_0.5 <- (logistic_train_df$KIDSDRIV)^0.5
logistic_train_df$HOMEKIDS_0.5 <- (logistic_train_df$HOMEKIDS)^0.5
logistic_train_df$MVR_PTS_3 <- (logistic_train_df$MVR_PTS)^3
logistic_train_df$TRAVTIME_0.33 <- (logistic_train_df$TRAVTIME)^0.33

logistic_train_df$KIDSDRIV <- NULL
logistic_train_df$HOMEKIDS <- NULL
logistic_train_df$MVR_PTS <- NULL
logistic_train_df$TRAVTIME <- NULL
```

After all the transformations, the test model now fits our data well

```
test_model <- glm(TARGET_FLAG~.-TARGET_AMT, family = binomial, logistic_train_df)
```

```
marginalModelPlots(test_model,~KIDSDRIV_0.5 + AGE + HOMEKIDS_0.5 + YOJ  + BLUEBOOK_LOG +
                     HOME_VAL_LOG + TRAVTIME_0.33 + INCOME_LOG + TIF + OLDCLAIM_LOG +
                      CLM_FREQ + MVR_PTS_3 + CAR_AGE, layout =c(4,4))
```



Building Models

Full Model

First we build a full model with all predictors

```
logi_full <- glm(TARGET_FLAG~.-TARGET_AMT, family = binomial, logistic_train_df)
```

```
summary(logi_full)
```

```
##
## Call:
## glm(formula = TARGET_FLAG ~ . - TARGET_AMT, family = binomial,
##     data = logistic_train_df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6436  -0.6982  -0.3922   0.5864   3.0620
##
## Coefficients:
##                                Estimate Std. Error z value Pr(>|z|)
## (Intercept)                   4.1926153  0.8919197   4.701 2.59e-06 ***
## AGE                          -0.2075838  0.0261411  -7.941 2.01e-15 ***
## YOJ                           0.0182481  0.0122245   1.493 0.135503
## PARENT1Yes                    0.2521664  0.1179172   2.139 0.032476 *
## MSTATUSYes                   -0.5504781  0.0908327  -6.060 1.36e-09 ***
## SEXM                          0.0225075  0.1087970   0.207 0.836107
## EDUCATIONHigh School          0.4454570  0.0916508   4.860 1.17e-06 ***
## EDUCATIONLess ThanHigh School 0.4187462  0.1191435   3.515 0.000440 ***
## EDUCATIONMasters              0.0839088  0.1270096   0.661 0.508837
## EDUCATIONPhD                  0.2149583  0.1716921   1.252 0.210570
## JOBClerical                   0.1011075  0.1079331   0.937 0.348881
## JOBDoctor                    -0.9055614  0.2424070  -3.736 0.000187 ***
## JOBHome Maker                -0.4034693  0.1777965  -2.269 0.023252 *
## JOBLawyer                    -0.1394737  0.1722292  -0.810 0.418047
## JOBManager                   -0.7362374  0.1317069  -5.590 2.27e-08 ***
## JOBProfessional              -0.1338279  0.1183506  -1.131 0.258150
## JOBStudent                   -0.5372155  0.1673175  -3.211 0.001324 **
## CAR_USEPrivate               -0.8110081  0.0901443  -8.997  < 2e-16 ***
## TIF                          -0.0559849  0.0073971  -7.568 3.78e-14 ***
## CAR_TYPEPanel Truck           0.5723457  0.1506465   3.799 0.000145 ***
## CAR_TYPEPickup                0.5932190  0.1013916   5.851 4.89e-09 ***
## CAR_TYPESports Car            0.8821918  0.1294136   6.817 9.31e-12 ***
## CAR_TYPESUV                   0.7138015  0.1078867   6.616 3.69e-11 ***
## CAR_TYPEVan                   0.6931168  0.1261036   5.496 3.88e-08 ***
## RED_CARyes                   -0.0568190  0.0874897  -0.649 0.516056
## CLM_FREQ                      0.0508621  0.0449174   1.132 0.257489
## REVOKEDYes                    0.8509978  0.0893069   9.529  < 2e-16 ***
## CAR_AGE                      -0.0045643  0.0075999  -0.601 0.548128
## URBANICITYUrban               2.3710113  0.1136549  20.862  < 2e-16 ***
## YOJ_Y1                        0.1880298  0.3223031   0.583 0.559628
## INCOME_Y1                     0.0176654  0.5757048   0.031 0.975521
## HOME_VAL_Y1                   1.4025301  1.4003605   1.002 0.316561
## OLDCLAIM_Y1                   1.7780707  0.4180172   4.254 2.10e-05 ***
## INCOME_LOG                   -0.1177478  0.0550460  -2.139 0.032429 *
## HOME_VAL_LOG                 -0.1420093  0.1153172  -1.231 0.218148
## BLUEBOOK_LOG                 -0.3267276  0.0596210  -5.480 4.25e-08 ***
## OLDCLAIM_LOG                 -0.1614049  0.0459442  -3.513 0.000443 ***
## AGE_Squared                   0.0022742  0.0002857   7.959 1.73e-15 ***
## KIDSDRIV_0.5                  0.6931103  0.0860736   8.053 8.11e-16 ***
## HOMEKIDS_0.5                 -0.0019231  0.0695976  -0.028 0.977956
```

```
## MVR_PTS_3                      0.0016496  0.0002616   6.306 2.86e-10 ***
## TRAVTIME_0.33                  0.4474503  0.0557612   8.024 1.02e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 9418  on 8160  degrees of freedom
## Residual deviance: 7190  on 8119  degrees of freedom
## AIC: 7274
##
## Number of Fisher Scoring iterations: 5
```

Backward Elimination by AIC

Starting with our full model, perform backward elimination by comparing the **AIC** of the models.

```
logi_AIC <- step(logi_full,trace=0)

summary(logi_AIC)
```

```
##
## Call:
## glm(formula = TARGET_FLAG ~ AGE + YOJ + PARENT1 + MSTATUS + EDUCATION +
##     JOB + CAR_USE + TIF + CAR_TYPE + REVOKED + URBANICITY + OLDCLAIM_Y +
##     INCOME_LOG + HOME_VAL_LOG + BLUEBOOK_LOG + OLDCLAIM_LOG +
##     AGE_Squared + KIDSDRIV_0.5 + MVR_PTS_3 + TRAVTIME_0.33, family = binomial,
##     data = logistic_train_df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6324  -0.6991  -0.3948   0.5856   3.0545
##
## Coefficients:
##                                Estimate Std. Error z value Pr(>|z|)
## (Intercept)                   4.2053043  0.8075640   5.207 1.92e-07 ***
## AGE                          -0.2075348  0.0251563  -8.250  < 2e-16 ***
## YOJ                           0.0226659  0.0104885   2.161 0.030694 *
## PARENT1Yes                    0.2410410  0.1020298   2.362 0.018154 *
## MSTATUSYes                   -0.5598577  0.0870345  -6.433 1.25e-10 ***
## EDUCATIONHigh School          0.4809054  0.0838658   5.734 9.80e-09 ***
## EDUCATIONLess ThanHigh School 0.4689556  0.1084051   4.326 1.52e-05 ***
## EDUCATIONMasters              0.0543500  0.1199101   0.453 0.650364
## EDUCATIONPhD                  0.1696062  0.1655181   1.025 0.305505
## JOBClerical                   0.1229697  0.1054132   1.167 0.243392
## JOBDoctor                    -0.9021506  0.2418323  -3.730 0.000191 ***
## JOBHome Maker                -0.3542443  0.1652394  -2.144 0.032047 *
## JOBLawyer                    -0.1390494  0.1720254  -0.808 0.418913
## JOBManager                   -0.7415811  0.1316323  -5.634 1.76e-08 ***
## JOBProfessional              -0.1386627  0.1181586  -1.174 0.240583
## JOBStudent                   -0.5438290  0.1500289  -3.625 0.000289 ***
## CAR_USEPrivate               -0.8113768  0.0900785  -9.007  < 2e-16 ***
## TIF                          -0.0558884  0.0073937  -7.559 4.06e-14 ***
```

```
## CAR_TYPEPanel Truck            0.5634100  0.1442941   3.905 9.44e-05 ***
## CAR_TYPEPickup                 0.5956822  0.1013114   5.880 4.11e-09 ***
## CAR_TYPESports Car             0.8940186  0.1093123   8.179 2.87e-16 ***
## CAR_TYPESUV                    0.7218283  0.0865293   8.342  < 2e-16 ***
## CAR_TYPEVan                    0.6834504  0.1224992   5.579 2.42e-08 ***
## REVOKEDYes                     0.8489262  0.0892323   9.514  < 2e-16 ***
## URBANICITYUrban                2.3764239  0.1136672  20.907  < 2e-16 ***
## OLDCLAIM_Y1                    1.8959019  0.4065049   4.664 3.10e-06 ***
## INCOME_LOG                    -0.1072836  0.0178796  -6.000 1.97e-09 ***
## HOME_VAL_LOG                  -0.0267723  0.0070144  -3.817 0.000135 ***
## BLUEBOOK_LOG                  -0.3294624  0.0555927  -5.926 3.10e-09 ***
## OLDCLAIM_LOG                  -0.1625225  0.0459131  -3.540 0.000400 ***
## AGE_Squared                    0.0022675  0.0002781   8.154 3.52e-16 ***
## KIDSDRIV_0.5                   0.6906316  0.0750086   9.207  < 2e-16 ***
## MVR_PTS_3                      0.0016379  0.0002609   6.279 3.41e-10 ***
## TRAVTIME_0.33                  0.4486130  0.0557318   8.049 8.31e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 9418.0  on 8160  degrees of freedom
## Residual deviance: 7193.8  on 8127  degrees of freedom
## AIC: 7261.8
##
## Number of Fisher Scoring iterations: 5
```

Backward Elimination by BIC

Starting with our full model, perform backward elimination by comparing the **BIC** of the models.

```
logi_BIC <- step(logi_full,trace=0,k=log(nrow(logistic_train_df)))

summary(logi_BIC)
```

```
##
## Call:
## glm(formula = TARGET_FLAG ~ AGE + MSTATUS + JOB + CAR_USE + TIF +
##     CAR_TYPE + REVOKED + CAR_AGE + URBANICITY + OLDCLAIM_Y +
##     INCOME_LOG + HOME_VAL_LOG + BLUEBOOK_LOG + OLDCLAIM_LOG +
##     AGE_Squared + KIDSDRIV_0.5 + MVR_PTS_3 + TRAVTIME_0.33, family = binomial,
##     data = logistic_train_df)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -2.706  -0.704  -0.400   0.602   3.028
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)          5.2883645  0.7840917   6.745 1.53e-11 ***
## AGE                 -0.2239648  0.0245516  -9.122  < 2e-16 ***
## MSTATUSYes          -0.5978100  0.0766304  -7.801 6.13e-15 ***
## JOBClerical          0.1663320  0.1046862   1.589 0.112091
```

```
## JOBDoctor           -0.9410548  0.1878314  -5.010 5.44e-07 ***
## JOBHome Maker       -0.4250346  0.1597190  -2.661 0.007788 **
## JOBLawyer           -0.2723200  0.1334668  -2.040 0.041315 *
## JOBManager          -0.8749803  0.1165556  -7.507 6.05e-14 ***
## JOBProfessional     -0.3206208  0.1087251  -2.949 0.003189 **
## JOBStudent          -0.5212863  0.1490870  -3.497 0.000471 ***
## CAR_USEPrivate      -0.7695062  0.0852531  -9.026  < 2e-16 ***
## TIF                 -0.0553312  0.0073640  -7.514 5.75e-14 ***
## CAR_TYPEPanel Truck  0.5902660  0.1411788   4.181 2.90e-05 ***
## CAR_TYPEPickup       0.6123297  0.1001314   6.115 9.64e-10 ***
## CAR_TYPESports Car   0.8864395  0.1088492   8.144 3.83e-16 ***
## CAR_TYPESUV          0.7252899  0.0861403   8.420  < 2e-16 ***
## CAR_TYPEVan          0.6745191  0.1214291   5.555 2.78e-08 ***
## REVOKEDYes           0.8470189  0.0889217   9.525  < 2e-16 ***
## CAR_AGE             -0.0196706  0.0063170  -3.114 0.001846 **
## URBANICITYUrban      2.3496569  0.1131686  20.762  < 2e-16 ***
## OLDCLAIM_Y1          1.8562086  0.4045177   4.589 4.46e-06 ***
## INCOME_LOG          -0.0882868  0.0141498  -6.239 4.39e-10 ***
## HOME_VAL_LOG        -0.0274088  0.0069772  -3.928 8.55e-05 ***
## BLUEBOOK_LOG        -0.3390947  0.0553941  -6.121 9.27e-10 ***
## OLDCLAIM_LOG        -0.1574864  0.0456965  -3.446 0.000568 ***
## AGE_Squared          0.0024336  0.0002731   8.912  < 2e-16 ***
## KIDSDRIV_0.5         0.7547176  0.0711852  10.602  < 2e-16 ***
## MVR_PTS_3            0.0016241  0.0002609   6.225 4.82e-10 ***
## TRAVTIME_0.33        0.4322588  0.0554208   7.800 6.21e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 9418.0  on 8160  degrees of freedom
## Residual deviance: 7233.3  on 8132  degrees of freedom
## AIC: 7291.3
##
## Number of Fisher Scoring iterations: 5
```

Backward Elimination with Chi-square test

Starting with our full model, perform backward elimination with Chi-square test.

```
#Define a function to perform backward elimination with Chi-square test
#using the significancy / alpha as one of the parameters

backward_chi <- function (train_df, significancy) {
  glm_string <- "TARGET_FLAG~.-TARGET_AMT"
  glm_formula <- as.formula(glm_string)

  repeat{
    drop1_chi <- drop1(glm(glm_formula, family=binomial, train_df), test="Chi")

    chi_result <- data.frame(preditors = rownames(drop1_chi)[-1],
              p_value = drop1_chi[-1,5])
    chi_result <- chi_result[order(chi_result$p_value,decreasing=TRUE),]
```

```
    if(chi_result[1,2] < significancy){
        break
    }
    else {
        glm_string <- paste0(glm_string,"-",chi_result[1,1])
        glm_formula <- as.formula(glm_string)
    }
  }

  return(glm_formula)
}
```

model with alpha 0.001 (based on Chi-square test)**

```
logi_chi_0.001 <- backward_chi(logistic_train_df, 0.001)
logi_chi_0.001 <- glm(logi_chi_0.001, family=binomial, logistic_train_df)
summary(logi_chi_0.001)
```

```
##
## Call:
## glm(formula = logi_chi_0.001, family = binomial, data = logistic_train_df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6119  -0.7000  -0.3978   0.5964   3.0622
##
## Coefficients:
##                                Estimate Std. Error z value Pr(>|z|)
## (Intercept)                   4.6596809  0.7906273   5.894 3.78e-09 ***
## AGE                          -0.2238191  0.0246070  -9.096  < 2e-16 ***
## MSTATUSYes                   -0.6217649  0.0769517  -8.080 6.48e-16 ***
## EDUCATIONHigh School          0.4851558  0.0837106   5.796 6.81e-09 ***
## EDUCATIONLess ThanHigh School  0.4818459  0.1081007   4.457 8.30e-06 ***
## EDUCATIONMasters              0.0424128  0.1197831   0.354 0.723279
## EDUCATIONPhD                  0.1548882  0.1654297   0.936 0.349130
## JOBClerical                   0.1369476  0.1051832   1.302 0.192919
## JOBDoctor                    -0.9139680  0.2417115  -3.781 0.000156 ***
## JOBHome Maker                -0.3475757  0.1650003  -2.107 0.035160 *
## JOBLawyer                    -0.1368241  0.1718765  -0.796 0.425997
## JOBManager                   -0.7440571  0.1314774  -5.659 1.52e-08 ***
## JOBProfessional              -0.1396181  0.1179796  -1.183 0.236647
## JOBStudent                   -0.5207675  0.1494837  -3.484 0.000494 ***
## CAR_USEPrivate               -0.8071656  0.0899564  -8.973  < 2e-16 ***
## TIF                          -0.0557846  0.0073849  -7.554 4.22e-14 ***
## CAR_TYPEPanel Truck           0.5490942  0.1441420   3.809 0.000139 ***
## CAR_TYPEPickup                0.5900486  0.1012617   5.827 5.64e-09 ***
## CAR_TYPESports Car            0.8936469  0.1090828   8.192 2.56e-16 ***
## CAR_TYPESUV                   0.7249045  0.0863638   8.394  < 2e-16 ***
## CAR_TYPEVan                   0.6699005  0.1224277   5.472 4.45e-08 ***
## REVOKEDYes                    0.8493823  0.0891338   9.529  < 2e-16 ***
## URBANICITYUrban               2.3685690  0.1134732  20.873  < 2e-16 ***
## OLDCLAIM_Y1                   1.8771837  0.4061273   4.622 3.80e-06 ***
```

```
## INCOME_LOG                   -0.0846883  0.0142323  -5.950 2.67e-09 ***
## HOME_VAL_LOG                 -0.0262103  0.0069893  -3.750 0.000177 ***
## BLUEBOOK_LOG                 -0.3270923  0.0555339  -5.890 3.86e-09 ***
## OLDCLAIM_LOG                 -0.1603339  0.0458632  -3.496 0.000472 ***
## AGE_Squared                   0.0024321  0.0002737   8.887  < 2e-16 ***
## KIDSDRIV_0.5                  0.7579709  0.0714426  10.610  < 2e-16 ***
## MVR_PTS_3                     0.0016352  0.0002610   6.265 3.72e-10 ***
## TRAVTIME_0.33                 0.4425301  0.0556105   7.958 1.75e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 9418.0  on 8160  degrees of freedom
## Residual deviance: 7205.1  on 8129  degrees of freedom
## AIC: 7269.1
##
## Number of Fisher Scoring iterations: 5
```

Model Selection

Since the data is imbalanced, we would not use the threshold for our model predictions.

In business, we don't want to misclassify a person with high risk to be low risk. We also don't want to lose customers by charging low risk people at a high-risk rate. Practically, we should use a cost matrix to determine the threshold for our classification. Since we don't know the cost here, we will weight the Sensitivity and Specificity equally. We will find our optimal threshold that maximize the sum of Sensitivity and Specificity

```
logi_models <- data.frame(model=c(""),DF=c(0),AIC=c(0.0000),AUC=c(0.0000),
                          Optimal_Threhold=c(0.0000),Sensitivity=c(0.0000),
                          Specificity=c(0.0000),Sum_Sens_Spec=c(0.0000))

models <- list(logi_full, logi_AIC, logi_BIC, logi_chi_0.001)
model_names <- c("logi_full", "logi_AIC", "logi_BIC", "logi_chi_0.001")
for (i in c(1:length(models))) {
    logi_models[i,"model"] <- model_names[i]
    logi_models[i,"DF"] <- models[[i]]$df.residual
    logi_models[i,"AIC"] <- round(models[[i]]$aic,4)
    rocCurve <- roc(logistic_train_df$TARGET_FLAG, models[[i]]$fitted.values)
    logi_models[i,"AUC"] <- round(rocCurve$auc,4)
    roc_df <- data.frame(Sensitivity = rocCurve$sensitivities,
                         Specificity = rocCurve$specificities,
                         Sum_Sens_Spec = rocCurve$sensitivities+rocCurve$specificities,
                         Thresholds = rocCurve$thresholds)
    roc_df <- roc_df[which.max(roc_df$Sum_Sens_Spec),]
    logi_models[i,"Optimal_Threhold"] <- roc_df$Thresholds
    logi_models[i,"Sensitivity"] <- roc_df$Sensitivity
    logi_models[i,"Specificity"] <- roc_df$Specificity
    logi_models[i,"Sum_Sens_Spec"] <- roc_df$Sum_Sens_Spec
}
logi_models
```

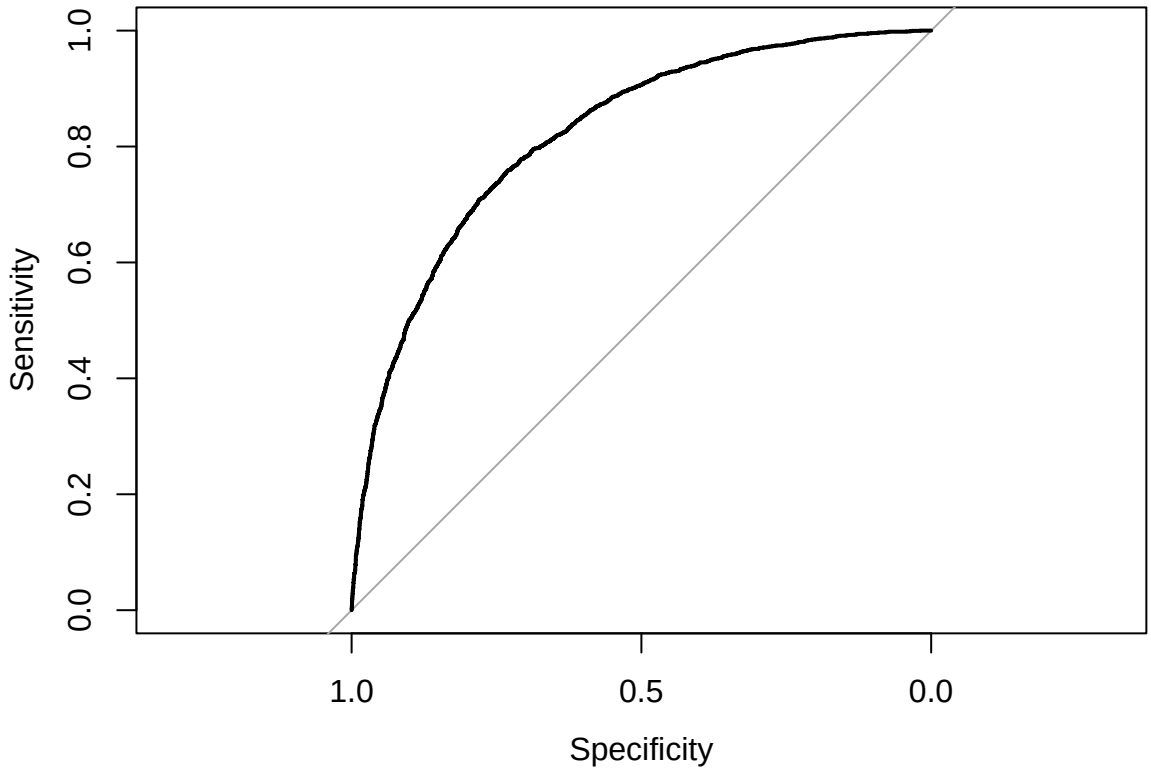
```
##           model   DF      AIC    AUC Optimal_Threhold Sensitivity Specificity
```

```
## 1      logi_full 8119 7273.990 0.8201        0.2651635   0.7570831   0.7333555
## 2       logi_AIC 8127 7261.818 0.8200        0.2628578   0.7589410   0.7305260
## 3       logi_BIC 8132 7291.271 0.8173        0.3008026   0.7101719   0.7724700
## 4 logi_chi_0.001 8129 7269.081 0.8192        0.3041777   0.7120297   0.7767976
##   Sum_Sens_Spec
## 1      1.490439
## 2      1.489467
## 3      1.482642
## 4      1.488827
```

By comparing the AUC and the sum of Sensitivity and Specificity, the best model is logi_AIC. The performance of logi_AIC is very close to the full model. The full model should not be selected since it is less parsimonious.

The following is the ROC of the logi_AIC model

```
rocCurve <- roc(logistic_train_df$TARGET_FLAG, logi_AIC$fitted.values)
plot(rocCurve)
```



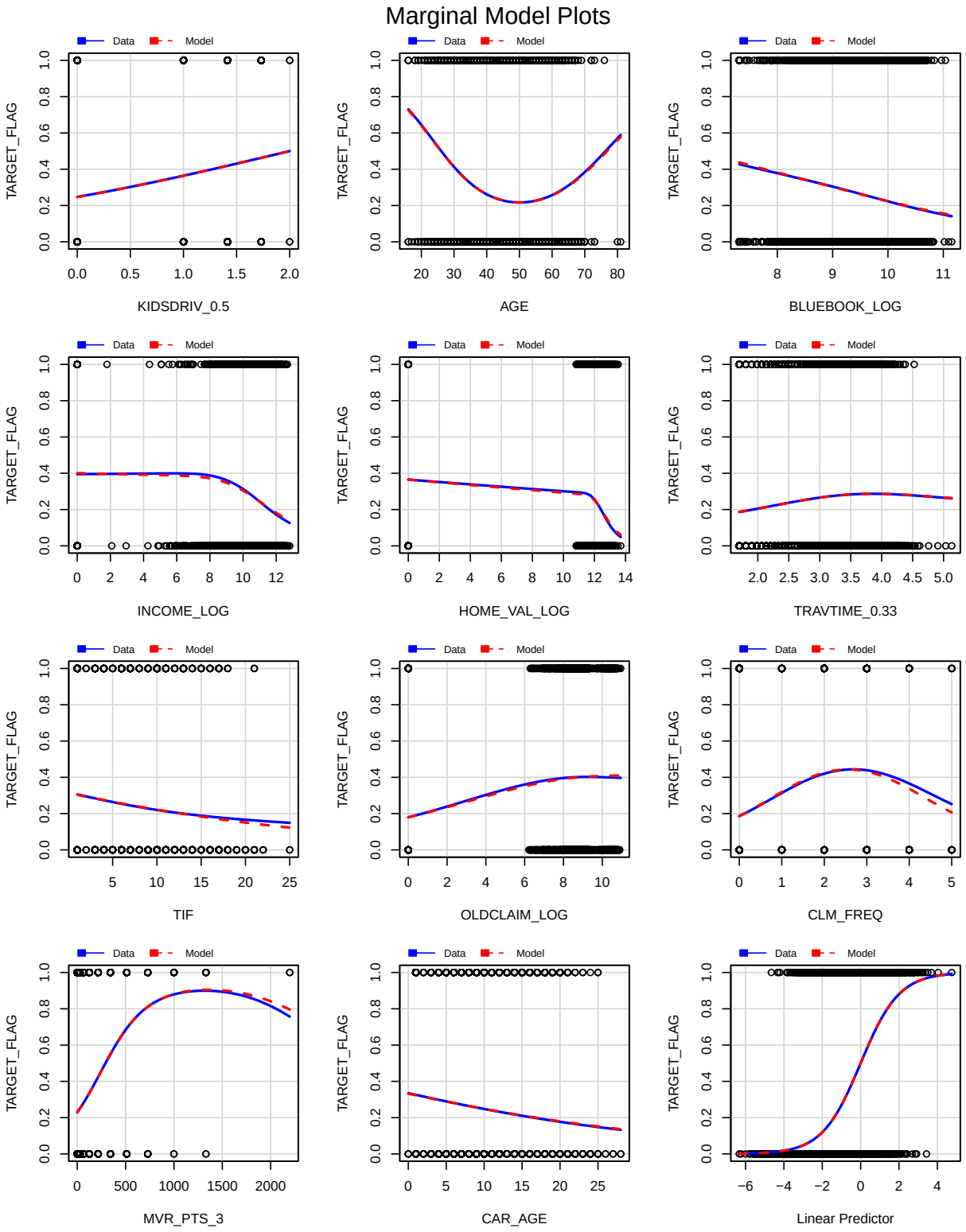
The following is the confusion matrix of the logi_AIC model

```
predicted_class <- ifelse(logi_AIC$fitted.values>logi_models[2,"Optimal_Threhold"],1,0)
confusion_matrix <- confusionMatrix(as.factor(predicted_class),
                                    as.factor(train_df$TARGET_FLAG),positive = "1")
confusion_matrix
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 4389  519
##          1 1619 1634
##
##                Accuracy : 0.738
##                  95% CI : (0.7283, 0.7475)
##     No Information Rate : 0.7362
##     P-Value [Acc > NIR] : 0.3585
##
##                   Kappa : 0.4205
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.7589
##             Specificity : 0.7305
##          Pos Pred Value : 0.5023
##          Neg Pred Value : 0.8943
##              Prevalence : 0.2638
##          Detection Rate : 0.2002
##    Detection Prevalence : 0.3986
##       Balanced Accuracy : 0.7447
##
##        'Positive' Class : 1
##
```

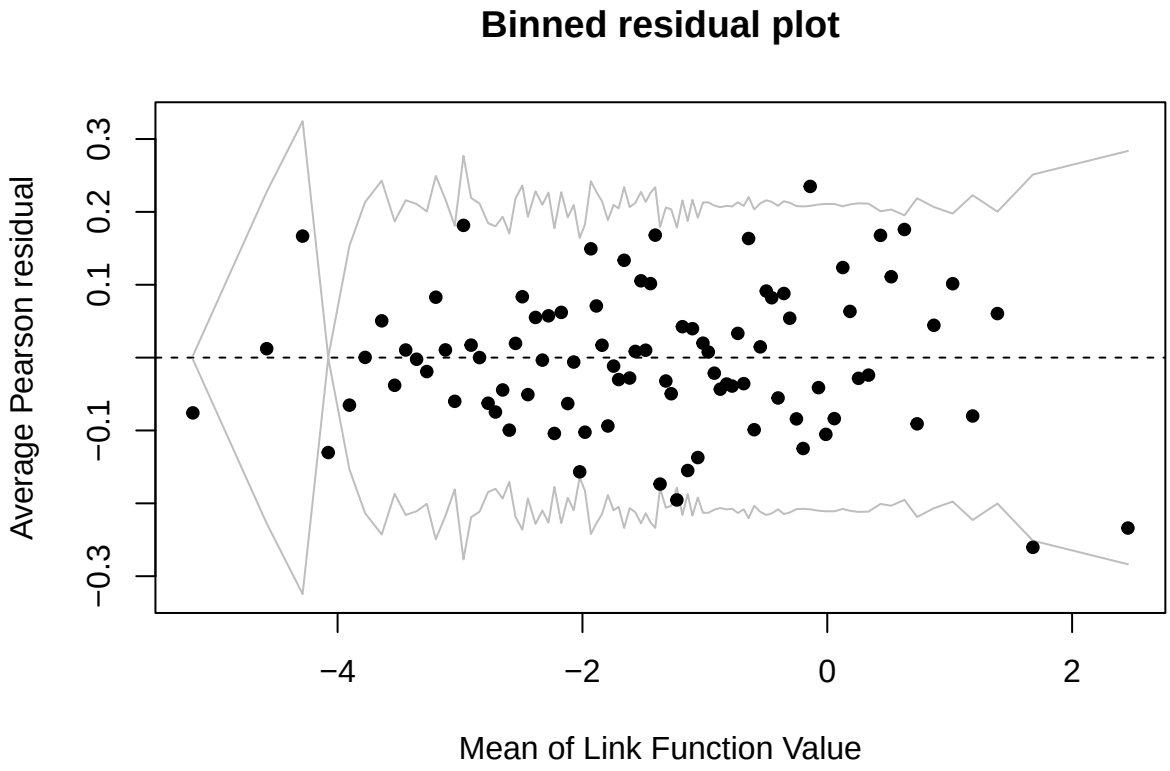
From the below marginal plots, we see no lack of fit of our model

```
marginalModelPlots(logi_AIC,~KIDSDRIV_0.5 + AGE + BLUEBOOK_LOG + INCOME_LOG +
                    HOME_VAL_LOG + TRAVTIME_0.33 + TIF + OLDCLAIM_LOG + CLM_FREQ +
                     MVR_PTS_3 + CAR_AGE, layout =c(4,3))
```



The residual plot below also shows that the pearson residuals are independent with approximately constant variance, with only a few outliers.

```
#arm::binnedplot(x = fitted(logi_AIC), y = residuals(logi_AIC, type="pearson"),
arm::binnedplot(x = predict(logi_AIC, type="link"), y = residuals(logi_AIC, type="pearson"),
                nclass = NULL,
                xlab = "Mean of Link Function Value",
                ylab = "Average Pearson residual",
                main = "Binned residual plot",
                cex.pts = 0.8,
                col.pts = 1,
                col.int = "gray")
```



We conclude that our optimal logistic model logi_AIC is valid

Linear Model

Preliminary model

First, let's build a test model to check if any additional transformations are needed to build a valid model

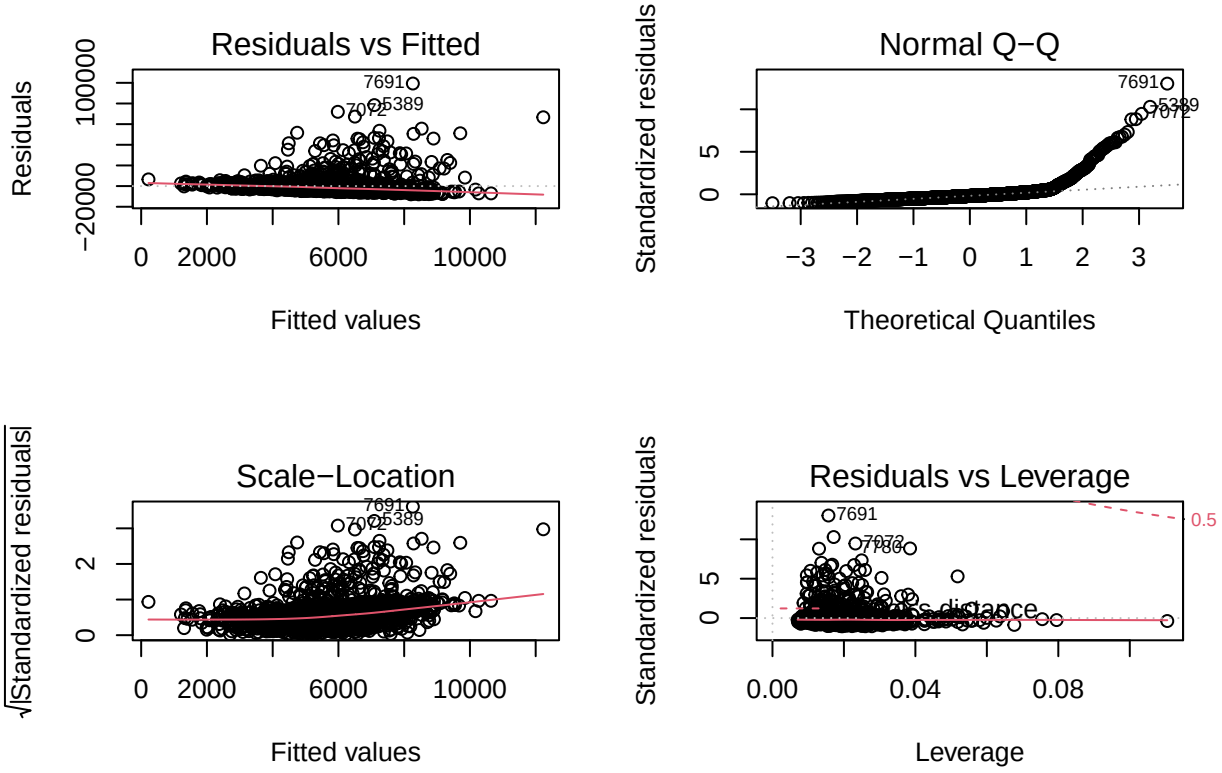
```
lm_train_df <- train_df[train_df$TARGET_FLAG==1,]
lm_train_df$TARGET_FLAG <- NULL
```

```
lm_full <- lm(TARGET_AMT~., lm_train_df)
summary(lm_full)
```

```
##
## Call:
## lm(formula = TARGET_AMT ~ ., data = lm_train_df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -7957  -3194  -1454    468  99322
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  -7990.1949  3515.8668  -2.273   0.0231 *
## KIDSDRIV                      -206.6577   317.7924  -0.650   0.5156
## AGE                             19.0736    21.7404   0.877   0.3804
## HOMEKIDS                       253.2798   214.4126   1.181   0.2376
## YOJ                            -10.6920    72.5708  -0.147   0.8829
## PARENT1Yes                     215.8562   589.8602   0.366   0.7144
## MSTATUSYes                    -973.8075   519.7251  -1.874   0.0611 .
## SEXM                          1117.2957   633.8365   1.763   0.0781 .
## EDUCATIONHigh School          -571.9285   514.8222  -1.111   0.2667
## EDUCATIONLess ThanHigh School  -67.8244   652.2848  -0.104   0.9172
## EDUCATIONMasters               211.0096   815.6847   0.259   0.7959
## EDUCATIONPhD                  2134.7035  1094.1512   1.951   0.0512 .
## JOBClerical                      3.3761   589.4189   0.006   0.9954
## JOBDoctor                    -2855.8022  1558.2757  -1.833   0.0670 .
## JOBHome Maker                 -227.6621  1026.2562  -0.222   0.8245
## JOBLawyer                      924.8903  1061.3919   0.871   0.3836
## JOBManager                    -833.8549   845.2089  -0.987   0.3240
## JOBProfessional                513.0563   672.0814   0.763   0.4453
## JOBStudent                    -126.2174   930.5024  -0.136   0.8921
## TRAVTIME                        -0.8145    11.0817  -0.073   0.9414
## CAR_USEPrivate                -569.2914   511.2572  -1.114   0.2656
## TIF                            -14.1601    42.5279  -0.333   0.7392
## CAR_TYPEPanel Truck            116.6428   875.5160   0.133   0.8940
## CAR_TYPEPickup                -169.4210   597.0312  -0.284   0.7766
## CAR_TYPESports Car             921.9720   736.3837   1.252   0.2107
## CAR_TYPESUV                    632.8410   644.3983   0.982   0.3262
## CAR_TYPEVan                    222.2604   759.6910   0.293   0.7699
## RED_CARyes                    -142.7977   497.6165  -0.287   0.7742
## CLM_FREQ                       -23.0045   238.0563  -0.097   0.9230
## REVOKEDYes                    -884.6297   492.6599  -1.796   0.0727 .
## MVR_PTS                        124.8456    70.1147   1.781   0.0751 .
## CAR_AGE                        -85.2471    44.0676  -1.934   0.0532 .
## URBANICITYUrban                 49.6246   757.4807   0.066   0.9478
## YOJ_Y1                        1264.6493  1702.4529   0.743   0.4577
## INCOME_Y1                     1389.6837  3196.6231   0.435   0.6638
## HOME_VAL_Y1                  -7324.2326  8042.1610  -0.911   0.3625
## OLDCLAIM_Y1                   -883.6704  2438.2781  -0.362   0.7171
## INCOME_LOG                    -262.1702   318.5151  -0.823   0.4105
## HOME_VAL_LOG                   666.1726   665.9492   1.000   0.3173
## BLUEBOOK_LOG                  1398.2887   329.3060   4.246 2.27e-05 ***
## OLDCLAIM_LOG                    96.8219   268.1332   0.361   0.7181
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 7688 on 2112 degrees of freedom
## Multiple R-squared:  0.03247,    Adjusted R-squared:  0.01414
## F-statistic: 1.772 on 40 and 2112 DF,  p-value: 0.002143
```

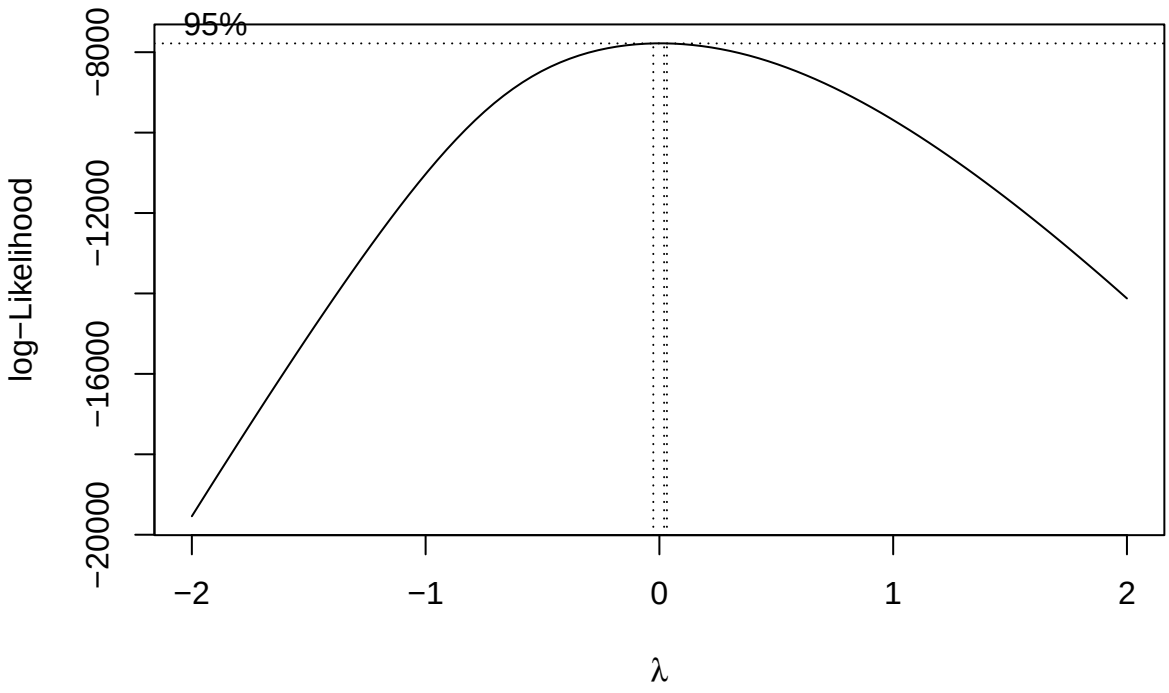
```
par(mfrow=c(2,2))
plot(lm_full)
```



Additioanl Transformation

The plots show that there is a non-linear relationship between the response variable and the predictors. Let's see what transformation Box-Cox would suggest for the response variable.

```
bc <- boxcox(lm_full)
```




```
lambda <- bc$x[which.max(bc$y)]
lambda
```

```
## [1] 0.02020202
```

It result indicates a log-transformation is appropriate.

```
lm_train_df$TARGET_AMT_LOG <- log(lm_train_df$TARGET_AMT)
lm_train_df$TARGET_AMT <- NULL
```

Buidling Models

Full Model

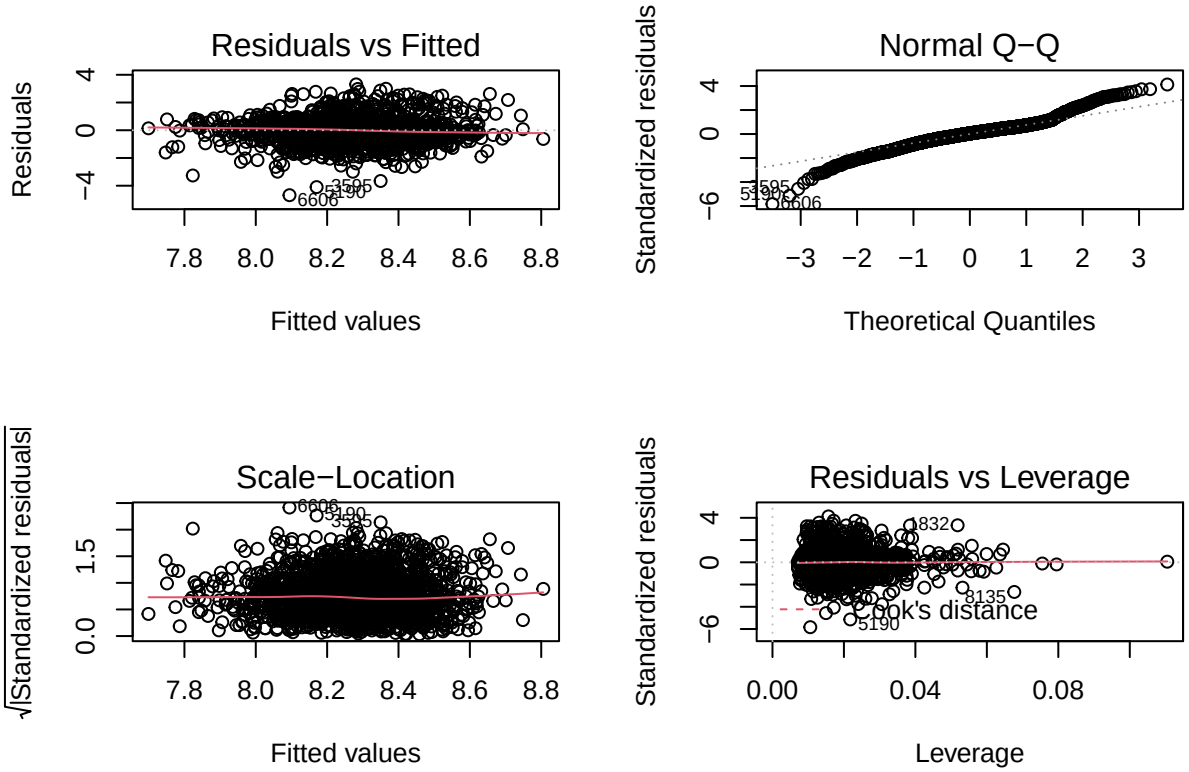
First we build a full model with all predictors

```
lm_full <- lm(TARGET_AMT_LOG~., lm_train_df)
summary(lm_full)
```

```
##
## Call:
## lm(formula = TARGET_AMT_LOG ~ ., data = lm_train_df)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6841 -0.4071  0.0392  0.4100  3.3047
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   6.3958408  0.3691039  17.328  < 2e-16 ***
## KIDSDRIV                     -0.0366703  0.0333626  -1.099   0.2718
## AGE                           0.0021320  0.0022824   0.934   0.3503
## HOMEKIDS                      0.0285277  0.0225095   1.267   0.2052
## YOJ                          -0.0087757  0.0076187  -1.152   0.2495
## PARENT1Yes                    0.0266487  0.0619249   0.430   0.6670
## MSTATUSYes                   -0.1040317  0.0545619  -1.907   0.0567 .
## SEXM                          0.0925483  0.0665416   1.391   0.1644
## EDUCATIONHigh School          0.0456222  0.0540472   0.844   0.3987
## EDUCATIONLess ThanHigh School 0.0455763  0.0684784   0.666   0.5058
## EDUCATIONMasters              0.1198844  0.0856325   1.400   0.1617
## EDUCATIONPhD                  0.2604448  0.1148665   2.267   0.0235 *
## JOBClerical                   0.0067850  0.0618786   0.110   0.9127
## JOBDoctor                    -0.1788178  0.1635914  -1.093   0.2745
## JOBHome Maker                -0.0712698  0.1077388  -0.662   0.5084
## JOBLawyer                    -0.0059555  0.1114274  -0.053   0.9574
## JOBManager                   -0.0497520  0.0887320  -0.561   0.5751
## JOBProfessional               0.0330490  0.0705567   0.468   0.6395
## JOBStudent                    0.0205032  0.0976863   0.210   0.8338
## TRAVTIME                     -0.0004997  0.0011634  -0.430   0.6676
## CAR_USEPrivate               -0.0049188  0.0536730  -0.092   0.9270
## TIF                          -0.0016968  0.0044647  -0.380   0.7039
## CAR_TYPEPanel Truck           0.0270259  0.0919137   0.294   0.7688
## CAR_TYPEPickup                0.0277256  0.0626777   0.442   0.6583
## CAR_TYPESports Car            0.0721836  0.0773073   0.934   0.3506
## CAR_TYPESUV                   0.0905003  0.0676504   1.338   0.1811
## CAR_TYPEVan                  -0.0277286  0.0797541  -0.348   0.7281
## RED_CARyes                    0.0283447  0.0522409   0.543   0.5875
## CLM_FREQ                     -0.0459350  0.0249917  -1.838   0.0662 .
## REVOKEDYes                   -0.0615658  0.0517206  -1.190   0.2340
## MVR_PTS                       0.0146954  0.0073608   1.996   0.0460 *
## CAR_AGE                      -0.0014951  0.0046263  -0.323   0.7466
## URBANICITYUrban               0.0508973  0.0795221   0.640   0.5222
## YOJ_Y1                        0.0407488  0.1787275   0.228   0.8197
## INCOME_Y1                     0.2357074  0.3355889   0.702   0.4825
## HOME_VAL_Y1                   0.2861214  0.8442848   0.339   0.7347
## OLDCLAIM_Y1                  -0.1428745  0.2559761  -0.558   0.5768
## INCOME_LOG                   -0.0200451  0.0334385  -0.599   0.5489
## HOME_VAL_LOG                 -0.0185913  0.0699129  -0.266   0.7903
## BLUEBOOK_LOG                  0.1755319  0.0345713   5.077 4.16e-07 ***
## OLDCLAIM_LOG                  0.0252454  0.0281492   0.897   0.3699
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8071 on 2112 degrees of freedom
## Multiple R-squared:  0.03167,    Adjusted R-squared:  0.01333
## F-statistic: 1.727 on 40 and 2112 DF,  p-value: 0.003273
```

```
par(mfrow=c(2,2))
plot(lm_full)
```



The residual plots show that the relationship is now linear. The only problem is that the distribution of the residuals is over-dispersed and not normal. Since the optimal transformation suggested by Box-Cox would not fix this problem, a GLM regression would be more appropriate to fit the data in this case. As requested by this assignment, we would keep the linear models. Since the normality of the residuals is violated, we would not judge the significance of the coefficient by the t-values. We will compare the performance of different models by the adjusted R-squared and the Root of Mean Square Errors.

Backward Elinmination By AIC

Starting with our full model, perform backward elimination by comparing the **AIC** of the models.

```
lm_AIC <- step(lm_full, trace = 0)
summary(lm_AIC)
```

```
##
## Call:
## lm(formula = TARGET_AMT_LOG ~ MSTATUS + SEX + CLM_FREQ + MVR_PTS +
##     BLUEBOOK_LOG, data = lm_train_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6994 -0.4043  0.0425  0.4106  3.2129
```

```
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.807358   0.248943  27.345  < 2e-16 ***
## MSTATUSYes   -0.073271   0.034682  -2.113   0.0347 *
## SEXM          0.054122   0.034990   1.547   0.1221
## CLM_FREQ     -0.022761   0.014548  -1.565   0.1178
## MVR_PTS       0.017331   0.007044   2.460   0.0140 *
## BLUEBOOK_LOG  0.156247   0.026314   5.938 3.36e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.804 on 2147 degrees of freedom
## Multiple R-squared:  0.02314,    Adjusted R-squared:  0.02087
## F-statistic: 10.17 on 5 and 2147 DF,  p-value: 1.201e-09
```

Backward Elinmination By BIC

Starting with our full model, perform backward elimination by comparing the **BIC** of the models.

```
lm_BIC <- step(lm_full, trace = 0, k = log(nrow(lm_train_df)))
summary(lm_BIC)
```

```
##
## Call:
## lm(formula = TARGET_AMT_LOG ~ BLUEBOOK_LOG, data = lm_train_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7888 -0.3907  0.0425  0.3914  3.2417
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.77803    0.24679  27.465  < 2e-16 ***
## BLUEBOOK_LOG  0.15976    0.02626   6.085 1.38e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8058 on 2151 degrees of freedom
## Multiple R-squared:  0.01692,    Adjusted R-squared:  0.01646
## F-statistic: 37.02 on 1 and 2151 DF,  p-value: 1.377e-09
```

Model with only characteristics of the cars

The model produced by backward elimination comparing BIC showed that the book value of the car is the most important factor determining the claim value. It is reasonable that the claim value is highly related to the car's value, but what about the other characteristics of the cars?
Let's build a model with only the characteristics of the cars, which means the behaviors of the drivers are not considered.

```
lm_car <- lm(TARGET_AMT_LOG~CAR_USE+BLUEBOOK_LOG+CAR_TYPE+RED_CAR+CAR_AGE, data = lm_train_df)
summary(lm_car)
```

```
##
## Call:
## lm(formula = TARGET_AMT_LOG ~ CAR_USE + BLUEBOOK_LOG + CAR_TYPE +
##     RED_CAR + CAR_AGE, data = lm_train_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7803 -0.4007  0.0391  0.3961  3.2331
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          6.770e+00  3.033e-01  22.323  < 2e-16 ***
## CAR_USEPrivate       6.668e-03  4.139e-02   0.161    0.872
## BLUEBOOK_LOG         1.557e-01  3.151e-02   4.941 8.38e-07 ***
## CAR_TYPEPanel Truck  7.093e-02  8.474e-02   0.837    0.403
## CAR_TYPEPickup       3.613e-02  6.079e-02   0.594    0.552
## CAR_TYPESports Car   2.133e-02  6.679e-02   0.319    0.749
## CAR_TYPESUV          3.486e-02  5.708e-02   0.611    0.541
## CAR_TYPEVan          1.624e-03  7.599e-02   0.021    0.983
## RED_CARyes           5.498e-02  4.631e-02   1.187    0.235
## CAR_AGE             -6.747e-05  3.230e-03  -0.021    0.983
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8068 on 2143 degrees of freedom
## Multiple R-squared:  0.0183, Adjusted R-squared:  0.01417
## F-statistic: 4.438 on 9 and 2143 DF,  p-value: 8.849e-06
```

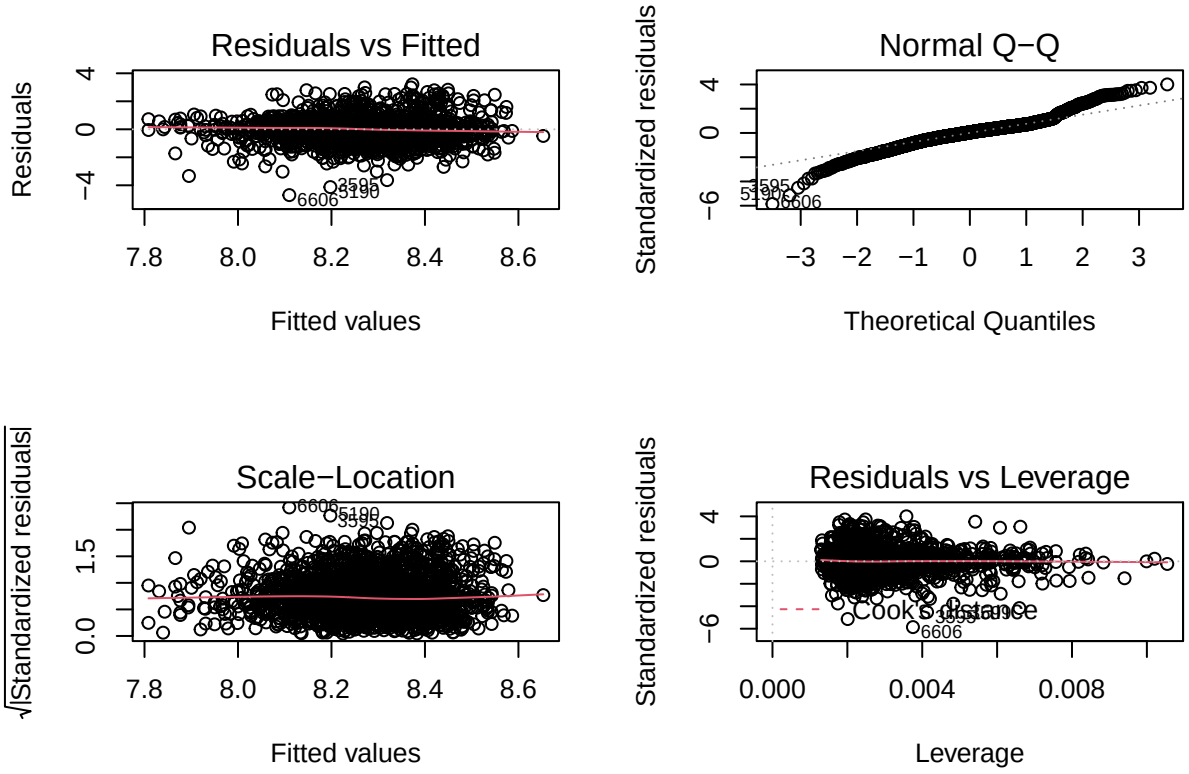
Model Selection

```
lm_models <- data.frame(model=c(""),Num_of_Coefficients=c(0),
                        R_squared_adj=c(0.0000), RMSE=c(0.0000))
models <- list(lm_full, lm_AIC, lm_BIC, lm_car)
model_names <- c("lm_full", "lm_AIC", "lm_BIC", "lm_car")
for (i in c(1:length(models))) {
    lm_models[i,"model"] <- model_names[i]
    lm_models[i,"Num_of_Coefficients"] <- length(models[[i]]$coefficients) - 1
    lm_models[i,"R_squared_adj"] <- summary(models[[i]])$adj.r.squared
    lm_models[i,"RMSE"] <- sqrt(mean(models[[i]]$residuals^2))
}
lm_models
```

```
##     model Num_of_Coefficients R_squared_adj      RMSE
## 1 lm_full                  40    0.01332863 0.7994062
## 2  lm_AIC                   5    0.02086660 0.8029181
## 3  lm_BIC                   1    0.01646443 0.8054703
## 4  lm_car                   9    0.01417248 0.8049073
```

Model lm_AIC has the highest adjusted R-squared and the RMSE is very close to the full model. Our optimal linear model is lm_AIC. The performance of the model using only characteristics of the cars is the lowest. The behaviors of the driver do affect the amount of a claim.

```
par(mfrow=c(2,2))
plot(lm_AIC)
```



Double checking the diagnostic plots of lm_AIC. There is no serious problem except the non-normal residuals.
The coefficients are unbiased since the relationship is linear and the residuals are independent.
To check the significance of the coefficients, bootstrap simulation may be used but we won't go further in this analysis.

Evaluation Data Prediction

Data Clean Up and Transformation

```
test_df$INDEX <- NULL
test_df$INCOME <- as.numeric(gsub('[$,]', '', test_df$INCOME))
test_df$HOME_VAL <- as.numeric(gsub('[$,]', '', test_df$HOME_VAL))
test_df$BLUEBOOK <- as.numeric(gsub('[$,]', '', test_df$BLUEBOOK))
test_df$OLDCLAIM <- as.numeric(gsub('[$,]', '', test_df$OLDCLAIM))
test_df$PARENT1 <- gsub("z_", "", test_df$PARENT1)
test_df$MSTATUS <- gsub("z_", "", test_df$MSTATUS)
test_df$SEX <- gsub("z_", "", test_df$SEX)
test_df$EDUCATION <- gsub("z_", "", test_df$EDUCATION)
test_df$EDUCATION <- gsub("<", "Less Than", test_df$EDUCATION)
```

```
test_df$JOB <- gsub("z_", "", test_df$JOB)
test_df$CAR_TYPE <- gsub("z_", "", test_df$CAR_TYPE)
test_df$URBANICITY <- ifelse(test_df$URBANICITY == "Highly Urban/ Urban", "Urban","Rural")

test_df$JOB[test_df$JOB == ""] <- NA

test_df[c("TARGET_FLAG","PARENT1","MSTATUS", "SEX", "EDUCATION", "JOB","CAR_TYPE",
          "RED_CAR", "URBANICITY", "CAR_USE","REVOKED")] <-
              lapply(test_df[c("TARGET_FLAG","PARENT1","MSTATUS", "SEX",
                               "EDUCATION", "JOB","CAR_TYPE", "RED_CAR",
                               "URBANICITY", "CAR_USE","REVOKED")], factor)
```

```
test_df$TARGET_FLAG <- NULL
test_df$TARGET_AMT <- NULL

test_df <- mice.reuse(mickey, test_df, maxit = 5, printFlag = FALSE, seed = 2022)[[1]]
```

```
summary(test_df)
```

```
##     KIDSDRIV           AGE           HOMEKIDS           YOJ
##  Min.   :0.0000   Min.   :17.00   Min.   :0.0000   Min.   : 0.00
##  1st Qu.:0.0000   1st Qu.:39.00   1st Qu.:0.0000   1st Qu.: 9.00
##  Median :0.0000   Median :45.00   Median :0.0000   Median :11.00
##  Mean   :0.1625   Mean   :45.01   Mean   :0.7174   Mean   :10.36
##  3rd Qu.:0.0000   3rd Qu.:51.00   3rd Qu.:1.0000   3rd Qu.:13.00
##  Max.   :3.0000   Max.   :73.00   Max.   :5.0000   Max.   :19.00
##
##      INCOME       PARENT1       HOME_VAL      MSTATUS    SEX
##  Min.   :     0   No :1875   Min.   :     0   No : 847   F:1170
##  1st Qu.: 25632   Yes: 266   1st Qu.:     0   Yes:1294   M: 971
##  Median : 51524              Median :158432
##  Mean   : 60092              Mean   :153341
##  3rd Qu.: 85727              3rd Qu.:237062
##  Max.   :291182              Max.   :669271
##
##                 EDUCATION             JOB         TRAVTIME
##  Bachelors           :581   Blue Collar :474   Min.   :  5.00
##  High School         :622   Clerical    :323   1st Qu.: 22.00
##  Less ThanHigh School:312   Manager     :319   Median : 33.00
##  Masters             :420   Professional:299   Mean   : 33.15
##  PhD                 :206   Lawyer      :241   3rd Qu.: 43.00
##                             Home Maker  :203   Max.   :105.00
##                             (Other)     :282
##        CAR_USE        BLUEBOOK          TIF                CAR_TYPE
##  Commercial: 760   Min.   : 1500   Min.   : 1.000   Minivan    :549
##  Private   :1381   1st Qu.: 8870   1st Qu.: 1.000   Panel Truck:177
##                    Median :14170   Median : 4.000   Pickup     :383
##                    Mean   :15469   Mean   : 5.245   Sports Car :272
##                    3rd Qu.:21050   3rd Qu.: 7.000   SUV        :589
##                    Max.   :49940   Max.   :25.000   Van        :171
##
##  RED_CAR       OLDCLAIM        CLM_FREQ     REVOKED       MVR_PTS
##  no :1543   Min.   :    0   Min.   :0.000   No :1880   Min.   : 0.000
```

```
##  yes: 598   1st Qu.:    0   1st Qu.:0.000   Yes: 261   1st Qu.: 0.000
##             Median :    0   Median :0.000              Median : 1.000
##             Mean   : 4022   Mean   :0.809              Mean   : 1.766
##             3rd Qu.: 4718   3rd Qu.:2.000              3rd Qu.: 3.000
##             Max.   :54399   Max.   :5.000              Max.   :12.000
##
##     CAR_AGE       URBANICITY
##  Min.   : 0.000   Rural: 403
##  1st Qu.: 1.000   Urban:1738
##  Median : 8.000
##  Mean   : 8.181
##  3rd Qu.:13.000
##  Max.   :26.000
##
```

```
test_df$YOJ_Y <- as.factor(ifelse(test_df$YOJ == 0,0,1))
test_df$INCOME_Y <- as.factor(ifelse(test_df$INCOME == 0,0,1))
test_df$HOME_VAL_Y <- as.factor(ifelse(test_df$HOME_VAL == 0,0,1))
test_df$OLDCLAIM_Y <- as.factor(ifelse(test_df$OLDCLAIM == 0,0,1))

test_df$INCOME_LOG <- log(test_df$INCOME+1)
test_df$HOME_VAL_LOG <- log(test_df$HOME_VAL+1)
test_df$BLUEBOOK_LOG <- log(test_df$BLUEBOOK)
test_df$OLDCLAIM_LOG <- log(test_df$OLDCLAIM+1)

test_df$INCOME <- NULL
test_df$HOME_VAL <- NULL
test_df$BLUEBOOK <- NULL
test_df$OLDCLAIM <- NULL

logistic_test_df <- test_df
```

```
logistic_test_df$AGE_Squared <- logistic_test_df$AGE^2
```

```
logistic_test_df$KIDSDRIV_0.5 <- (logistic_test_df$KIDSDRIV)^0.5
logistic_test_df$HOMEKIDS_0.5 <- (logistic_test_df$HOMEKIDS)^0.5
logistic_test_df$MVR_PTS_3 <- (logistic_test_df$MVR_PTS)^3
logistic_test_df$TRAVTIME_0.33 <- (logistic_test_df$TRAVTIME)^0.33

logistic_test_df$KIDSDRIV <- NULL
logistic_test_df$HOMEKIDS <- NULL
logistic_test_df$MVR_PTS <- NULL
logistic_test_df$TRAVTIME <- NULL
```

Claim classification Prediction

```
logistic_test_df$TARGET_FLAG <-
  ifelse(predict(logi_AIC,logistic_test_df, type="response")>
           logi_models[2,"Optimal_Threhold"],1,0)
```

```
test_predict <- logistic_test_df$TARGET_FLAG
train_predict <- ifelse(logi_AIC$fitted.values>logi_models[2,"Optimal_Threhold"],1,0)
```

```
dist_df <- data.frame(rbind(
      cbind(train_predict,"train_predict"),
      cbind(test_predict,"test_predict")
      ))
colnames(dist_df) <- c("value","data")
dist_df <- table(dist_df)
dist_df[,1] <- dist_df[,1]/sum(dist_df[,1])
dist_df[,2] <- dist_df[,2]/sum(dist_df[,2])
dist_df
```

```
##      data
## value test_predict train_predict
##     0    0.5833723     0.6013969
##     1    0.4166277     0.3986031
```

The model produces similar result for both the training and testing data. Around 60% of the cases are classified as no crash and 40% of the cases are classified as having a crash. Our logistic model has similar performance for predicting unseen results.

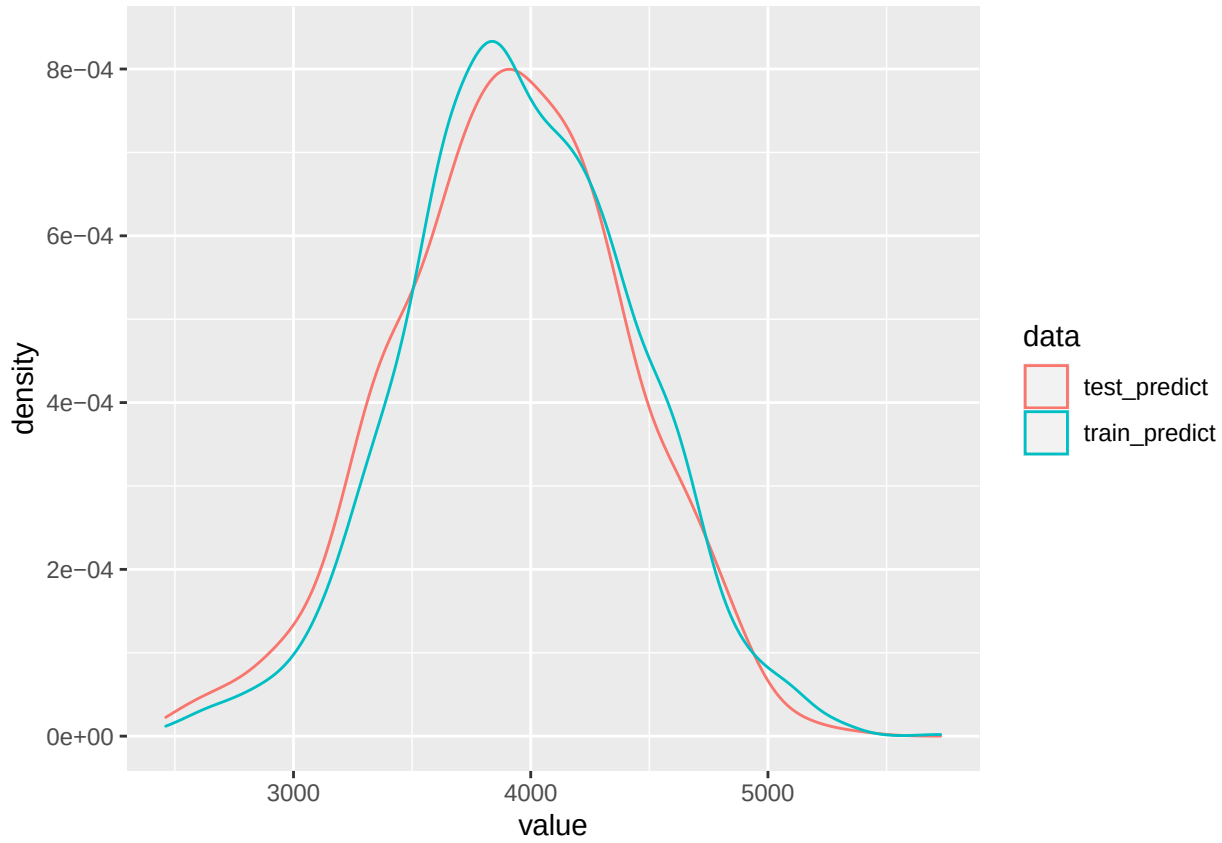
Claim Amount Prediction

```
lm_test_df <- test_df
lm_test_df$TARGET_FLAG <- NULL
```

```
lm_test_df$TARGET_AMT <- predict(lm_AIC,lm_test_df)
lm_test_df$TARGET_AMT <- exp(lm_test_df$TARGET_AMT)
```

```
test_df$TARGET_FLAG <- logistic_test_df$TARGET_FLAG
test_df$TARGET_AMT <- lm_test_df$TARGET_AMT * logistic_test_df$TARGET_FLAG
```

```
train_predict <- exp(lm_AIC$fitted.values)
test_predict <- test_df$TARGET_AMT[test_df$TARGET_AMT > 0]

dist_df <- data.frame(rbind(
      cbind(train_predict,"train_predict"),
      cbind(test_predict,"test_predict")
      ),stringsAsFactors=FALSE)
colnames(dist_df) <- c("value","data")
dist_df$value <- as.numeric(dist_df$value)


ggplot(dist_df, aes(x=value, color=data)) +
  geom_density()
```



The prediction of claim amounts have similar distributions for the training and testing data. Our linear model has stable performance in predicting unseen results.