Critical Thinking Group 4: DATA621 Homework 4

Table of Contents

[TEAM Members: 2](#_Toc38761704)

[**Overview** 2](#_Toc38761705)

[**Deliverables** 3](#_Toc38761706)

[**Data Exploration** 3](#_Toc38761707)

[**Data Preparation** 6](#_Toc38761708)

[**Build Models** 7](#_Toc38761709)

[**Select Models** 8](#_Toc38761710)

[**Make Predictions** 9](#_Toc38761711)

[**Appendix** 9](#_Toc38761712)

## TEAM Members:

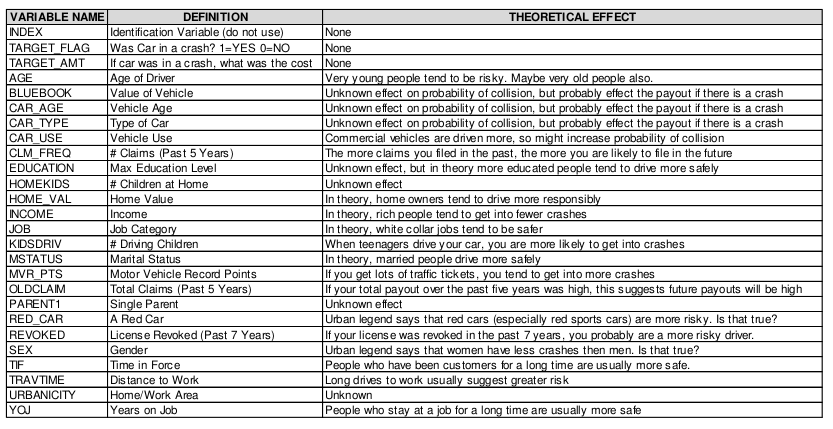
Rajwant Mishra  
Priya Shaji  
Debabrata Kabiraj  
Isabel Ramesar  
Sin Ying Wong  
Fan Xu

-------------------------------------------------

**Overview**

In this homework assignment, you 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.

Your 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:

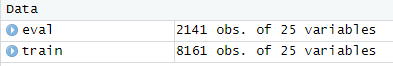


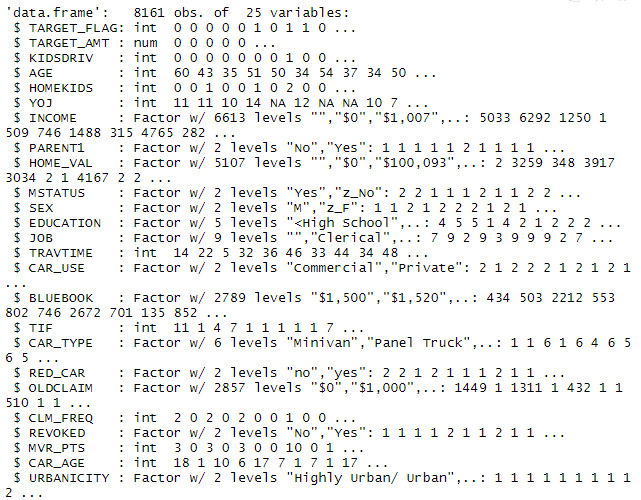
**Deliverables**

A write-up of your solutions submitted in PDF format. Assigned prediction (probabilities, classifications) for the evaluation dataset. Use 0.5 threshold.

**Data** **Exploration**

The first step we did was to import the data from GitHub, remove the index and look at the structure of the data.





We removed special characters then converted variables to numbers for both the Training and Evaluation data.

```{r}

train$INCOME<-gsub("[\\$,]", "", train$INCOME)

train$HOME\_VAL<-gsub("[\\$,]", "", train$HOME\_VAL)

train$BLUEBOOK<-gsub("[\\$,]", "", train$BLUEBOOK)

train$OLDCLAIM<-gsub("[\\$,]", "",train$OLDCLAIM)

eval$INCOME<-gsub("[\\$,]", "", eval$INCOME)

eval$HOME\_VAL<-gsub("[\\$,]", "", eval$HOME\_VAL)

eval$BLUEBOOK<-gsub("[\\$,]", "", eval$BLUEBOOK)

eval$OLDCLAIM<-gsub("[\\$,]", "",eval$OLDCLAIM)

train$INCOME<-as.numeric(train$INCOME)

train$HOME\_VAL<-as.numeric(train$HOME\_VAL)

train$BLUEBOOK<-as.numeric(train$BLUEBOOK)

train$OLDCLAIM<-as.numeric(train$OLDCLAIM)

eval$INCOME<-as.numeric(eval$INCOME)

eval$HOME\_VAL<-as.numeric(eval$HOME\_VAL)

eval$BLUEBOOK<-as.numeric(eval$BLUEBOOK)

eval$OLDCLAIM<-as.numeric(eval$OLDCLAIM)

```

We then split the training data into a train and test data set.

```{r}

set.seed(123)

sample <- sample.split(train,SplitRatio = 0.80)

train <- subset(train, sample == TRUE)

test <- subset(train, sample == FALSE)

```

We then ran the summary for ‘Train’ as follows:

```{r}

summary(train)

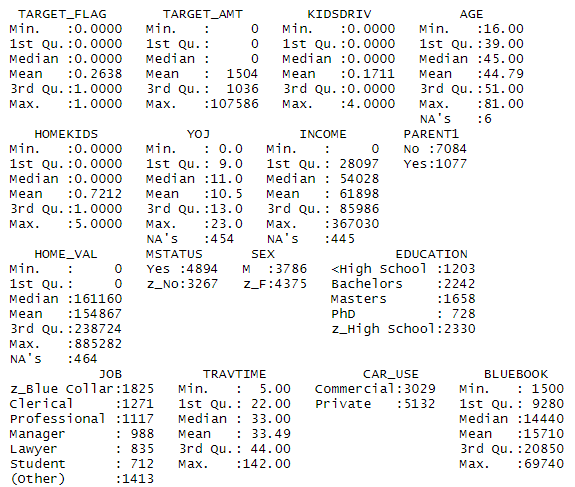
status <- df\_status(train, print\_results = TRUE)

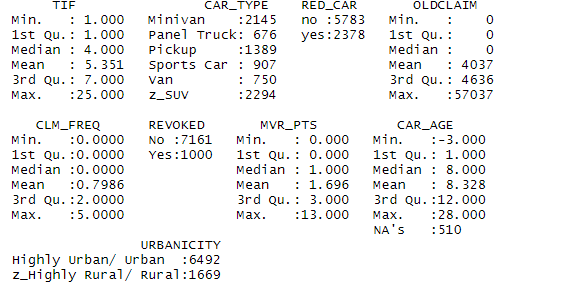
filter(status, p\_zeros > 60) %>% .$variable

train2 <- select(train, -c(KIDSDRIV,HOMEKIDS,OLDCLAIM,CLM\_FREQ))

test <- select(test, -c(KIDSDRIV,HOMEKIDS,OLDCLAIM,CLM\_FREQ))

```





```{r}

freq(train2)

```

We then determine the skewness and kurtosis of the data.

```{r}

plot\_num(train2)

profiling\_num(train2)

```

**Data Preparati****on**

The missing NA values were imputed with the median using the Hmisc package:

```{r}

train2$AGE<-impute(train2$AGE, median)

train2$YOJ<-impute(train2$YOJ, median)

train2$INCOME<-impute(train2$INCOME, median)

train2$CAR\_AGE<-impute(train2$CAR\_AGE, median)

eval$AGE<-impute(eval$AGE, median)

eval$YOJ<-impute(eval$YOJ, median)

eval$INCOME<-impute(eval$INCOME, median)

eval$CAR\_AGE<-impute(eval$CAR\_AGE, median)

```

We created new variable which is PTSAGE = MVR\_PTS/AGE.

```{r}

train2$PTSAGE <- train2$MVR\_PTS/train2$AGE

test$PTSAGE <- test$MVR\_PTS/test$AGE

train2 <- select(train2, -c(MVR\_PTS,AGE))

test <- select(test, -c(MVR\_PTS,AGE))

```

**Build Models**

Predicting car crash

In the model, we selected the following variables.

```{r}

model1 = glm(TARGET\_FLAG ~ YOJ + INCOME + PARENT1 + HOME\_VAL + MSTATUS + SEX + EDUCATION + JOB + TRAVTIME + CAR\_USE + TIF + CAR\_TYPE + RED\_CAR + REVOKED + URBANICITY + PTSAGE,data = train2, family = 'binomial')

summary(model1)

```

However, we removed variables that deemed insufficient.

```{r}

model2 = glm(TARGET\_FLAG ~ INCOME + PARENT1 + HOME\_VAL + MSTATUS + EDUCATION + TRAVTIME + CAR\_USE + TIF + CAR\_TYPE + REVOKED + URBANICITY + PTSAGE, data = train2, family = 'binomial')

summary(model2)

```

After removing the unnecessary variables, all coefficients fall in line with their theoretical effects.

The model has a majority of the variables with significant p-values, with the exception of 2 categories of education (high school) and car type (truck). All of the coefficients of the variables also fall in line with theoretical effects.

Amount Predicted

```{r}

train2\_claims = train2 %>% filter(TARGET\_FLAG == 1)

test\_claims = test %>% filter(TARGET\_FLAG == 1)

linearmodel1 = lm(TARGET\_AMT ~ .-TARGET\_FLAG, data = train2\_claims)

summary(linearmodel1)

```

A lot of the variables are insignificant so we will limit the variables in the next model to make it more significant.

```{r}

linearmodel2 = lm(TARGET\_AMT ~ MSTATUS + BLUEBOOK + CAR\_AGE, data = train2\_claims)

summary(linearmodel2)

```

The coefficients are in line with theoretical effects in this model.

**Select Models**

Linear Models

```{r linearmodel1 plots}

par(mfrow = c(2,2))

plot(linearmodel1)

```

```{r linearmodel2 plots}

par(mfrow = c(2,2))

plot(linearmodel2)

```

```{r linearmodel2 plots}

par(mfrow = c(2,2))

plot(linearmodel2)

```

```{r mse}

amt = test\_claims$TARGET\_AMT

summary(test\_claims)

as.matrix(c(mean((amt - predict.lm(linearmodel1, newdata = test\_claims))^2, na.rm = TRUE), mean((amt - predict.lm(linearmodel2, newdata = test\_claims))^2, na.rm = TRUE), mean((amt - predict.lm(linearmodel2, newdata = test\_claims))^2, na.rm = TRUE)))

```

Logit Models

```{r}

anova(model1, test = 'Chisq')

anova(model2, test = 'Chisq')

```

```{r}

pR2(model1);

pR2(model2);

```

**Make Predictions**

```{r}

fitted.results = predict(model2, test, type = 'response')

fitted.results = ifelse(fitted.results > 0.5, 1, 0)

misClasificError = mean(fitted.results != test$TARGET\_FLAG, na.rm = TRUE)

print(paste('Accurancy', round(1-misClasificError, 3)))

```

```{r}

eval$PTSAGE = eval$MVR\_PTS/eval$AGE

summary(eval)

eval\_results = predict(model2, eval, type = 'response')

eval\_results = ifelse(eval\_results > 0.5, 1, 0)

eval\_amt = predict(linearmodel2, eval)

```

**Appendix**

<https://github.com/Rajwantmishra/DATA621_CR4/blob/master/HW4/Homework4_Final.Rmd>

**Thank you**