Exposition

We've got about 8k observations concerning car insurance customer attributes. We will be using this data to predict two target variables:

TARGET_FLAG, a binary value indicating if the customer crashed their vehicle,

TARGET_AMT, a value equal to 0 if TARGET_FLAG is zero, else it is positive.

To accomplish this task, we will be training multiple linear regression and binary logistic regression models. First, we will use a function to clean the data and explicitly declare data types on all variables. Then, we will impute the missing information before training the models.

VARIABLE NAME	DEFINITION	THEORETICAL EFFECT
INDEX	Identification Variable (do not use)	None
TARGET_FLAG	Was Car in a crash? 1=YES 0=NO	None
TARGET_AMT	If car was in a crash, what was the cost	None
AGE	Age of Driver	Very young people tend to be risky. Maybe very old people also.
BLUEBOOK	Value of Vehicle	Unknown effect on probability of collision, but probably effect the payout if there is a crash
CAR_AGE	Vehicle Age	Unknown effect on probability of collision, but probably effect the payout if there is a crash
CAR_TYPE	Type of Car	Unknown effect on probability of collision, but probably effect the payout if there is a crash
CAR_USE	Vehicle Use	Commercial vehicles are driven more, so might increase probability of collision
CLM_FREQ	# Claims (Past 5 Years)	The more claims you filed in the past, the more you are likely to file in the future
EDUCATION	Max Education Level	Unknown effect, but in theory more educated people tend to drive more safely
HOMEKIDS	# Children at Home	Unknown effect
HOME_VAL	Home Value	In theory, home owners tend to drive more responsibly
INCOME	Income	In theory, rich people tend to get into fewer crashes
JOB	Job Category	In theory, white collar jobs tend to be safer
KIDSDRIV	# Driving Children	When teenagers drive your car, you are more likely to get into crashes
MSTATUS	Marital Status	In theory, married people drive more safely
MVR_PTS	Motor Vehicle Record Points	If you get lots of traffic tickets, you tend to get into more crashes
OLDCLAIM	Total Claims (Past 5 Years)	If your total payout over the past five years was high, this suggests future payouts will be high
PARENT1	Single Parent	Unknown effect
RED_CAR	A Red Car	Urban legend says that red cars (especially red sports cars) are more risky. Is that true?
REVOKED	License Revoked (Past 7 Years)	If your license was revoked in the past 7 years, you probably are a more risky driver.
SEX	Gender	Urban legend says that women have less crashes then men. Is that true?
TIF	Time in Force	People who have been customers for a long time are usually more safe.
TRAVTIME	Distance to Work	Long drives to work usually suggest greater risk
URBANICITY	Home/Work Area	Unknown
YOJ	Years on Job	People who stay at a job for a long time are usually more safe

Data Exploration

```
summary(df.t)
```

```
##  TARGET_FLAG    TARGET_AMT        KIDSDRIV           AGE
##  FALSE:6008   Min.   :     0   Min.   :0.0000   Min.   :16.00
##  TRUE :2153   1st Qu.:     0   1st Qu.:0.0000   1st Qu.:39.00
##               Median :     0   Median :0.0000   Median :45.00
##               Mean   :  1504   Mean   :0.1711   Mean   :44.79
##               3rd Qu.:  1036   3rd Qu.:0.0000   3rd Qu.:51.00
##               Max.   :107586   Max.   :4.0000   Max.   :81.00
##                                                 NA's   :6
##     HOMEKIDS           YOJ           INCOME        PARENT1
##  Min.   :0.0000   Min.   : 0.0   Min.   :     0   Mode :logical
##  1st Qu.:0.0000   1st Qu.: 9.0   1st Qu.: 28097   FALSE:7084
##  Median :0.0000   Median :11.0   Median : 54028   TRUE :1077
```

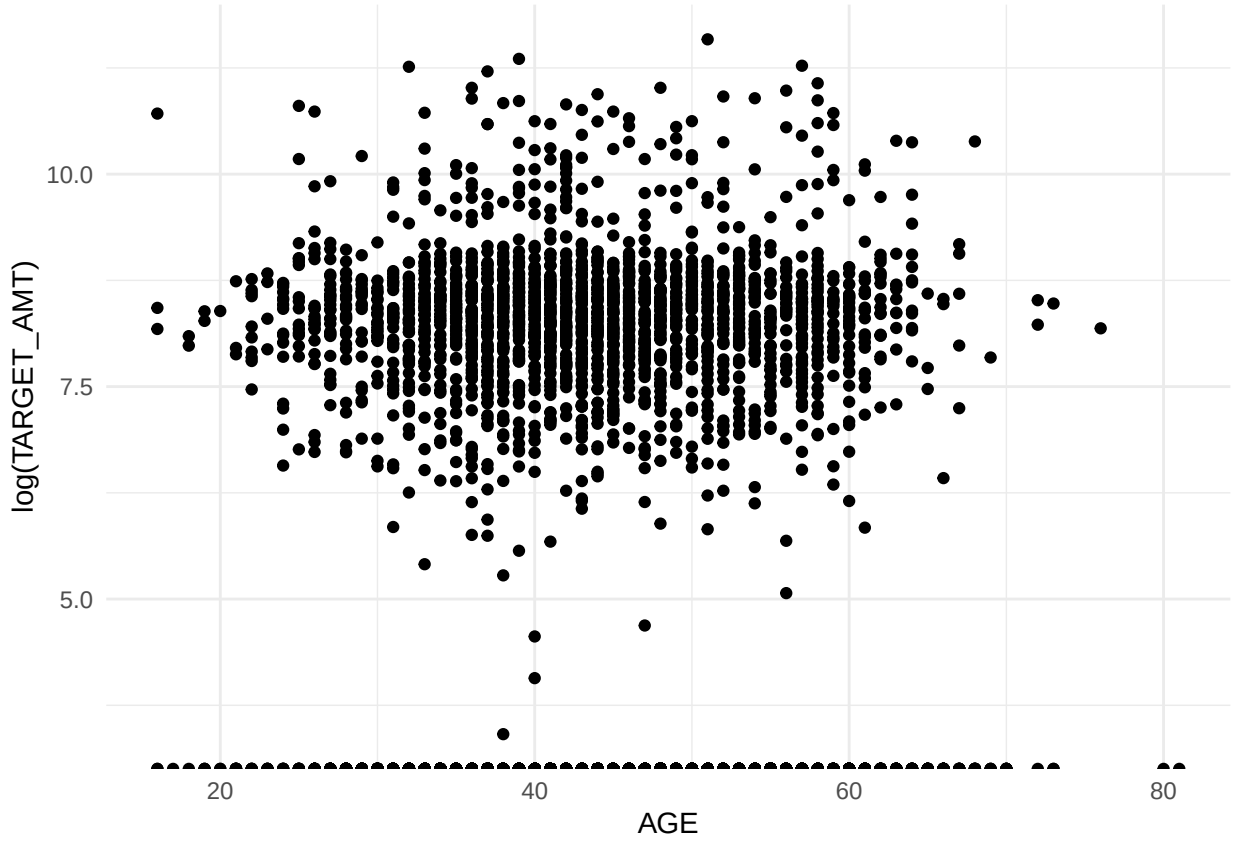
```
##  Mean   :0.7212   Mean   :10.5   Mean   : 61898
##  3rd Qu.:1.0000   3rd Qu.:13.0   3rd Qu.: 85986
##  Max.   :5.0000   Max.   :23.0   Max.   :367030
##                   NA's   :454    NA's   :445
##     HOME_VAL       MSTATUS           SEX          EDUCATION           JOB
##  Min.   :     0   Mode :logical   Mode :logical   0:1203    Blue Collar :1825
##  1st Qu.:     0   FALSE:3267      FALSE:4375      2:2242    Clerical    :1271
##  Median :161160   TRUE :4894      TRUE :3786      1:2330    Professional:1117
##  Mean   :154867                                   3:1658    Manager     : 988
##  3rd Qu.:238724                                   4: 728    Lawyer      : 835
##  Max.   :885282                                             Student     : 712
##  NA's   :464                                                (Other)     :1413
##     TRAVTIME       CAR_USE           BLUEBOOK          TIF
##  Min.   :  5.00   Mode :logical   Min.   : 1500   Min.   : 1.000
##  1st Qu.: 22.00   FALSE:3029      1st Qu.: 9280   1st Qu.: 1.000
##  Median : 33.00   TRUE :5132      Median :14440   Median : 4.000
##  Mean   : 33.49                   Mean   :15710   Mean   : 5.351
##  3rd Qu.: 44.00                   3rd Qu.:20850   3rd Qu.: 7.000
##  Max.   :142.00                   Max.   :69740   Max.   :25.000
##
##         CAR_TYPE     RED_CAR           OLDCLAIM        CLM_FREQ
##  Minivan    :2145   Mode :logical   Min.   :    0   Min.   :0.0000
##  Panel Truck: 676   FALSE:5783      1st Qu.:    0   1st Qu.:0.0000
##  Pickup     :1389   TRUE :2378      Median :    0   Median :0.0000
##  Sports Car : 907                   Mean   : 4037   Mean   :0.7986
##  SUV        :2294                   3rd Qu.: 4636   3rd Qu.:2.0000
##  Van        : 750                   Max.   :57037   Max.   :5.0000
##
##   REVOKED           MVR_PTS          CAR_AGE       URBANICITY
##  Mode :logical   Min.   : 0.000   Min.   :-3.000   Mode :logical
##  FALSE:7161      1st Qu.: 0.000   1st Qu.: 1.000   FALSE:1669
##  TRUE :1000      Median : 1.000   Median : 8.000   TRUE :6492
##                  Mean   : 1.696   Mean   : 8.328
##                  3rd Qu.: 3.000   3rd Qu.:12.000
##                  Max.   :13.000   Max.   :28.000
##                                   NA's   :510
```

```
colSums(is.na(df.t))
```

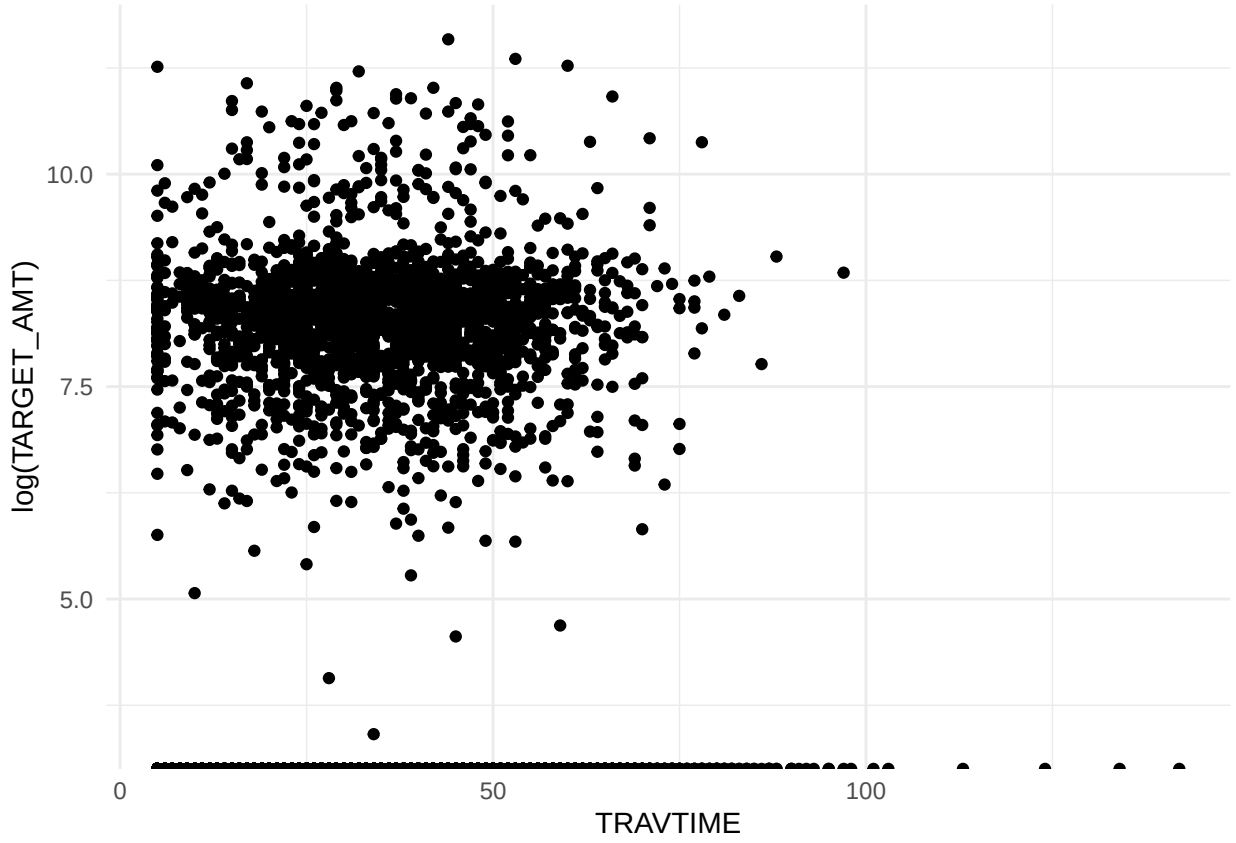
```
## TARGET_FLAG  TARGET_AMT    KIDSDRIV         AGE    HOMEKIDS         YOJ
##           0           0           0           6           0         454
##      INCOME     PARENT1    HOME_VAL     MSTATUS         SEX   EDUCATION
##         445           0         464           0           0           0
##         JOB    TRAVTIME     CAR_USE    BLUEBOOK         TIF    CAR_TYPE
##           0           0           0           0           0           0
##     RED_CAR    OLDCLAIM    CLM_FREQ     REVOKED     MVR_PTS     CAR_AGE
##           0           0           0           0           0         510
##  URBANICITY
##           0
```

```
ggplot(df.t, aes(x = AGE)) + geom_point(aes(y = log(TARGET_AMT))) + theme_minimal()
```

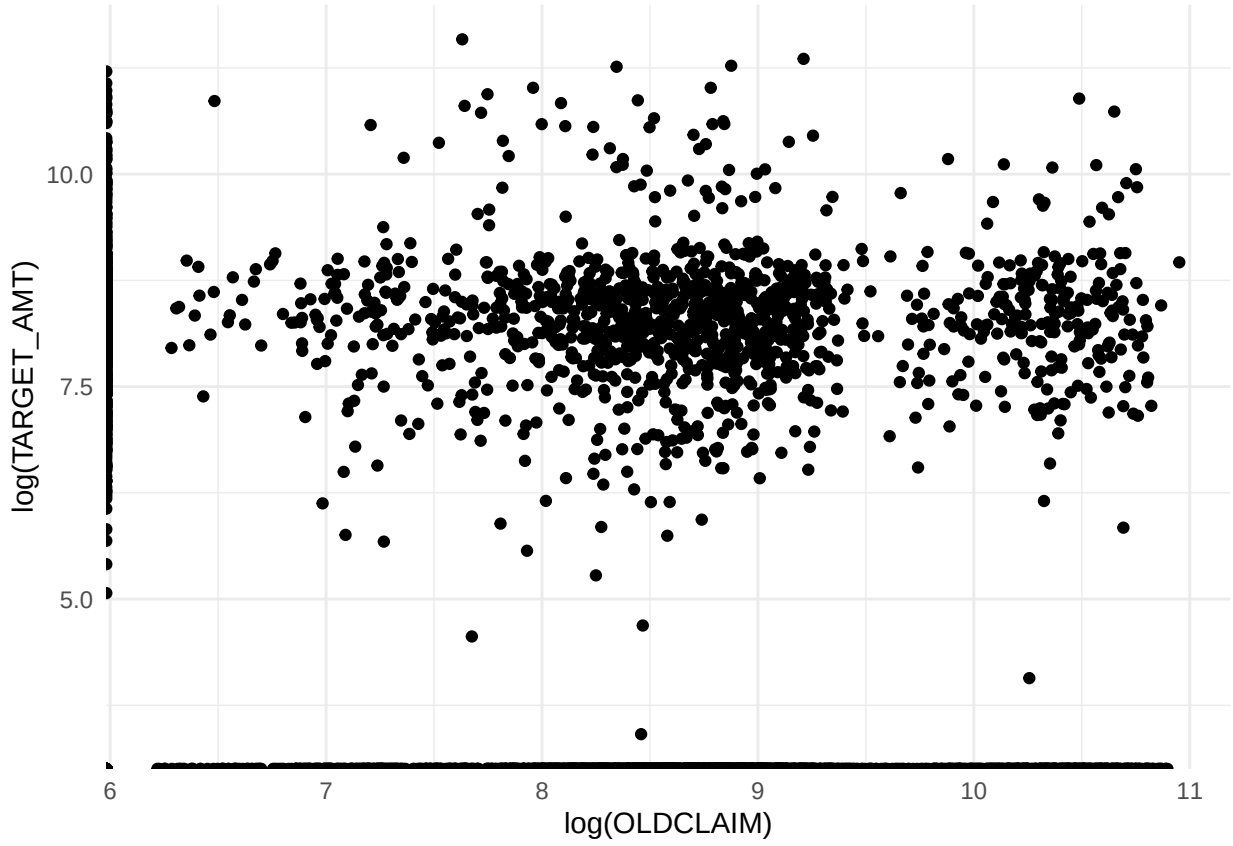
```
## Warning: Removed 6 rows containing missing values (geom_point).
```




```
ggplot(df.t, aes(x = TRAVTIME)) + geom_point(aes(y = log(TARGET_AMT))) + theme_minimal()
```

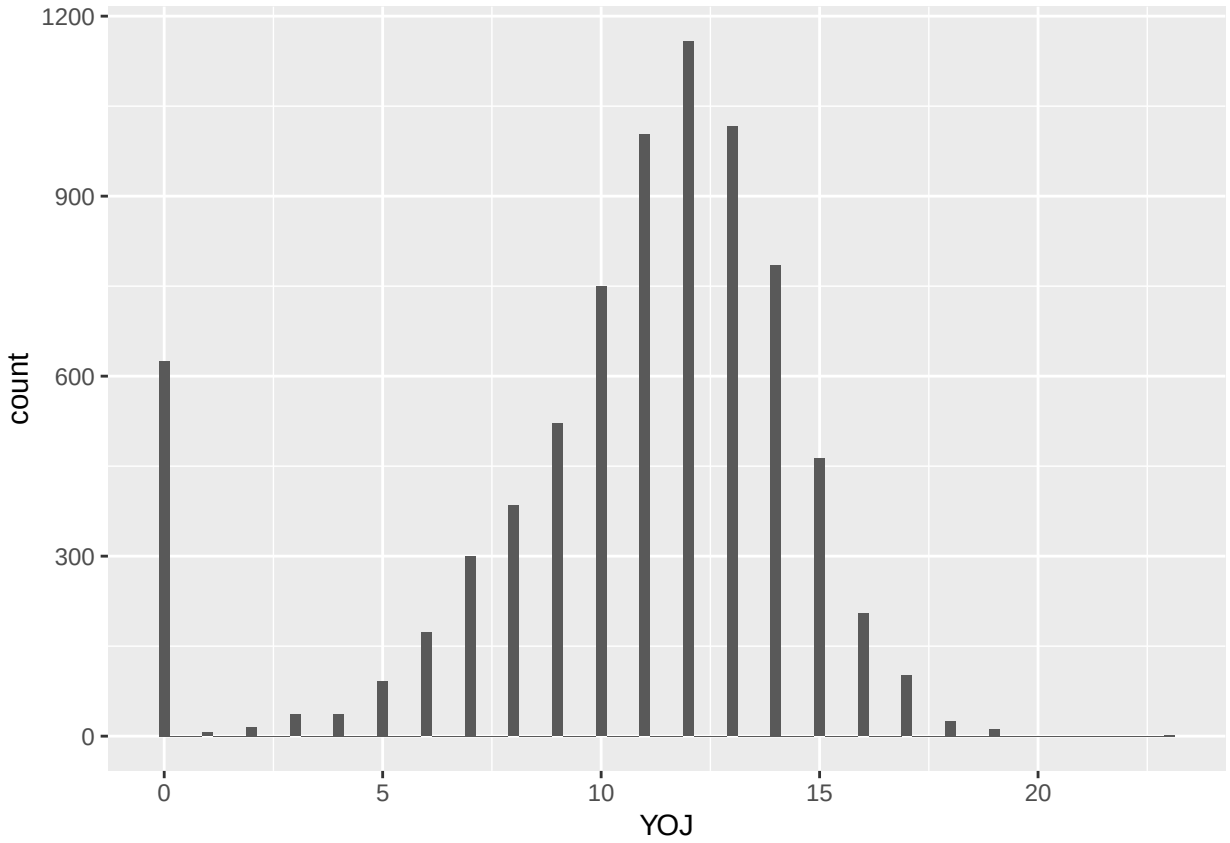



```
ggplot(df.t, aes(x = log(OLDCLAIM))) + geom_point(aes(y = log(TARGET_AMT))) + theme_minimal()
```




```
ggplot(df.t, aes(x = YOJ)) + geom_histogram(binwidth = 0.25)
```

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



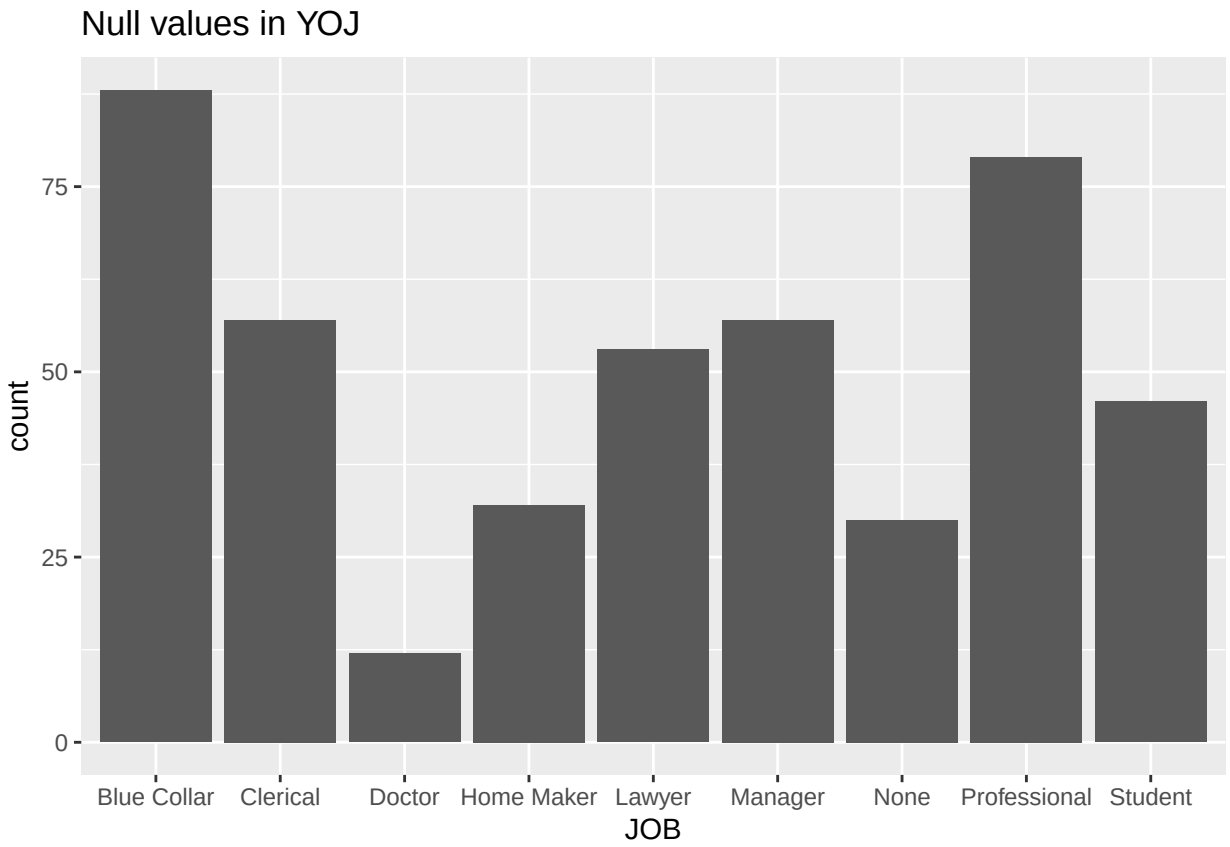
There is missing information in AGE, YOJ, INCOME, HOME_VAL, and CAR_AGE. We will chain the equations starting from AGE to impute the missing data.

Data Preparation

Here we can see the distributions of some of the missing information.

```
df.t %>% filter(is.na(YOJ)) %>% ggplot(aes(x = JOB)) +
  geom_histogram(stat = 'count') + ggtitle('Null values in YOJ')
```

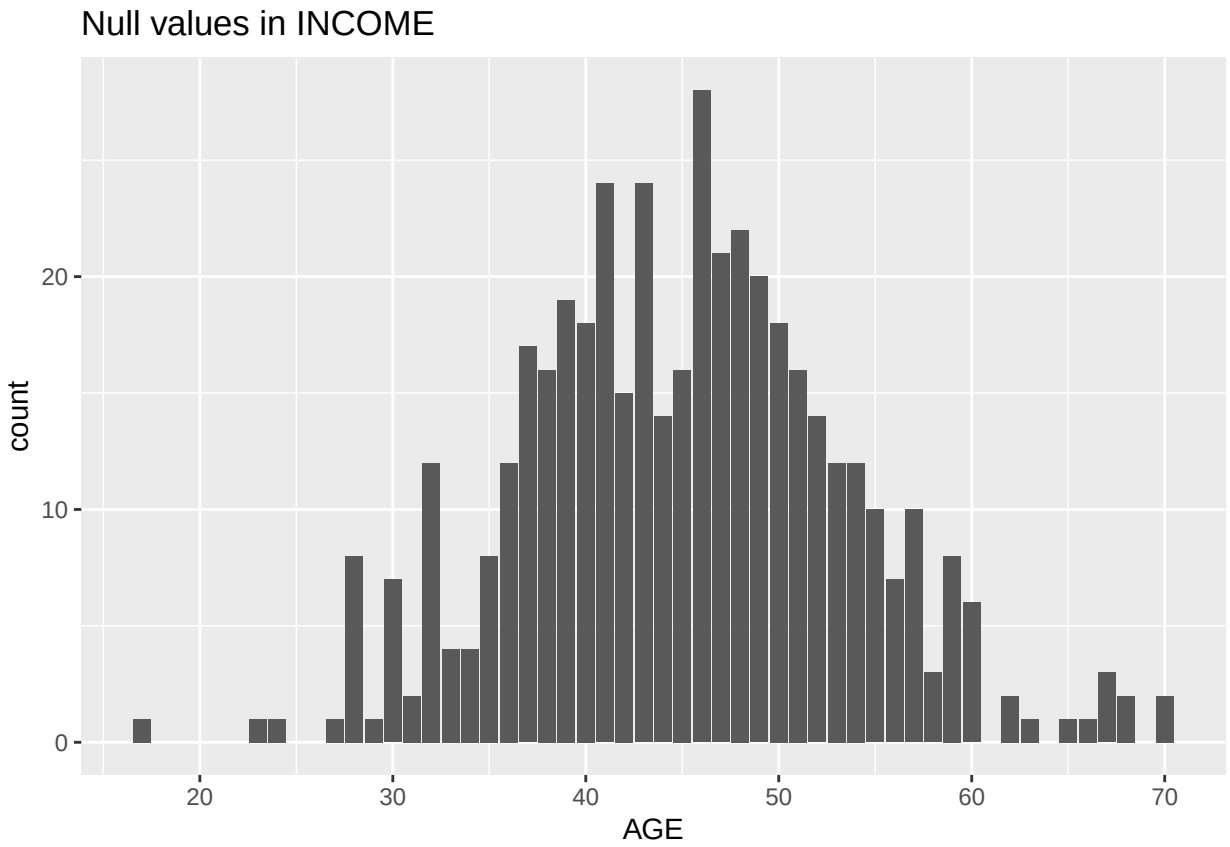
```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```




```
df.t %>% filter(is.na(INCOME)) %>% ggplot(aes(x = AGE)) +
  geom_histogram(stat = 'count') + ggtitle('Null values in INCOME')
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
## Warning: Removed 1 rows containing non-finite values (stat_count).
```



Build Models

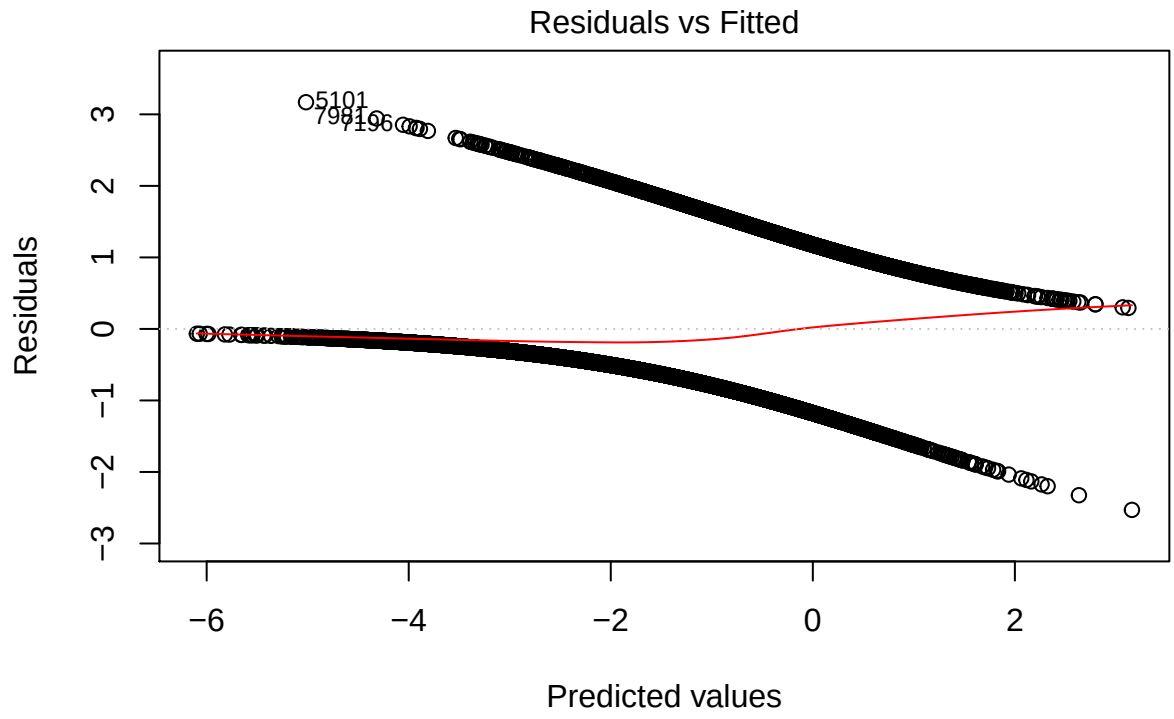
We begin with a binomial GLM containing all variables, and then use bidirectional stepwise selection to come with the best model relatively. Then we train the multiple regression with a gaussian glm and use bidirectional model selection again.

It's important to remember to include each term multiplied by the logical value TARGET_FLAG, to encourage the model not to predict a TARGET_AMT when TARGET_FLAG is false.

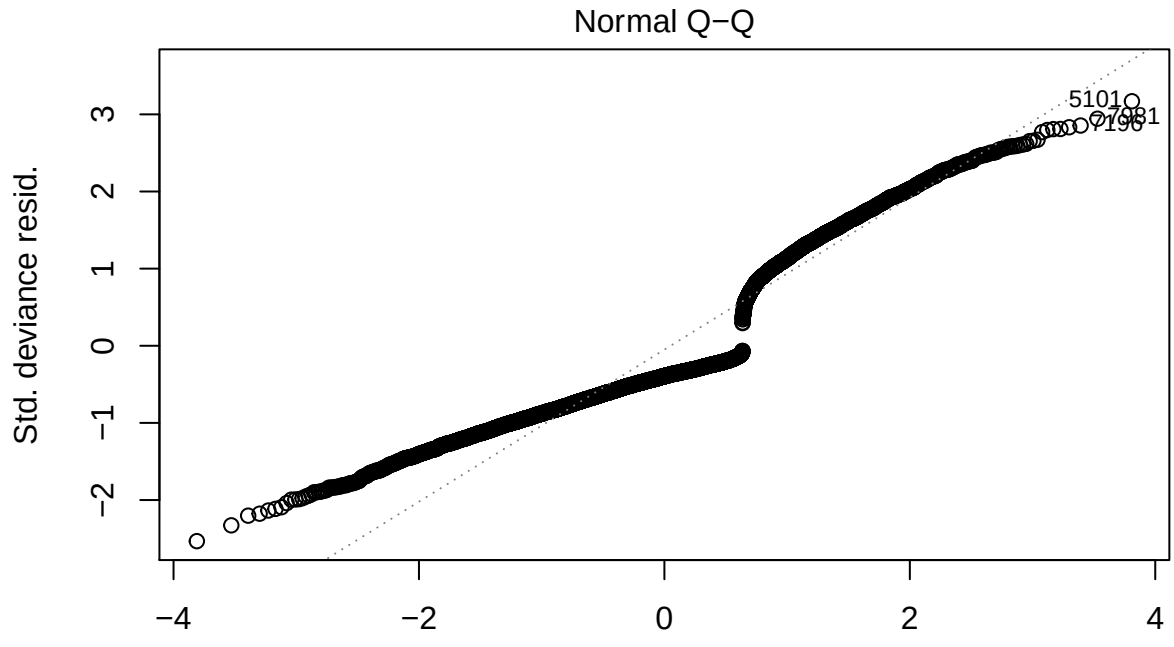
Predict values

To predict values, we use the binomial model on TARGET_FLAG, and add it to the evaluation data. Then, we run this through the multiple regression model to find TARGET_AMT.

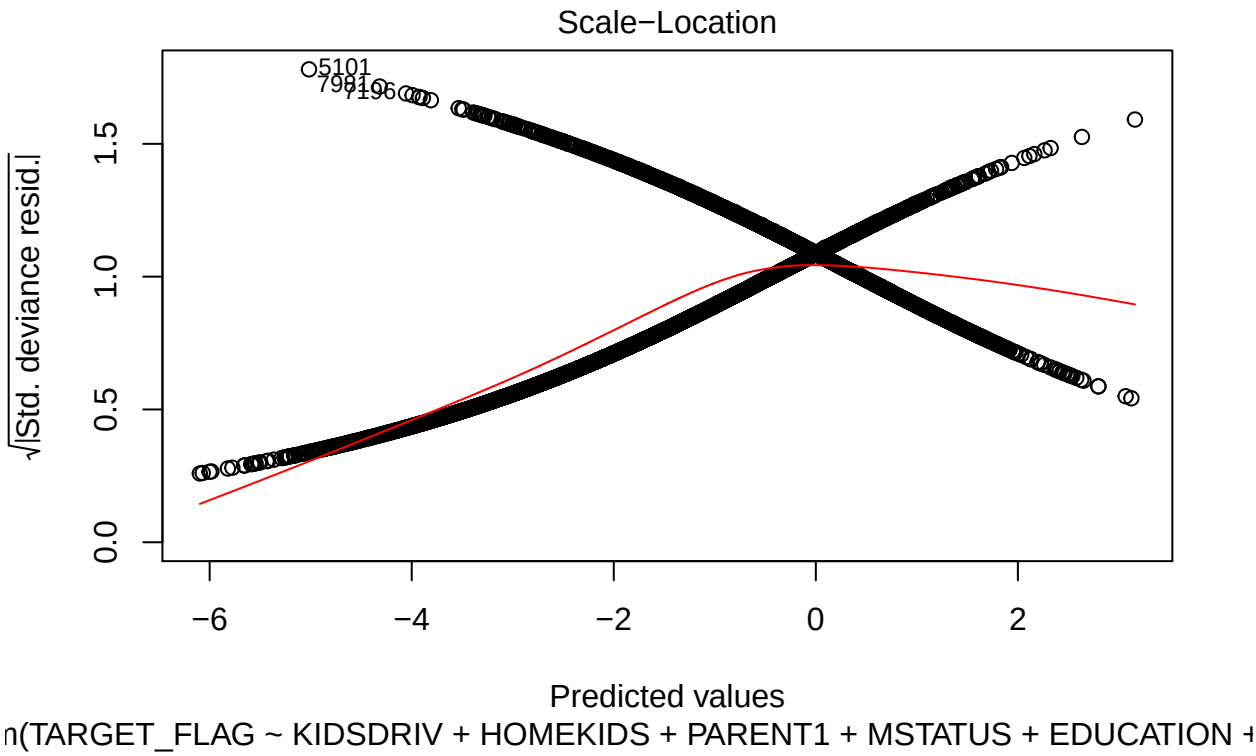
Judging the models



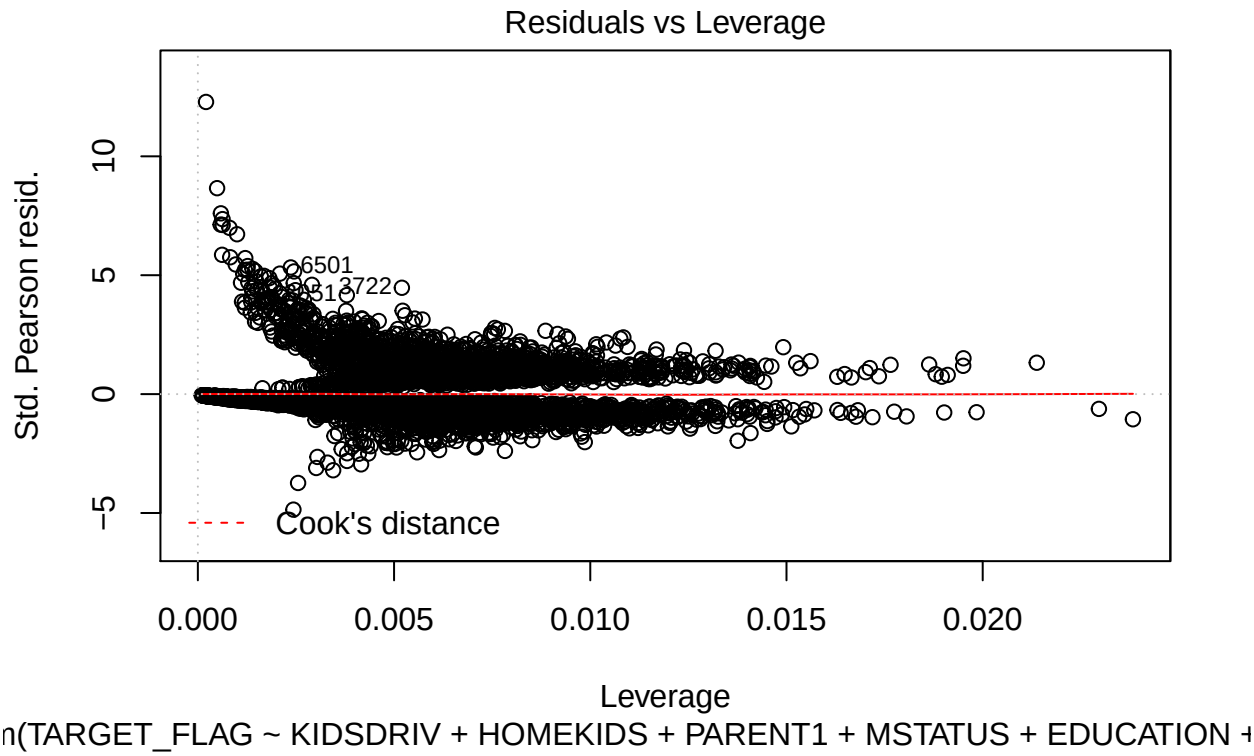
n(TARGET_FLAG ~ KIDSDRIV + HOMEKIDS + PARENT1 + MSTATUS + EDUCATION +



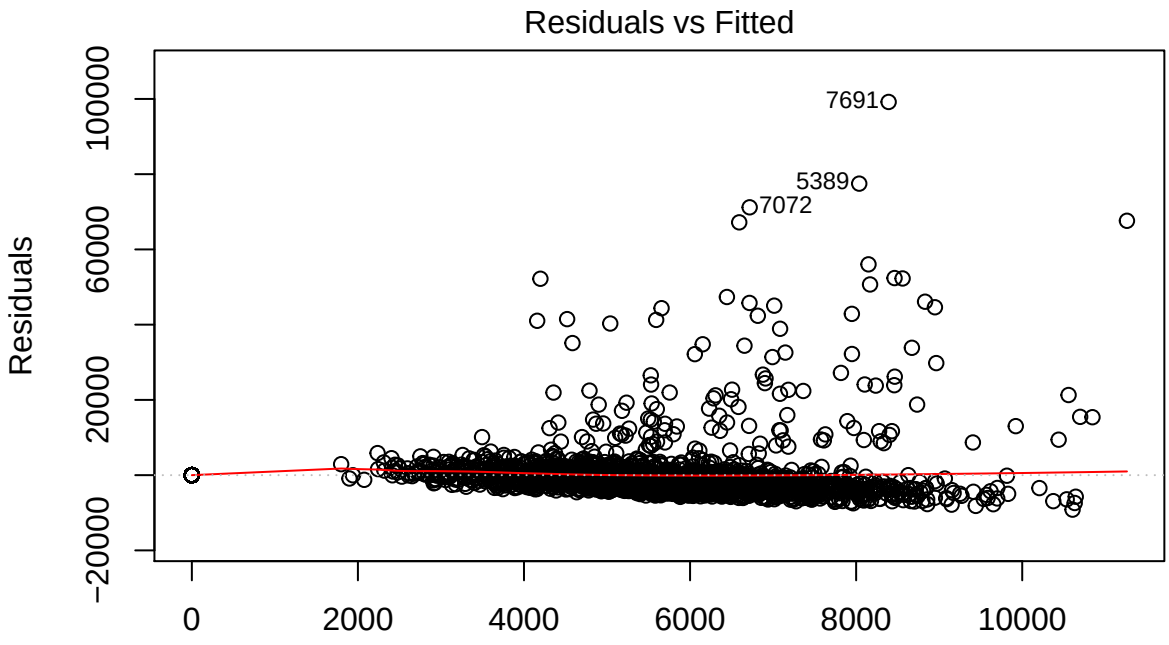
n(TARGET_FLAG ~ KIDSDRIV + HOMEKIDS + PARENT1 + MSTATUS + EDUCATION +



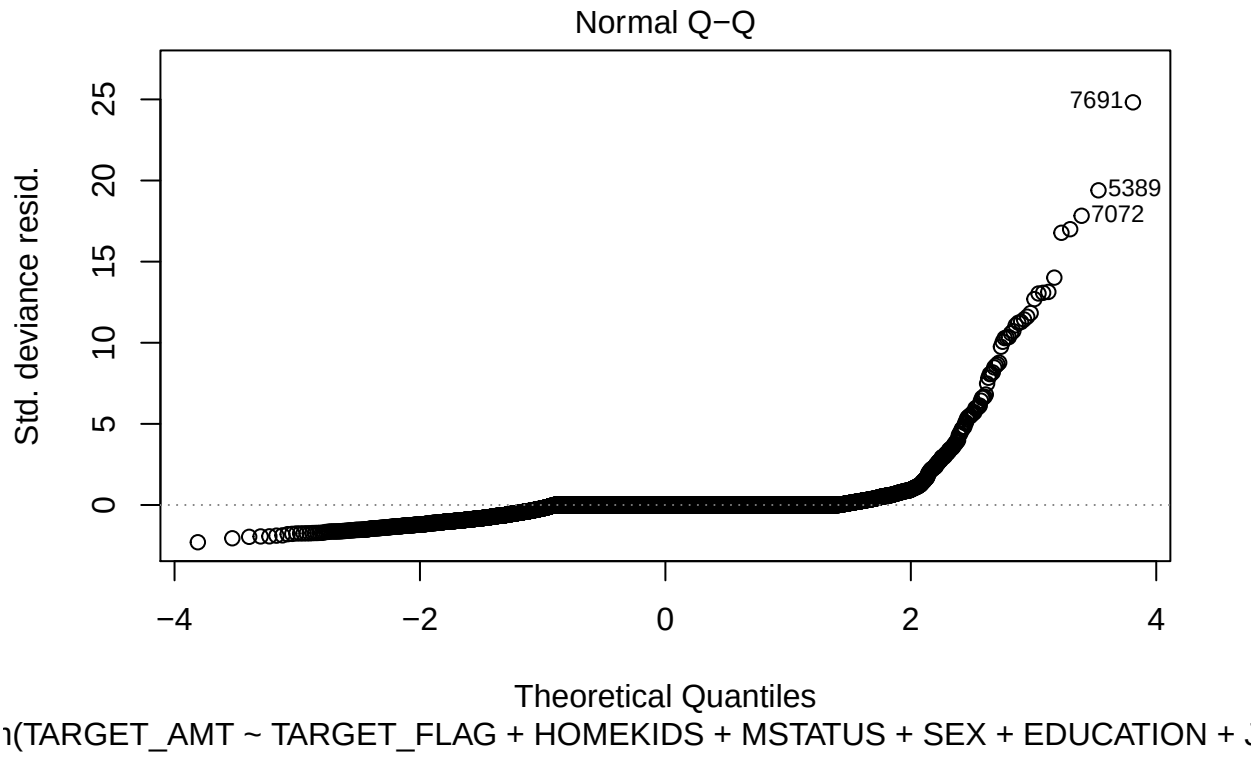
n(TARGET_FLAG ~ KIDSDRIV + HOMEKIDS + PARENT1 + MSTATUS + EDUCATION +



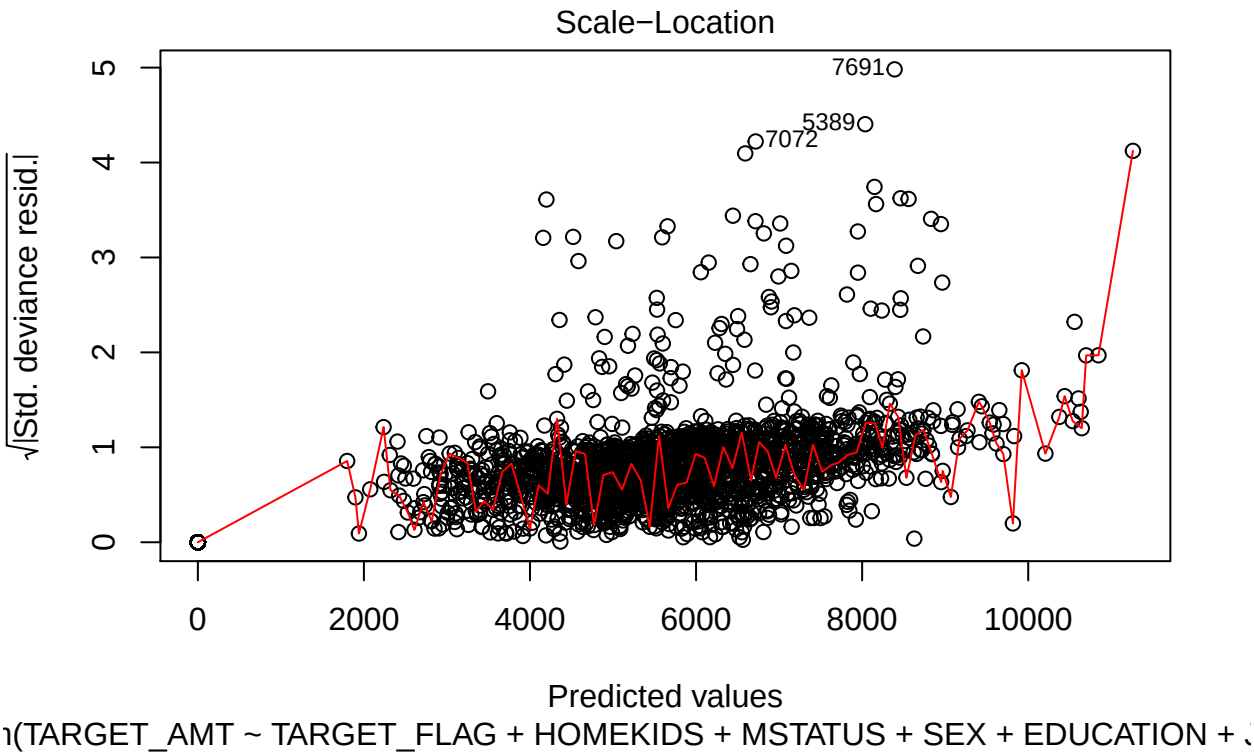
n(TARGET_FLAG ~ KIDSDRIV + HOMEKIDS + PARENT1 + MSTATUS + EDUCATION +



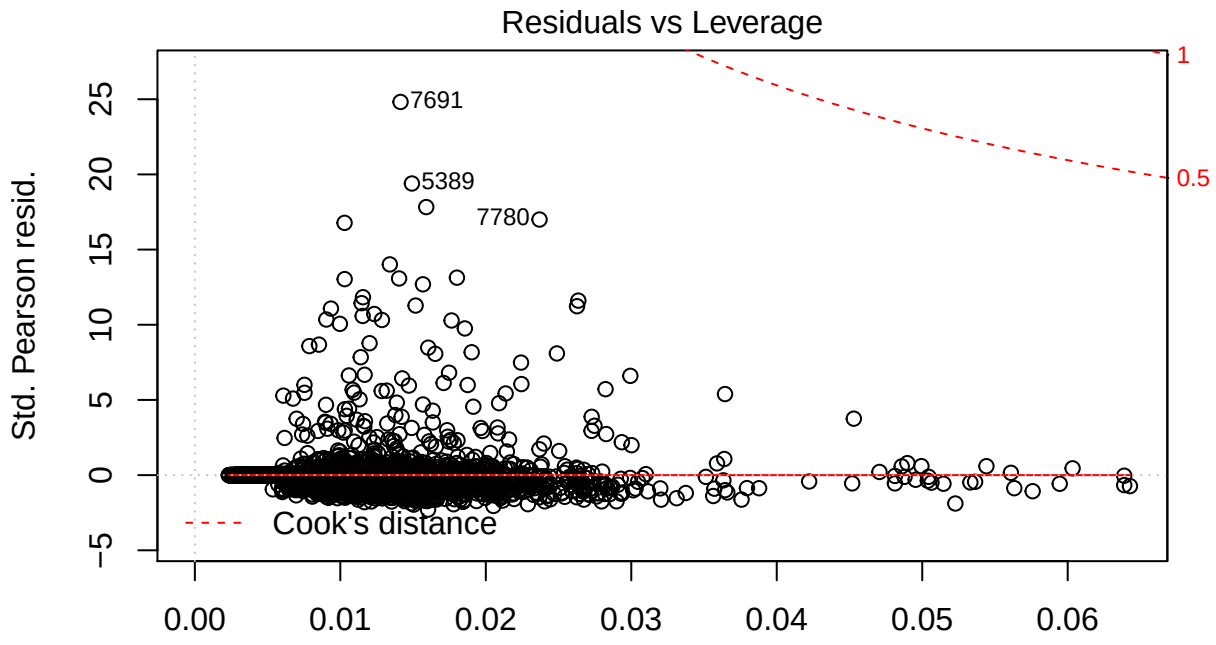
ı(TARGET_AMT ~ TARGET_FLAG + HOMEKIDS + MSTATUS + SEX + EDUCATION + J



ı(TARGET_AMT ~ TARGET_FLAG + HOMEKIDS + MSTATUS + SEX + EDUCATION + J



ı(TARGET_AMT ~ TARGET_FLAG + HOMEKIDS + MSTATUS + SEX + EDUCATION + J



ı(TARGET_AMT ~ TARGET_FLAG + HOMEKIDS + MSTATUS + SEX + EDUCATION + J

```
truth.bin <- df.t$TARGET_FLAG[t.e1]
fit.bin <- tr1$TARGET_FLAG[t.e1]
confusionMatrix(fit.bin, truth.bin)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FALSE TRUE
##      FALSE   641  166
##      TRUE     54  100
##
##                Accuracy : 0.7711
##                  95% CI : (0.7432, 0.7973)
##     No Information Rate : 0.7232
##     P-Value [Acc > NIR] : 0.000421
##
##                   Kappa : 0.3428
##
##  Mcnemar's Test P-Value : 7.23e-14
##
##             Sensitivity : 0.9223
##             Specificity : 0.3759
##          Pos Pred Value : 0.7943
##          Neg Pred Value : 0.6494
##              Prevalence : 0.7232
```

```
##          Detection Rate : 0.6670
##    Detection Prevalence : 0.8398
##       Balanced Accuracy : 0.6491
##
##        'Positive' Class : FALSE
##
```

```
truth.reg <- df.t$TARGET_AMT[t.e2]
fit.reg <- tr1$TARGET_AMT[t.e2]
confusionMatrix(fit.bin, truth.bin)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FALSE TRUE
##      FALSE   641  166
##      TRUE     54  100
##
##                Accuracy : 0.7711
##                  95% CI : (0.7432, 0.7973)
##     No Information Rate : 0.7232
##     P-Value [Acc > NIR] : 0.000421
##
##                   Kappa : 0.3428
##
##  Mcnemar's Test P-Value : 7.23e-14
##
##             Sensitivity : 0.9223
##             Specificity : 0.3759
##          Pos Pred Value : 0.7943
##          Neg Pred Value : 0.6494
##              Prevalence : 0.7232
##          Detection Rate : 0.6670
##    Detection Prevalence : 0.8398
##       Balanced Accuracy : 0.6491
##
##        'Positive' Class : FALSE
##
```

Both models seem to be working effectively, considerably better than a coin toss. Still, I would never use these models to make accurate spot predictions. Rather, this information should be used to group customers low or high risk, and the aggregate predictions can maybe help estimate overall yearly costs. If used properly, this could be used to set lower rates across the entire customer base or to return more money to the company.

Code Appendix

```
library(tidyverse)
library(mice)
library(MASS)
library(caret)
library(pROC)
```

```
set.seed(1337)

scrub <- function(df) {
  f1 <- function(x) {
  return(as.numeric(gsub('[\\$,]', '', x)))}

  f2 <- function(x) {
  return(as.character(gsub('[z_,]', '', x)))}

  df <- tibble(df)[,2:26] %>%
    mutate(
      TARGET_FLAG = as.factor(as.logical(df$TARGET_FLAG)),
         TARGET_AMT = as.numeric(df$TARGET_AMT),
         KIDSDRIV = as.numeric(df$KIDSDRIV),
         AGE = as.numeric(df$AGE),
         HOMEKIDS = as.numeric(df$HOMEKIDS),
         YOJ = as.numeric(df$YOJ),
         INCOME = as.numeric(f1(df$INCOME)),
         PARENT1 = as.logical(lapply(df$PARENT1,
                      function(x) if(x == 'Yes') 1 else 0)),
         HOME_VAL = as.numeric(f1(df$HOME_VAL)),
         MSTATUS = as.logical(lapply(f2(df$MSTATUS),
                      function(x) if(x == 'Yes') 1 else 0)),
         SEX = as.logical(lapply(f2(df$SEX),
                   function(x) if(x == 'M') 1 else 0)),
         EDUCATION = as.factor(f2(df$EDUCATION)),
         JOB = as.factor(as.character(lapply(f2(df$JOB),
                      function(x) if(x == '') 'None' else x))),
         TRAVTIME = as.numeric(df$TRAVTIME),
         CAR_USE = as.logical(lapply(df$CAR_USE,
                      function(x) if(x == 'Private') 1 else 0)),
         BLUEBOOK = as.numeric(f1(df$BLUEBOOK)),
         TIF = as.numeric(df$TIF),
         CAR_TYPE = as.factor(f2(df$CAR_TYPE)),
         RED_CAR = as.logical(lapply(df$RED_CAR,
                      function(x) if(x == 'yes') 1 else 0)),
         OLDCLAIM = as.numeric(f1(df$OLDCLAIM)),
         CLM_FREQ = as.numeric(df$CLM_FREQ),
         REVOKED = as.logical(lapply(df$REVOKED,
                      function(x) if(x == 'Yes') 1 else 0)),
         MVR_PTS = as.numeric(df$MVR_PTS),
         CAR_AGE = as.numeric(df$CAR_AGE),
         URBANICITY = as.logical(lapply(f2(df$URBANICITY),
                                        function(x) if(x == 'Highly Urban/ Urban')
                                          1 else 0)))
  levels(df$EDUCATION) <- c(0,2,1,3,4)
  return(df)
}

df.t <- scrub(read.csv('insurance_training_data.csv'))
df.e <- scrub(read.csv('insurance-evaluation-data.csv'))

summary(df.t)
colSums(is.na(df.t))
```

```
ggplot(df.t, aes(x = AGE)) + geom_point(aes(y = log(TARGET_AMT))) + theme_minimal()
ggplot(df.t, aes(x = TRAVTIME)) + geom_point(aes(y = log(TARGET_AMT))) + theme_minimal()
ggplot(df.t, aes(x = log(OLDCLAIM))) + geom_point(aes(y = log(TARGET_AMT))) + theme_minimal()
ggplot(df.t, aes(x = YOJ)) + geom_histogram(binwidth = 0.25)

df.t %>% filter(is.na(YOJ)) %>% ggplot(aes(x = JOB)) +
  geom_histogram(stat = 'count') + ggtitle('Null values in YOJ')

df.t %>% filter(is.na(INCOME)) %>% ggplot(aes(x = AGE)) +
  geom_histogram(stat = 'count') + ggtitle('Null values in INCOME')

 fill.missing <- function(df1, df2) {
  x <- rbind(df.t, df.e)[,3:25]
  y <- mice(x, maxit = 20)
  z <- complete(y, 1)

  imp <- c(2,4,5,7,22)
  complete <- cbind(x[,-imp], z[,imp])
  return(complete)
}

full <- fill.missing(df.t, df.e)

tr <- full[1:8161,]
ev <- full[8162:10302,]

anyNA(tr)
anyNA(ev)

tr1 <- cbind(df.t[,1], tr)

x <- 1:length(tr1$AGE)
t.t1 <- sample(x, 7200)
t.e1  <- setdiff(x, t.t1)

m <- glm(TARGET_FLAG ~ ., tr1[t.t1,], family = 'binomial')
bin <- stepAIC(m, direction = 'both', steps = 1000000, k = 2)

tr2 <- cbind(df.t[,1:2], tr)
t.t2 <- sample(x, 7200)
t.e2  <- setdiff(x, t.t2)

m3 <- glm(TARGET_AMT ~ TARGET_FLAG * ., tr2[t.t2,], family = 'gaussian')
reg <- stepAIC(m3, direction = 'both', steps = 1000000, k = 2)

tr1$TARGET_FLAG[t.e1] <- predict.glm(bin, tr1[t.e1,]) %>%
  lapply(function(x) if (x <= 0) 0 else 1) %>%
  as.logical()

tr2$TARGET_AMT[t.e2] <- predict.glm(reg, tr2[t.e2,]) %>%
  lapply(function(x) if (x <= 0) 0 else x)

plot(bin)
plot(reg)
```

```
truth.bin <- df.t$TARGET_FLAG[t.e1]
fit.bin <- tr1$TARGET_FLAG[t.e1]
confusionMatrix(fit.bin, truth.bin)

truth.reg <- df.t$TARGET_AMT[t.e2]
fit.reg <- tr1$TARGET_AMT[t.e2]
confusionMatrix(fit.bin, truth.bin)

ev$TARGET_FLAG <- predict.glm(bin, ev) %>%
  lapply(function(x) if (x <= 0) 0 else 1) %>%
  as.logical() %>% as.factor()

ev$TARGET_AMT <- predict.glm(reg, ev) %>%
  lapply(function(x) if (x <= 0) 0 else x)

write.csv(as.matrix(ev), 'answers.csv')
```