Business Analytics and Data Mining

DATA621 Homework 04

William Outcault, Mengqin Cai, Philip Tanofsky, Robert Welk, Zhi Ying Chen

22 November 2020

Contents

Overview

This assignment attempts to 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.

The objective is to build both a 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.

The structure of the training data indicates 8161 records with 26 variables, 23 predictor variables and 2 target variables along with the index variable. Before diving into the data, unwanted characters in the dataset (dollar signs, commas, etc) were removed and datatypes were changed. Code for changes made to the dataset are available in the Appendix at the end of this report.

DATA EXPLORATION

The following exploratory data methods present a picture of the data to capture the distribution of the data and potential correlation with the target variable. The techniques used explore a summary of the variables,the distribution of each predictor variable against the target variable, density plot of each predictor variable against the target variable, along with a correlation plot across all the features.

Structure of Data

```
str(raw_training)
```

```
## 'data.frame':    8161 obs. of  25 variables:
##  $ 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 "Yes","z_No": 2 2 1 1 1 2 1 1 2 2 ...
##  $ SEX        : Factor w/ 2 levels "M","z_F": 1 1 2 1 2 2 2 1 2 1 ...
##  $ EDUCATION  : Factor w/ 4 levels "Bachelors","High School",..: 4 2 2 2 4 1 2 1 1 1 ...
##  $ JOB        : Factor w/ 8 levels "Blue Collar",..: 7 1 2 1 3 1 1 1 2 7 ...
##  $ 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 Urban/ Urban",..: 1 1 1 1 1 1 1 1 1 2 ...
```

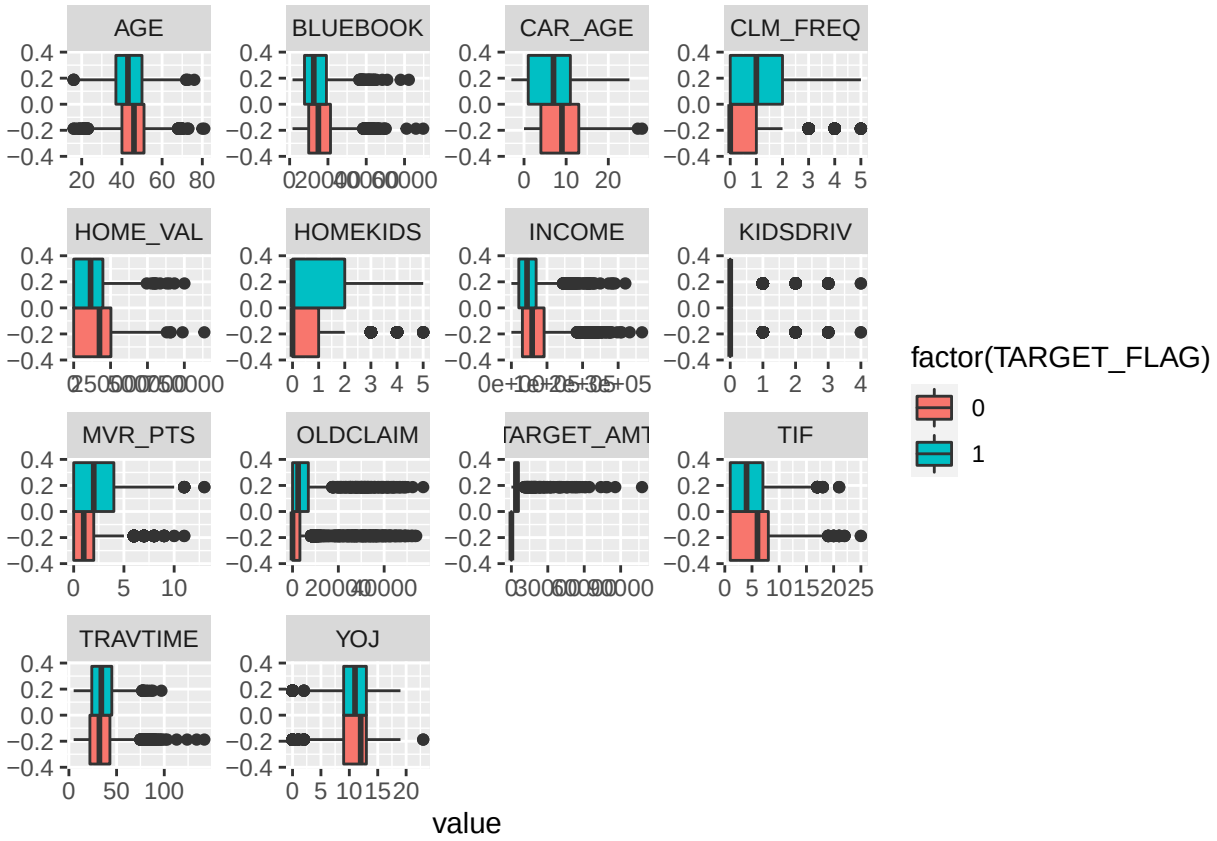
Missing Values were present in several of the variables:

-AGE -YOJ -CAR_AGE -HOME_VAL -INCOME

EDA for logistic regression

Boxplots

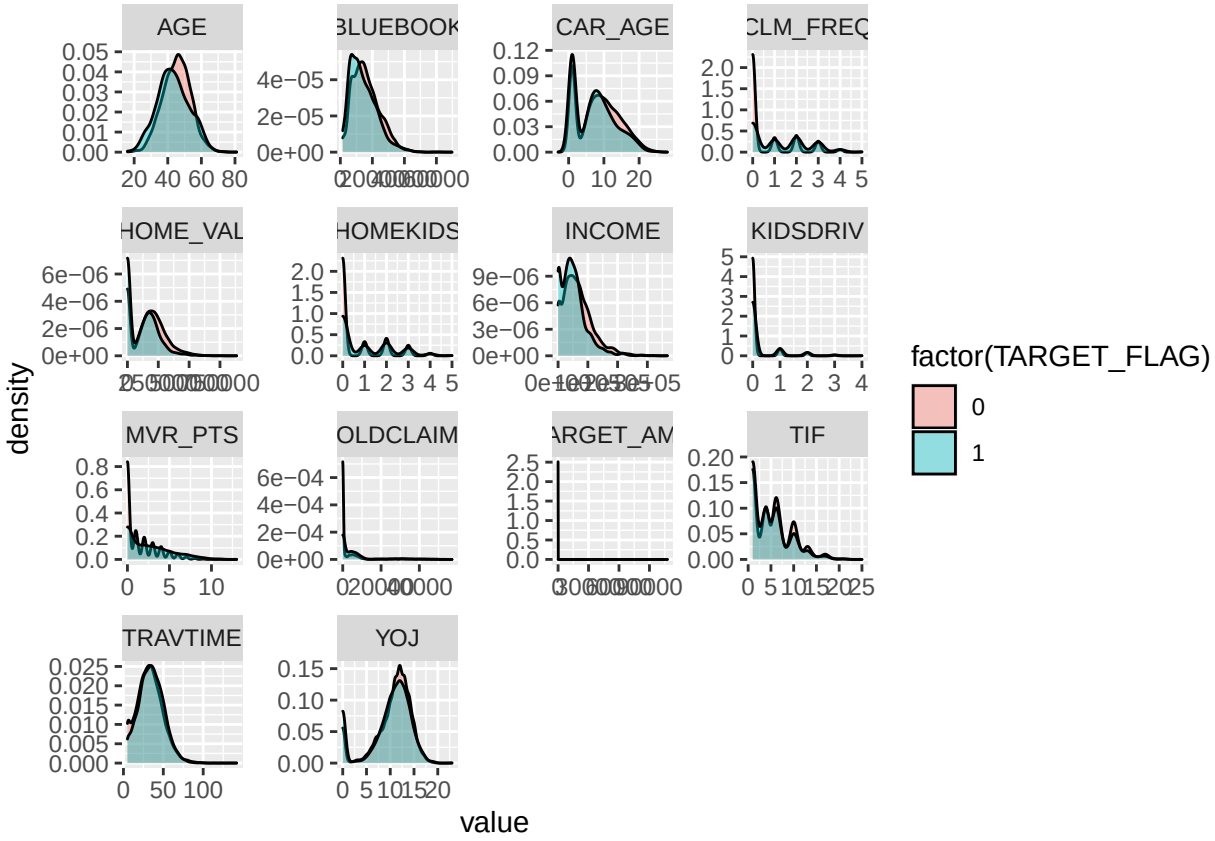
The dodged boxplot of each numeric variable against the target variable highlights differences between target boxes which could mean the variable is useful for prediction of the logistic model. A dodged boxplot without overlapping boxes likely indicates a correlation in the value of the predictor variable to the target classes.



Not many of the variables show distinct differences in response value. CLM_FREQ has the largest discrepency.

Density plots

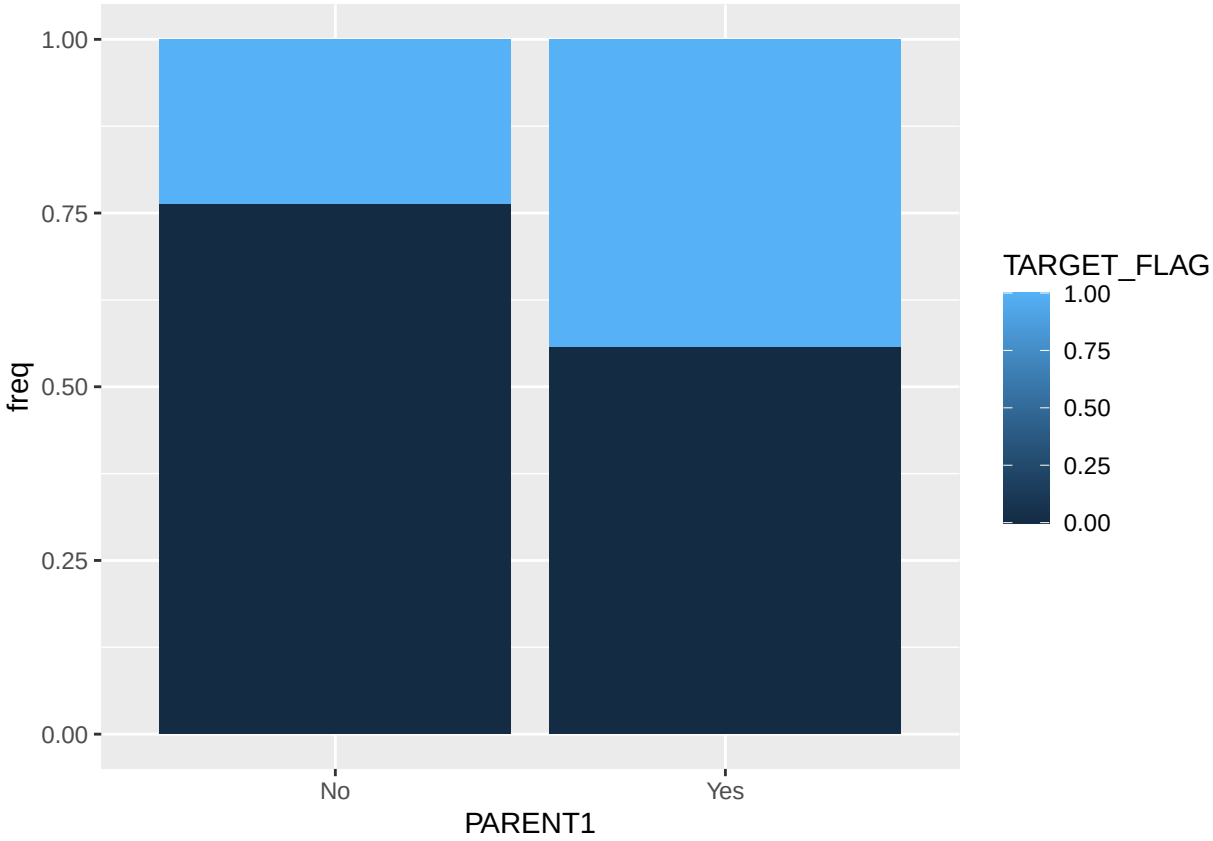
Similar to the boxplots, the density plots are another tool to identify which numeric predictor variables likely have a strong correlation with the target variable, and can suggest which variables are good to include in the logistic regression model.

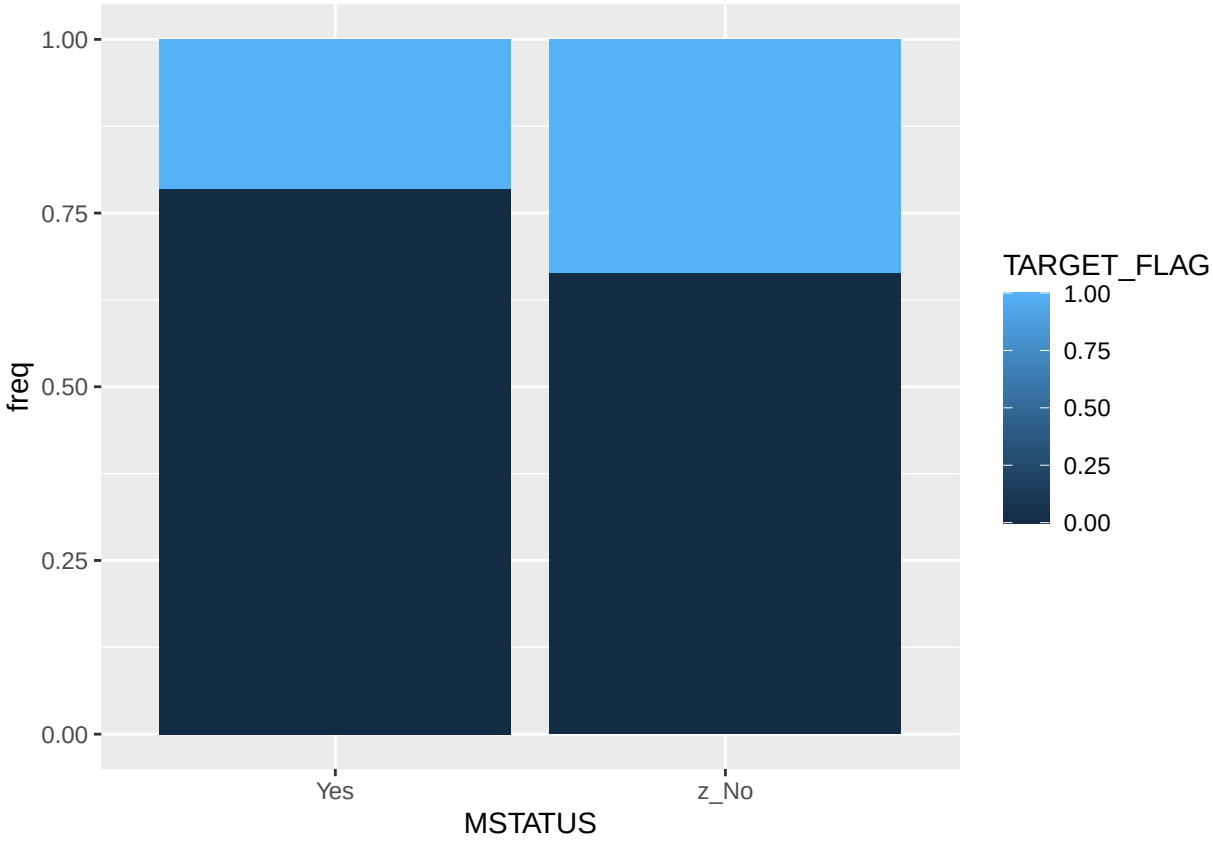


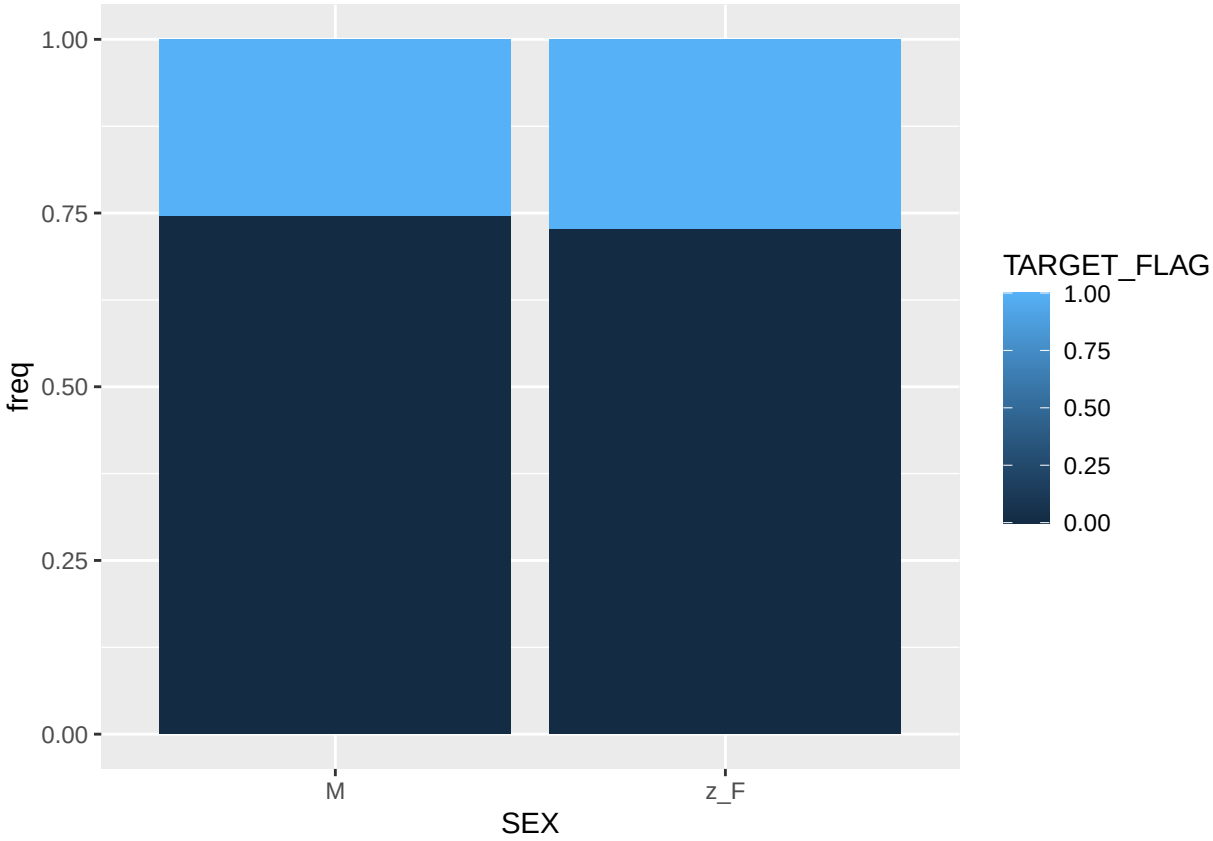
The density plots show distributions that are almost identical for each variable comparing the target. Based on this visualization, none of the numeric variables would make good predictors for the logistic regression model.

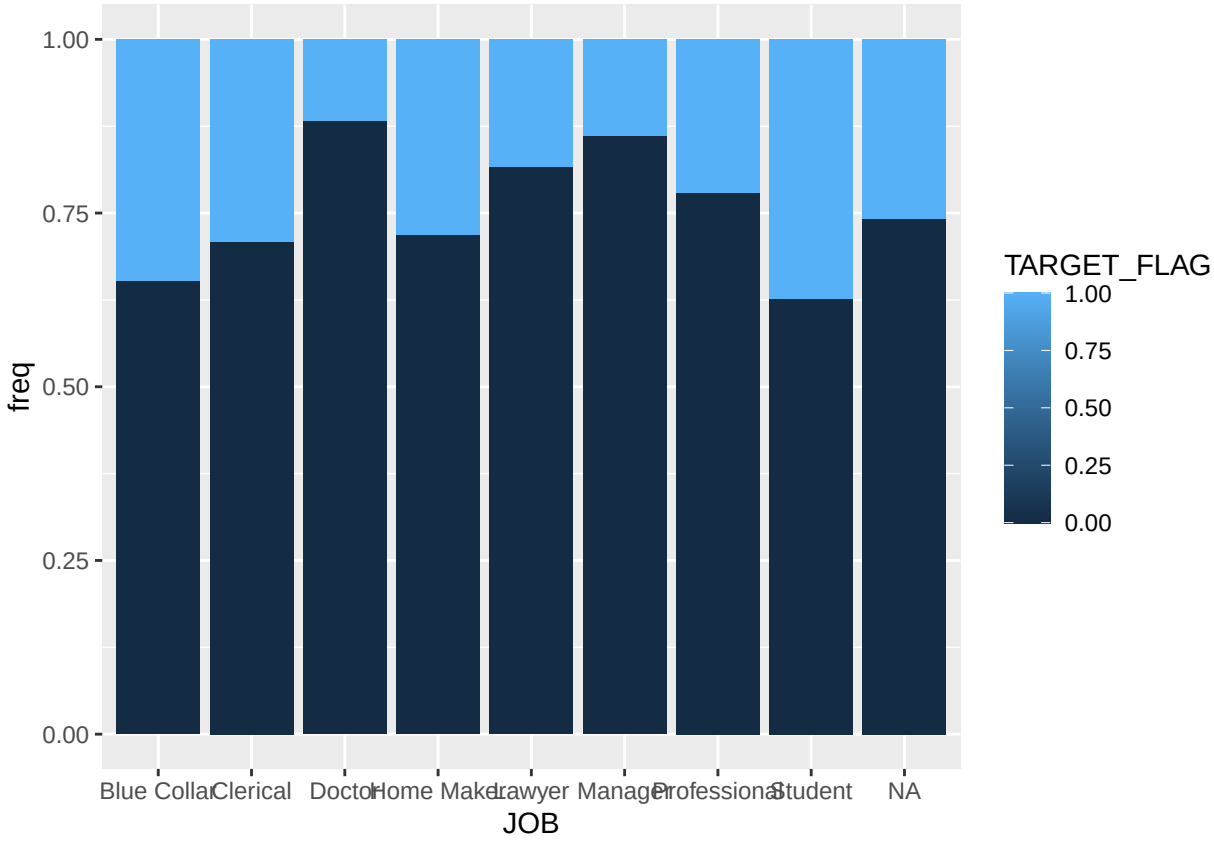
Barplots

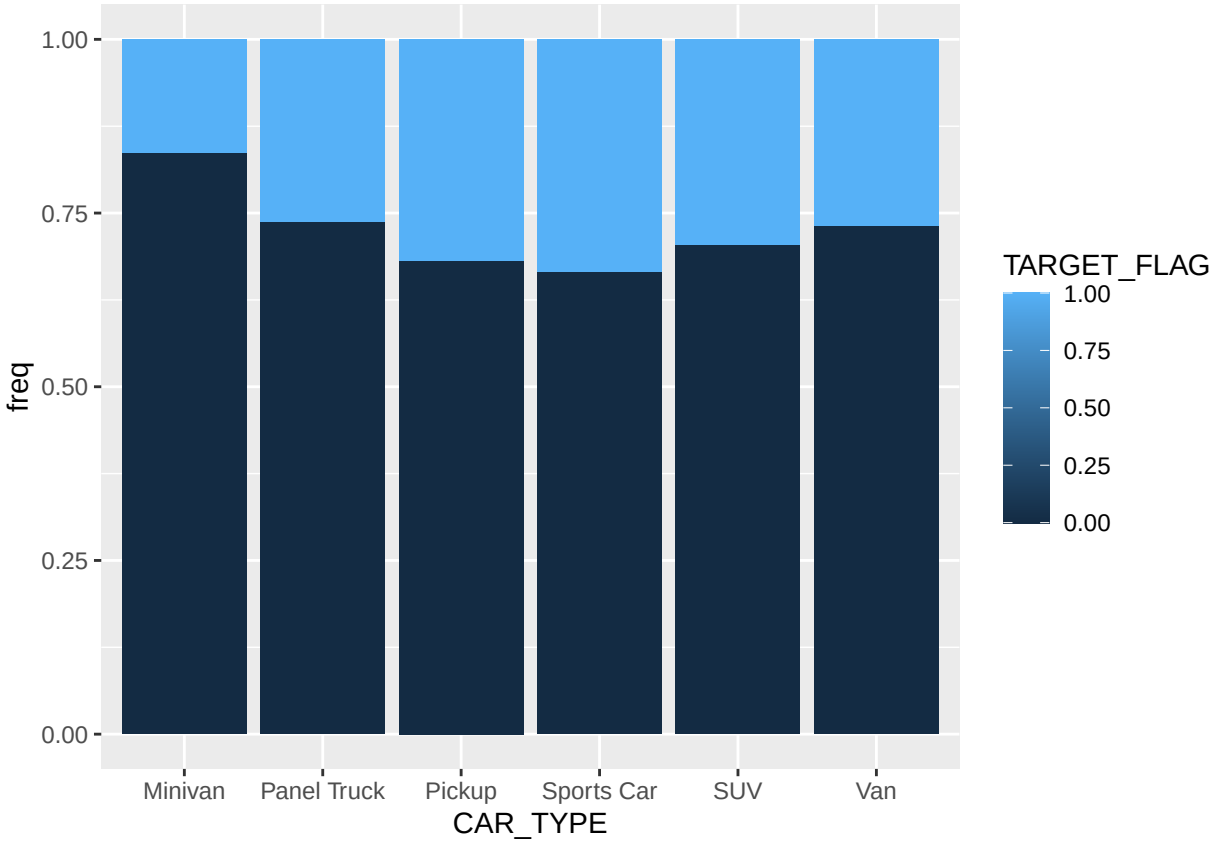
Next, factor variables were evaluated for logistic regression suitability using stacked barplots

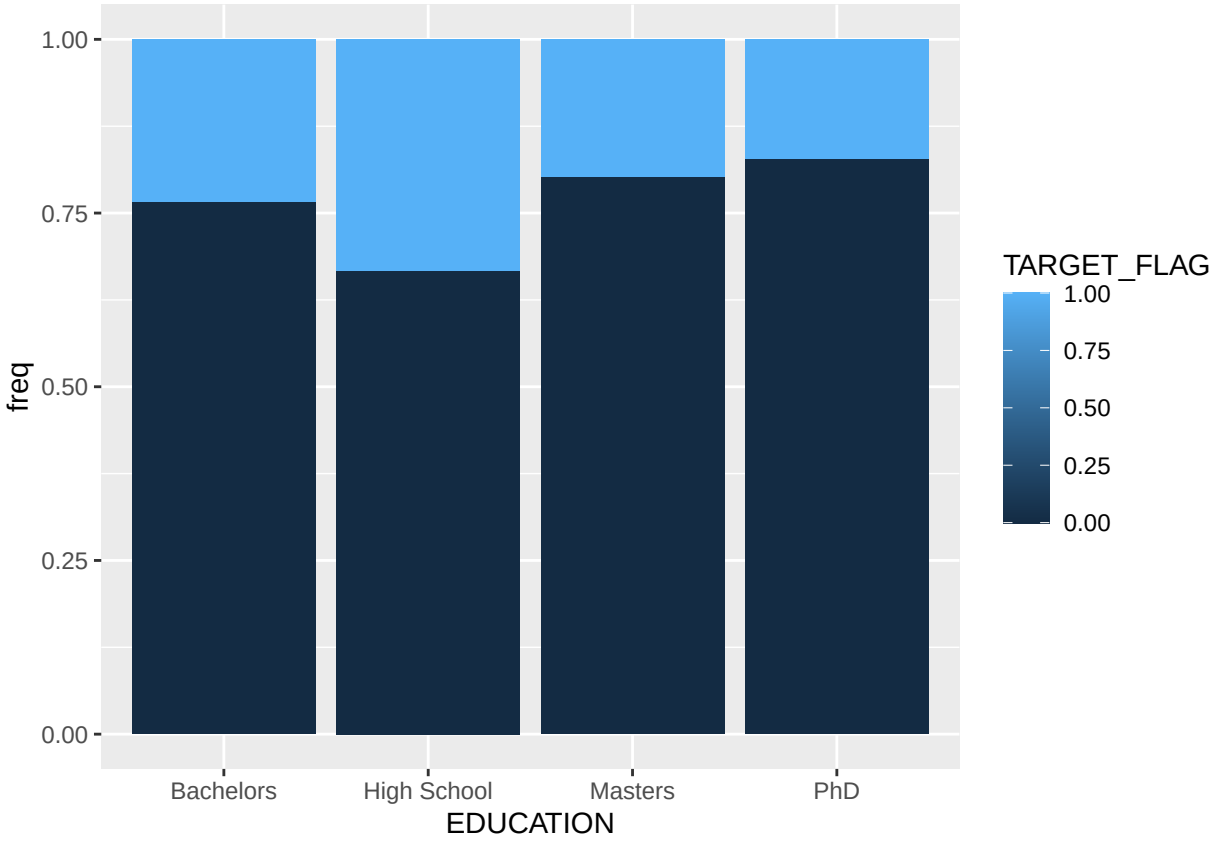

1.00
0.75
0.50
0.25
0.00
freq
No
Yes
PARENT1
TARGET_FLAG
1.00
0.75
0.50
0.25
0.00

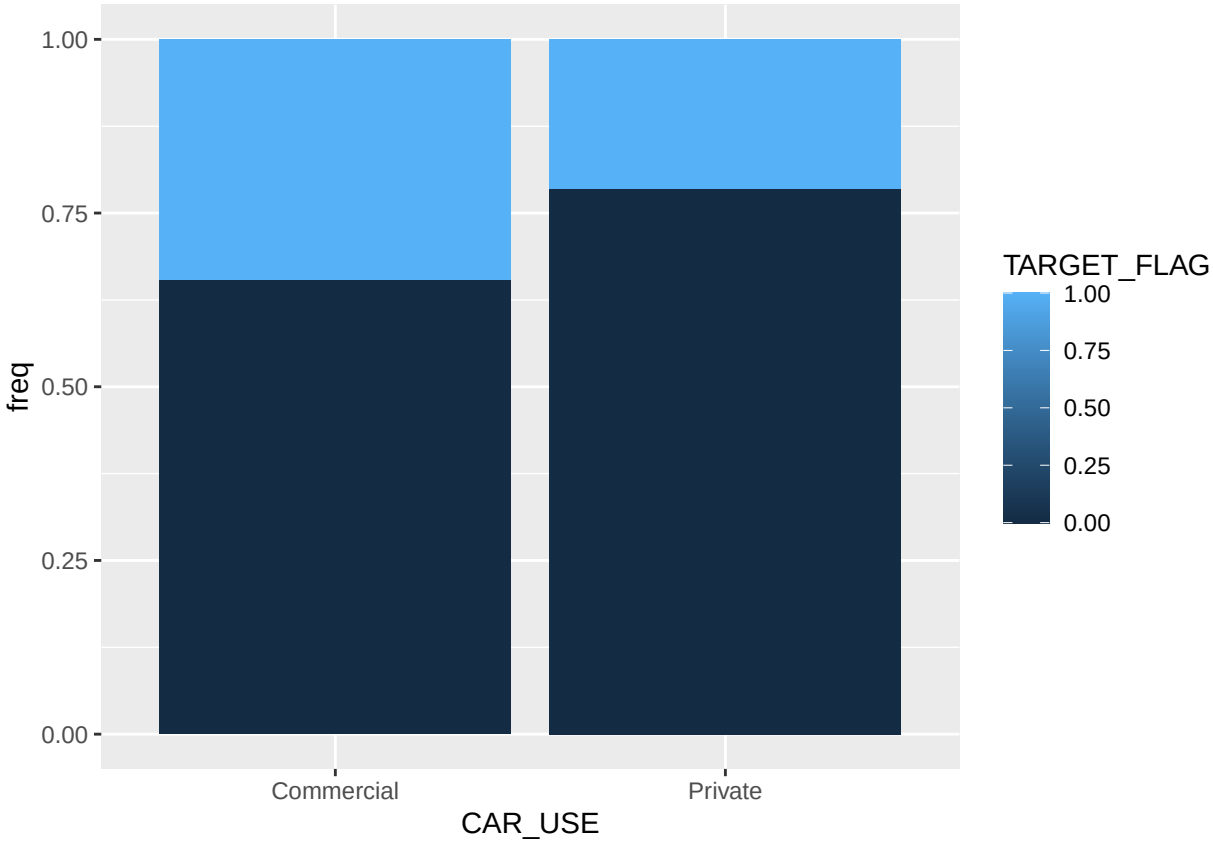

1.00
0.75
0.50
0.25
0.00
freq
Yes
z_No
MSTATUS
TARGET_FLAG
1.00
0.75
0.50
0.25
0.00

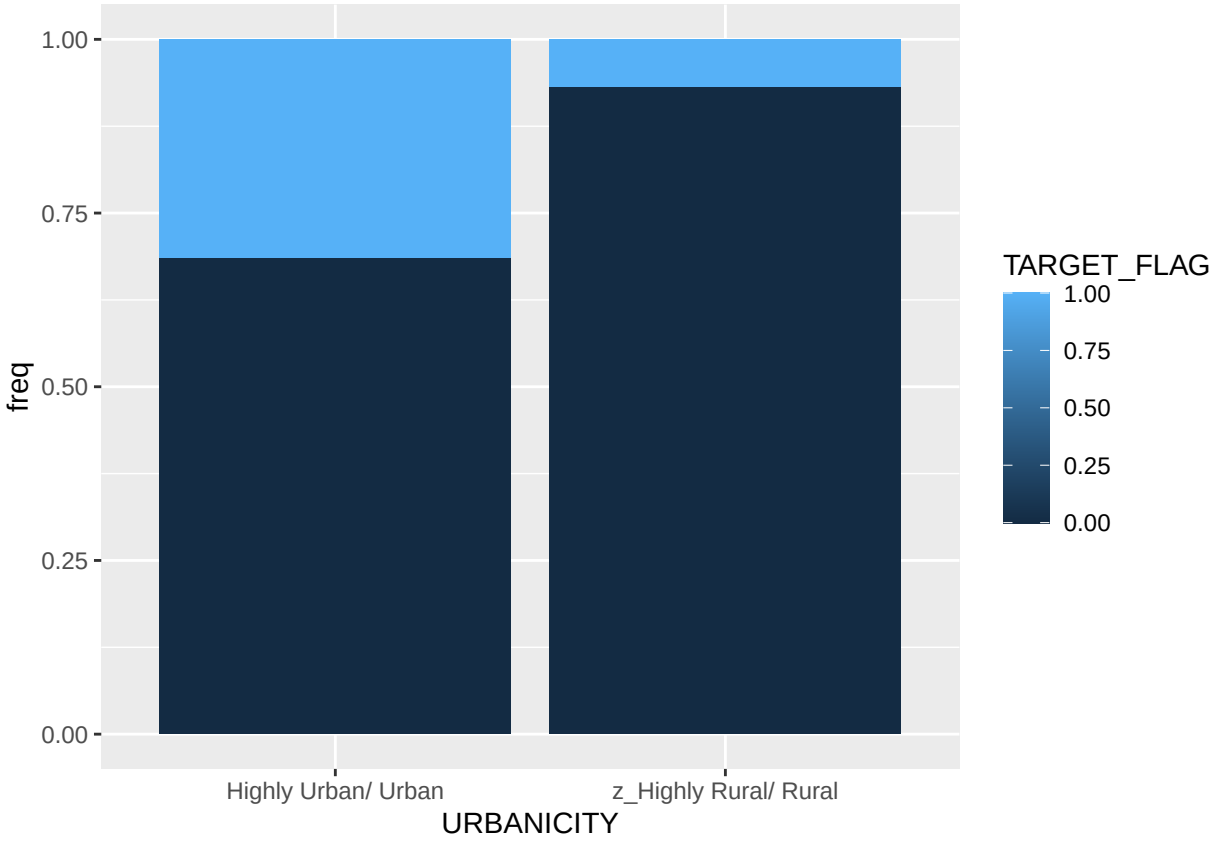

1.00
0.75
0.50
0.25
0.00
freq
M
z_F
SEX
TARGET_FLAG
1.00
0.75
0.50
0.25
0.00

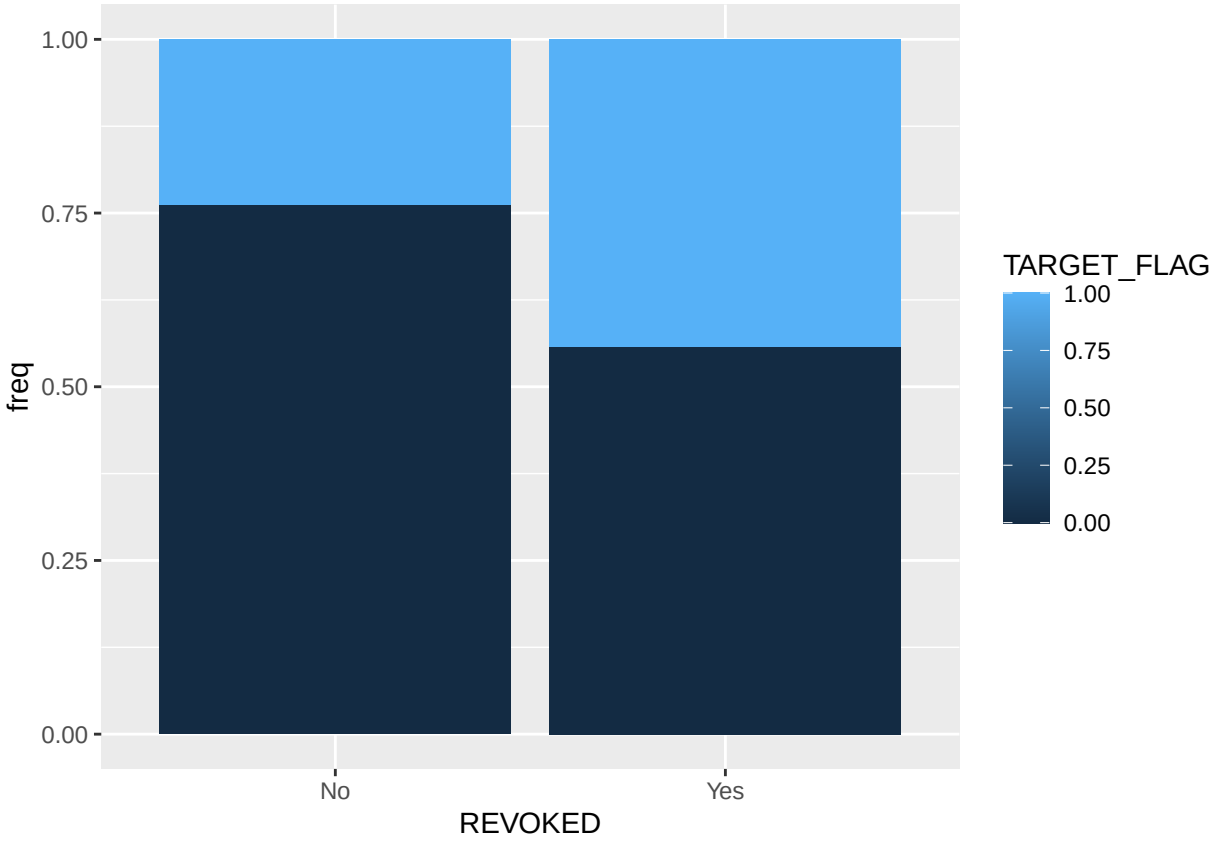

1.00
0.75
0.50
0.25
0.00
freq
Blue Collar
Clerical
Doctor
Home Maker
Lawyer
Manager
Professional
Student
NA
JOB
TARGET_FLAG
1.00
0.75
0.50
0.25
0.00

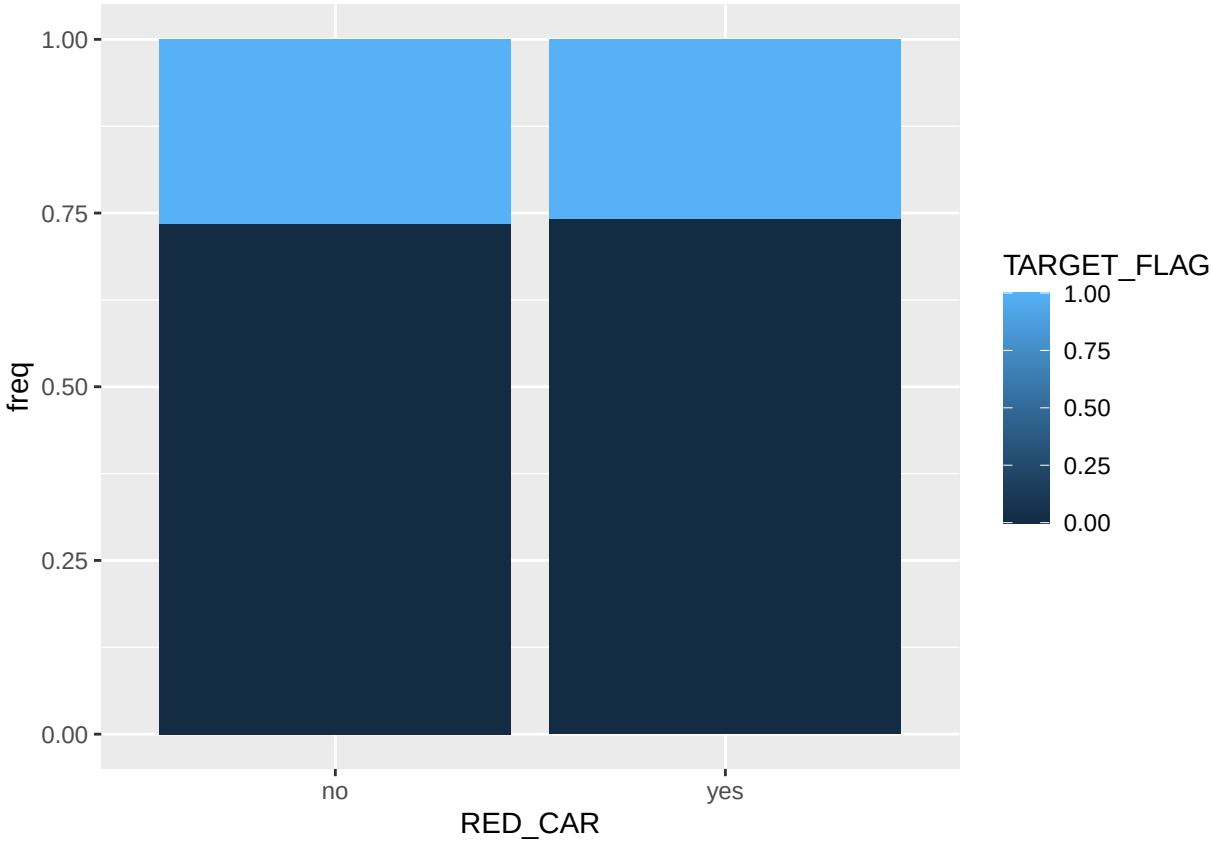

1.00
0.75
0.50
0.25
0.00
freq
Minivan
Panel Truck
Pickup
Sports Car
SUV
Van
CAR_TYPE
TARGET_FLAG
1.00
0.75
0.50
0.25
0.00


1.00
0.75
0.50
0.25
0.00
freq
Bachelors
High School
Masters
PhD
EDUCATION
TARGET_FLAG
1.00
0.75
0.50
0.25
0.00


1.00
0.75
0.50
0.25
0.00
freq
Commercial
Private
CAR_USE
TARGET_FLAG
1.00
0.75
0.50
0.25
0.00


1.00
0.75
0.50
0.25
0.00
freq
Highly Urban/ Urban
z_Highly Rural/ Rural
URBANICITY
TARGET_FLAG
1.00
0.75
0.50
0.25
0.00


1.00
0.75
0.50
0.25
0.00
freq
No
Yes
REVOKED
TARGET_FLAG
1.00
0.75
0.50
0.25
0.00



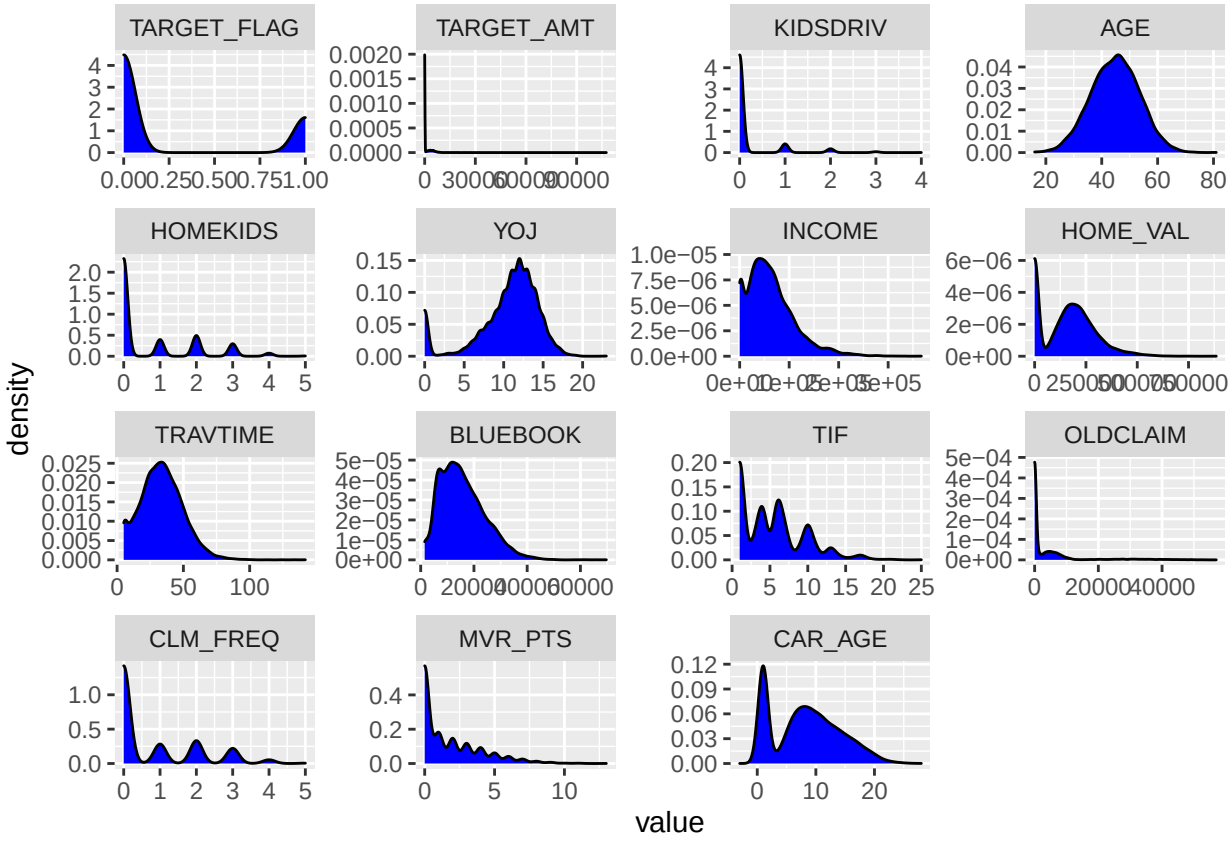
Factor variables that might make good predictors for logistic regression: -PARENT1 -MSTATUS -EDUCATION -CAR_USE -URBANCITY

It might also be useful to combine levels for factors with more than two levels. That will be done in the Data Preparation section.

EDA for linear regression

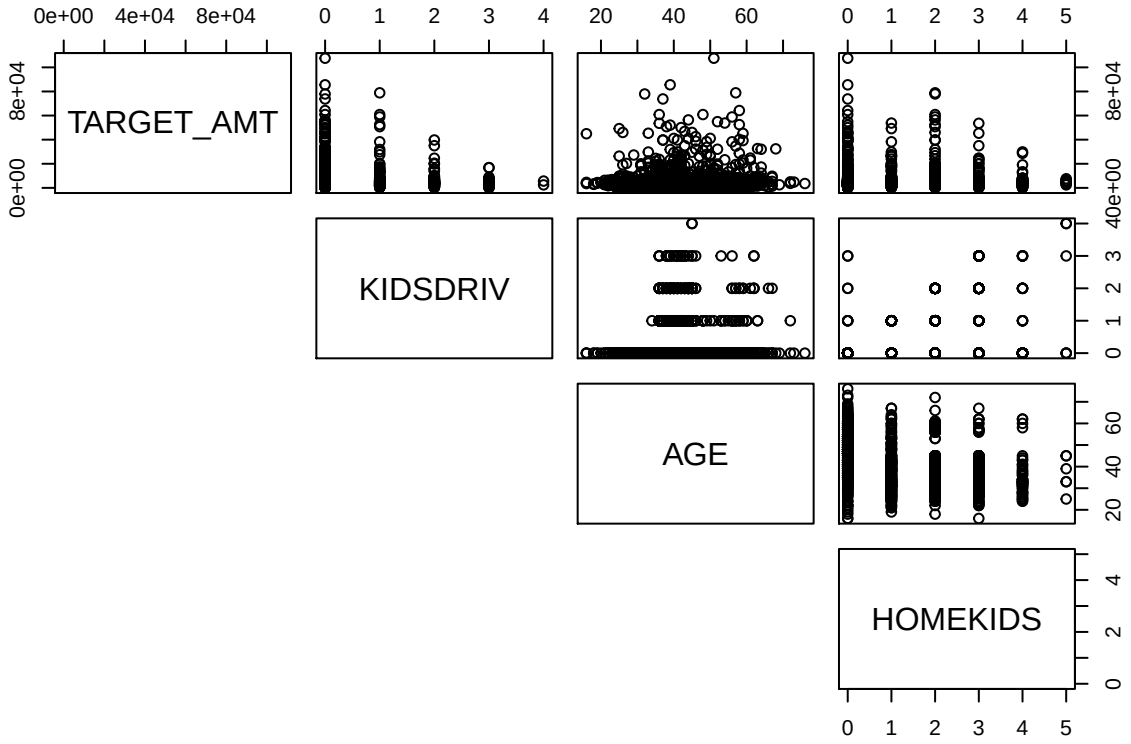
For the linear regression model, only the data where a claim was actually made is analyzed (ie TARGET_FLAG=0)

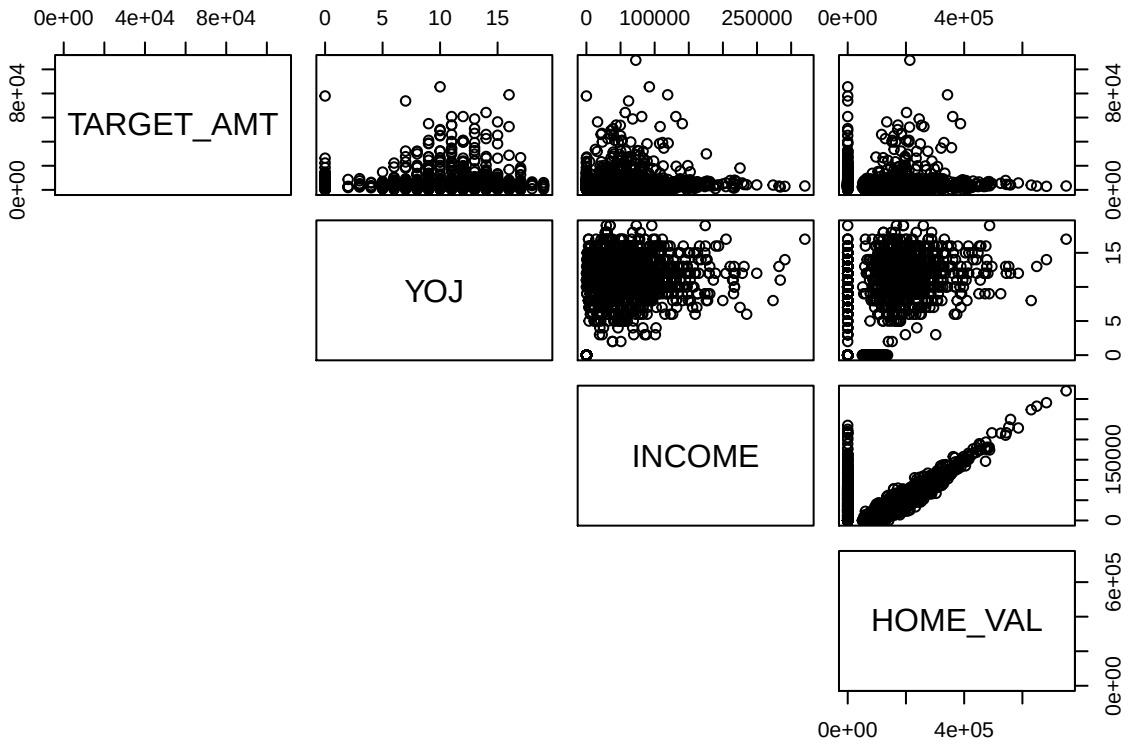
Density plots

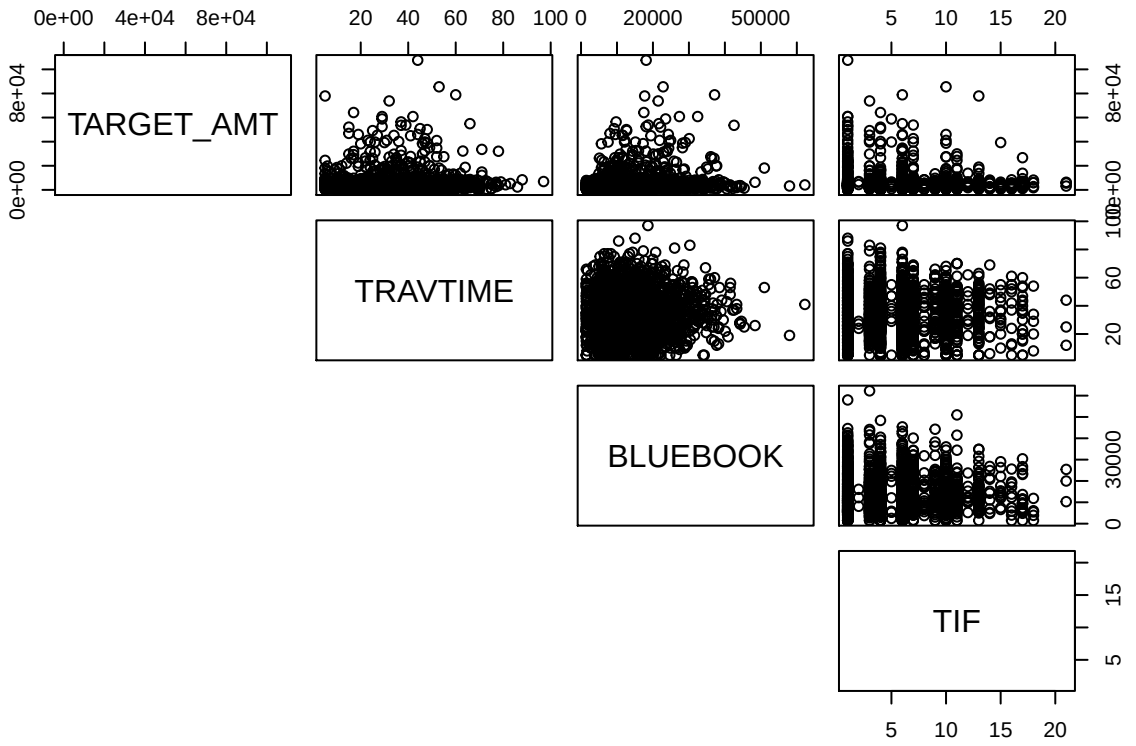


Scatterplot matrix

A scatterplot matrix is generated to evaluate the relationship between the numeric variables and the linear regression target. The matrix is divided over three plots for clarity.

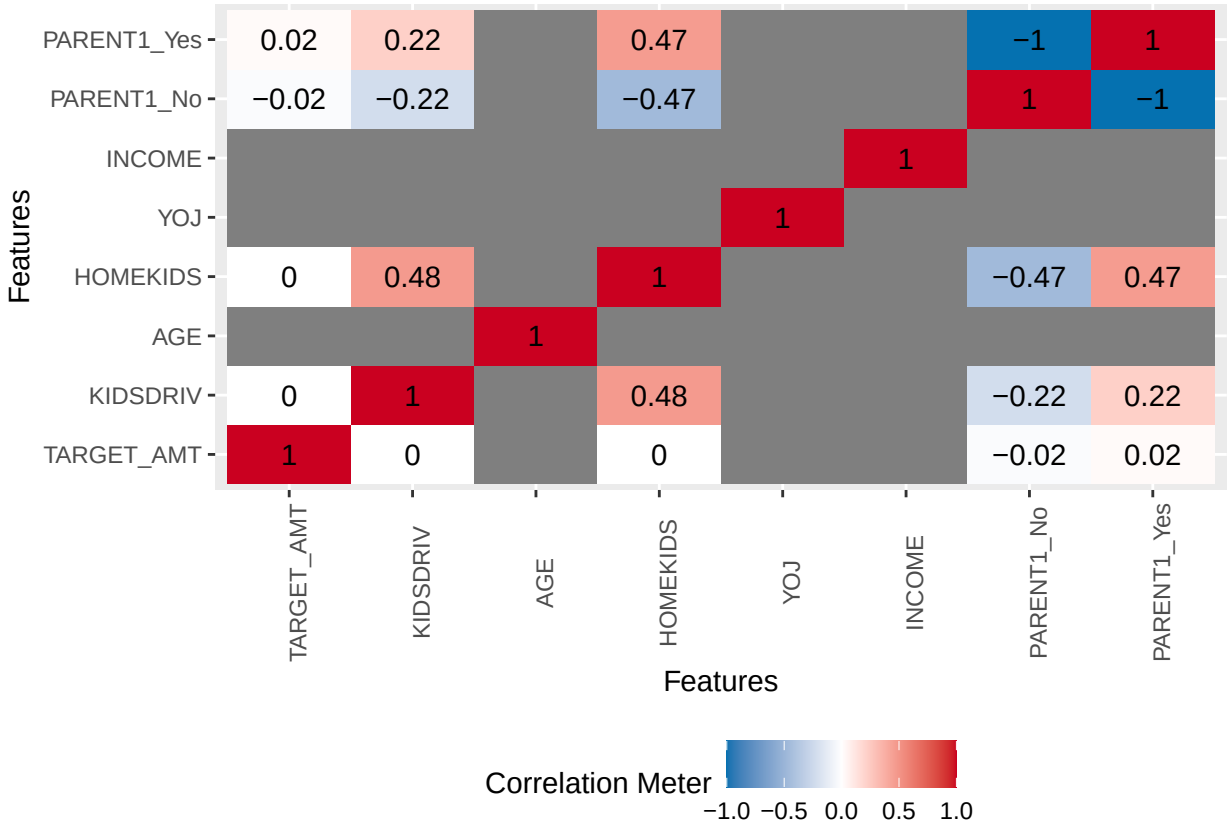

TARGET_AMT
KIDSDRIV
AGE
HOMEKIDS
0e+00
4e+04
8e+04
0
1
2
3
4
20
40
60
5


TARGET_AMT
YOJ
INCOME
HOME_VAL
0e+00
4e+04
8e+04
0
5
10
15
0
100000
250000
0e+00
4e+05
0e+00
8e+04
0
5
15
0
150000
0e+00
6e+05
0e+00
4e+05

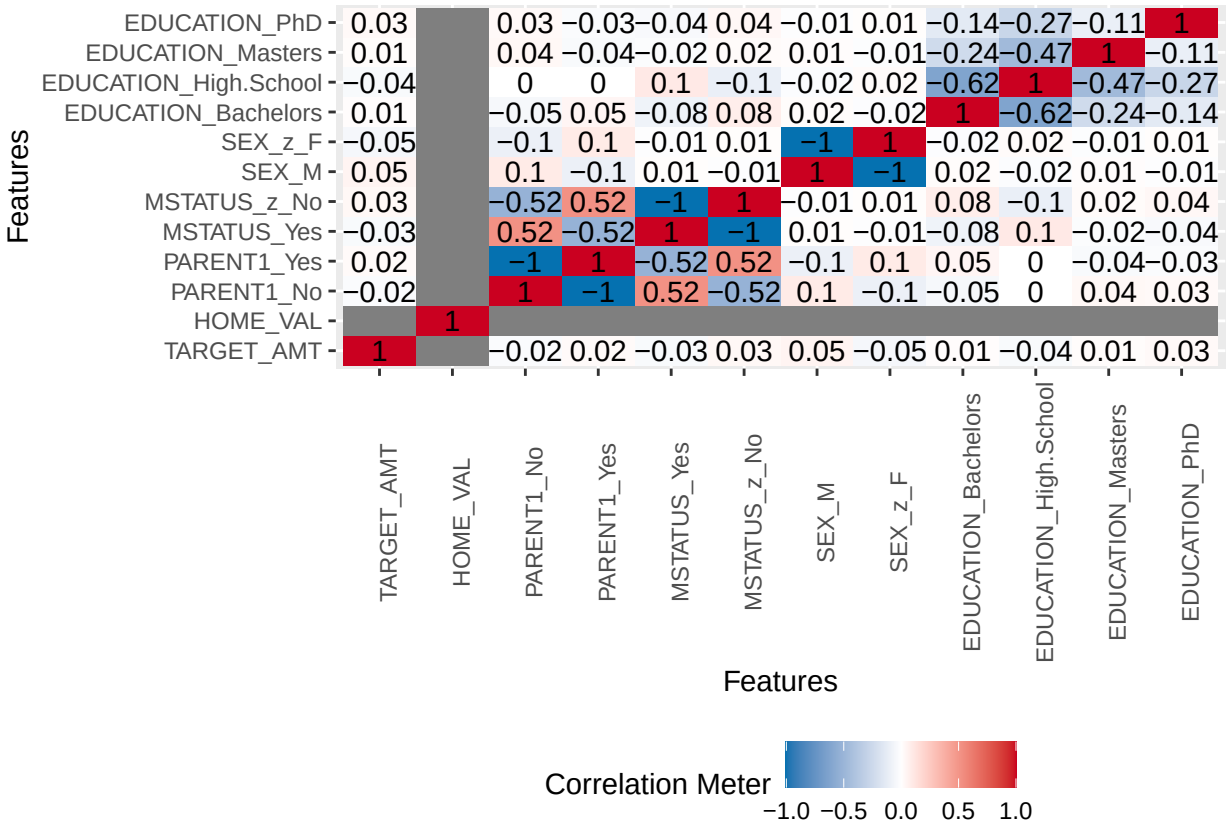

TARGET_AMT
TRAVTIME
BLUEBOOK
TIF
0e+00
4e+04
8e+04
20
40
60
80
100
0
20000
50000
5
10
15
20
30000

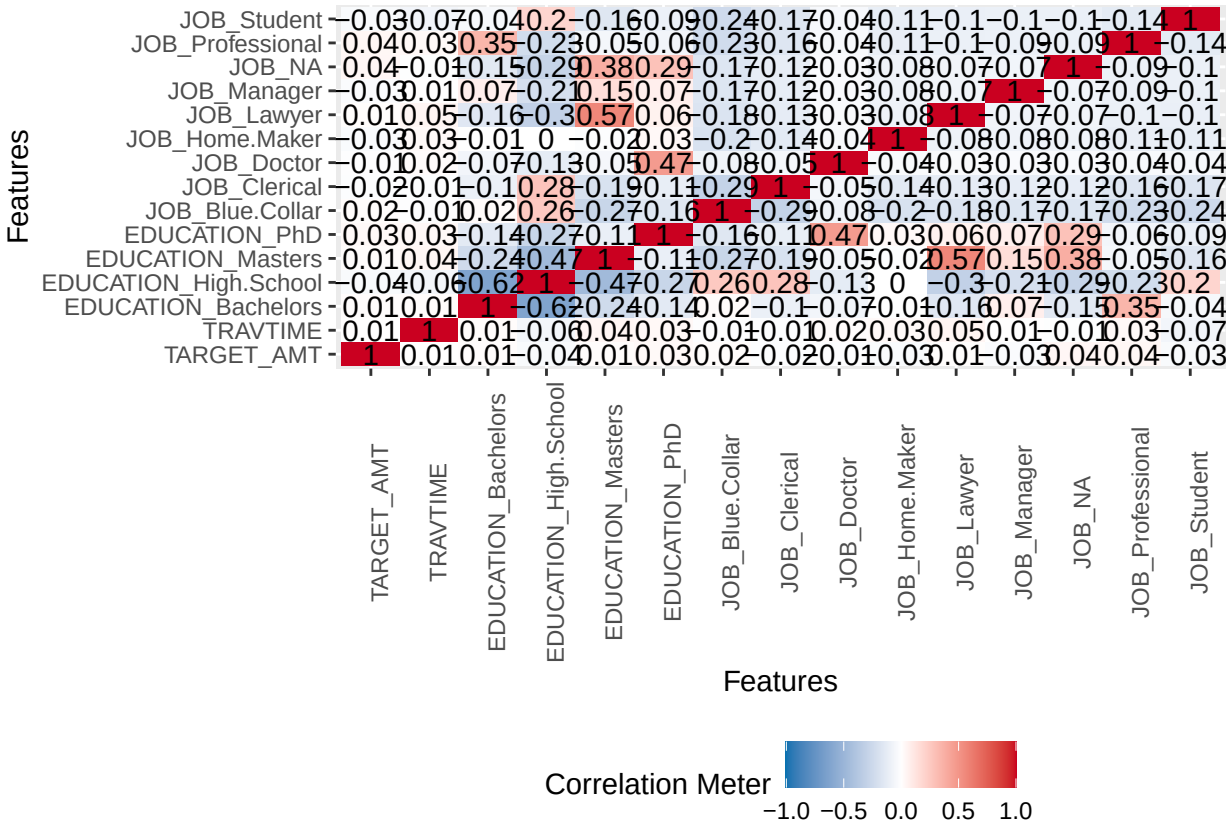
Correlation matrix

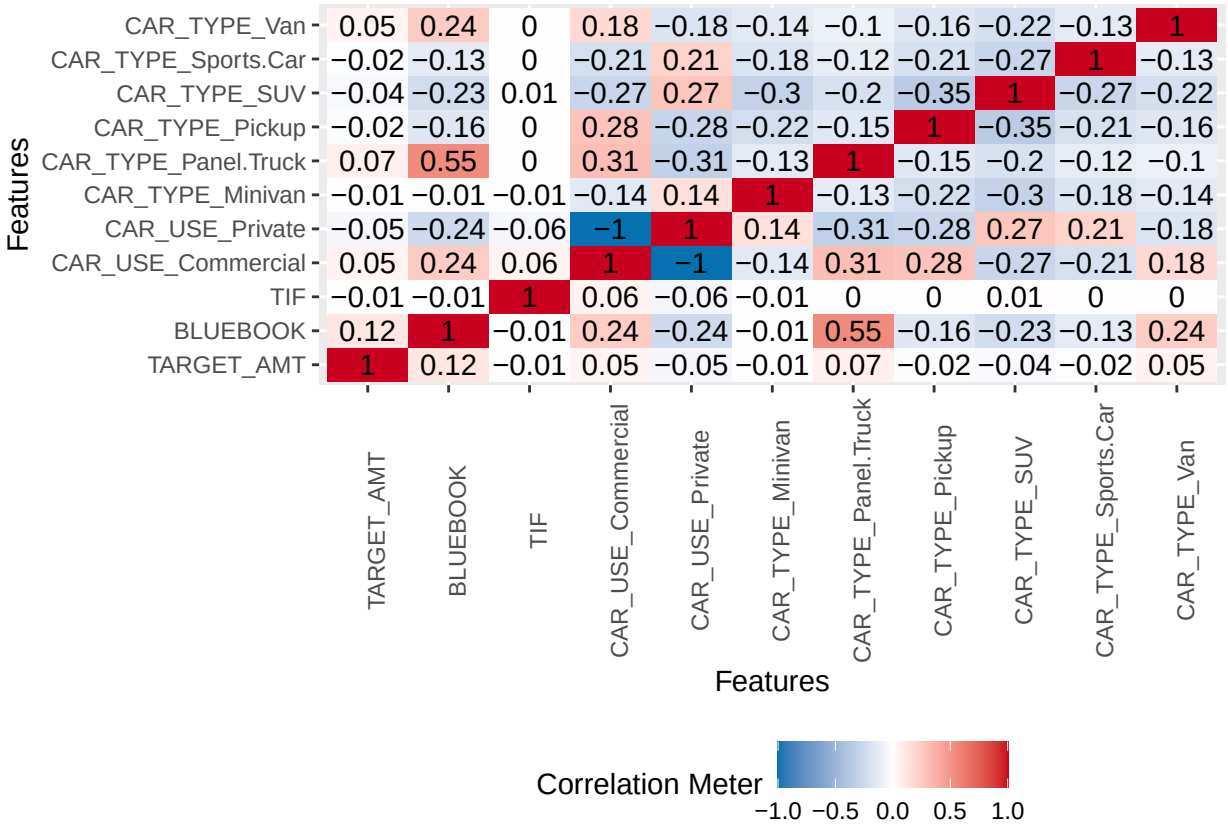
Similarly, a series of correlation matrices are given to quantify the linear regression variable's relationship with

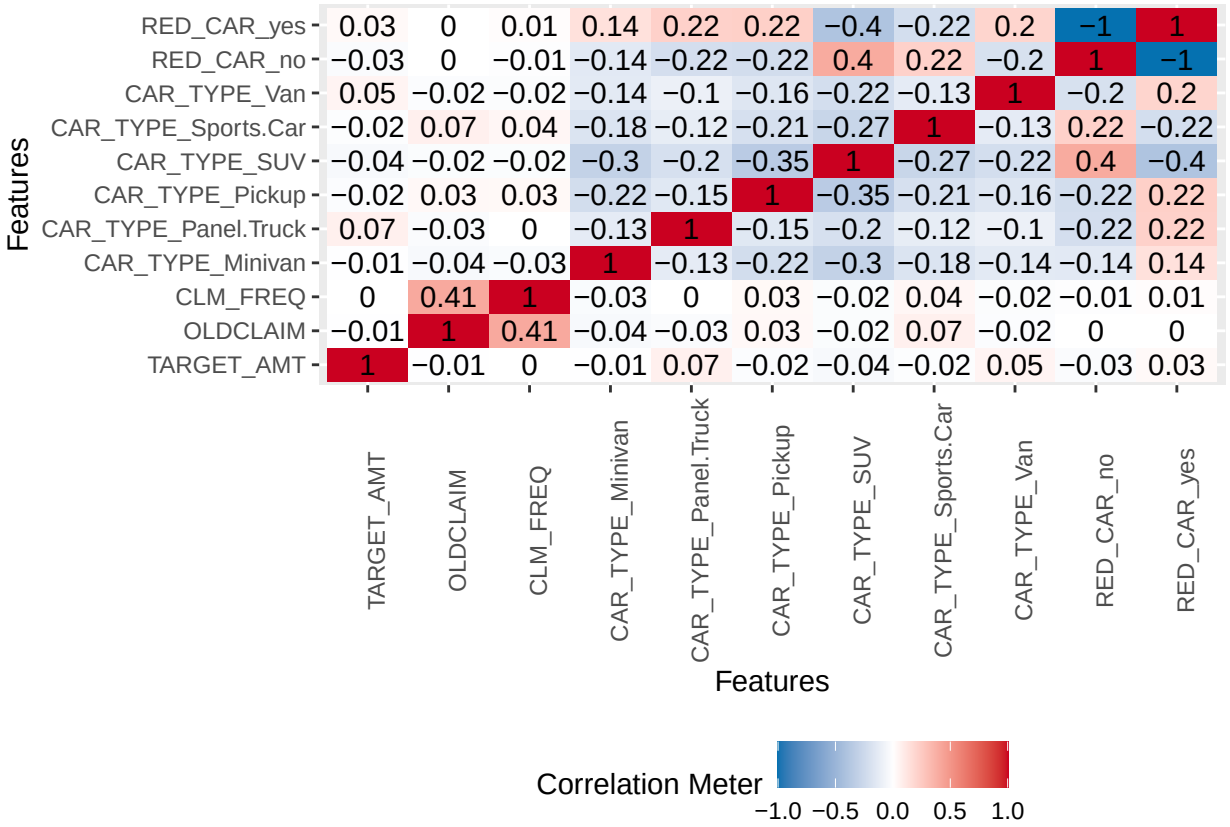


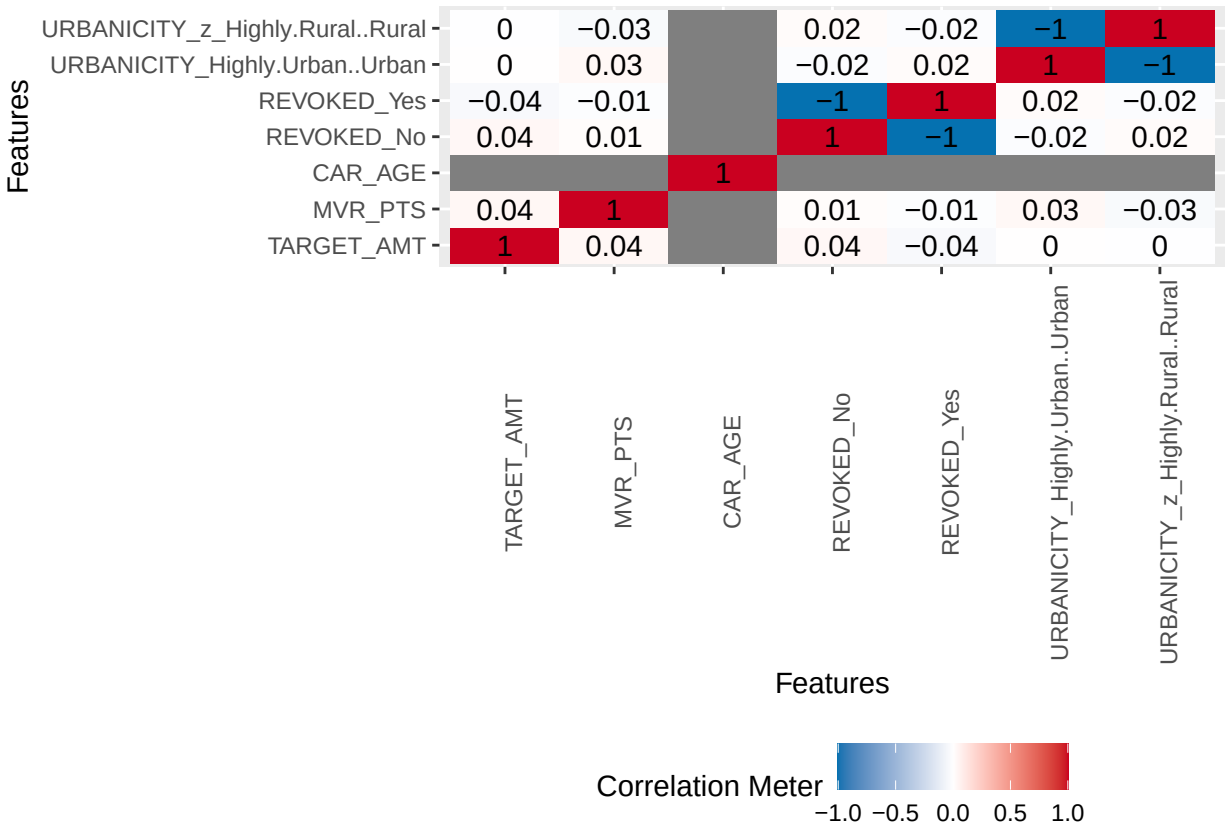
the target.


Features
EDUCATION_PhD 0.03 0.03 −0.03 −0.04 0.04 −0.01 0.01 −0.14 −0.27 −0.11 1
EDUCATION_Masters 0.01 0.04 −0.04 −0.02 0.02 0.01 −0.01 −0.24 −0.47 1 −0.11
EDUCATION_High.School −0.04 0 0 0.1 −0.1 −0.02 0.02 −0.62 1 −0.47 −0.27
EDUCATION_Bachelors 0.01 −0.05 0.05 −0.08 0.08 0.02 −0.02 1 −0.62 −0.24 −0.14
SEX_z_F −0.05 −0.1 0.1 −0.01 0.01 −1 1 −0.02 0.02 −0.01 0.01
SEX_M 0.05 0.1 −0.1 0.01 −0.01 1 −1 0.02 −0.02 0.01 −0.01
MSTATUS_z_No 0.03 −0.52 0.52 −1 1 −0.01 0.01 0.08 −0.1 0.02 0.04
MSTATUS_Yes −0.03 0.52 −0.52 1 −1 0.01 −0.01 −0.08 0.1 −0.02 −0.04
PARENT1_Yes 0.02 −1 1 −0.52 0.52 −0.1 0.1 0.05 0 −0.04 −0.03
PARENT1_No −0.02 1 −1 0.52 −0.52 0.1 −0.1 −0.05 0 0.04 0.03
HOME_VAL 1
TARGET_AMT 1 −0.02 0.02 −0.03 0.03 0.05 −0.05 0.01 −0.04 0.01 0.03
TARGET_AMT
HOME_VAL
PARENT1_No
PARENT1_Yes
MSTATUS_Yes
MSTATUS_z_No
SEX_M
SEX_z_F
EDUCATION_Bachelors
EDUCATION_High.School
EDUCATION_Masters
EDUCATION_PhD
Features
Correlation Meter
−1.0 −0.5 0.0 0.5 1.0


Features
JOB_Student
JOB_Professional
JOB_NA
JOB_Manager
JOB_Lawyer
JOB_Home.Maker
JOB_Doctor
JOB_Clerical
JOB_Blue.Collar
EDUCATION_PhD
EDUCATION_Masters
EDUCATION_High.School
EDUCATION_Bachelors
TRAVTIME
TARGET_AMT
Features
Correlation Meter
−1.0 −0.5 0.0 0.5 1.0


Features
CAR_TYPE_Van 0.05 0.24 0 0.18 −0.18 −0.14 −0.1 −0.16 −0.22 −0.13 1
CAR_TYPE_Sports.Car −0.02 −0.13 0 −0.21 0.21 −0.18 −0.12 −0.21 −0.27 1 −0.13
CAR_TYPE_SUV −0.04 −0.23 0.01 −0.27 0.27 −0.3 −0.2 −0.35 1 −0.27 −0.22
CAR_TYPE_Pickup −0.02 −0.16 0 0.28 −0.28 −0.22 −0.15 1 −0.35 −0.21 −0.16
CAR_TYPE_Panel.Truck 0.07 0.55 0 0.31 −0.31 −0.13 1 −0.15 −0.2 −0.12 −0.1
CAR_TYPE_Minivan −0.01 −0.01 −0.01 −0.14 0.14 1 −0.13 −0.22 −0.3 −0.18 −0.14
CAR_USE_Private −0.05 −0.24 −0.06 −1 1 0.14 −0.31 −0.28 0.27 0.21 −0.18
CAR_USE_Commercial 0.05 0.24 0.06 1 −1 −0.14 0.31 0.28 −0.27 −0.21 0.18
TIF −0.01 −0.01 1 0.06 −0.06 −0.01 0 0 0.01 0 0
BLUEBOOK 0.12 1 −0.01 0.24 −0.24 −0.01 0.55 −0.16 −0.23 −0.13 0.24
TARGET_AMT 1 0.12 −0.01 0.05 −0.05 −0.01 0.07 −0.02 −0.04 −0.02 0.05
TARGET_AMT
BLUEBOOK
TIF
CAR_USE_Commercial
CAR_USE_Private
CAR_TYPE_Minivan
CAR_TYPE_Panel.Truck
CAR_TYPE_Pickup
CAR_TYPE_SUV
CAR_TYPE_Sports.Car
CAR_TYPE_Van
Features
Correlation Meter
−1.0 −0.5 0.0 0.5 1.0


Features
RED_CAR_yes 0.03 0 0.01 0.14 0.22 0.22 −0.4 −0.22 0.2 −1 1
RED_CAR_no −0.03 0 −0.01 −0.14 −0.22 −0.22 0.4 0.22 −0.2 1 −1
CAR_TYPE_Van 0.05 −0.02 −0.02 −0.14 −0.1 −0.16 −0.22 −0.13 1 −0.2 0.2
CAR_TYPE_Sports.Car −0.02 0.07 0.04 −0.18 −0.12 −0.21 −0.27 1 −0.13 0.22 −0.22
CAR_TYPE_SUV −0.04 −0.02 −0.02 −0.3 −0.2 −0.35 1 −0.27 −0.22 0.4 −0.4
CAR_TYPE_Pickup −0.02 0.03 0.03 −0.22 −0.15 1 −0.35 −0.21 −0.16 −0.22 0.22
CAR_TYPE_Panel.Truck 0.07 −0.03 0 −0.13 1 −0.15 −0.2 −0.12 −0.1 −0.22 0.22
CAR_TYPE_Minivan −0.01 −0.04 −0.03 1 −0.13 −0.22 −0.3 −0.18 −0.14 −0.14 0.14
CLM_FREQ 0 0.41 1 −0.03 0 0.03 −0.02 0.04 −0.02 −0.01 0.01
OLDCLAIM −0.01 1 0.41 −0.04 −0.03 0.03 −0.02 0.07 −0.02 0 0
TARGET_AMT 1 −0.01 0 −0.01 0.07 −0.02 −0.04 −0.02 0.05 −0.03 0.03
TARGET_AMT
OLDCLAIM
CLM_FREQ
CAR_TYPE_Minivan
CAR_TYPE_Panel.Truck
CAR_TYPE_Pickup
CAR_TYPE_SUV
CAR_TYPE_Sports.Car
CAR_TYPE_Van
RED_CAR_no
RED_CAR_yes
Features
Correlation Meter
−1.0 −0.5 0.0 0.5 1.0



DATA PREPARATION

Based on the EDA several steps will be taken to prepare the dataset for modeling.

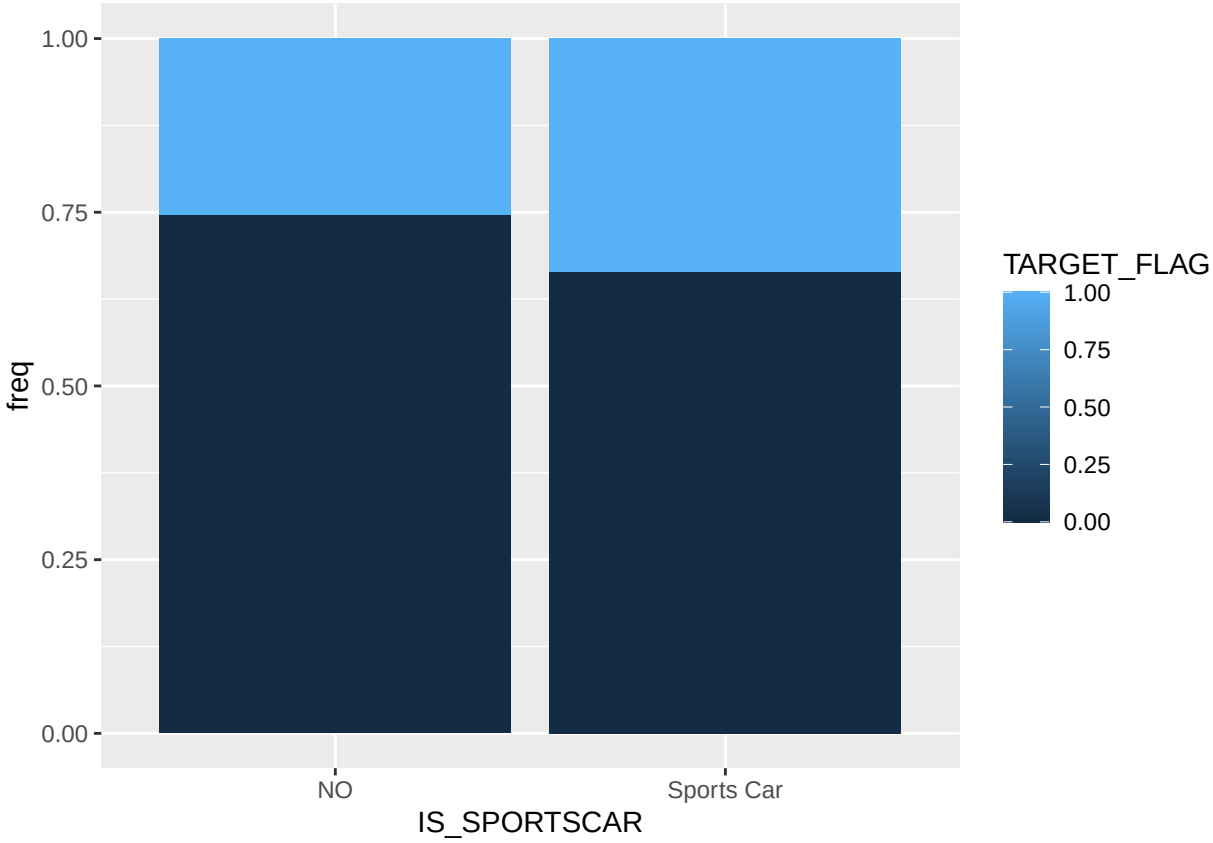
Imputate missing values.

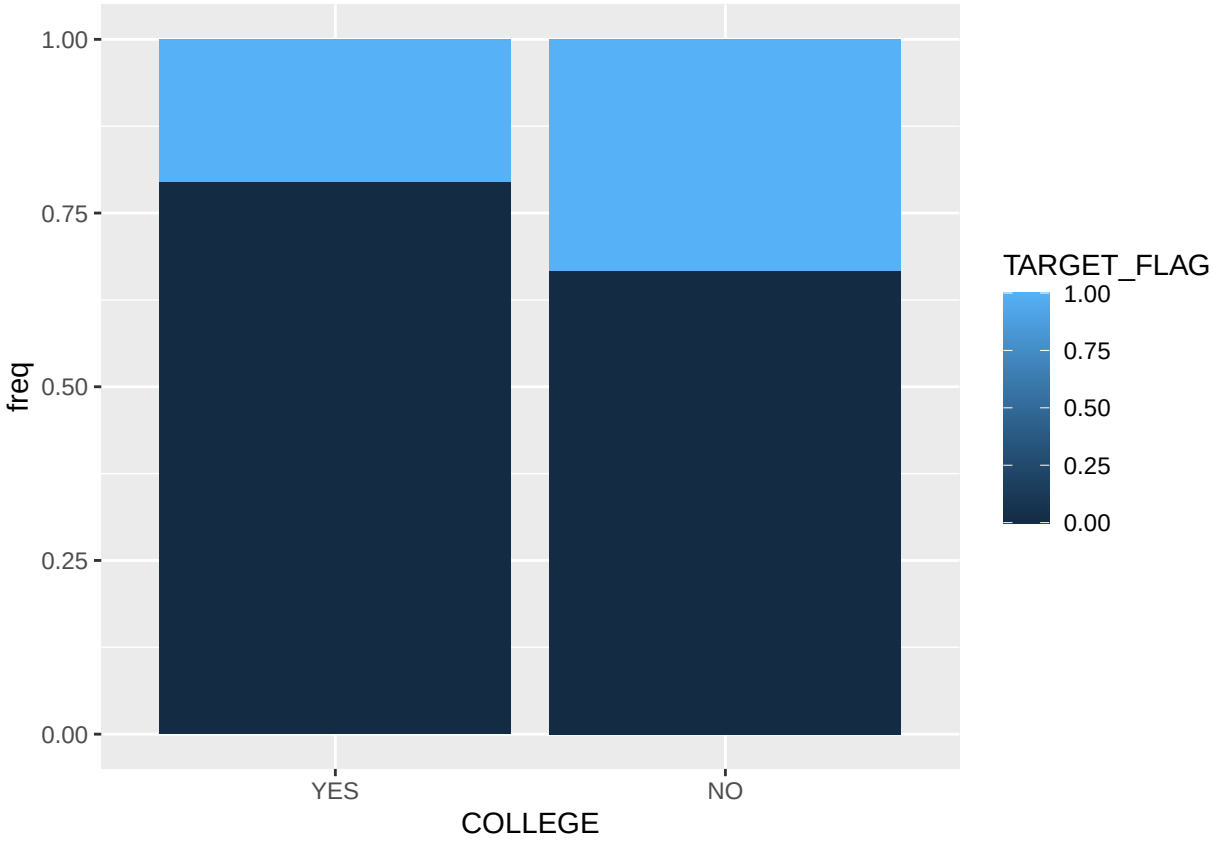
In this case, we will impute the mean for predictor variable.

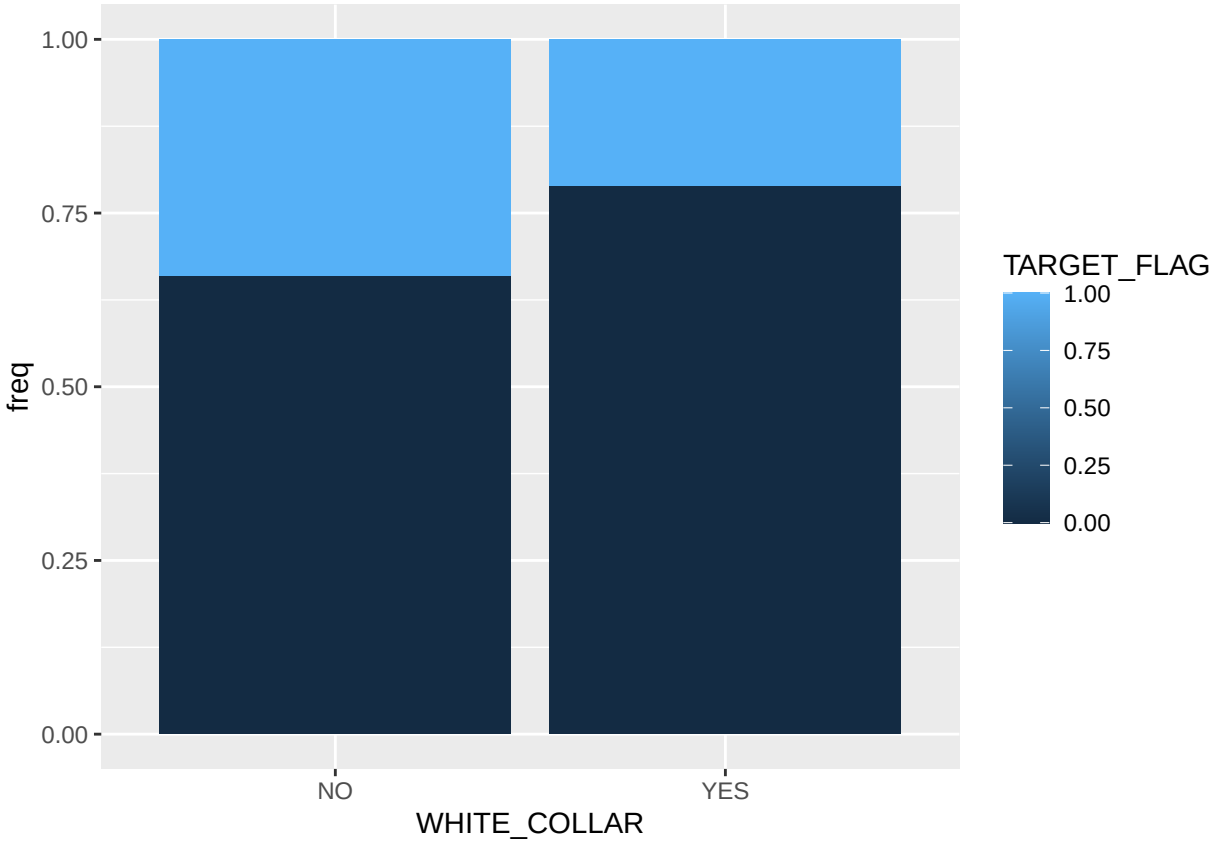
```
raw_training$AGE[is.na(raw_training$AGE)] <- mean(raw_training$AGE, na.rm=TRUE)
raw_training$YOJ[is.na(raw_training$YOJ)] <- mean(raw_training$YOJ, na.rm=TRUE)
raw_training$HOME_VAL[is.na(raw_training$HOME_VAL)] <- mean(raw_training$HOME_VAL, na.rm=TRUE)
raw_training$CAR_AGE[is.na(raw_training$CAR_AGE)] <- mean(raw_training$CAR_AGE, na.rm=TRUE)
raw_training$INCOME[is.na(raw_training$INCOME)] <- mean(raw_training$INCOME, na.rm=TRUE)
raw_training <- raw_training %>% na.omit()
```

Combine levels for factors with more than two levels

These new variables will be used in one of the models.




1.00
0.75
0.50
0.25
0.00
freq
YES
NO
COLLEGE
TARGET_FLAG
1.00
0.75
0.50
0.25
0.00



BUILD MODELS

The team created many binary logistic and linear regression models, using along with initial forays into data transformation. For each of the three models presented, the same 70 percent of the training data is used to evaluate the model and measure the predictions on the remaining 30 percent of the training data. The following sections explore the team's three primary methods for creating a model.

Logistic Regression:

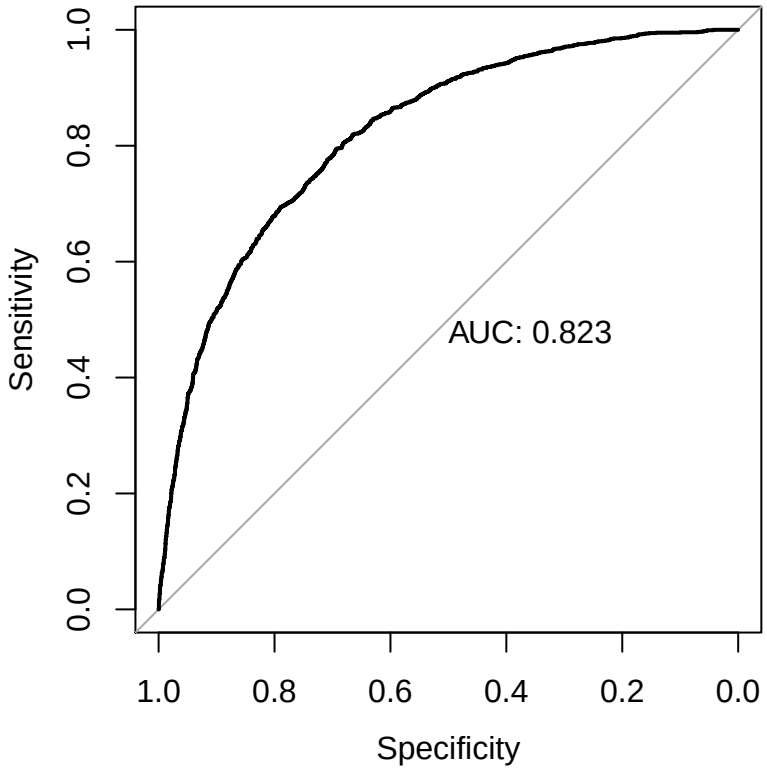
Three logistic regression models were built and were evaluated based on the metrics AIC, AUC, accuracy, precision, recall, sensitivity and specificity.

Logistic Model 1: Full Model

The raw model simply uses all the predictor variables in order to create a baseline for evaluation. The raw model uses the `glm` function to create the generalized linear model based on the `binomial` family and the link function `logit`.

```
##
## Call:
## glm(formula = TARGET_FLAG ~ ., family = binomial(link = "logit"),
##     data = train_flag)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5192  -0.6988  -0.3758   0.6213   2.9055
##
## Coefficients:
##                                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)                     -4.654e-01  3.348e-01  -1.390 0.164464
## KIDSDRIV                         3.446e-01  7.548e-02   4.566 4.97e-06 ***
## AGE                             -3.291e-03  4.986e-03  -0.660 0.509299
## HOMEKIDS                         3.674e-02  4.515e-02   0.814 0.415867
## YOJ                             -9.530e-03  1.060e-02  -0.899 0.368514
## INCOME                          -4.308e-06  1.498e-06  -2.876 0.004026 **
## PARENT1Yes                       3.930e-01  1.353e-01   2.904 0.003682 **
## HOME_VAL                        -1.630e-06  4.589e-07  -3.551 0.000384 ***
## MSTATUSz_No                      4.687e-01  1.049e-01   4.469 7.88e-06 ***
## SEXz_F                          -2.053e-01  1.417e-01  -1.448 0.147482
## EDUCATIONHigh School             3.616e-01  1.066e-01   3.391 0.000697 ***
## EDUCATIONMasters                 1.417e-02  1.743e-01   0.081 0.935188
## EDUCATIONPhD                     5.286e-01  2.314e-01   2.285 0.022333 *
## JOBClerical                      1.151e-01  1.266e-01   0.909 0.363483
## JOBDoctor                       -1.238e+00  3.690e-01  -3.354 0.000796 ***
## JOBHome Maker                   -2.034e-01  1.816e-01  -1.120 0.262789
## JOBLawyer                       -2.165e-01  2.255e-01  -0.960 0.336983
## JOBManager                      -8.561e-01  1.668e-01  -5.132 2.87e-07 ***
## JOBProfessional                 -1.624e-01  1.419e-01  -1.145 0.252406
## JOBStudent                      -2.474e-01  1.590e-01  -1.557 0.119581
## TRAVTIME                         1.294e-02  2.362e-03   5.479 4.28e-08 ***
## CAR_USEPrivate                  -8.312e-01  1.063e-01  -7.821 5.24e-15 ***
## BLUEBOOK                        -2.530e-05  6.607e-06  -3.829 0.000129 ***
## TIF                             -6.256e-02  9.187e-03  -6.810 9.77e-12 ***
## CAR_TYPEPanel Truck              4.774e-01  2.114e-01   2.258 0.023943 *
## CAR_TYPEPickup                   5.242e-01  1.218e-01   4.303 1.68e-05 ***
## CAR_TYPESports Car               1.049e+00  1.600e-01   6.555 5.55e-11 ***
## CAR_TYPESUV                      7.920e-01  1.371e-01   5.776 7.66e-09 ***
## CAR_TYPEVan                      4.124e-01  1.632e-01   2.527 0.011499 *
## RED_CARyes                      -3.736e-02  1.105e-01  -0.338 0.735301
## OLDCLAIM                        -1.231e-05  4.859e-06  -2.532 0.011331 *
## CLM_FREQ                         1.867e-01  3.584e-02   5.209 1.90e-07 ***
## REVOKEDYes                       9.205e-01  1.151e-01   7.997 1.28e-15 ***
## MVR_PTS                          1.102e-01  1.722e-02   6.397 1.58e-10 ***
## CAR_AGE                         -4.897e-04  9.456e-03  -0.052 0.958697
## URBANICITYz_Highly Rural/ Rural -2.365e+00  1.351e-01 -17.506  < 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: 6182.4  on 5344  degrees of freedom
## Residual deviance: 4697.7  on 5309  degrees of freedom
## AIC: 4769.7
##
## Number of Fisher Scoring iterations: 5
```



The output represents the raw model based on the `logit` link function. The resulting AIC for the raw model is 4769.7260152.

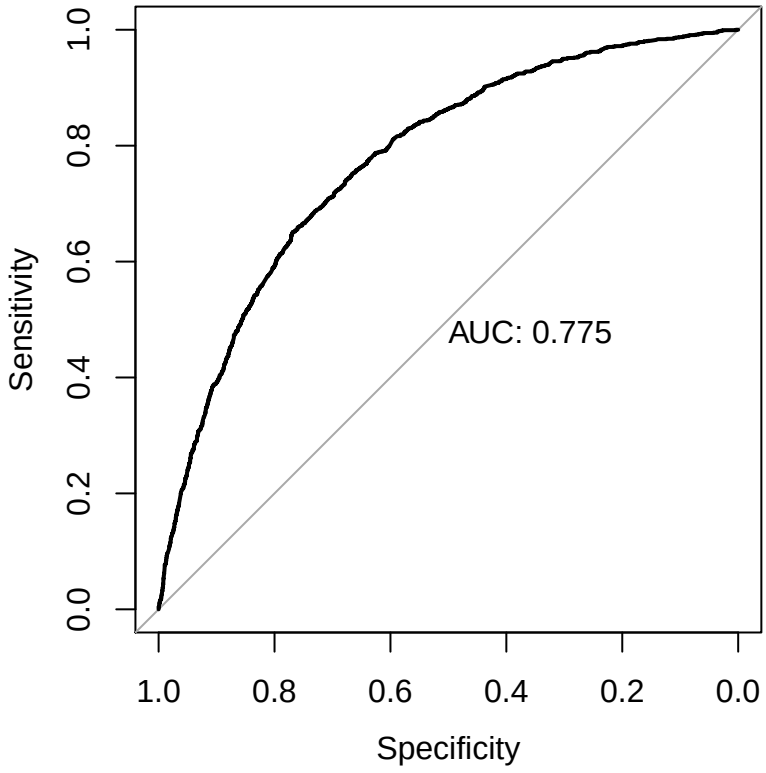
The AIC (Akaike's Information Criteria) statistic is used to compare different models to determine the best fit for the data. The AIC is based on the count of independent variables as input into the model in addition to the how well the model reproduces the data. The purpose of the AIC best-fit model is to explain the greatest amount of variation with the fewest number of independent predictor variables.

Logistic Model 2: Manual variable selection

For the second logistic model, variables were selected manually based on the figures provided in the Data Exploration section above.

```
##
## Call:
## glm(formula = TARGET_FLAG ~ INCOME + PARENT1 + MSTATUS + CAR_USE +
##     URBANICITY + IS_SPORTSCAR + COLLEGE, family = binomial(link = "logit"),
##     data = train, na.action = na.exclude)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0504  -0.7701  -0.4864   0.8358   3.0852
##
## Coefficients:
##                                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)                   -2.493e-01  1.103e-01  -2.260   0.0238 *
```

```
## INCOME                           -1.014e-05  1.063e-06  -9.540  < 2e-16 ***
## PARENT1Yes                        6.617e-01  1.056e-01   6.263 3.77e-10 ***
## MSTATUSz_No                       5.382e-01  7.988e-02   6.737 1.62e-11 ***
## CAR_USEPrivate                   -8.972e-01  7.251e-02 -12.374  < 2e-16 ***
## URBANICITYz_Highly Rural/ Rural -2.367e+00  1.275e-01 -18.559  < 2e-16 ***
## IS_SPORTSCARSports Car            5.397e-01  1.026e-01   5.258 1.46e-07 ***
## COLLEGENO                         5.459e-01  7.875e-02   6.932 4.14e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6182.4  on 5344  degrees of freedom
## Residual deviance: 5167.4  on 5337  degrees of freedom
## AIC: 5183.4
##
## Number of Fisher Scoring iterations: 5
```



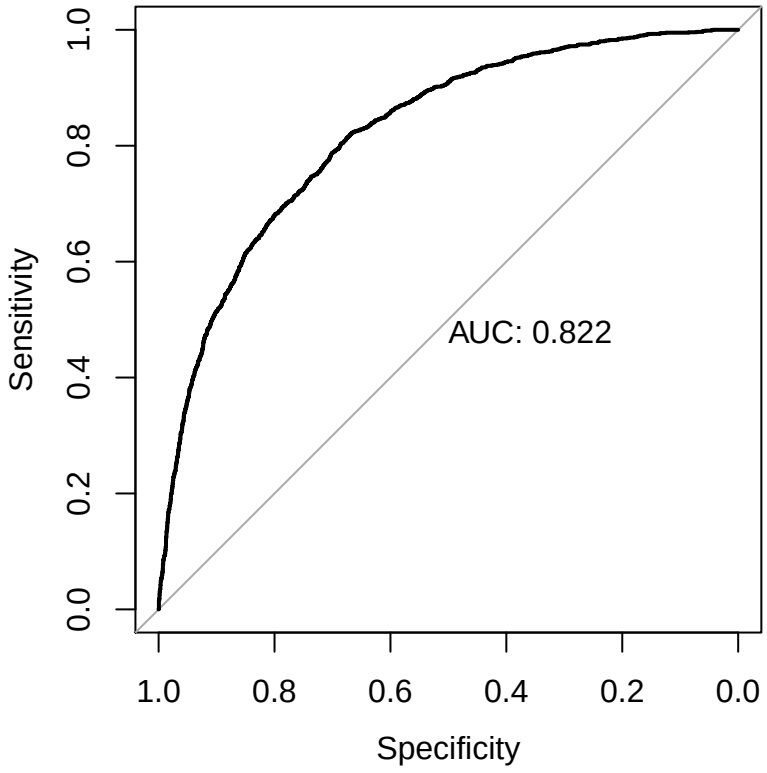
Logistic Model 3: Stepwise Model

This model uses the raw model created above with the addition of the `stepAIC` function from the `MASS` package. `stepAIC` is a common package used to help with feature selection. This version of the model uses this package with no additional constraints to train and evaluate model performance.

```
##
```

```
## Call:
## glm(formula = TARGET_FLAG ~ KIDSDRIV + INCOME + PARENT1 + HOME_VAL +
##     MSTATUS + EDUCATION + JOB + TRAVTIME + CAR_USE + BLUEBOOK +
##     TIF + CAR_TYPE + OLDCLAIM + CLM_FREQ + REVOKED + MVR_PTS +
##     URBANICITY, family = binomial(link = "logit"), data = train_flag)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5388  -0.6962  -0.3789   0.6222   2.9027
##
## Coefficients:
##                                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)                     -7.040e-01  2.172e-01  -3.242 0.001188 **
## KIDSDRIV                         3.647e-01  6.867e-02   5.312 1.09e-07 ***
## INCOME                          -4.400e-06  1.488e-06  -2.958 0.003101 **
## PARENT1Yes                       4.664e-01  1.158e-01   4.028 5.62e-05 ***
## HOME_VAL                        -1.675e-06  4.569e-07  -3.667 0.000246 ***
## MSTATUSz_No                      4.504e-01  9.942e-02   4.530 5.90e-06 ***
## EDUCATIONHigh School             3.662e-01  9.879e-02   3.707 0.000210 ***
## EDUCATIONMasters                 3.368e-03  1.669e-01   0.020 0.983905
## EDUCATIONPhD                     5.175e-01  2.269e-01   2.280 0.022585 *
## JOBClerical                      1.167e-01  1.262e-01   0.924 0.355359
## JOBDoctor                       -1.236e+00  3.682e-01  -3.356 0.000791 ***
## JOBHome Maker                   -1.902e-01  1.729e-01  -1.100 0.271200
## JOBLawyer                       -2.192e-01  2.251e-01  -0.974 0.330152
## JOBManager                      -8.580e-01  1.665e-01  -5.154 2.54e-07 ***
## JOBProfessional                 -1.663e-01  1.416e-01  -1.175 0.240023
## JOBStudent                      -2.015e-01  1.519e-01  -1.326 0.184700
## TRAVTIME                         1.291e-02  2.358e-03   5.476 4.35e-08 ***
## CAR_USEPrivate                  -8.308e-01  1.061e-01  -7.833 4.77e-15 ***
## BLUEBOOK                        -2.985e-05  5.995e-06  -4.980 6.36e-07 ***
## TIF                             -6.211e-02  9.175e-03  -6.770 1.29e-11 ***
## CAR_TYPEPanel Truck              5.893e-01  1.990e-01   2.961 0.003066 **
## CAR_TYPEPickup                   5.227e-01  1.216e-01   4.299 1.72e-05 ***
## CAR_TYPESports Car               9.183e-01  1.301e-01   7.056 1.71e-12 ***
## CAR_TYPESUV                      6.652e-01  1.041e-01   6.392 1.64e-10 ***
## CAR_TYPEVan                      4.804e-01  1.574e-01   3.053 0.002264 **
## OLDCLAIM                        -1.250e-05  4.856e-06  -2.575 0.010030 *
## CLM_FREQ                         1.866e-01  3.581e-02   5.211 1.88e-07 ***
## REVOKEDYes                       9.269e-01  1.150e-01   8.057 7.80e-16 ***
## MVR_PTS                          1.113e-01  1.719e-02   6.475 9.45e-11 ***
## URBANICITYz_Highly Rural/ Rural -2.364e+00  1.350e-01 -17.511  < 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: 6182.4  on 5344  degrees of freedom
## Residual deviance: 4702.0  on 5315  degrees of freedom
## AIC: 4762
##
## Number of Fisher Scoring iterations: 5

## [1] 4761.967
```



Linear Regression:

In this section models are built to try to predict the amount of claims that were made. We start by filtering out observations were there was no claim made (ie TARGET_FLAG=0). The training data decreases from 5345 cases to 1417 cases.

Linear Model #1

```
##
## Call:
## lm(formula = TARGET_AMT ~ . - TARGET_FLAG, data = train2_claims)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -7964  -3079  -1490    407  99792
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    4.013e+03  1.610e+03   2.493  0.01279 *
## KIDSDRIV                      -2.261e+02  3.765e+02  -0.600  0.54828
## AGE                            9.827e+00  2.454e+01   0.400  0.68885
## HOMEKIDS                      -1.066e+02  2.424e+02  -0.440  0.66007
## YOJ                            2.031e+01  5.308e+01   0.383  0.70209
## INCOME                         3.839e-03  7.756e-03   0.495  0.62072
```

```
## PARENT1Yes                         6.814e+02  7.022e+02   0.970  0.33199
## HOME_VAL                           1.055e-03  2.543e-03   0.415  0.67840
## MSTATUSz_No                       -6.172e+00  5.949e+02  -0.010  0.99172
## SEXz_F                            -3.116e+02  5.842e+02  -0.533  0.59392
## TRAVTIME                           3.770e+00  1.324e+01   0.285  0.77594
## CAR_USEPrivate                    -1.430e+02  5.023e+02  -0.285  0.77586
## BLUEBOOK                           9.165e-02  2.887e-02   3.175  0.00153 **
## TIF                               -9.381e+00  5.103e+01  -0.184  0.85418
## RED_CARyes                         2.534e+02  6.021e+02   0.421  0.67393
## OLDCLAIM                           2.112e-02  2.626e-02   0.804  0.42133
## CLM_FREQ                          -1.432e+02  1.901e+02  -0.753  0.45146
## REVOKEDYes                        -1.030e+03  6.170e+02  -1.670  0.09521 .
## MVR_PTS                            1.212e+02  8.425e+01   1.438  0.15057
## CAR_AGE                           -6.446e+01  4.929e+01  -1.308  0.19115
## URBANICITYz_Highly Rural/ Rural    9.575e+00  8.830e+02   0.011  0.99135
## IS_SPORTSCARSports Car             8.164e+01  6.043e+02   0.135  0.89256
## COLLEGENO                         -3.049e+02  5.572e+02  -0.547  0.58429
## WHITE_COLLARYES                    2.248e+02  4.998e+02   0.450  0.65303
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7510 on 1393 degrees of freedom
## Multiple R-squared:  0.02059,    Adjusted R-squared:  0.004423
## F-statistic: 1.274 on 23 and 1393 DF,  p-value: 0.1736
```

In the full linear regression model 1, we notice that the f-statistic appeared to be significant, but r-squared value was 0.004423 which indicates poor predictive ability of the variables to the target.

Other metrics to evaluate the full linear model were calculated and will be compared to second linear model in the next section.

Linear Model #2

This model uses the raw model created above with the addition of the `stepAIC` function from the `MASS` package. `stepAIC` is a common package used to help with feature selection. This version of the model uses this package with no additional constraints to train and evaluate model performance.

```
##
## Call:
## lm(formula = TARGET_AMT ~ BLUEBOOK + REVOKED, data = train2_claims)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -7780  -3013  -1533    254 101341
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4327.09860  417.18104  10.372  < 2e-16 ***
## BLUEBOOK       0.10592    0.02605   4.065 5.06e-05 ***
## REVOKEDYes  -788.11956  488.96686  -1.612    0.107
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 7480 on 1414 degrees of freedom
## Multiple R-squared:  0.01356,    Adjusted R-squared:  0.01217
## F-statistic: 9.721 on 2 and 1414 DF,  p-value: 6.415e-05
```

In the linear regression model 2, we notice that r-squared value was 0.01217 which is also suggests a poor fit. Only two variables were determined to be significant.

Metrics for the second model were calculated and will be evaluated in the next section.

SELECT MODELS

Logistic Model

All the models will be compared in order to select the model with the best fit in order to produce the most accurate results. The metrics we will be focused on are accuracy, AIC, and AUC (area under the curve). The models compared were the original raw model which included all variables. Next was a stepwise model which minimizes AIC in order to determine the variables which are necessary to include. The last model was a manual backwards stepwise model with only one variable different than the stepwise model.

Accuracy and Classicition error rate

We see that the full model and the stepwise model perform similarly.

	Full Model	Manuallly selected	AIC Stepwise
Accuracy	0.7864629	0.7633188	0.7860262
Classification Error Rate	0.2135371	0.2366812	0.2139738
Sensitivity	0.4016667	0.2716667	0.4000000
Specificity	0.9230769	0.9378698	0.9230769
Precision	0.6495957	0.6082090	0.6486486
Recall	0.4016667	0.2716667	0.4000000
F1	0.4963955	0.3755760	0.4948454
AIC	4769.7260152	5183.4176901	4761.9668851
Deviance	4697.7260152	5167.4176901	4701.9668851
AUC	0.8227562	0.7750307	0.8224540

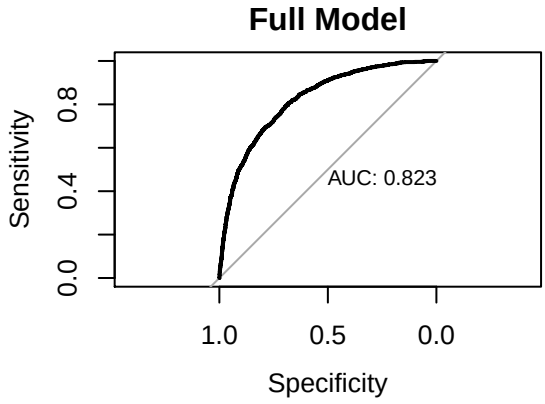
Deviance residuals

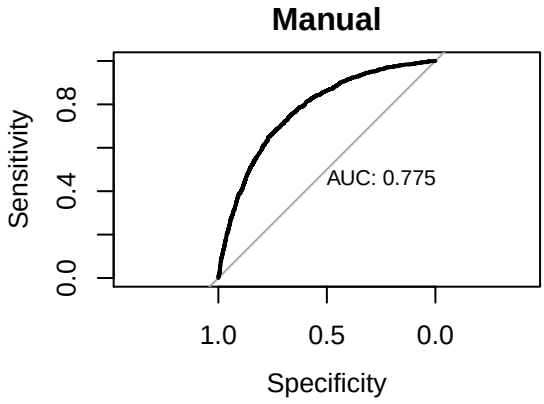
To further assess the models, the deviance residuals are compared. Based on the distribution, the better model will produce deviance residuals centered at zero and more symmetrical. There is right skew in each of the models suggesting possible outliers, but the stepwise model performs best.

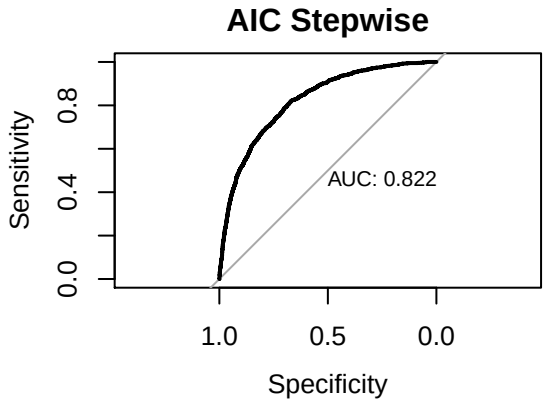
```
##                Min.   1st Qu.    Median        Mean  3rd Qu.      Max.
## full     -23.881864 -1.276561 -1.073165 -0.01926109 1.212861  68.10191
## manual    -8.183076 -1.345194 -1.125547  0.03503387 1.418024 116.66871
## stepwise -25.095542 -1.274275 -1.074427 -0.01451063 1.213556  67.54150
```

ROC Plot

The ROC plots for each of the models indicates very similar performance among the models.







The selected model is the AIC stepwise model. It performs comparably to the full model in each metric and is a better model since it contains fewer variables and avoids overfitting.

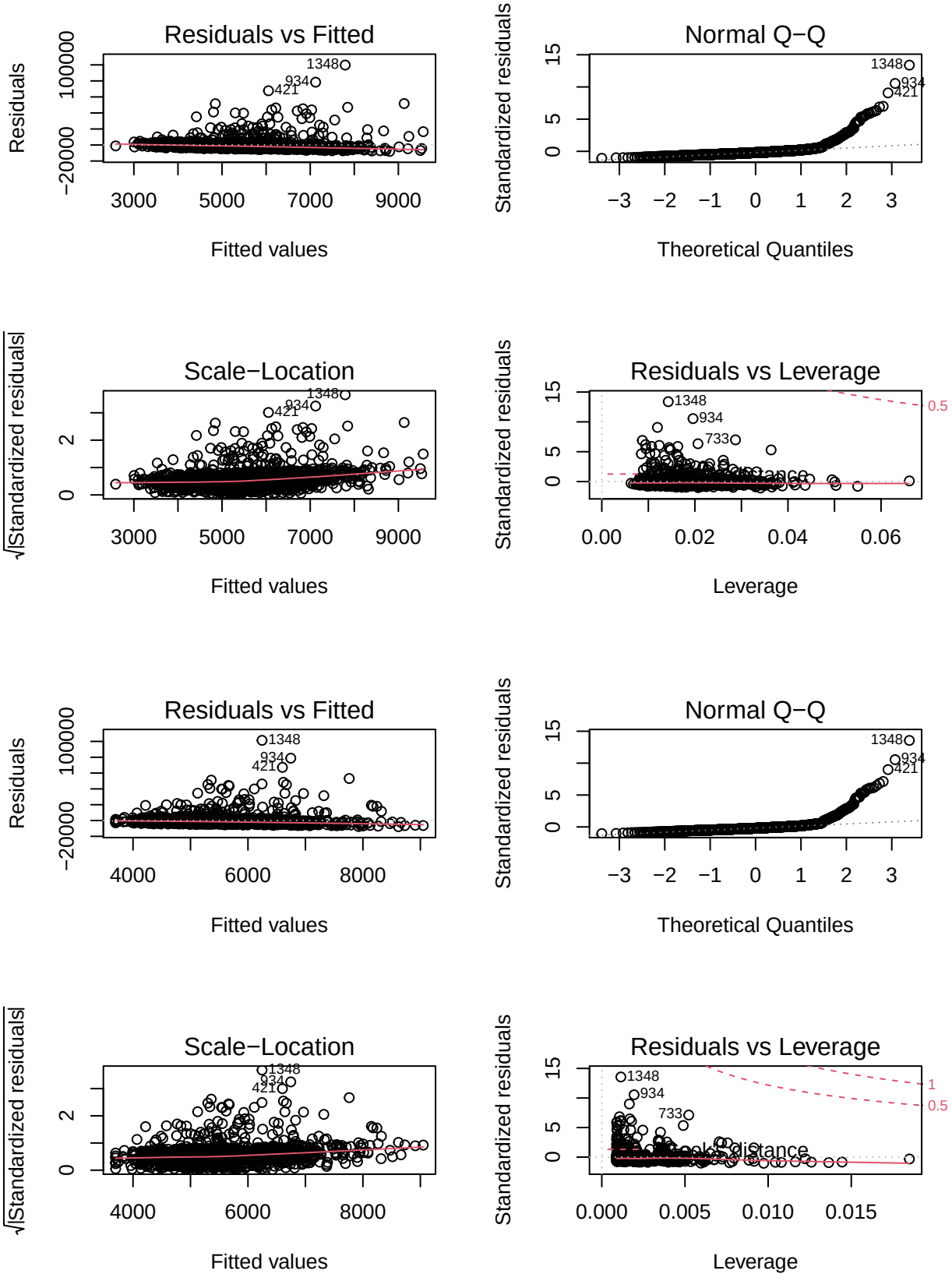
Linear Model

The linear models were evaluated in terms of R-squared and Root Mean Square Error(RMSE), as well as diagnostic plots that evaluate constant variance of residuals, normality of errors, homoscedasticity, and leverage of outliers.

R-squared and RMSE

```
##              rmse        R.sq
## Full     6452.592 0.004422936
## Stepwise 6449.390 0.012167336
```

Diagnostic Plots



The stepwise model uses performs better in RMSE, R-squared, and in all diagnostic plots and is a better model than the full model, and uses only two variables, BLUEBOOK and REVOKED. There are issues with the model - residuals are not normally distributed and only a small fraction of the target can be explained by the predictors. The variables of this dataset can be used to predict whether or not a claim will be made, but struggle with predicting how much the claim will be for.

Predictions

Output of the logistic model is available in file `insurance_predictions_Logistic.csv`. Output of the linear available in file `insurance_predictions_Linear.csv`.

Appendix

R statistical programming code:

```
library(recommenderlab)
library(tidyverse)
library(Metrics)
library(kableExtra)
library(gridExtra)
library(rmdformats)
library(caTools)
library(formattable)
library(mice)
library(naniar)
library(reshape)
library(corrplot)
library(caret)
library(knitr)
library(scales)
library(gplots)
library(MASS)
library(pROC)
library(Hmisc)
library(DataExplorer)
```

Overview

This assignment attempts to 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.

The objective is to build both a 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.

```r
# Load dataframes without index column
raw_training <- read.csv("https://raw.githubusercontent.com/Zchen116/data-621/master/insurance_training_
evaluation <- read.csv("https://raw.githubusercontent.com/Zchen116/data-621/master/insurance-evaluation-
```

The structure of the training data indicates 8161 records with 26 variables, 23 predictor variables and 2 target variables along with the index variable. Before diving into the data, unwanted characters in the dataset (dollar signs, commas, etc) were removed and datatypes were changed. Code for changes made to the dataset are available in the Appendix at the end of this report.

DATA EXPLORATION

The following exploratory data methods present a picture of the data to capture the distribution of the data and potential correlation with the target variable. The techniques used explore a summary of the variables,the distribution of each predictor variable against the target variable, density plot of each predictor variable against the target variable, along with a correlation plot across all the features.

Structure of Data

```
str(raw_training)
```

Missing Values were present in several of the variables:

-AGE -YOJ -CAR_AGE -HOME_VAL -INCOME

EDA for logistic regression

Boxplots

The dodged boxplot of each numeric variable against the target variable highlights differences between target boxes which could mean the variable is useful for prediction of the logistic model. A dodged boxplot without overlapping boxes likely indicates a correlation in the value of the predictor variable to the target classes.

```
raw_training %>%
dplyr::select_if(is.numeric) %>%
  gather("attribute", "value", -TARGET_FLAG) %>%
  ggplot(aes(x=value, fill=factor(TARGET_FLAG)))+
    geom_boxplot(position = 'dodge')+
    facet_wrap(~attribute, scales="free")
```

Not many of the variables show distinct differences in response value. CLM_FREQ has the largest discrepency.

Density plots

Similar to the boxplots, the density plots are another tool to identify which numeric predictor variables likely have a strong correlation with the target variable, and can suggest which variables are good to include in the logistic regression model.

```
# density plots
raw_training %>%
  dplyr::select_if(is.numeric) %>%
  gather("attribute", "value", -TARGET_FLAG) %>%
  ggplot(aes(x=value, fill=factor(TARGET_FLAG)))+
    geom_density(position = 'dodge', alpha=0.4)+
    facet_wrap(~attribute, scales="free")
```

The density plots show distributions that are almost identical for each variable comparing the target. Based on this visualization, none of the numeric variables would make good predictors for the logistic regression model.

Barplots

Next, factor variables were evaluated for logistic regression suitability using stacked barplots

```
###PARENT1
raw_training %>%
  group_by(PARENT1, TARGET_FLAG) %>%  dplyr::summarize(n=n()) %>% mutate(freq=n/sum(n)) %>%
  ggplot(aes(x=PARENT1, y=freq, fill=TARGET_FLAG)) + geom_col()

### MSTATUS
raw_training %>%
  group_by(MSTATUS, TARGET_FLAG) %>%  dplyr::summarize(n=n()) %>% mutate(freq=n/sum(n)) %>%
  ggplot(aes(x=MSTATUS, y=freq, fill=TARGET_FLAG)) + geom_col()

### SEX
raw_training %>%
  group_by(SEX, TARGET_FLAG) %>%  dplyr::summarize(n=n()) %>% mutate(freq=n/sum(n)) %>%
  ggplot(aes(x=SEX, y=freq, fill=TARGET_FLAG)) + geom_col()

### JOB
raw_training %>%
  group_by(JOB, TARGET_FLAG) %>%  dplyr::summarize(n=n()) %>% mutate(freq=n/sum(n)) %>%
  ggplot(aes(x=JOB, y=freq, fill=TARGET_FLAG)) + geom_col()

### Car Type
raw_training %>%
  group_by(CAR_TYPE, TARGET_FLAG) %>%  dplyr::summarize(n=n()) %>% mutate(freq=n/sum(n)) %>%
  ggplot(aes(x=CAR_TYPE, y=freq, fill=TARGET_FLAG)) + geom_col()

### Education
raw_training %>%
  group_by(EDUCATION, TARGET_FLAG) %>%  dplyr::summarize(n=n()) %>% mutate(freq=n/sum(n)) %>%
  ggplot(aes(x=EDUCATION, y=freq, fill=TARGET_FLAG)) + geom_col()

### CAR USE
```

```
raw_training %>%
  group_by(CAR_USE, TARGET_FLAG) %>%  dplyr::summarize(n=n()) %>% mutate(freq=n/sum(n)) %>%
  ggplot(aes(x=CAR_USE, y=freq, fill=TARGET_FLAG)) + geom_col()

### URBANICITY
raw_training %>%
  group_by(URBANICITY, TARGET_FLAG) %>%  dplyr::summarize(n=n()) %>% mutate(freq=n/sum(n)) %>%
  ggplot(aes(x=URBANICITY, y=freq, fill=TARGET_FLAG)) + geom_col()

### REVOKED
raw_training %>%
  group_by(REVOKED, TARGET_FLAG) %>%  dplyr::summarize(n=n()) %>% mutate(freq=n/sum(n)) %>%
  ggplot(aes(x=REVOKED, y=freq, fill=TARGET_FLAG)) + geom_col()

### RED_CAR
raw_training %>%
  group_by(RED_CAR, TARGET_FLAG) %>%  dplyr::summarize(n=n()) %>% mutate(freq=n/sum(n)) %>%
  ggplot(aes(x=RED_CAR, y=freq, fill=TARGET_FLAG)) + geom_col()
```

Factor variables that might make good predictors for logistic regression: -PARENT1 -MSTATUS -EDUCATION -CAR_USE -URBANCITY

It might also be useful to combine levels for factors with more than two levels. That will be done in the Data Preparation section.

EDA for linear regression

For the linear regression model, only the data where a claim was actually made is analyzed (ie TARGET_FLAG=0)

```
train.lm <- raw_training %>% filter(TARGET_FLAG==1) %>%
  dplyr::select(-TARGET_FLAG)
```

Density plots

```
par(mfrow = c(3, 3))

datasub = melt(raw_training)
ggplot(datasub, aes(x= value)) +
    geom_density(fill='blue') + facet_wrap(~variable, scales = 'free')
```

Scatterplot matrix

A scatterplot matrix is generated to evaluate the relationship between the numeric variables and the linear regression target. The matrix is divided over three plots for clarity.

```
### Scatter plot matrix
train.lm %>%
  dplyr::select_if(is.numeric) %>%
  dplyr::select(1:4) %>%
```

```
  pairs(lower.panel=NULL)

train.lm %>%
  dplyr::select_if(is.numeric) %>%
  dplyr::select(c(1,5:7)) %>%
  pairs(lower.panel=NULL)

train.lm %>%
  dplyr::select_if(is.numeric) %>%
  dplyr::select(c(1,8:10)) %>%
  pairs(lower.panel=NULL)
```

Correlation matrix

Similarly, a series of correlation matrices are given to quantify the linear regression variable's relationship with the target.

```
# corr matrix
train.lm %>%
  dplyr::select(c(1,2:7)) %>%
plot_correlation()

train.lm %>%
  dplyr::select(c(1,7:11)) %>%
plot_correlation()

train.lm %>%
  dplyr::select(c(1,11:13)) %>%
plot_correlation()

train.lm %>%
  dplyr::select(c(1, 14:17)) %>%
plot_correlation()

train.lm %>%
  dplyr::select(c(1,17:20)) %>%
plot_correlation()

train.lm %>%
  dplyr::select(c(1,21:24)) %>%
plot_correlation()
```

DATA PREPARATION

Based on the EDA several steps will be taken to prepare the dataset for modeling.

Imputate missing values.

In this case, we will impute the mean for predictor variable.

```
raw_training$AGE[is.na(raw_training$AGE)] <- mean(raw_training$AGE, na.rm=TRUE)
raw_training$YOJ[is.na(raw_training$YOJ)] <- mean(raw_training$YOJ, na.rm=TRUE)
raw_training$HOME_VAL[is.na(raw_training$HOME_VAL)] <- mean(raw_training$HOME_VAL, na.rm=TRUE)
raw_training$CAR_AGE[is.na(raw_training$CAR_AGE)] <- mean(raw_training$CAR_AGE, na.rm=TRUE)
raw_training$INCOME[is.na(raw_training$INCOME)] <- mean(raw_training$INCOME, na.rm=TRUE)
raw_training <- raw_training %>% na.omit()
```

Combine levels for factors with more than two levels

These new variables will be used in one of the models.

```
raw_training$IS_SPORTSCAR <- fct_collapse(raw_training$CAR_TYPE, YES="Sport_Car", NO=c("Minivan", "Panel

raw_training$COLLEGE <- fct_collapse(raw_training$EDUCATION, YES=c("Bachelors","Masters", "PhD"), NO="Hi

raw_training$WHITE_COLLAR <- fct_collapse(raw_training$JOB, YES=c("Clerical","Doctor", "Lawyer", "Manage
par(mfrow=c(2,2))
### IS_SPORTSCAR
raw_training %>%
  group_by(IS_SPORTSCAR, TARGET_FLAG) %>%  dplyr::summarize(n=n()) %>% mutate(freq=n/sum(n)) %>%
  ggplot(aes(x=IS_SPORTSCAR, y=freq, fill=TARGET_FLAG)) + geom_col()

### COLLEGE
raw_training %>%
  group_by(COLLEGE, TARGET_FLAG) %>%  dplyr::summarize(n=n()) %>% mutate(freq=n/sum(n)) %>%
  ggplot(aes(x=COLLEGE, y=freq, fill=TARGET_FLAG)) + geom_col()

### WHITE_COLLAR
raw_training %>%
  group_by(WHITE_COLLAR, TARGET_FLAG) %>%  dplyr::summarize(n=n()) %>% mutate(freq=n/sum(n)) %>%
  ggplot(aes(x=WHITE_COLLAR, y=freq, fill=TARGET_FLAG)) + geom_col()
```

BUILD MODELS

The team created many binary logistic and linear regression models, using along with initial forays into data transformation. For each of the three models presented, the same 70 percent of the training data is used to evaluate the model and measure the predictions on the remaining 30 percent of the training data. The following sections explore the team's three primary methods for creating a model.

```
set.seed(123)
trainIndex <-createDataPartition(raw_training$TARGET_FLAG, p = 0.7,list = FALSE,times = 1)
train <- raw_training[trainIndex,]
test <- raw_training[-trainIndex,]
```

Logistic Regression:

Three logistic regression models were built and were evaluated based on the metrics AIC, AUC, accuracy, precision, recall, sensitivity and specificity.

Logistic Model 1: Full Model

The raw model simply uses all the predictor variables in order to create a baseline for evaluation. The raw model uses the `glm` function to create the generalized linear model based on the `binomial` family and the link function `logit`.

```
train_flag <- train %>% dplyr::select(-c(IS_SPORTSCAR,COLLEGE,WHITE_COLLAR))
train_flag <- train_flag[ ,-which(names(train)%in% c("TARGET_AMT", "TARGET_FLAG_FAC"))]

glm.full <- glm(formula = TARGET_FLAG ~ ., data = train_flag, family = "binomial" (link="logit"))
summary(glm.full)
```

```
## AIC
#glm.full$aic

## use the test data set to make predicts and calculate metrics from the confusion matrix
glm_full.probs <- predict(glm.full,type="response", newdata=test)
glm_predict.full <- ifelse(glm_full.probs > 0.5, '1','0')
attach(test)
#table(glm_predict.full, test$TARGET_FLAG)

# now can use the caret function
cm.full <- caret::confusionMatrix(factor(glm_predict.full), factor(test$TARGET_FLAG), positive='1')
#cm.full$table

# print metrics
#c(cm.full$overall[c(1)], cm.full$byClass[c(1,2,5,6,7)])

# ROC and AUC
par(pty="s")
roc.full <- roc(train$TARGET_FLAG, glm.full$fitted.values, plot=TRUE, print.auc=TRUE)
#glm.full$aic
```

The output represents the raw model based on the `logit` link function. The resulting AIC for the raw model is 4769.7260152.

The AIC (Akaike's Information Criteria) statistic is used to compare different models to determine the best fit for the data. The AIC is based on the count of independent variables as input into the model in addition to the how well the model reproduces the data. The purpose of the AIC best-fit model is to explain the greatest amount of variation with the fewest number of independent predictor variables.

Logistic Model 2: Manual variable selection

For the second logistic model, variables were selected manually based on the figures provided in the Data Exploration section above.

```
glm.manual <-  glm(TARGET_FLAG ~ INCOME + PARENT1 +  MSTATUS+ CAR_USE+ URBANICITY+ IS_SPORTSCAR+ COLLEGE
                   family = "binomial"(link="logit"), na.action=na.exclude,
                   data=train)
summary(glm.manual)
```

```
## AIC
#glm.manual$aic

## use the test data set to make predicts and calculate metrics from the confusion matrix
glm_manual.probs <- predict(glm.manual,type="response", newdata=test)
glm_predict.manual <- ifelse(glm_manual.probs > 0.5, '1','0')
attach(test)
#table(glm_predict.manual, test$TARGET_FLAG)

# now can use the caret function
cm.manual <- caret::confusionMatrix(factor(glm_predict.manual), factor(test$TARGET_FLAG), positive='1')
#cm.manual$table

# print metrics
#c(cm.manual$overall[c(1)], cm.manual$byClass[c(1,2,5,6,7)])

# ROC and AUC
par(pty="s")
roc.manual <- roc(train$TARGET_FLAG, glm.manual$fitted.values, plot=TRUE, print.auc=TRUE)
#glm.manual$aic
```

Logistic Model 3: Stepwise Model

This model uses the raw model created above with the addition of the `stepAIC` function from the `MASS` package. `stepAIC` is a common package used to help with feature selection. This version of the model uses this package with no additional constraints to train and evaluate model performance.

```
train_flag <- train %>% dplyr::select(-c(IS_SPORTSCAR,COLLEGE,WHITE_COLLAR))


train_flag <- train_flag[ ,-which(names(train)%in% c("TARGET_AMT", "TARGET_FLAG_FAC"))]
glm.stepwise <- glm(TARGET_FLAG~., data = train_flag, family = "binomial"(link="logit"))%>%
  stepAIC(trace = F)
summary(glm.stepwise)
```

```
## AIC
glm.stepwise$aic

## use the test data set to make predicts and calculate metrics from the confusion matrix
glm_stepwise.probs <- predict(glm.stepwise,type="response", newdata=test)
glm_stepwise.manual <- ifelse(glm_stepwise.probs > 0.5, '1','0')
attach(test)
#table(glm_stepwise.manual, test$TARGET_FLAG)

# now can use the caret function
cm.stepwise <- caret::confusionMatrix(factor(glm_stepwise.manual), factor(test$TARGET_FLAG), positive='
#cm.stepwise$table

# print metrics
#c(cm.stepwise$overall[c(1)], cm.stepwise$byClass[c(1,2,5,6,7)])

# ROC and AUC
```

```
par(pty="s")
roc.stepwise <- roc(train$TARGET_FLAG, glm.stepwise$fitted.values, plot=TRUE, print.auc=TRUE)
#glm.stepwise$aic
```

Linear Regression:

In this section models are built to try to predict the amount of claims that were made. We start by filtering out observations were there was no claim made (ie TARGET_FLAG=0). The training data decreases from 5345 cases to 1417 cases.

```
train2_claims <- train %>% filter(TARGET_FLAG == 1) %>%
  dplyr::select(-c(JOB,EDUCATION,CAR_TYPE))
test_claims <- test %>% filter(TARGET_FLAG == 1) %>%
  dplyr::select(-c(JOB,EDUCATION,CAR_TYPE))
```

Linear Model #1

```
linearmodel1 <- lm(TARGET_AMT ~ .-TARGET_FLAG, data = train2_claims)
summary(linearmodel1)
```

In the full linear regression model 1, we notice that the f-statistic appeared to be significant, but r-squared value was 0.004423 which indicates poor predictive ability of the variables to the target.

Other metrics to evaluate the full linear model were calculated and will be compared to second linear model in the next section.

```
#Calculate RMSE and R.Squared for the raw model
test_amt <- test[ ,-which(names(test)%in% c("TARGET_FLAG", "TARGET_FLAG_FAC"))]
predictions <- predict.lm(linearmodel1, newdata = test_amt)
rmse <-rmse(pull(test_amt, TARGET_AMT), predictions)
R.sq <-summary(linearmodel1)$adj.r.squared
lm.full <-cbind(rmse, R.sq)
```

Linear Model #2

This model uses the raw model created above with the addition of the `stepAIC` function from the `MASS` package. `stepAIC` is a common package used to help with feature selection. This version of the model uses this package with no additional constraints to train and evaluate model performance.

```
linearmodel2 <- stepAIC(linearmodel1, trace = F)
summary(linearmodel2)
```

In the linear regression model 2, we notice that r-squared value was 0.01217 which is also suggests a poor fit. Only two variables were determined to be significant.

Metrics for the second model were calculated and will be evaluated in the next section.

```
#Calculate RMSE and R.Squared for the raw model
predictions_2 <-predict.lm(linearmodel2, newdata = test_amt)
rmse_2 <-rmse(pull(test_amt, TARGET_AMT), predictions_2)
R.sq_2 <-summary(linearmodel2)$adj.r.squared
lm.stepwise <-cbind(rmse_2, R.sq_2)
```

SELECT MODELS

Logistic Model

All the models will be compared in order to select the model with the best fit in order to produce the most accurate results. The metrics we will be focused on are accuracy, AIC, and AUC (area under the curve). The models compared were the original raw model which included all variables. Next was a stepwise model which minimizes AIC in order to determine the variables which are necessary to include. The last model was a manual backwards stepwise model with only one variable different than the stepwise model.

Accuracy and Classicition error rate

We see that the full model and the stepwise model perform similarly.

```
temp <- data.frame(cm.full$overall,
                   cm.manual$overall,
                   cm.stepwise$overall) %>%
  t() %>%
  data.frame() %>%
  dplyr::select(Accuracy) %>%
  mutate(`Classification Error Rate` = 1-Accuracy)
  rownames(temp)<- c("full","manual","stepwise")
```

```
eval <- data.frame(cm.full$byClass,
                   cm.manual$byClass,
                   cm.stepwise$byClass)
eval <- data.frame(t(eval)) %>%
  cbind(temp) %>%
  mutate(eval = c("Full Model", "AIC Stepwise", "Manual Backwards"))
rownames(eval)<- c("full","manual","stepwise")
```

```
eval <- dplyr::select(eval, Accuracy, `Classification Error Rate`, Sensitivity, Specificity, Precision,

# AIC is lower in the stepwise model suggesting it is closer to the "true" model
AIC.combined <- c(glm.full$aic, glm.manual$aic, glm.stepwise$aic)

# Residual Deviance are lower in the stepwise model
DEV.combined <- c(glm.full$deviance, glm.manual$deviance, glm.stepwise$deviance)

# Area under the curve is slightly better for the stepwise model
AUC.combined <- c(roc.full$auc, roc.manual$auc, roc.stepwise$auc)

eval <- cbind(eval, AIC=AIC.combined, Deviance=DEV.combined, AUC=AUC.combined)
```

```
rownames(eval) = c("Full Model", "Manuallly selected", "AIC Stepwise")

t_eval <- t(eval)
colnames(t_eval) <- rownames(eval)
rownames(t_eval) <- colnames(eval)

knitr::kable(t_eval)
```

Deviance residuals

To further assess the models, the deviance residuals are compared. Based on the distribution, the better model will produce deviance residuals centered at zero and more symmetrical. There is right skew in each of the models suggesting possible outliers, but the stepwise model performs best.

```
bind <- rbind(summary(glm.full$residuals), summary(glm.manual$residuals), summary(glm.stepwise$residuals
rownames(bind)<- c("full","manual","stepwise")
bind
```

ROC Plot

The ROC plots for each of the models indicates very similar performance among the models.

```
par(mfrow=c(2,2))
plot(roc.full, print.auc=TRUE, main="Full Model")
plot(roc.manual, print.auc=TRUE, main="Manual")
plot(roc.stepwise, print.auc=TRUE, main="AIC Stepwise")

par(mfrow=c(1,1))
```

The selected model is the AIC stepwise model. It performs comparably to the full model in each metric and is a better model since it contains fewer variables and avoids overfitting.

Linear Model

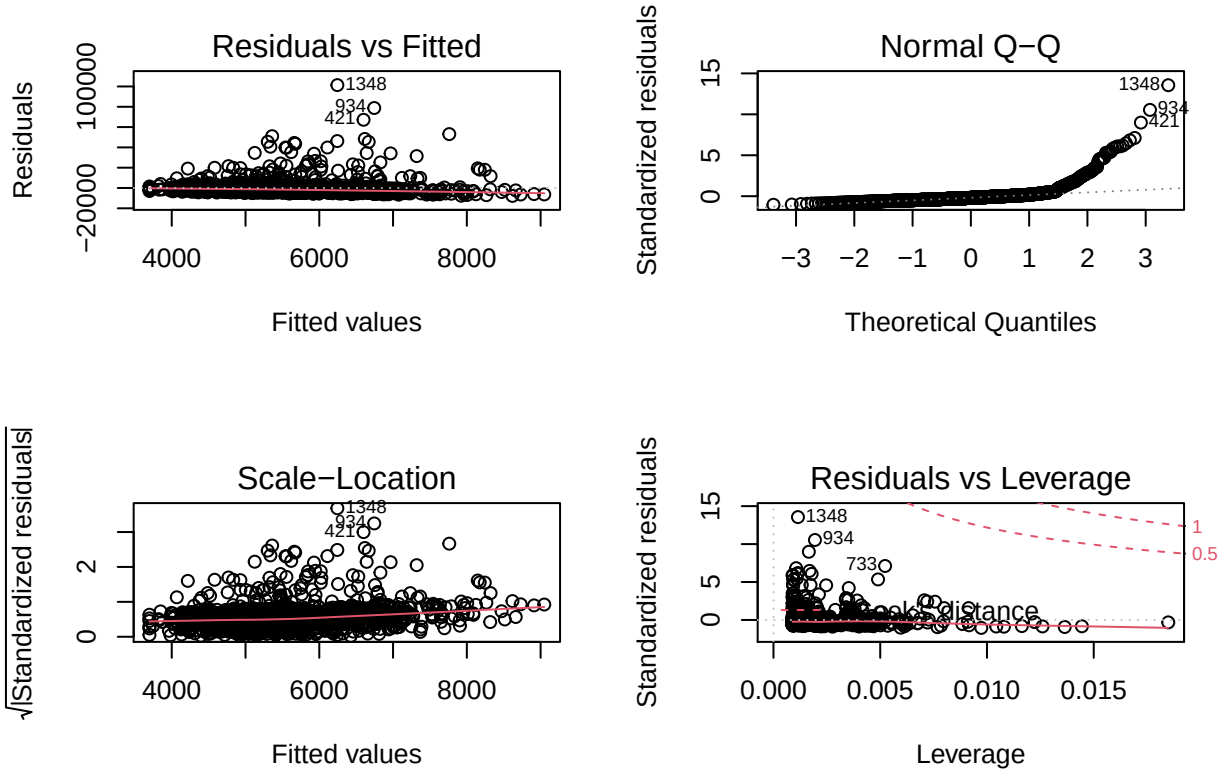
The linear models were evaluated in terms of R-squared and Root Mean Square Error(RMSE), as well as diagnostic plots that evaluate constant variance of residuals, normality of errors, homoscedasticity, and leverage of outliers.

R-squared and RMSE

```
bind <- rbind(lm.full,lm.stepwise)
rownames(bind) <- c("Full", "Stepwise")
bind
```

Diagnostic Plots

```
par(mfrow=c(2,2))
plot(linearmodel1)
```



The stepwise model uses performs better in RMSE, R-squared, and in all diagnostic plots and is a better model than the full model, and uses only two variables, BLUEBOOK and REVOKED. There are issues with the model - residuals are not normally distributed and only a small fraction of the target can be explained by the predictors. The variables of this dataset can be used to predict whether or not a claim will be made, but struggle with predicting how much the claim will be for.

Predictions

Output of the logistic model is available in file `insurance_predictions_Logistic.csv`. Output of the linear available in file `insurance_predictions_Linear.csv`.

```
evaluation <- evaluation[-c(1)]
evaluation$INCOME<-gsub("[\\$,]", "", evaluation$INCOME)
evaluation$HOME_VAL<-gsub("[\\$,]", "", evaluation$HOME_VAL)
evaluation$BLUEBOOK<-gsub("[\\$,]", "", evaluation$BLUEBOOK)
evaluation$OLDCLAIM<-gsub("[\\$,]", "", evaluation$OLDCLAIM)
evaluation$CAR_TYPE <- evaluation$CAR_TYPE %>%
  str_replace_all("z_","") %>%
  as.factor()
evaluation$EDUCATION <- evaluation$EDUCATION %>%
```

```
  str_replace_all("<","") %>%
  str_replace_all("z_","") %>%
  as.factor()
evaluation$JOB <- evaluation$JOB %>%
  str_replace_all("z_","") %>%
  as.factor()

evaluation$INCOME <- as.numeric(evaluation$INCOME)
evaluation$HOME_VAL <- as.numeric(evaluation$HOME_VAL)
evaluation$BLUEBOOK <- as.numeric(evaluation$BLUEBOOK)
evaluation$OLDCLAIM <- as.numeric(evaluation$OLDCLAIM)

evaluation$AGE<-impute(evaluation$AGE, median)
evaluation$YOJ<-impute(evaluation$YOJ, median)
evaluation$INCOME<-impute(evaluation$INCOME, median)
evaluation$CAR_AGE<-impute(evaluation$CAR_AGE, median)

# Remove "z_" prefix for clean up
levels(evaluation$MSTATUS)[levels(evaluation$MSTATUS)=="z_No"] <- "No"
levels(evaluation$SEX)[levels(evaluation$SEX)=="z_F"] <- "F"
levels(evaluation$EDUCATION)[levels(evaluation$EDUCATION)=="z_High School"] <- "High School"
levels(evaluation$JOB)[levels(evaluation$JOB)=="z_Blue Collar"] <- "Blue Collar"
levels(evaluation$CAR_TYPE)[levels(evaluation$CAR_TYPE)=="z_SUV"] <- "SUV"
levels(evaluation$URBANICITY)[levels(evaluation$URBANICITY)=="Highly Urban/ Urban"] <- "Urban"
levels(evaluation$URBANICITY)[levels(evaluation$URBANICITY)=="z_Highly Rural/ Rural"] <- "Rural"

evaluation$PTSAGE = evaluation$MVR_PTS/evaluation$AGE
eval_results = predict(glm.stepwise, evaluation, type = 'response')
eval_results = ifelse(eval_results > 0.5, 1, 0)
eval_amt = predict(linearmodel2, evaluation)

#write.csv(eval_results, "insurance_predictions_Logistic.csv", row.names = F)

#write.csv(eval_amt, "insurance_predictions_Linear.csv", row.names = F)
```