CUNY DATA 621 - Business Analytics and Data Mining

Homework 4 - Insurance

Walt Wells, 2018

Problem

Our goal is to explore, analyze and model a dataset represent customer records at an auto insurance company. The 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.

1. DATA EXPLORATION

First we'll do some basic data cleansing.

```
cleanMoney <- function(vector) {
    i <- gsub(",", "", vector)
    i <- as.numeric(gsub("[\\$,]", "", i))
    return(i)
}

train$INCOME <- cleanMoney(train$INCOME)
train$HOME_VAL <- cleanMoney(train$HOME_VAL)
train$BLUEBOOK <- cleanMoney(train$BLUEBOOK)
train$OLDCLAIM <- cleanMoney(train$OLDCLAIM)

test$INCOME <- cleanMoney(test$INCOME)
test$HOME_VAL <- cleanMoney(test$HOME_VAL)
test$BLUEBOOK <- cleanMoney(test$BLUEBOOK)
test$OLDCLAIM <- cleanMoney(test$OLDCLAIM)
```

Below we'll display a few basic EDA techniques to gain insight into our insurance dataset.

Basic Statistics

The data is 1.8 Mb in size. There are 8,161 rows and 25 columns (features). Of all 25 columns, 14 are discrete, 11 are continuous, and 0 are all missing. There are 970 missing values out of 204,025 data points.

	n	mean	sd	median	min	max	skew	kurtosis
TARGET_FLAG*	8161	1.263816e+00	4.407276e-01	1	1	2.0	1.0716614	-0.8516462
TARGET_AMT	8161	1.504325e+03	4.704027e+03	0	0	107586.1	8.7063034	112.2884386
KIDSDRIV	8161	1.710575e-01	5.115341e-01	0	0	4.0	3.3518374	11.7801916
AGE	8155	4.479031e+01	8.627589e+00	45	16	81.0	-0.0289889	-0.0617020
HOMEKIDS	8161	7.212351e-01	1.116323e+00	0	0	5.0	1.3411271	0.6489915
YOJ	7707	1.049929e+01	4.092474e+00	11	0	23.0	-1.2029676	1.1773410
INCOME	7716	6.189809e+04	4.757268e+04	54028	0	367030.0	1.1863166	2.1290163
PARENT1*	8161	1.131969e+00	3.384779e-01	1	1	2.0	2.1743561	2.7281589
HOME_VAL	7697	1.548673e+05	1.291238e+05	161160	0	885282.0	0.4885950	-0.0160838
MSTATUS*	8161	1.400319e+00	4.899929e-01	1	1	2.0	0.4068189	-1.8347231

	n	mean	sd	median	min	max	skew	kurtosis
SEX*	8161	1.536086e+00	4.987266e-01	2	1	2.0	-0.1446959	-1.9793056
EDUCATION*	8161	3.090675e+00	1.444856e+00	3	1	5.0	0.1162654	-1.3799674
JOB*	8161	5.687171e+00	2.681873e+00	6	1	9.0	-0.3067029	-1.2222635
TRAVTIME	8161	3.348572e+01	1.590833e+01	33	5	142.0	0.4468174	0.6643331
CAR_USE*	8161	1.628845e+00	4.831436e-01	2	1	2.0	-0.5332937	-1.7158080
BLUEBOOK	8161	1.570990e+04	8.419734e+03	14440	1500	69740.0	0.7942141	0.7913559
TIF	8161	5.351305e+00	4.146635e+00	4	1	25.0	0.8908120	0.4224940
CAR_TYPE*	8161	3.529714e+00	1.965357e+00	3	1	6.0	-0.0047181	-1.5165329
RED_CAR*	8161	1.291386e+00	4.544287e-01	1	1	2.0	0.9180255	-1.1573709
OLDCLAIM	8161	4.037076e+03	8.777139e+03	0	0	57037.0	3.1190400	9.8606583
CLM_FREQ	8161	7.985541e-01	1.158453e+00	0	0	5.0	1.2087985	0.2842890
REVOKED*	8161	1.122534e+00	3.279216e-01	1	1	2.0	2.3018899	3.2991013
MVR_PTS	8161	1.695503e+00	2.147112e+00	1	0	13.0	1.3478403	1.3754900
CAR_AGE	7651	8.328323e+00	5.700742e+00	8	-3	28.0	0.2819531	-0.7489756
URBANICITY*	8161	1.204509e+00	4.033673e-01	1	1	2.0	1.4649406	0.1460688

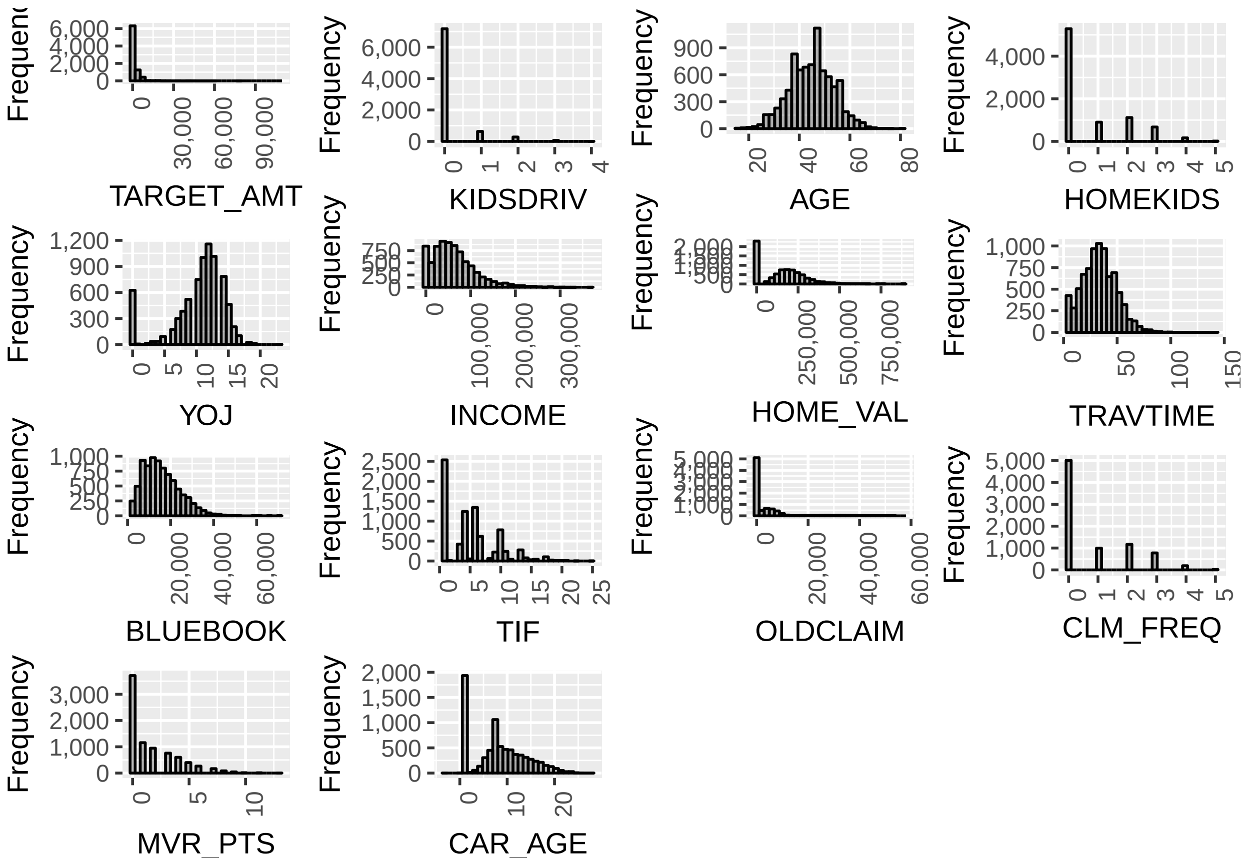
Compare Target in Training

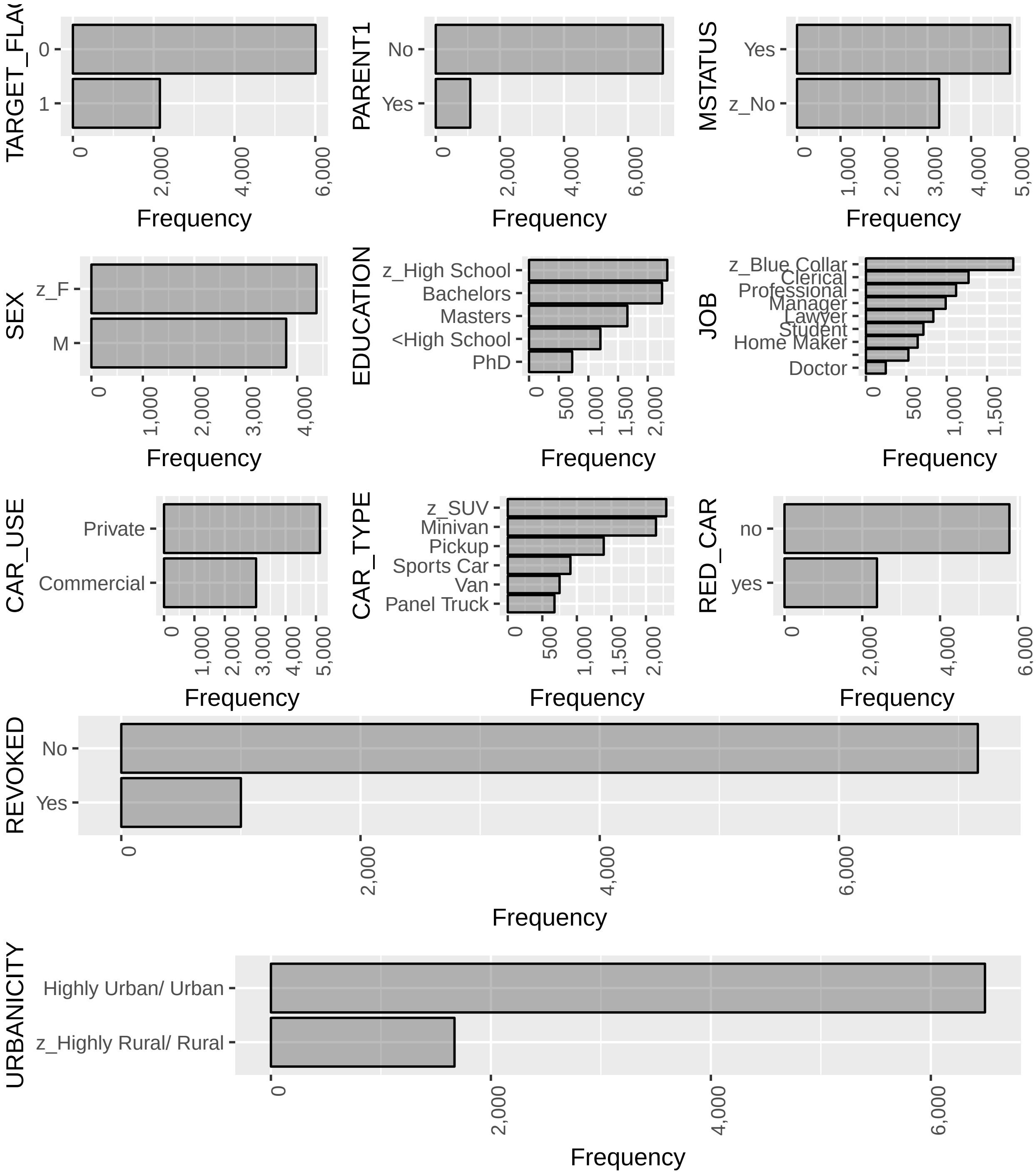
We make sure there are no issues with an inappropriate distribution of the target variable in our training data. We also confirm that the cases where the target amount is = 0 is equivalent to the number of target flags that were not claims.

Var1	Freq
0	6008
1	2153

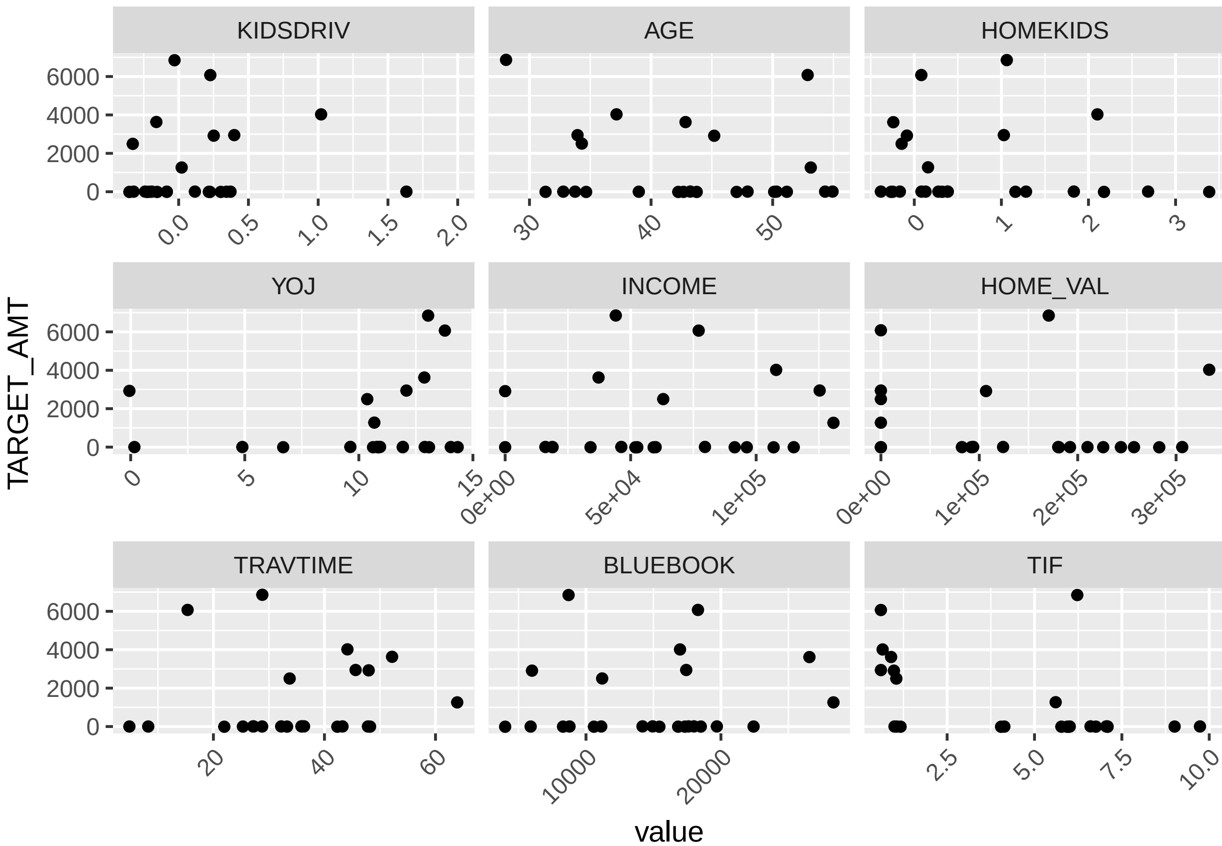
```
## [1] 6008
```

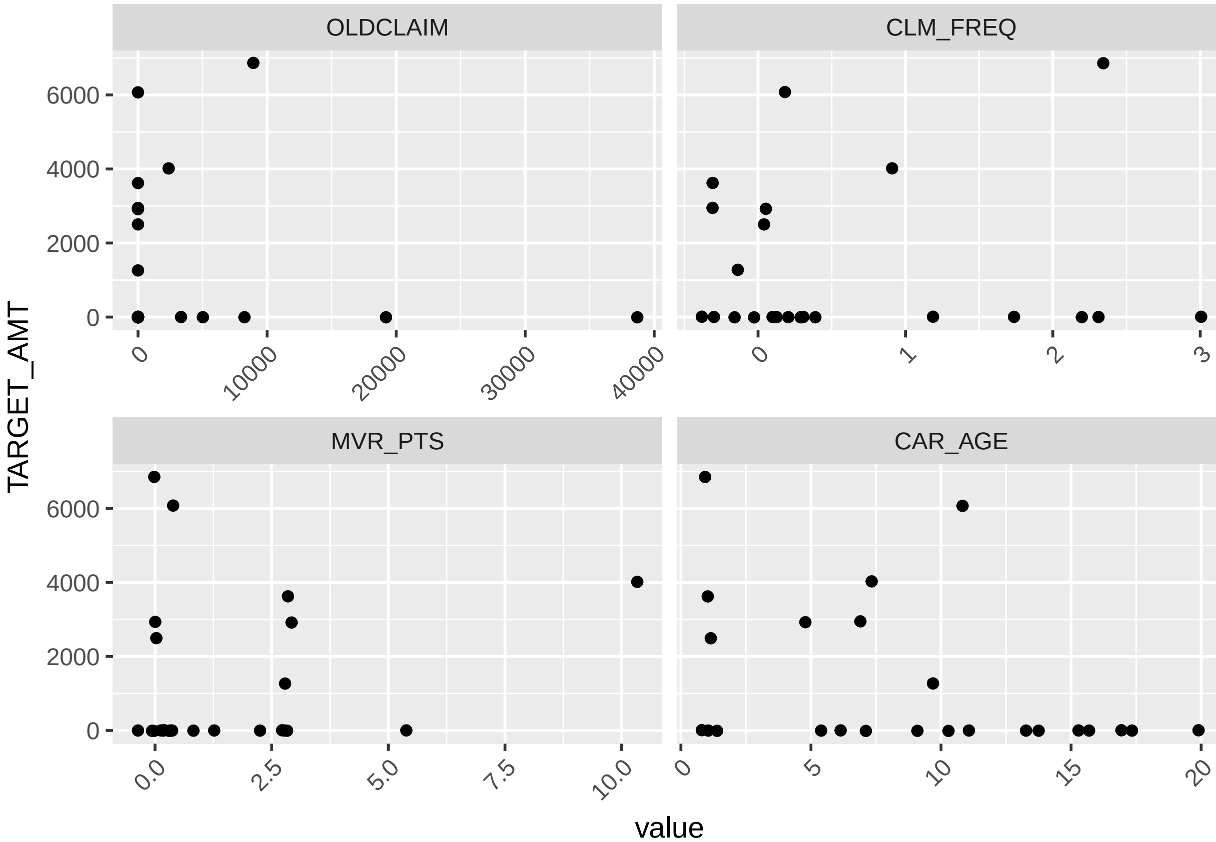
Histogram of Variables

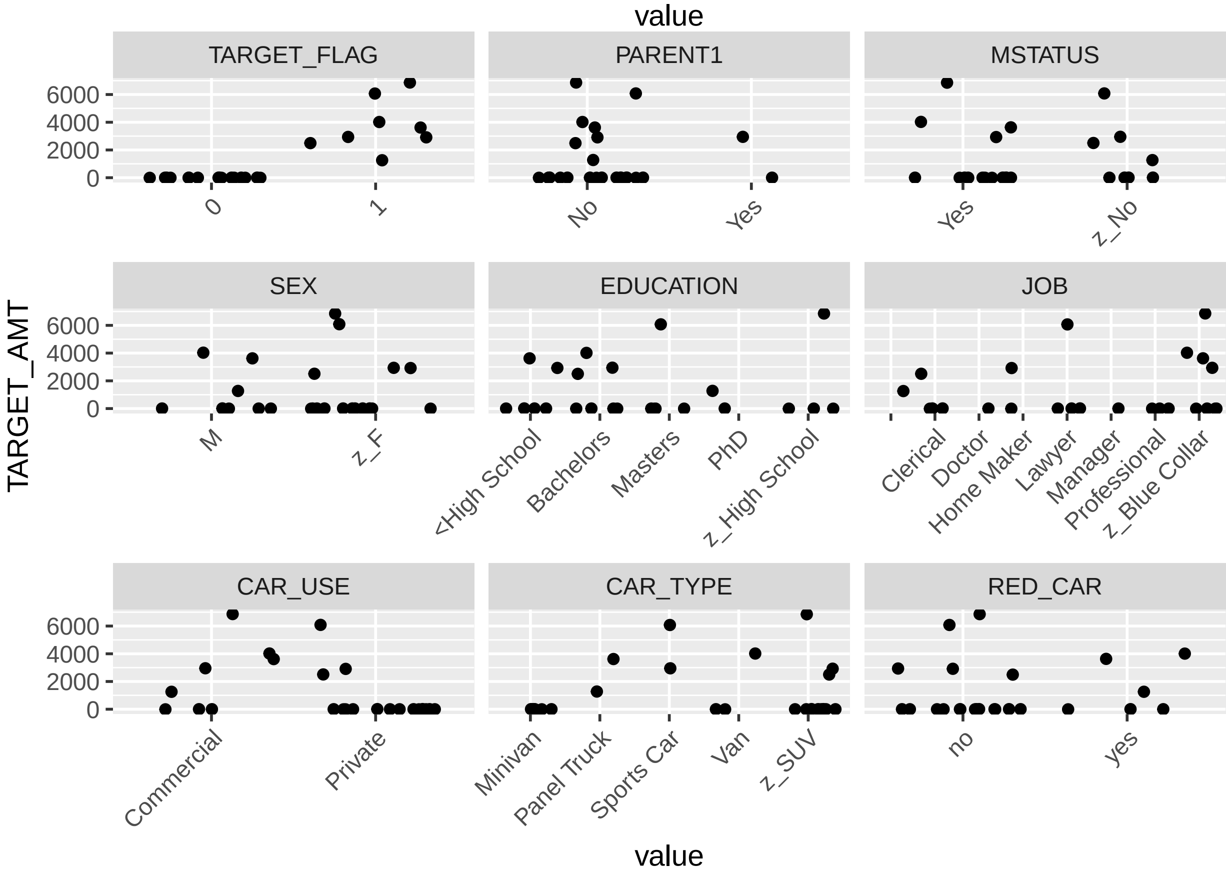


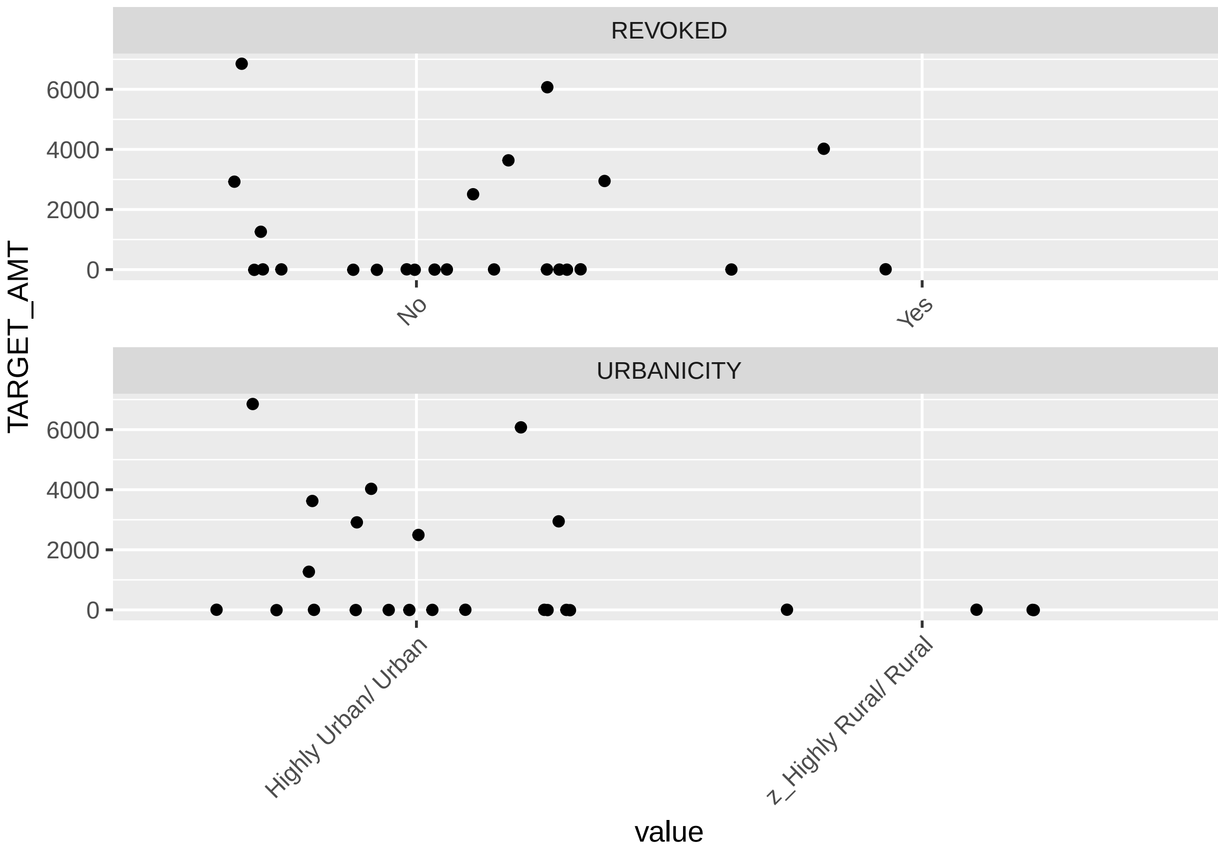

TARGET_FLAG
0
1
0
2,000
4,000
6,000
Frequency
PARENT1
No
Yes
0
2,000
4,000
6,000
Frequency
MSTATUS
Yes
z_No
0
1,000
2,000
3,000
4,000
5,000
Frequency
SEX
z_F
M
0
1,000
2,000
3,000
4,000
Frequency
EDUCATION
z_High School
Bachelors
Masters
<High School
PhD
0
500
1,000
1,500
2,000
Frequency
JOB
z_Blue Collar
Clerical
Professional
Manager
Lawyer
Student
Home Maker
Doctor
0
500
1,000
1,500
Frequency
CAR_USE
Private
Commercial
0
1,000
2,000
3,000
4,000
5,000
Frequency
CAR_TYPE
z_SUV
Minivan
Pickup
Sports Car
Van
Panel Truck
0
500
1,000
1,500
2,000
Frequency
RED_CAR
no
yes
0
2,000
4,000
6,000
Frequency
REVOKED
No
Yes
0
2,000
4,000
6,000
Frequency
URBANICITY
Highly Urban/ Urban
z_Highly Rural/ Rural
0
2,000
4,000
6,000
Frequency

Relationship of Predictors to Target




OLDCLAIM
CLM_FREQ
MVR_PTS
CAR_AGE
TARGET_AMT
6000
4000
2000
0
0
10000
20000
30000
40000
0
1
2
3
0.0
2.5
5.0
7.5
10.0
0
5
10
15
20
value


TARGET_FLAG
PARENT1
MSTATUS
SEX
EDUCATION
JOB
CAR_USE
CAR_TYPE
RED_CAR
TARGET_AMT
6000
4000
2000
0
0
1
No
Yes
Yes
z_No
M
z_F
<High School
Bachelors
Masters
PhD
z_High School
Clerical
Doctor
Home Maker
Lawyer
Manager
Professional
z_Blue Collar
Commercial
Private
Minivan
Panel Truck
Sports Car
Van
z_SUV
no
yes
value



2. DATA PREPARATION

Variable Adjustments

We'll make a few adjustments here based on some of the plotting we see.

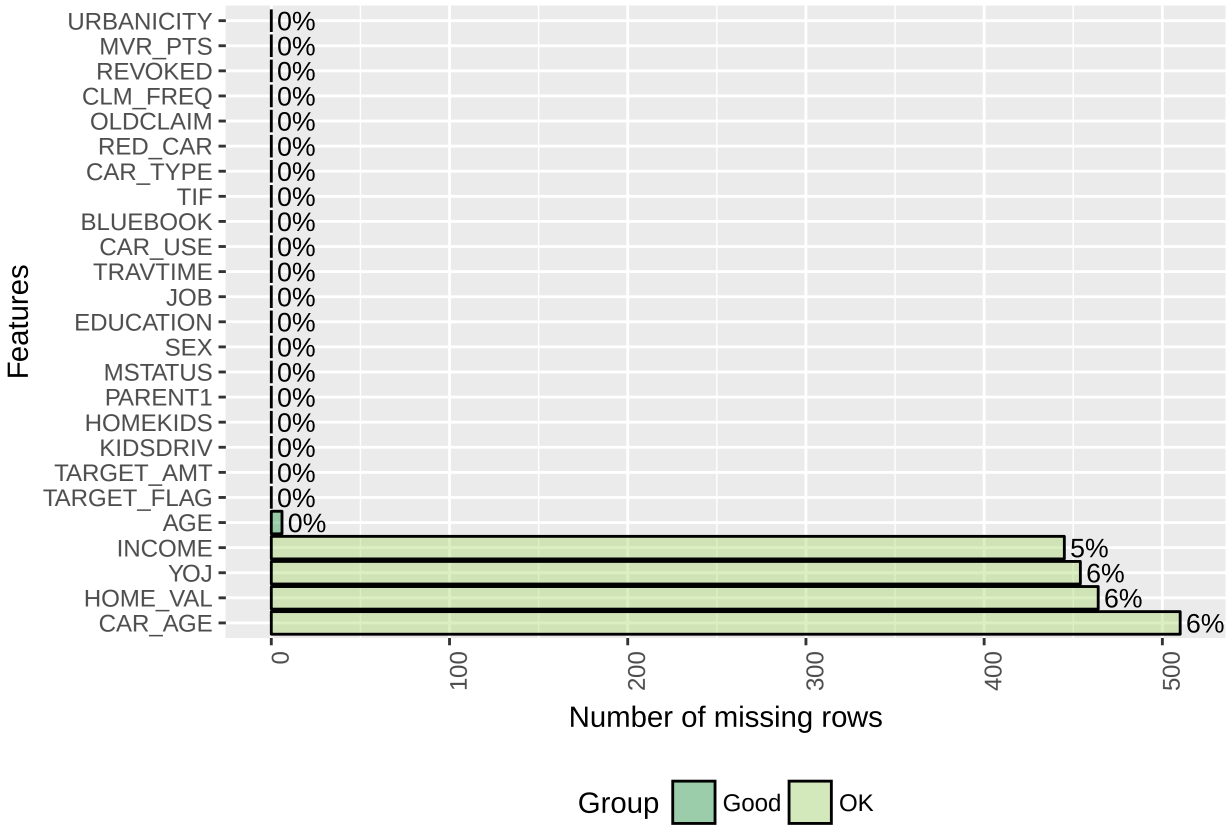
- Let's make HomeKids a Boolean instead of a factor.

- There are some CAR_AGE with - numbers. Let's make that 0.
- There are some blank Jobs. Let's code those as "Unknown".
- We'll change Education levels 1 if PhD and Masters.

```
train$HOMEKIDS[train$HOMEKIDS != 0 ] <- 1
test$HOMEKIDS[test$HOMEKIDS != 0 ] <- 1

train$CAR_AGE[train$CAR_AGE < 0 ] <- 0
test$CAR_AGE[test$CAR_AGE < 0 ] <- 0

train$JOB <- as.character(train$JOB)
train$JOB[train$JOB == ""] <- "Unknown"
train$JOB <- as.factor(train$JOB)

test$JOB <- as.character(test$JOB)
test$JOB[test$JOB == ""] <- "Unknown"
test$JOB <- as.factor(test$JOB)

train$EDUCATION <- ifelse(train$EDUCATION %in% c("PhD", "Masters"), 0, 1)
```

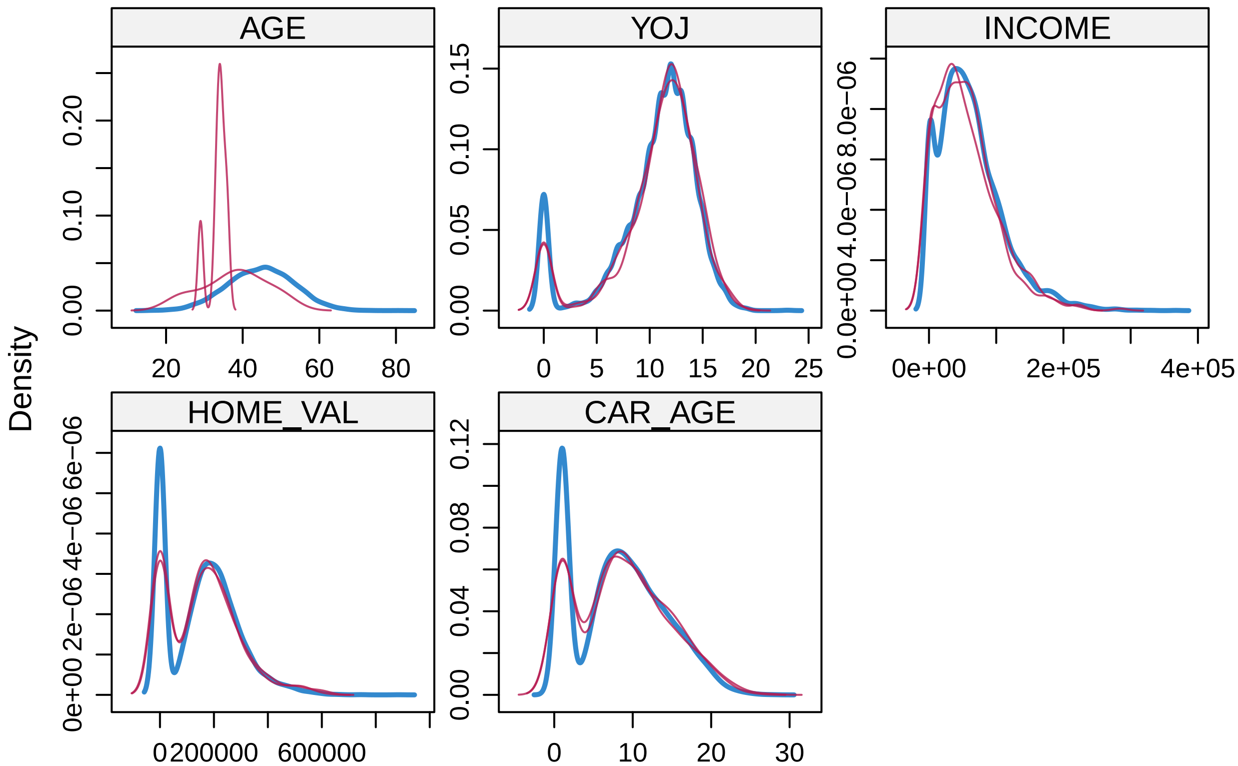
Plot and Review Missing

```
plot_missing(train)
```



We need to make decisions about Age, Income, YOJ, Home_Value, and Car_Age. After reviewing the test set, we are missing the same variables at about the same rate.

```
mice_imputes <- mice(train, m = 2, maxit = 2, print = FALSE)
densityplot(mice_imputes)
```



We can see that 4 of the variables with missing values seem to be MAR, as the mice imputation distributions roughly match the existing. The Age variable does not, which is interesting. Perhaps people lying about their age? We'll handle that differently, simply using median imputation for now.

We'll also run the mice imputation again on both the train and test set. Instead of using it for our models, however, we'll simplify our run and fill in our data. This is not a good method, as it doesn't account for variability, but it should do fine for the sake of this exercise.

```
m <- median(train$AGE, na.rm = T)
train$AGE[is.na(train$AGE)] <- m

mice_train <-  mice(train, m = 1, maxit = 1, print = FALSE)
train <- complete(mice_train)

mice_test <- mice(test, m = 1, maxit = 1, print = FALSE)
test <- complete(mice_test)
```

3. BUILD MODELS

Problem 1: Classification

Model 1

The first model fits includes all the variables. A review of the VIF output of the model suggests some points that are highly colinear and a number of variables that may not be necessary. Model 1 uses the formula:

target ~ .

	x
KIDSDRIV	7.709115

	x
AGE	10.359955
HOMEKIDS	17.553192
YOJ	9.612895
INCOME	22.531118
PARENT12	13.592953
HOME_VAL	16.935448
MSTATUS2	15.337781
SEX2	25.652783
EDUCATION	31.124277
JOB2	18.286278
JOB3	11.842945
JOB4	24.576359
JOB5	16.769393
JOB6	13.379737
JOB7	11.284504
JOB8	19.114372
JOB9	16.185896
TRAVTIME	7.249821
CAR_USE2	14.270424
BLUEBOOK	16.171712
TIF	7.536906
CAR_TYPE2	15.772020
CAR_TYPE3	11.555356
CAR_TYPE4	13.576152
CAR_TYPE5	10.683010
CAR_TYPE6	20.606204
RED_CAR2	12.736109
OLDCLAIM	9.677143
CLM_FREQ	8.862736
REVOKED2	7.327380
MVR_PTS	7.035772
CAR_AGE	12.648303
URBANICITY2	17.148135

Model 2

Our second model ignores the colinear issues, but removes models that seemed unnecessary in Model #1. Model 2 uses the formula:

TARGET_FLAG ~ KIDSDRIV + INCOME + PARENT1 + HOME_VAL + MSTATUS + JOB + TRAVTIME + CAR_USE + BLUEBOOK + TIF + CAR_TYPE + OLDCLAIM + CLM_FREQ + REVOKED + MVR_PTS + URBANICITY

	x
KIDSDRIV	6.490528
INCOME	21.689623
PARENT12	8.297485
HOME_VAL	16.796253
MSTATUS2	13.004965
JOB2	14.915291
JOB3	9.718152
JOB4	13.155597

	x
JOB5	14.667207
JOB6	12.713213
JOB7	10.200655
JOB8	12.276623
JOB9	16.082331
TRAVTIME	7.218974
CAR_USE2	14.129943
BLUEBOOK	12.984247
TIF	7.504643
CAR_TYPE2	13.695355
CAR_TYPE3	11.509005
CAR_TYPE4	9.263431
CAR_TYPE5	9.955548
CAR_TYPE6	12.290266
OLDCLAIM	9.651697
CLM_FREQ	8.840048
REVOKED2	7.295053
MVR_PTS	6.994880
URBANICITY2	17.213265

Model #3

Model #3 removes the variables with the 3 highest VIF values from model1, EDUCATION and SEX. The model formula is:

TARGET_FLAG ~ KIDSDRIV + AGE + HOMEKIDS + YOJ + INCOME + PARENT1 + HOME_VAL + MSTATUS + JOB + TRAVTIME + CAR_USE + BLUEBOOK + TIF + CAR_TYPE + RED_CAR + OLDCLAIM + CLM_FREQ + REVOKED + MVR_PTS + CAR_AGE + URBANICITY

	x
KIDSDRIV	7.667492
AGE	10.226240
HOMEKIDS	17.471045
YOJ	9.538255
PARENT12	13.533246
HOME_VAL	13.669385
MSTATUS2	14.423742
JOB2	14.820140
JOB3	11.197333
JOB4	14.856397
JOB5	14.645344
JOB6	12.559081
JOB7	11.152283
JOB8	12.043547
JOB9	15.719688
TRAVTIME	7.227292
CAR_USE2	14.223416
BLUEBOOK	13.367610
TIF	7.489318
CAR_TYPE2	14.148391
CAR_TYPE3	11.506891

	x
CAR_TYPE4	10.284687
CAR_TYPE5	10.098411
CAR_TYPE6	14.469053
RED_CAR2	10.114512
OLDCLAIM	9.663854
CLM_FREQ	8.844615
REVOKED2	7.295380
MVR_PTS	7.003844
CAR_AGE	11.005424
URBANICITY2	17.126198

Model #4

Model #4 takes the advances in model #3 and removes those values shown to be poor predictors.

TARGET_FLAG ~ KIDSDRIV + PARENT1 + HOME_VAL + MSTATUS + JOB + TRAVTIME + CAR_USE + BLUEBOOK + TIF + CAR_TYPE + OLDCLAIM + CLM_FREQ + REVOKED + MVR_PTS + CAR_AGE + URBANICITY

	x
KIDSDRIV	6.451379
PARENT12	8.285759
HOME_VAL	13.581243
MSTATUS2	12.245988
JOB2	14.658717
JOB3	9.725595
JOB4	14.570484
JOB5	14.463503
JOB6	12.466368
JOB7	10.054805
JOB8	11.992637
JOB9	15.677613
TRAVTIME	7.210038
CAR_USE2	14.182205
BLUEBOOK	12.439630
TIF	7.478665
CAR_TYPE2	13.685404
CAR_TYPE3	11.493483
CAR_TYPE4	9.243942
CAR_TYPE5	9.958285
CAR_TYPE6	12.280506
OLDCLAIM	9.659883
CLM_FREQ	8.824779
REVOKED2	7.276608
MVR_PTS	6.972136
CAR_AGE	10.964777
URBANICITY2	17.191949

Problem 2: Regression

Model #1

The first model fits includes all the variables.

target ~ .

```
set.seed(121)

train_regression <- train
train_regression <- train_regression[train_regression$TARGET_FLAG == 1, ]
train_regression$TARGET_FLAG <- NULL

mod1lm <- train(TARGET_AMT ~., data = train_regression,
              method = "lm",
              trControl = trainControl(
                  method = "cv", number = 10,
                  savePredictions = TRUE),
              tuneLength = 5,
              preProcess = c("center", "scale"))
```

Model #2

Model #2 we take start trimming out features with less impact.

TARGET_AMT ~ HOME_VAL + CAR_USE + BLUEBOOK + TIF + CAR_TYPE + OLDCLAIM + CLM_FREQ + REVOKED + MVR_PTS + CAR_AGE + URBANICITY

```
mod2lm <- train(TARGET_AMT ~ HOME_VAL +
                  CAR_USE + BLUEBOOK + TIF + CAR_TYPE +
                  OLDCLAIM + CLM_FREQ + REVOKED + MVR_PTS +
                  CAR_AGE + URBANICITY, data = train_regression,
              method = "lm",
              trControl = trainControl(
                  method = "cv", number = 10,
                  savePredictions = TRUE),
              tuneLength = 5,
              preProcess = c("center", "scale"))
```

Model 3

Model #3 is pretty bare-bones and only reflects generally issues related to the car value or driver's legal issues.

TARGET_AMT ~ BLUEBOOK + REVOKED + MVR_PTS + CAR_AGE

```
mod3lm <- train(TARGET_AMT ~ BLUEBOOK + REVOKED + MVR_PTS +
                  CAR_AGE, data = train_regression,
              method = "lm",
              trControl = trainControl(
                  method = "cv", number = 10,
                  savePredictions = TRUE),
              tuneLength = 5,
              preProcess = c("center", "scale"))
```

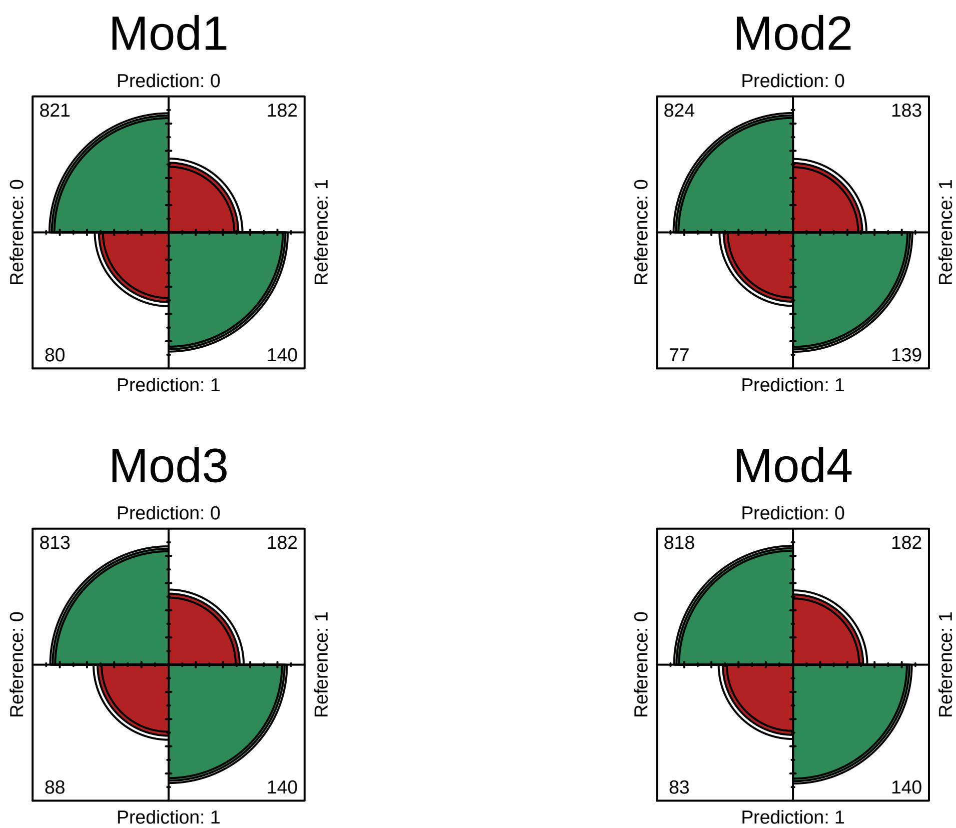
4. SELECT MODELS

To help aid in model selection for the classification problem, we'll review their accuracy by making predictions on our holdout validation set, and comparing their performance using a variety of confusion matrix adjacent functions like fourfold plots, summary statistics, and ROC / AUC plots.

To aid in model selection for the regression problem, we'll compare the error in the fit of our models in a table and select from there.

1: Classification

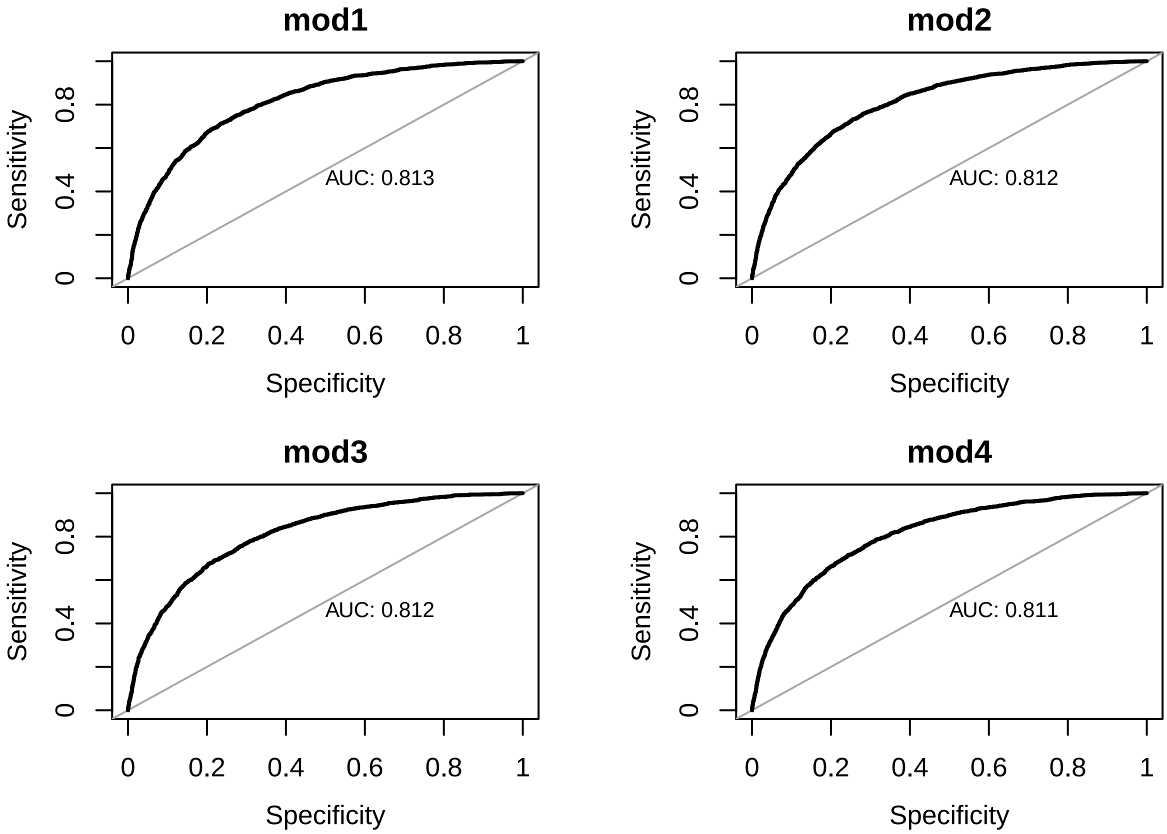
Fourfold Plots



Summary Statistics

	Sensitivity	Specificity	Precision	Recall	F1
Model1	0.9112098	0.4347826	0.8185444	0.9112098	0.8623950
Model2	0.9145394	0.4316770	0.8182721	0.9145394	0.8637317
Model3	0.9023307	0.4347826	0.8170854	0.9023307	0.8575949
Model4	0.9078801	0.4347826	0.8180000	0.9078801	0.8605997

ROC / AUC



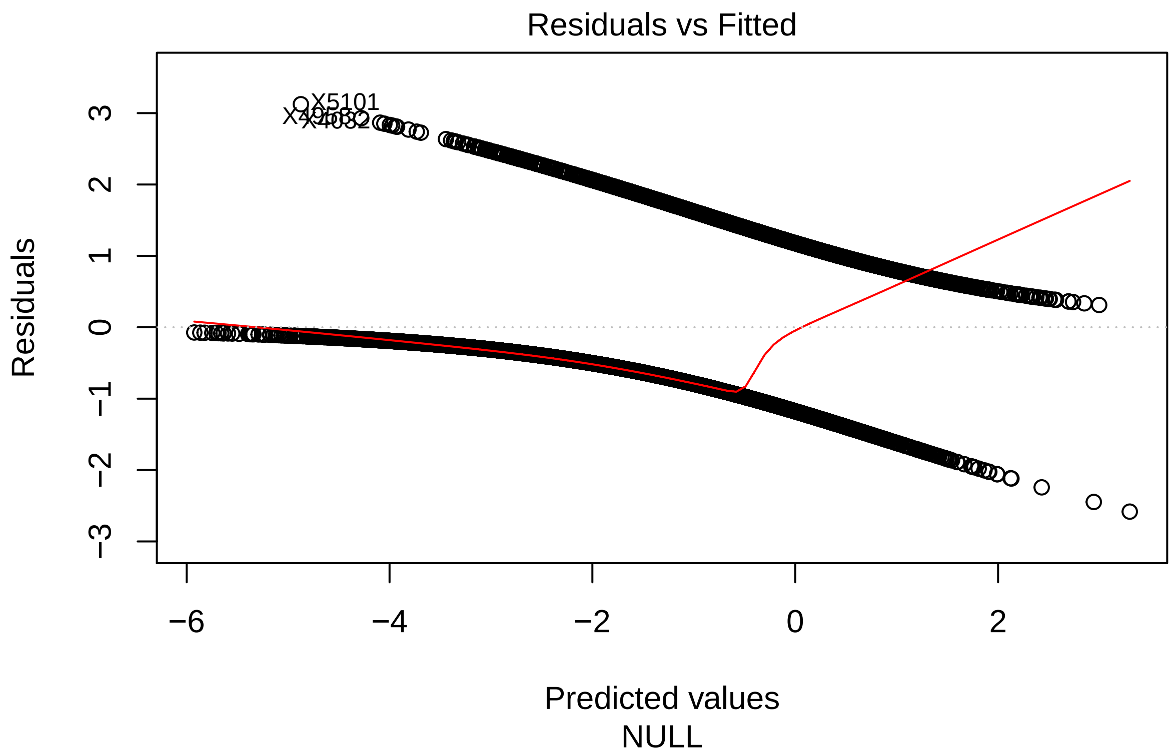
Model Selection - Classification

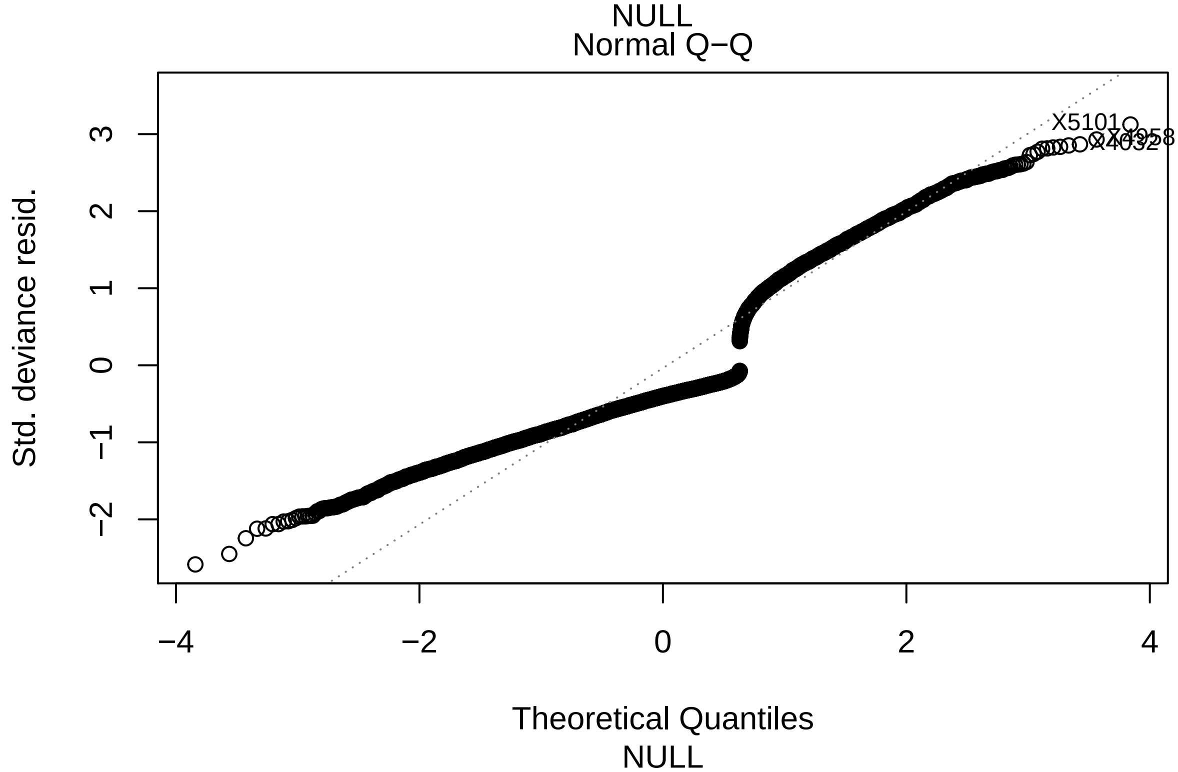
While the first 2 models may have the most information, they also suffer from so co-linearity issues as shown by the variance VIF output. Model #3 performs well, but has some additional variables that may be poor predictors of whether a neighborhood will be above or below the median crime rate. Instead, while stripped out, we'll use Model #4.

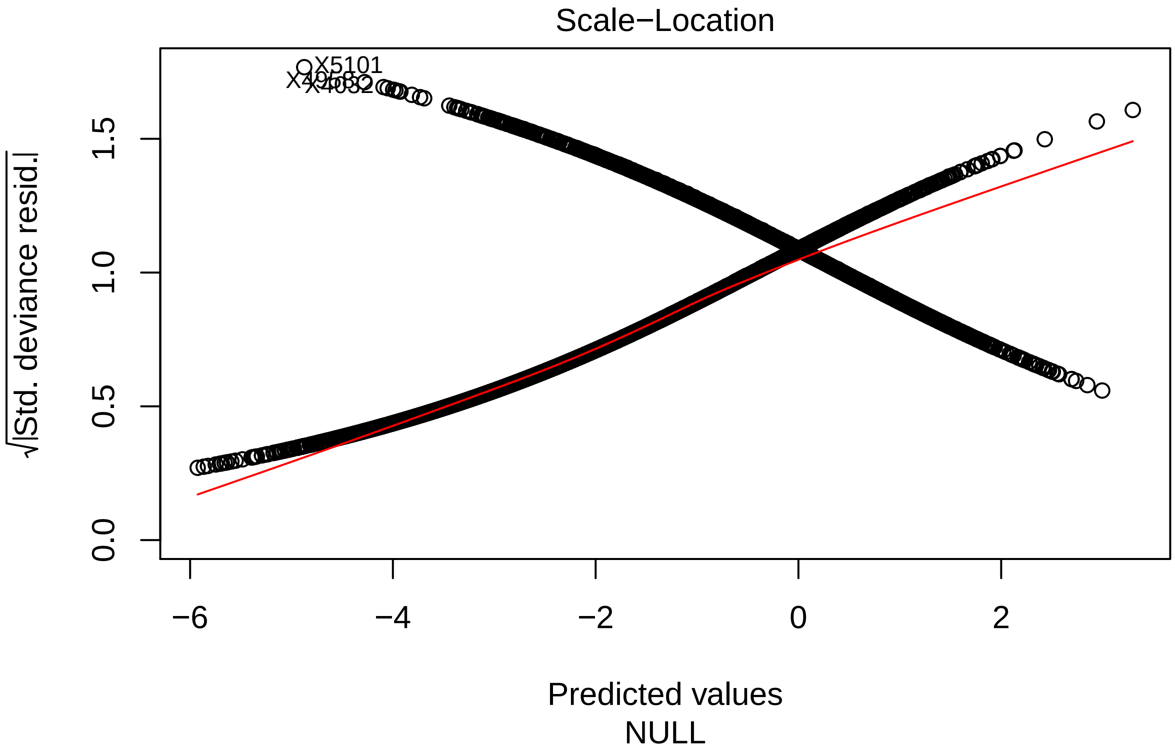
Before we make predictions, let's run this final model over our full dataset, and review some summary diagnostic plots and output.

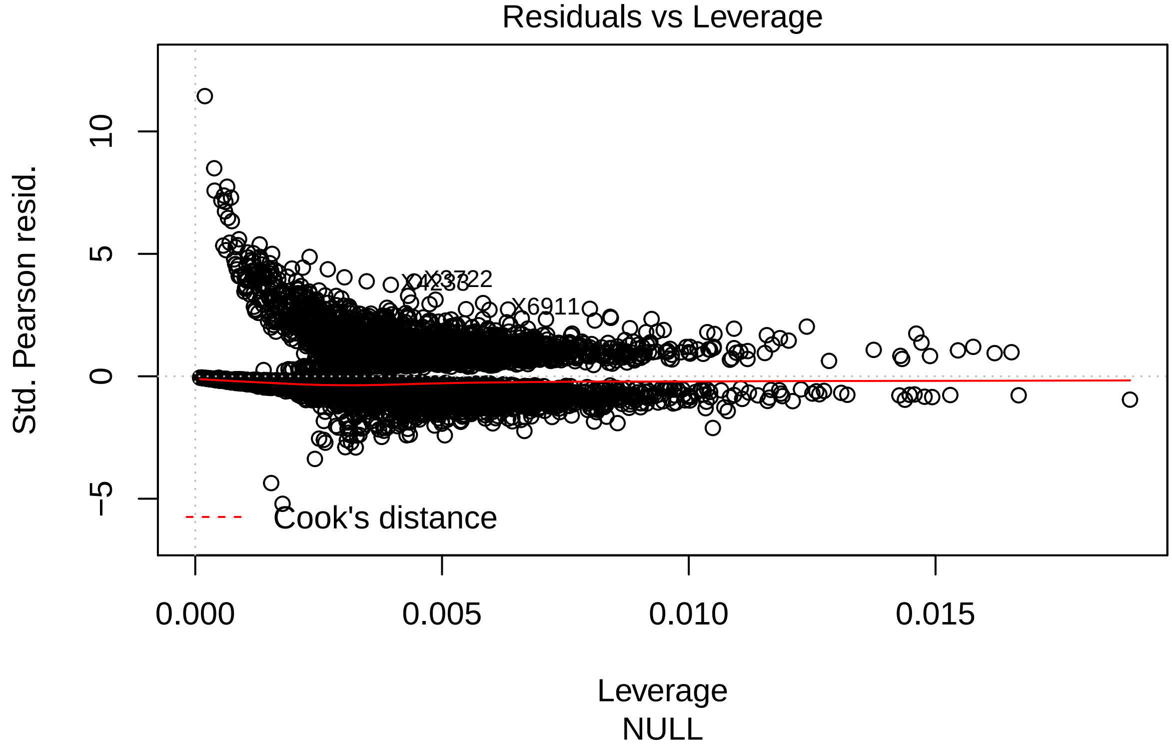
```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5825  -0.7204  -0.4001   0.6459   3.1247
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.42271    0.03475 -40.937  < 2e-16 ***
## KIDSDRIV     0.21167    0.02806   7.544 4.55e-14 ***
## PARENT12     0.15918    0.03182   5.003 5.66e-07 ***
## HOME_VAL    -0.25819    0.04068  -6.347 2.19e-10 ***
## MSTATUS2     0.17542    0.03856   4.549 5.38e-06 ***
```

```
## JOB2        -0.17603    0.04011  -4.389 1.14e-05 ***
## JOB3        -0.03632    0.03447  -1.054 0.291919    
## JOB4        -0.15330    0.04199  -3.651 0.000261 ***
## JOB5        -0.39201    0.04226  -9.276  < 2e-16 ***
## JOB6        -0.17088    0.03918  -4.361 1.29e-05 ***
## JOB7        -0.03670    0.03512  -1.045 0.296019    
## JOB8        -0.14923    0.03829  -3.898 9.70e-05 ***
## JOB9        -0.07042    0.04375  -1.609 0.107518    
## TRAVTIME     0.22525    0.02982   7.554 4.22e-14 ***
## CAR_USE2    -0.34683    0.04186  -8.285  < 2e-16 ***
## BLUEBOOK    -0.22240    0.03904  -5.696 1.23e-08 ***
## TIF         -0.22769    0.03034  -7.504 6.17e-14 ***
## CAR_TYPE2    0.17794    0.04108   4.332 1.48e-05 ***
## CAR_TYPE3    0.21541    0.03742   5.757 8.57e-09 ***
## CAR_TYPE4    0.30269    0.03369   8.986  < 2e-16 ***
## CAR_TYPE5    0.18791    0.03504   5.363 8.18e-08 ***
## CAR_TYPE6    0.31927    0.03855   8.283  < 2e-16 ***
## OLDCLAIM    -0.12104    0.03426  -3.533 0.000411 ***
## CLM_FREQ     0.22564    0.03294   6.849 7.43e-12 ***
## REVOKED2     0.29161    0.02982   9.779  < 2e-16 ***
## MVR_PTS      0.24772    0.02910   8.513  < 2e-16 ***
## CAR_AGE     -0.09591    0.03642  -2.633 0.008454 ** 
## URBANICITY2 -0.95456    0.04534 -21.052  < 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: 9418.0  on 8160  degrees of freedom
## Residual deviance: 7330.5  on 8133  degrees of freedom
## AIC: 7386.5
## 
## Number of Fisher Scoring iterations: 5
```


Residuals vs Fitted
Residuals
Predicted values
NULL
X5101
X4958
X4032
−6
−4
−2
0
2
−3
−2
−1
0
1
2
3


Normal Q−Q
Std. deviance resid.
Theoretical Quantiles
NULL
X5101
X4958
X4032
−4
−2
0
2
4
−2
−1
0
1
2
3





2: Regression

Here we see a preference again for the simpler model. We'll make our predictions using Model 3.

```
df <- data.frame()
df <- rbind(df, mod1lm$results)
```

```r
df <- rbind(df, mod2lm$results)
df <- rbind(df, mod3lm$results)
df$intercept <- c("Mod1", "Mod2", "Mod3")
colnames(df)[1] <- "model"
knitr::kable(df)
```

model	RMSE	Rsquared	MAE	RMSESD	RsquaredSD	MAESD
Mod1	7600.400	0.0096814	3758.356	1596.105	0.0091622	340.5952
Mod2	7517.020	0.0108574	3691.560	1880.665	0.0109962	465.7851
Mod3	7497.834	0.0195371	3660.723	1794.285	0.0198798	464.6214

Make Predictions

We make our final predictions, create a dataframe with the prediction and the predicted probabilities for our classification problem.

However, in case our predictive model got the classification portion wrong, we'll make a prediction on the target amount for all observations in the test set, regardless of whether we think they'll make a claim.

```r
finalpreds <- predict(finalmod, test)
finalpreds.probs <- predict(finalmod, test, type="prob")
finaldf <- cbind(finalpreds.probs, TARGET_FLAG=finalpreds)

finalAmountPreds <- predict(mod3lm, test)
finaldf <- cbind(finaldf, TARGET_AMT = finalAmountPreds)

write.csv(finaldf, 'HW4preds.csv', row.names = FALSE)
```

Appendix

- For full output code visit: https://github.com/wwells/CUNY_DATA_621/blob/master/HW/HW4/HW4_WWells.Rmd
- For predicted values over test set visit: https://github.com/wwells/CUNY_DATA_621/blob/master/HW/HW4/HW4preds.csv