DS 621 Fall2020: Homework 4 (Group3)

Auto Insurance Regression

Zach Alexander, Sam Bellows, Donny Lofland, Joshua Registe, Neil Shah, Aaron Zalki

Source code: https://github.com/djlofland/DS621_F2020_Group3/tree/master/Homework_4

Introduction

Group 3 created a multiple linear regression and binary logistic model to estimate the probability of a driver having an auto accident, and the monetary damage, for the customer Khansari Auto Insurance. As auto insurance and insurance in general stem off the pooling of individuals to mitigate risk, it's important to be able to predict accident rates to ensure funds are available for disbursement.

1. Data Exploration

Describe the size and the variables in the insurance training data set. Consider that too much detail will cause a manager to lose interest while too little detail will make the manager consider that you aren't doing your job. Some suggestions are given below. Please do NOT treat this as a checklist of things to do to complete the assignment.

You should have your own thoughts on what to tell the boss. These are just ideas.

Given that the Index column had no impact on the target variable, it was dropped as part of the initial cleaning function. Additionally, the fields "INCOME", "HOME_VAL", "OLDCLAIM", and, "BLUEBOOK", were imported as characters with "$" leaders and were converted to numeric as part of the initial cleaning function. both the training and evaluation datasets will pass through this treatment.

Now, with initial cleaning complete, we can take a quick look at the dimensions of both the training and evaluation datasets:

Dimensions of the training dataset:

```
## [1] 8161   25
```

Dimensions of evaluation dataset:

```
## [1] 2141   25
```

It looks like we have 8,161 cases and 25 variables in the training dataset and 2,141 cases and 25 variables in the evaluation dataset.

We can also provide a summary of each variable and the theoretical effect it'll have on our 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

Figure 1: Variables of Interest

Summary Stats

Next, we compiled summary statistics on our data set to better understand the data before modeling.

```
##  TARGET_FLAG   TARGET_AMT        KIDSDRIV           AGE           HOMEKIDS
##  0:6008      Min.   :     0   Min.   :0.0000   Min.   :16.00   Min.   :0.0000
##  1:2153      1st Qu.:     0   1st Qu.:0.0000   1st Qu.:39.00   1st Qu.:0.0000
##              Median :     0   Median :0.0000   Median :45.00   Median :0.0000
##              Mean   :  1504   Mean   :0.1711   Mean   :44.79   Mean   :0.7212
##              3rd Qu.:  1036   3rd Qu.:0.0000   3rd Qu.:51.00   3rd Qu.:1.0000
##              Max.   :107586   Max.   :4.0000   Max.   :81.00   Max.   :5.0000
##                                                NA's   :6
##       YOJ            INCOME         PARENT1             HOME_VAL
##  Min.   : 0.0   Min.   :     0   Length:8161        Min.   :     0
##  1st Qu.: 9.0   1st Qu.: 28097   Class :character   1st Qu.:     0
##  Median :11.0   Median : 54028   Mode  :character   Median :161160
##  Mean   :10.5   Mean   : 61898                      Mean   :154867
##  3rd Qu.:13.0   3rd Qu.: 85986                      3rd Qu.:238724
##  Max.   :23.0   Max.   :367030                      Max.   :885282
##  NA's   :454    NA's   :445                        NA's   :464
##    MSTATUS              SEX             EDUCATION             JOB
##  Length:8161        Length:8161        Length:8161        Length:8161
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##     TRAVTIME        CAR_USE             BLUEBOOK          TIF
##  Min.   :  5.00   Length:8161        Min.   : 1500   Min.   : 1.000
```

```
##  1st Qu.: 22.00   Class :character   1st Qu.: 9280   1st Qu.: 1.000
##  Median : 33.00   Mode  :character   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
##  Length:8161        Length:8161        Min.   :    0   Min.   :0.0000
##  Class :character   Class :character   1st Qu.:    0   1st Qu.:0.0000
##  Mode  :character   Mode  :character   Median :    0   Median :0.0000
##                                        Mean   : 4037   Mean   :0.7986
##                                        3rd Qu.: 4636   3rd Qu.:2.0000
##                                        Max.   :57037   Max.   :5.0000
##
##    REVOKED             MVR_PTS          CAR_AGE        URBANICITY
##  Length:8161        Min.   : 0.000   Min.   :-3.000   Length:8161
##  Class :character   1st Qu.: 0.000   1st Qu.: 1.000   Class :character
##  Mode  :character   Median : 1.000   Median : 8.000   Mode  :character
##                     Mean   : 1.696   Mean   : 8.328
##                     3rd Qu.: 3.000   3rd Qu.:12.000
##                     Max.   :13.000   Max.   :28.000
##                                      NA's   :510
```

Check Class Bias

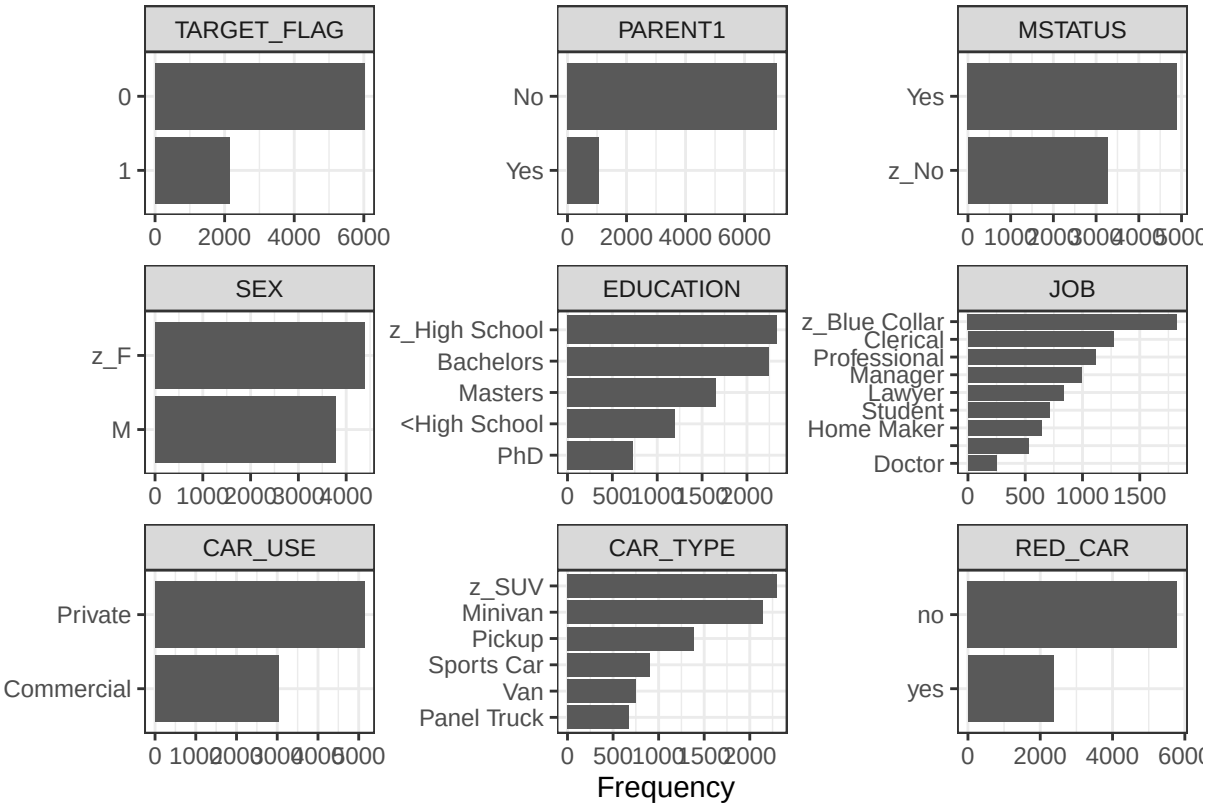
Next, we need to understand the distribution profiles for each of the variables. We have two target values, 0 and 1. When building models, we ideally want an equal representation of both classes. As class imbalance increases, favoring one class, our model performance will suffer both from the effects of differential variance between the classes and bias towards the more represented class. For logistic regression, to address a strong class imbalance, some approaches we can try include:

1. up-sample the smaller group,
2. down-sample the larger group, or
3. adjust our threshold for assigning the predicted value away from 0.5.

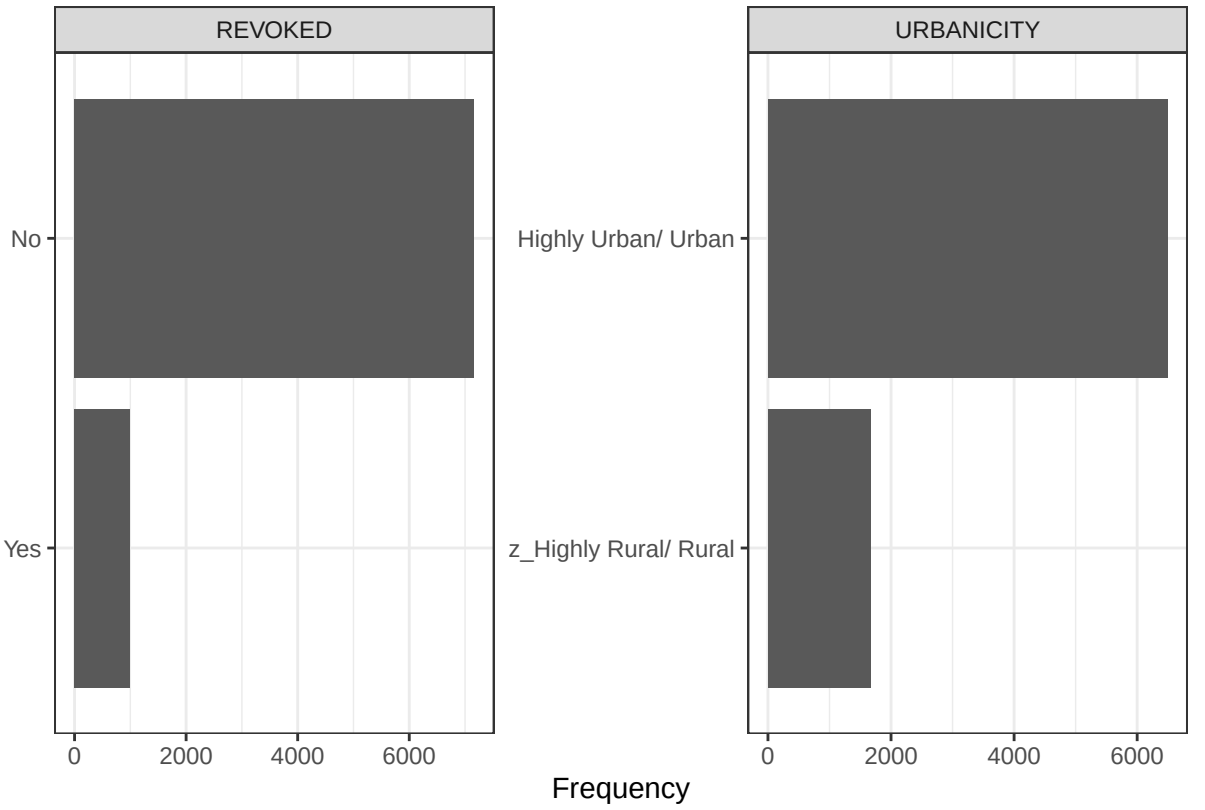
Classification of Target Flag	
Var1	Freq
0	0.7361843
1	0.2638157

The classes are not well balanced, with approximately 73.6% 0's and 26.4% 1's. Given the unbalanced class distributions, upsampling or downsampling may be required to achieve class balance with this dataset. We will evaluate model performance accordingly.

Distributions



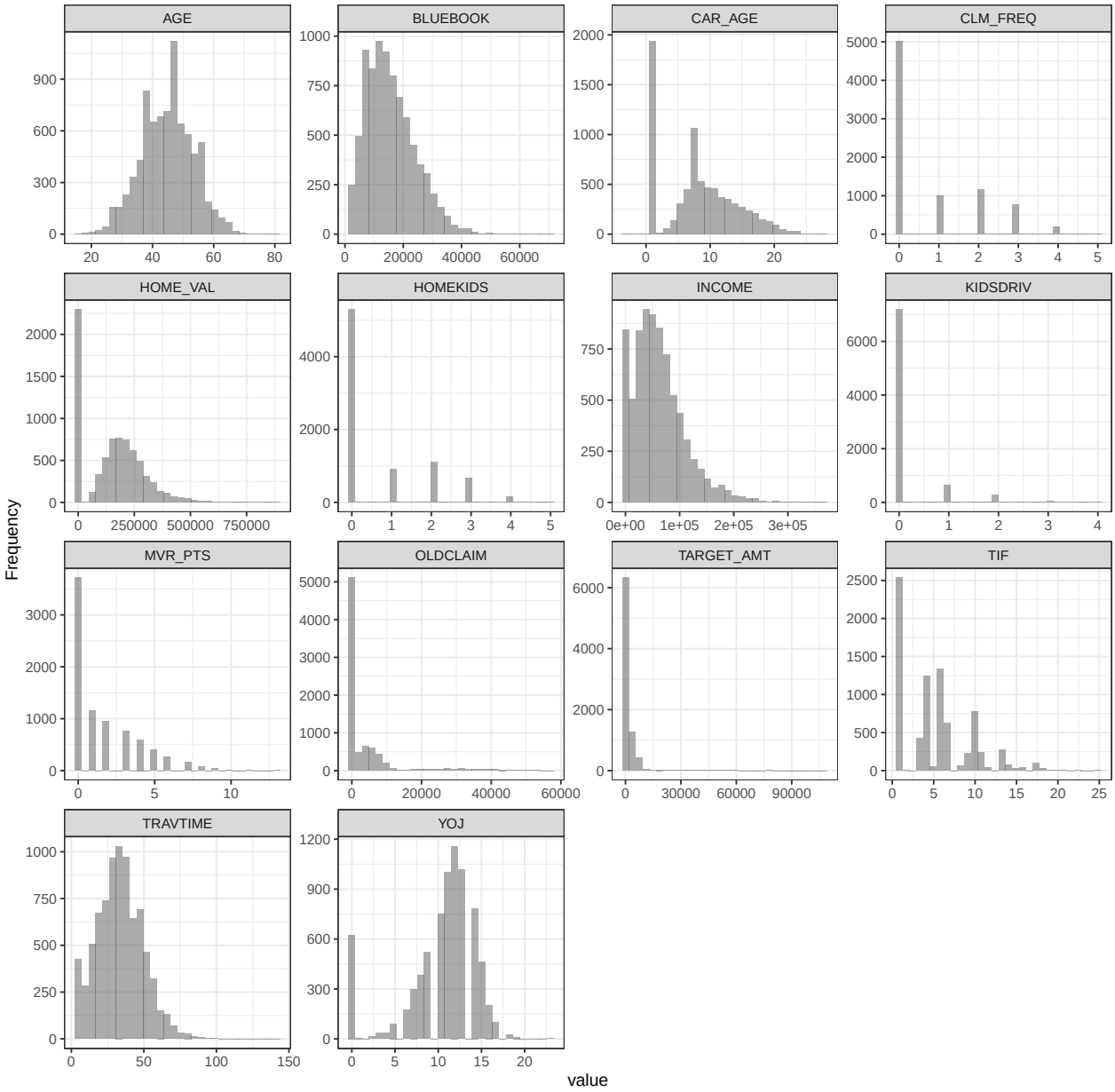
Page 1



Page 2

Next, we visualize the distribution profiles for each of the predictor variables. This will help us to make a plan on which variables to include, how they might be related to each other or the target, and finally identify outliers or transformations that might help improve model resolution.

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



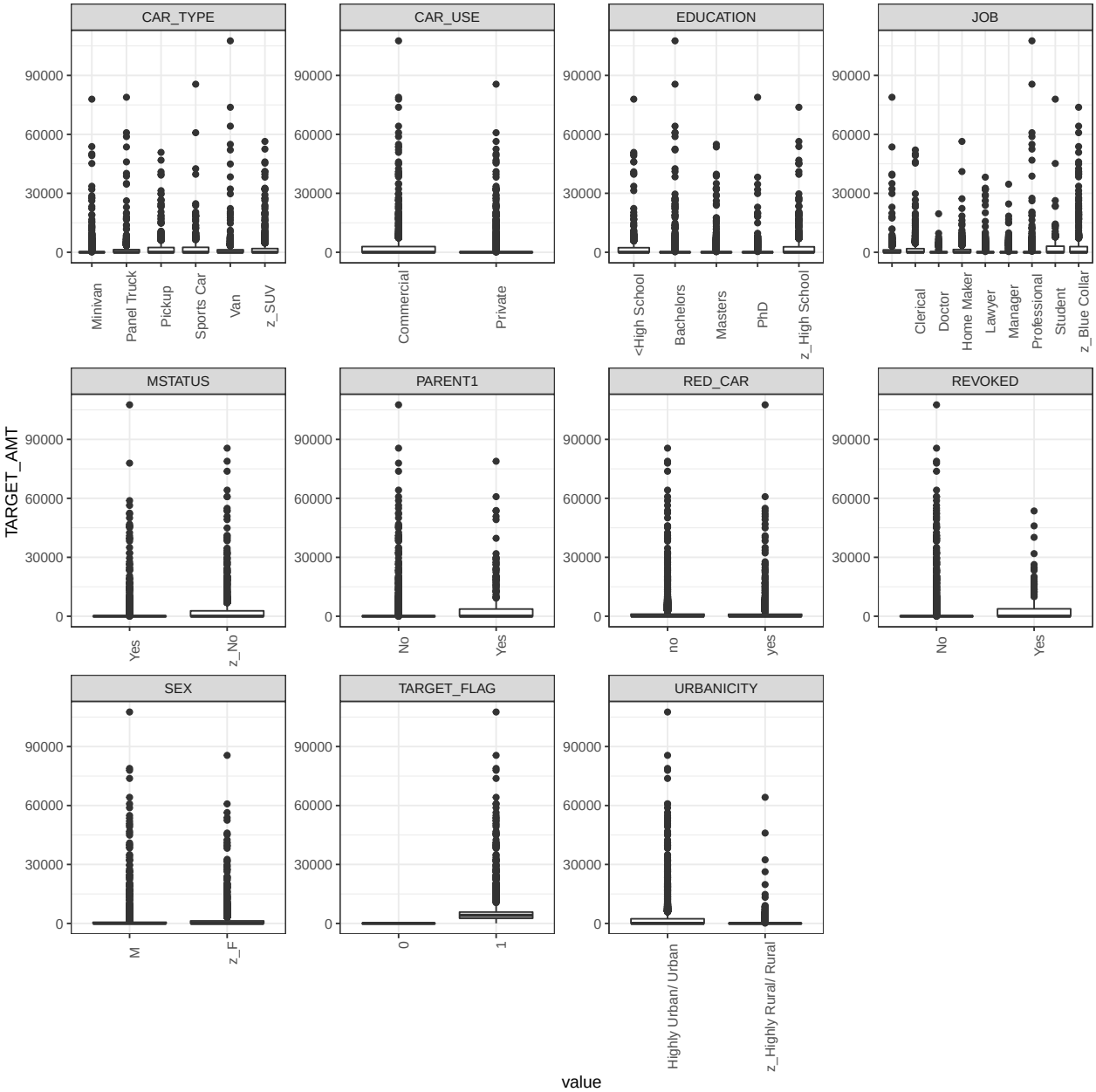
The distribution profiles show the prevalence of kurtosis, specifically right skew in variables `TRAVTIME`, `OLDCLAIM`, `MVR_PTS`, `TARGET_AMT`, `INCOME`, `BLUEBOOK`, and approximately normal distributions in `YOJ`, `CARAGE`, `HOME_VAL`, and `AGE`. When deviations are skewed from a traditional normal distribution, this can be problematic for regression assumptions, and thus we might need to transform the data. Under logistic regression, we will need to dummy factor-based variables for the model to understand the data.

While we don't tackle feature engineering in this analysis, if we were performing a more in-depth analysis, we could leverage the package, `mixtools` (see R Vignette). This package helps regress *mixed models* where data can be subdivided into subgroups.

Lastly, several features have both a distribution along with a high number of values at an extreme. However, based on the feature meanings and provided information, there is no reason to believe that any of these extreme values are mistakes, data errors, or otherwise inexplicable. As such, we will not remove the extreme values, as they represent valuable data and could be predictive of the target.

Boxplots

In addition to creating histogram distributions, we also elected to use box-plots to get an idea of the spread of the response variable `TARGET_AMT` in relation to all of the non-numeric variables. Two sets of boxplots are shown below due to the wide distribution of the response variable. The first set of boxplots highlights the entire range and shows how the cost of car crashes peaks relative to the specific category.

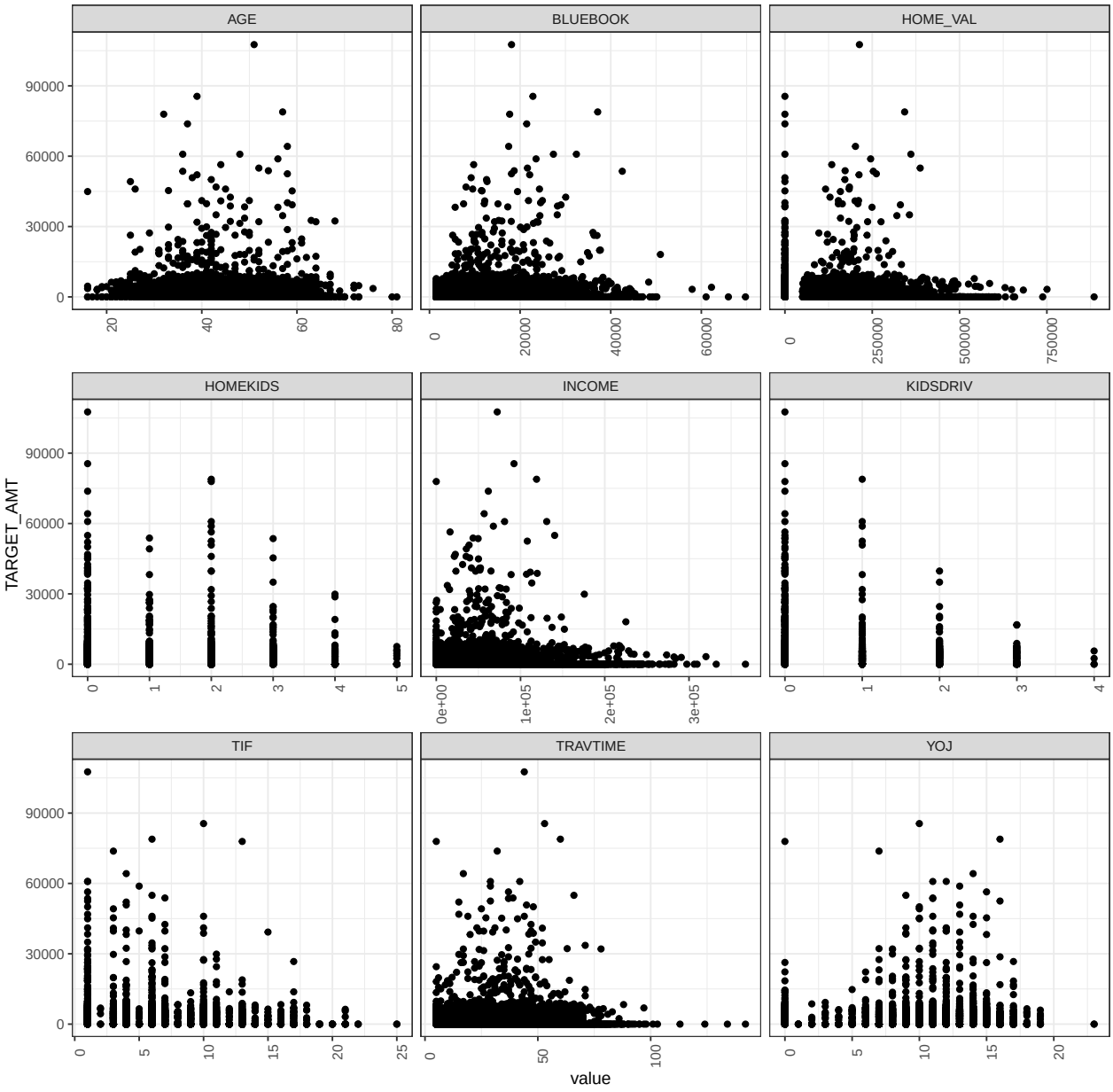


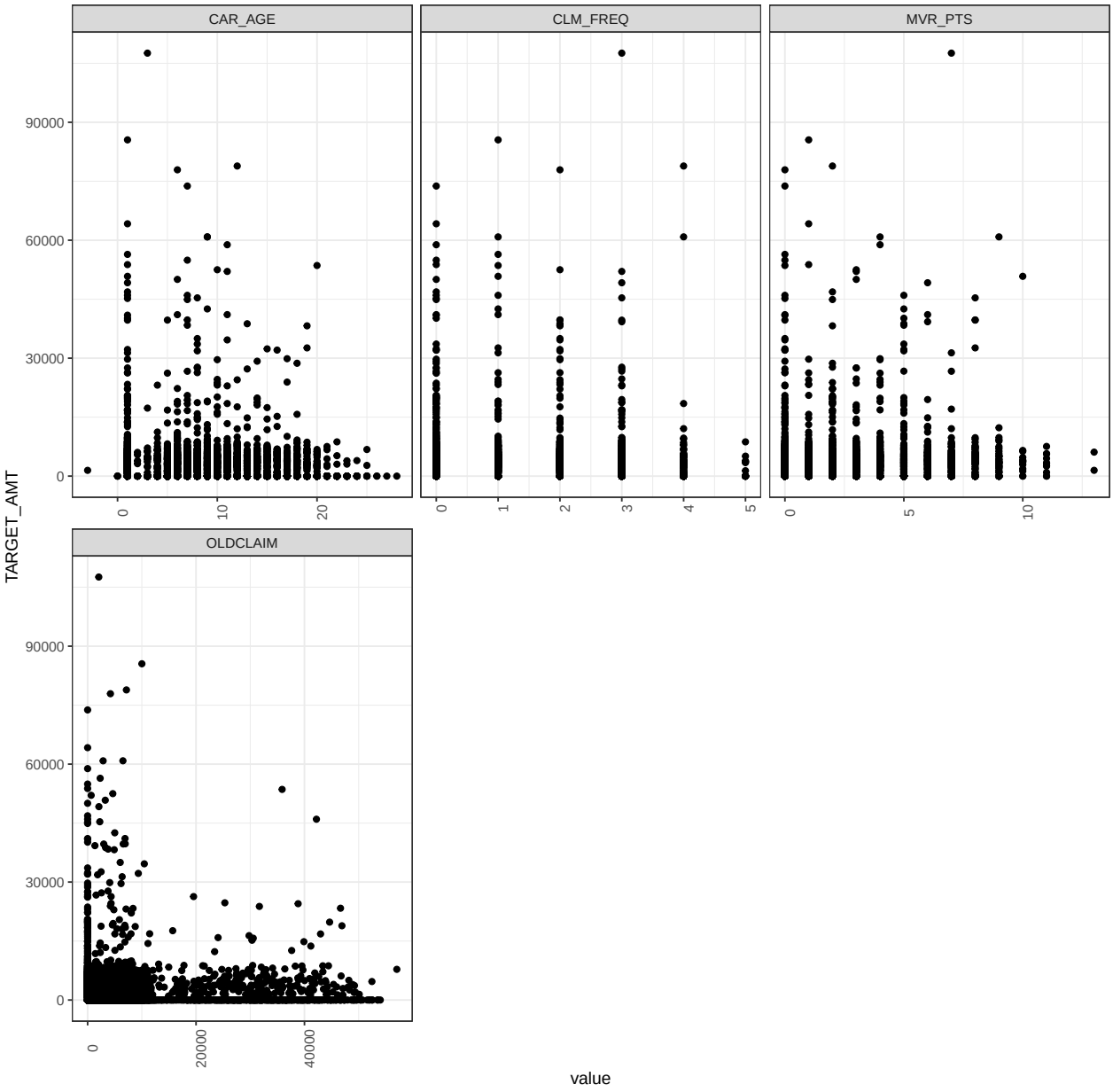
The second set of box plots simply shows these same distributions "zoomed in" by adjusting the axis to allow for a visual of the interquartile range of the response variable relative to each of the categorical predictors.

Variable Plots

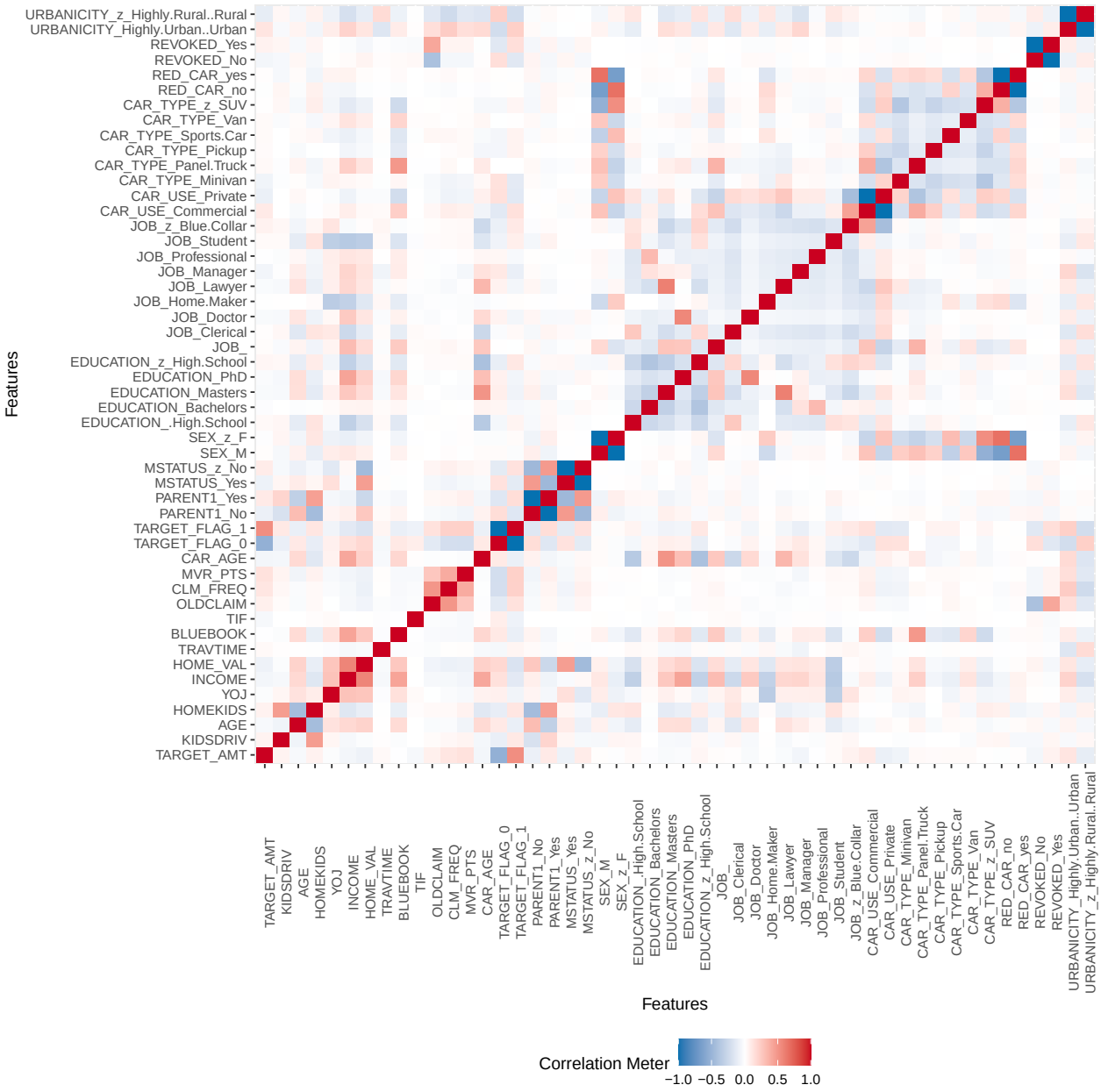
We generate scatter plots of each variable versus the target variable, `TARGET_AMT`, to get an idea of the relationship between them. We observe some notable trends in the scatterplots below such as our response variable `TARGET_AMT` is likely to be lower when individuals have more kids at home as indicated by the `HOMEKIDS` feature, and when they have more teenagers driving the car indicated by the feature `KIDSDRIV`.

Additionally, a pairwise comparison plot between all features, both numeric and non-numeric is shown following the scatterplot where this initially implies that there aren't a significant amount of correlated features and this can give some insight into the expected significance and performing dimensionality reduction on the datasets for the models.


AGE
BLUEBOOK
HOME_VAL
HOMEKIDS
INCOME
KIDSDRIV
TIF
TRAVTIME
YOJ
TARGET_AMT
value
0
30000
60000
90000
20
40
60
80
0
20000
40000
60000
250000
500000
750000
1
2
3
4
5
0e+00
1e+05
2e+05
3e+05
0
5
10
15
20
25
50
100

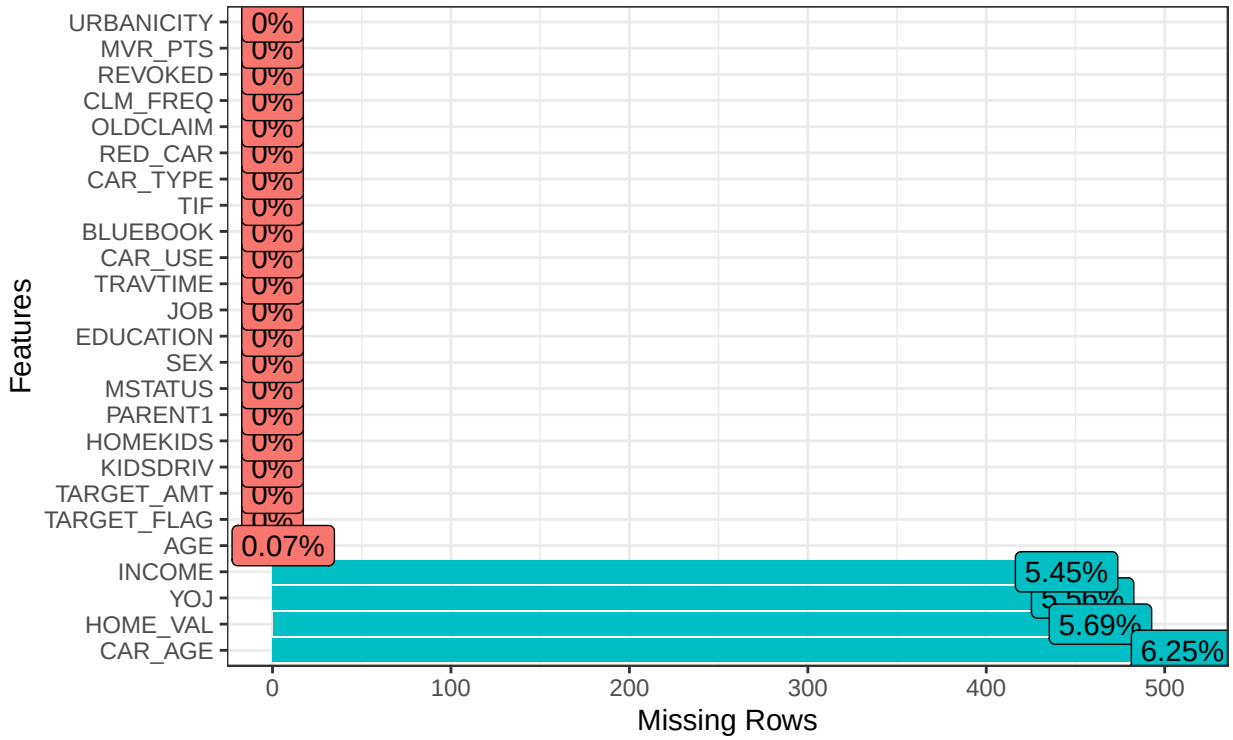

CAR_AGE
CLM_FREQ
MVR_PTS
OLDCLAIM
TARGET_AMT
value
90000
60000
30000
0
0
10
20
0
1
2
3
4
5
0
5
10
0
20000
40000

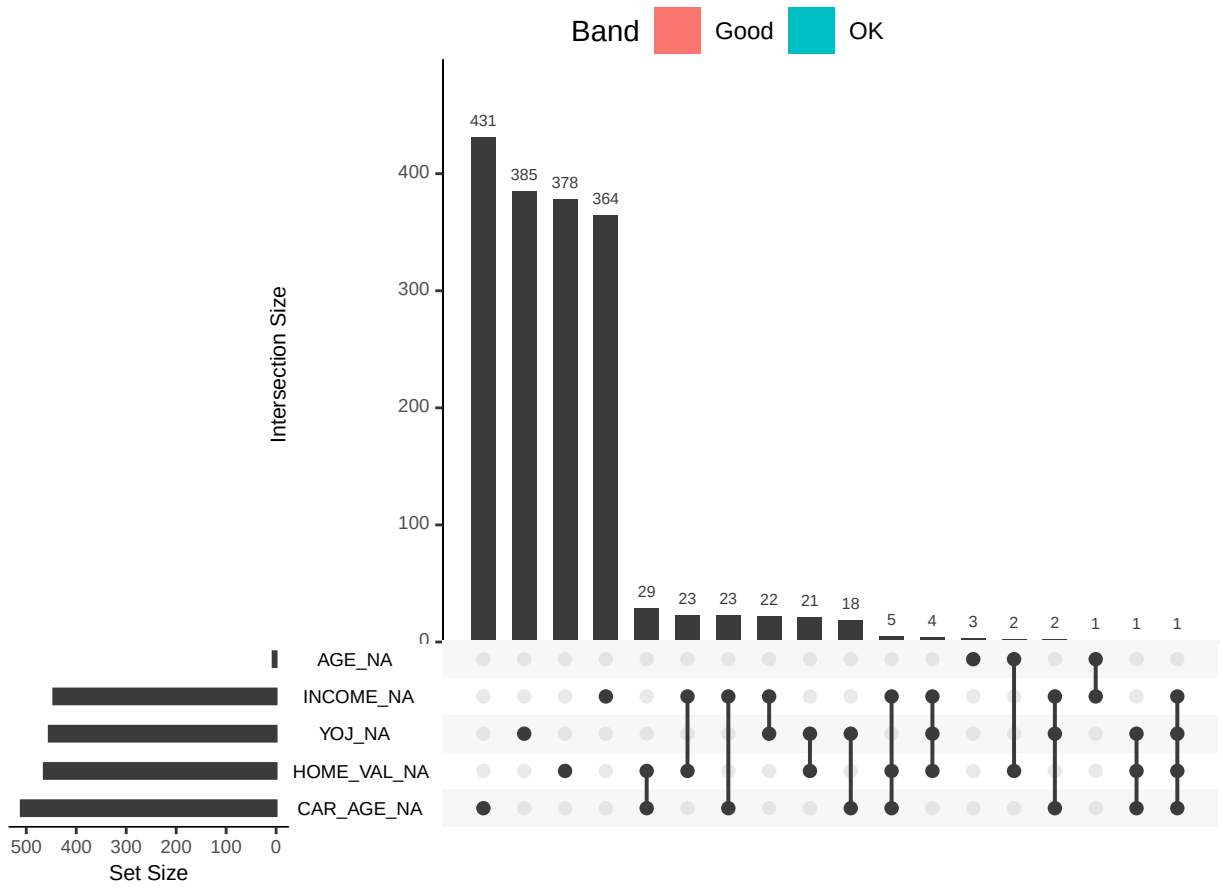
Page 2



Data Sparsity Check

Finally, we can observe the sparsity of information within our dataset by using the `DataExplorer` package to assess missing information and the `nanair` package to understand any patterns within the missing features.





We can see that generally, our dataset is in good shape, however, some imputation may be needed for `INCOME`,

`YOJ`, `HOME_VAL`, and `CAR_AGE`. Notice that there is also a correlation between these missing variables, when one is missing, we see many cases where a second (or more) are also missing. This does imply that our `knnImpute` approach to missing values may not be as accurate. That said, it's a low percentage, so probably won't have a meaningful impact on final model results. Note that a `knnImpute` approach may also be skewed by class imbalances with features. In a more rigorous analysis, we might consider testing a `median` imputing strategy and/or bootstrapping blocking strategies for better sampling to impute.

2. Data Preparation

To summarize our data preparation and exploration, we can distinguish our findings into a few categories below:

Removed Fields

All the predictor variables have a low enough percentage of missing values that they can be imputed and still provide useful information to the model. As such, we chose to keep all the fields.

Missing Values

Missing values will be imputed with the preprocess function in the caret package and the `knnImpute` method.

variable	n_miss	pct_miss
TARGET_FLAG	0	0
TARGET_AMT	0	0
KIDSDRIV	0	0
AGE	0	0
HOMEKIDS	0	0
YOJ	0	0
INCOME	0	0
PARENT1	0	0
HOME_VAL	0	0
MSTATUS	0	0
SEX	0	0
EDUCATION	0	0
JOB	0	0
TRAVTIME	0	0
CAR_USE	0	0
BLUEBOOK	0	0
TIF	0	0
CAR_TYPE	0	0
RED_CAR	0	0
OLDCLAIM	0	0
CLM_FREQ	0	0
REVOKED	0	0
MVR_PTS	0	0
CAR_AGE	0	0
URBANICITY	0	0

We see no missing values now.

Near Zero Variance

We leverage `nearZeroVar` to identify any features with little variance. Features with low or zero variance will have little impact on a model's resolution. Similarly, a feature with low variance could lead to undue influence by exceptions and/or zero variance when data are split into training/testing sets or when using bootstrapping or resampling techniques. These would be good candidates for removal.

```
## [1] freqRatio     percentUnique zeroVar       nzv
## <0 rows> (or 0-length row.names)
```

We see no candidates for removal.

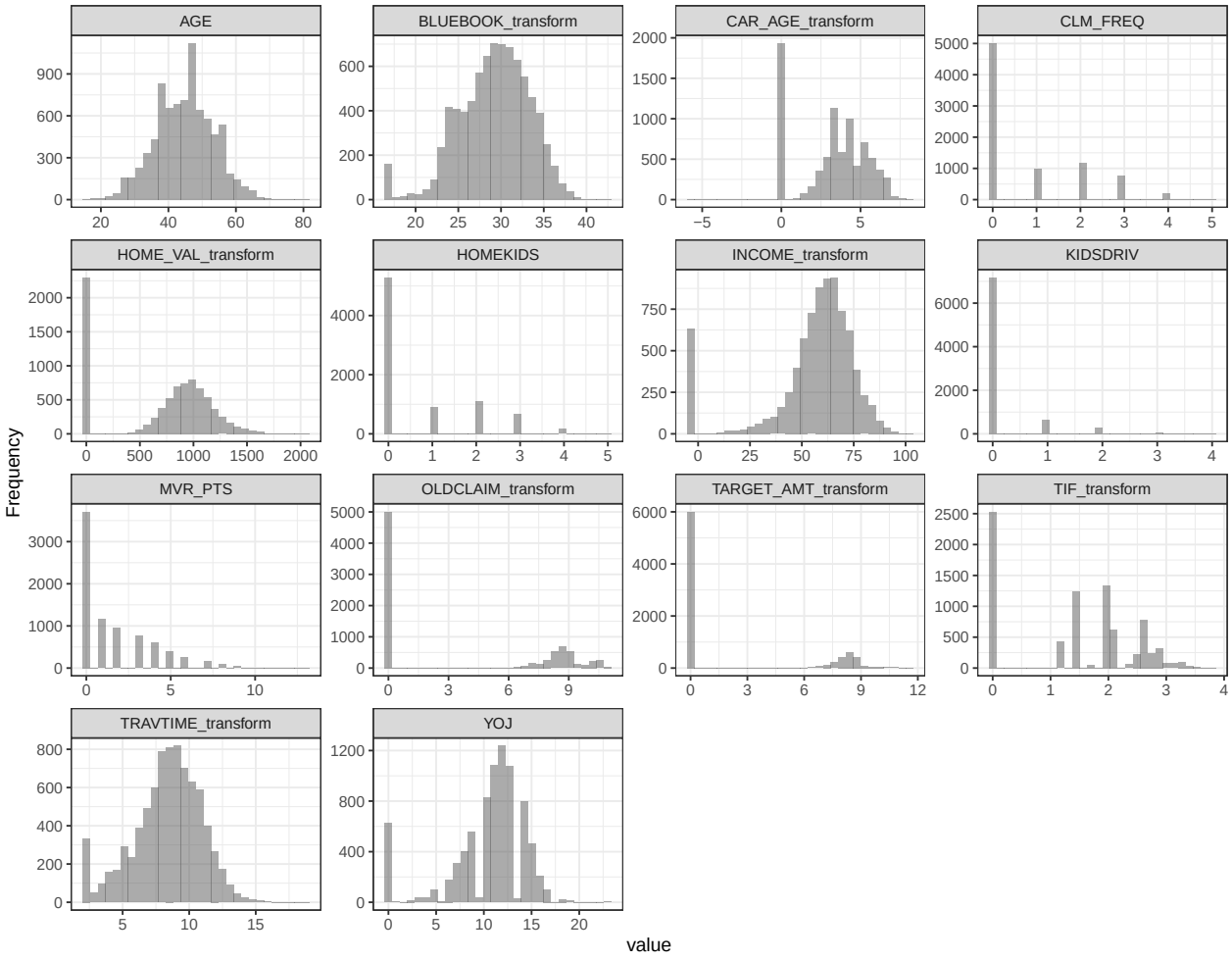
Outliers

No outliers were removed as all values seemed reasonable. We will rely on transformations to help address resolution issues introduced by extreme values.

Transform non-normal variables

Finally, as mentioned earlier in our data exploration, and our findings from our histogram plots, we can see that some of our variables are highly skewed. To address skew, we perform transformations to make the distributions more normally distributed. We apply BoxCox and log transformations.

Here are some plots to demonstrate the changes in distributions after the transformations:



As we can see from above, our transformations helped to alleviate some of the kurtosis in the distributions that were highly skewed. Now, we can see that `AGE_transform`, `BLUEBOOK_transform`, `CAR_AGE_transform`, `HOME_VAL_transform`, `INCOME_transform`, `TIF_transform`, and `TRAVTIME_transform` are all the result of Boxcox transformations on the original variables. Additionally, `TARGET_AMT_transform` and `OLDCLAIM_transform` are the results of log transformations.

3. Build Models

Using the training data, build at least two different multiple linear regression models and three different binary logistic models, using different variables (or the same variables with different transformations). You may select the variables manually, use an approach such as Forward or Stepwise, use a different approach, or use a combination of techniques. Describe the techniques you used. If you manually selected a variable for inclusion into the model or exclusion into the model, indicate why this was done.

Be sure to explain how you can make inferences from the model, as well as discuss other relevant model output. Discuss the coefficients in the models, do they make sense? Are you keeping the model even though it is counter-intuitive? Why? The boss needs to know.

Linear Regression

Linear Regression Model 1 The first model built was based on the entire dataset. The linear models will be attempting to predict the feature "TARGET_AMT" using the existing features within the dataset

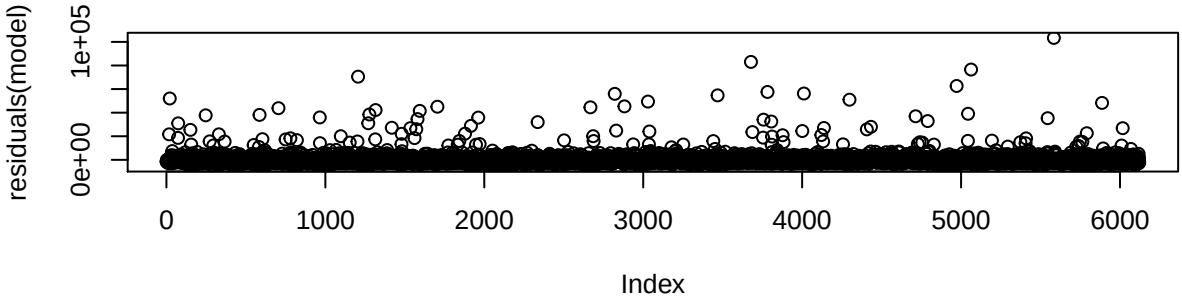
except for "TARGET_FLAG". These features are to be considered unknown.

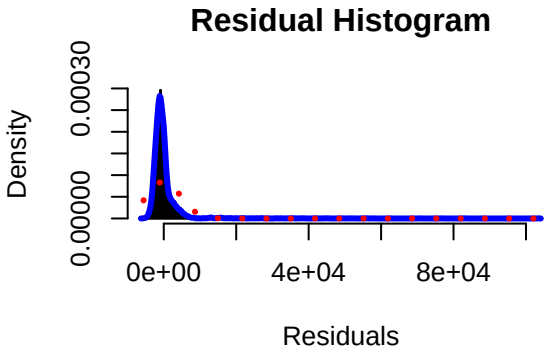
First, we split our cleaned dataset into training and testing sets (75% training, 25% testing). This was necessary as the provided holdout evaluation dataset didn't provide target values so we cannot measure our model performance against that dataset. This split was used on all models.

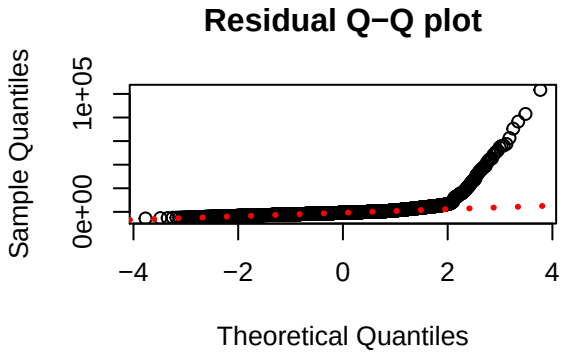
First linear regression model, no transformations, all numeric predictors included

```
##
## Call:
## stats::lm(formula = TARGET_AMT ~ AGE + BLUEBOOK + CAR_AGE + CLM_FREQ +
##     HOME_VAL + HOMEKIDS + INCOME + KIDSDRIV + MVR_PTS + OLDCLAIM +
##     TIF + TRAVTIME + YOJ + CAR_TYPE + CAR_USE + EDUCATION + JOB +
##     MSTATUS + PARENT1 + RED_CAR + REVOKED + SEX + URBANICITY,
##     data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -5586  -1697   -789    319 103372
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   1.196e+03  6.610e+02   1.809  0.07046 .
## AGE                           4.701e+00  8.397e+00   0.560  0.57562
## BLUEBOOK                      1.192e-02  1.037e-02   1.150  0.25027
## CAR_AGE                      -4.595e+01  1.519e+01  -3.025  0.00250 **
## CLM_FREQ                      1.216e+02  6.545e+01   1.858  0.06318 .
## HOME_VAL                     -1.122e-03  7.249e-04  -1.547  0.12190
## HOMEKIDS                      8.651e+01  7.783e+01   1.112  0.26639
## INCOME                       -3.342e-03  2.268e-03  -1.474  0.14064
## KIDSDRIV                      2.970e+02  1.350e+02   2.199  0.02791 *
## MVR_PTS                       1.796e+02  3.091e+01   5.809 6.59e-09 ***
## OLDCLAIM                     -1.154e-02  8.826e-03  -1.308  0.19105
## TIF                          -5.958e+01  1.454e+01  -4.097 4.24e-05 ***
## TRAVTIME                      1.116e+01  3.831e+00   2.912  0.00360 **
## YOJ                          -1.435e+01  1.792e+01  -0.801  0.42319
## CAR_TYPEPanel Truck           1.829e+02  3.302e+02   0.554  0.57967
## CAR_TYPEPickup                2.429e+02  2.025e+02   1.200  0.23026
## CAR_TYPESports Car            1.053e+03  2.594e+02   4.059 4.98e-05 ***
## CAR_TYPEVan                   7.123e+02  2.518e+02   2.829  0.00468 **
## CAR_TYPEz_SUV                 6.835e+02  2.146e+02   3.185  0.00145 **
## CAR_USEPrivate               -7.580e+02  1.942e+02  -3.904 9.56e-05 ***
## EDUCATIONBachelors            2.072e+00  2.426e+02   0.009  0.99319
## EDUCATIONMasters              3.645e+02  3.570e+02   1.021  0.30724
## EDUCATIONPhD                  6.416e+02  4.219e+02   1.521  0.12842
## EDUCATIONz_High School        1.805e+01  2.045e+02   0.088  0.92966
## JOBClerical                   6.944e+02  4.050e+02   1.715  0.08645 .
## JOBDoctor                    -3.627e+02  4.833e+02  -0.751  0.45297
## JOBHome Maker                 4.854e+02  4.339e+02   1.119  0.26329
## JOBLawyer                     2.991e+02  3.513e+02   0.851  0.39454
## JOBManager                   -2.915e+02  3.422e+02  -0.852  0.39434
## JOBProfessional               6.865e+02  3.658e+02   1.877  0.06058 .
## JOBStudent                    4.244e+02  4.461e+02   0.951  0.34144
## JOBz_Blue Collar              8.108e+02  3.815e+02   2.125  0.03362 *
## MSTATUSz_No                   5.423e+02  1.747e+02   3.104  0.00191 **
```

```
## PARENT1Yes                          3.826e+02  2.400e+02   1.594  0.11100
## RED_CARyes                         -1.143e+02  1.786e+02  -0.640  0.52221
## REVOKEDYes                          4.836e+02  2.052e+02   2.357  0.01843 *
## SEXz_F                             -3.626e+02  2.194e+02  -1.653  0.09846 .
## URBANICITYz_Highly Rural/ Rural -1.656e+03  1.658e+02  -9.989  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4675 on 6083 degrees of freedom
## Multiple R-squared:  0.06739,    Adjusted R-squared:  0.06172
## F-statistic: 11.88 on 37 and 6083 DF,  p-value: < 2.2e-16
```





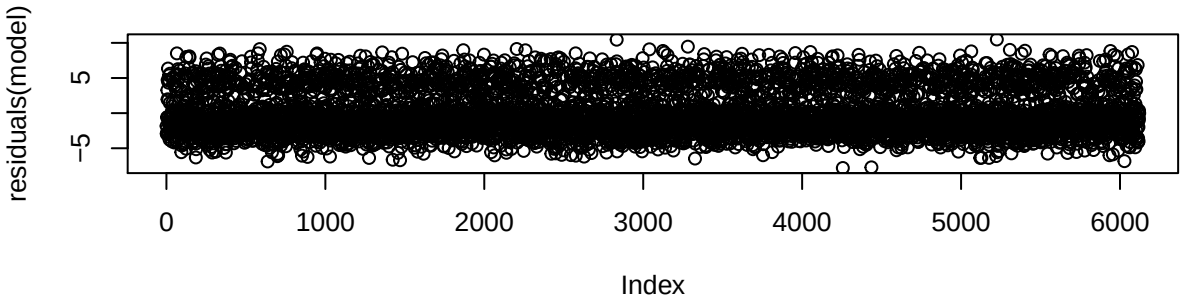


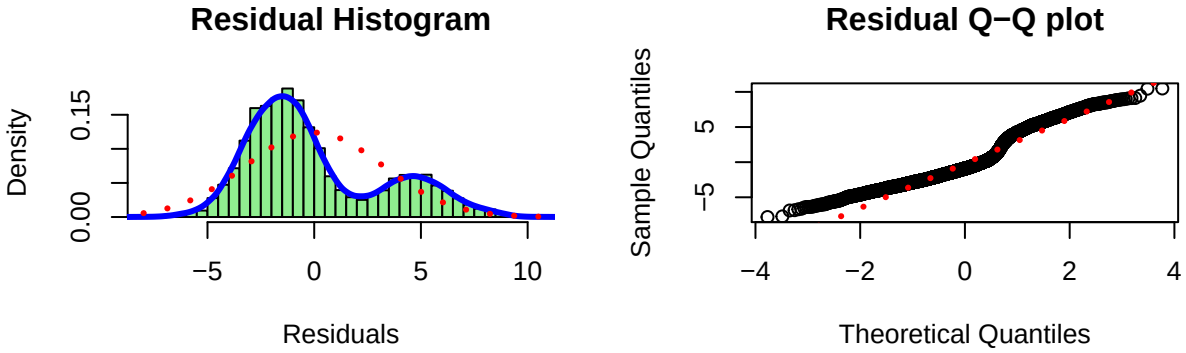
Results of the first model show that using all non-transformed data doesn't fully satisfy the assumptions for a linear regression model. We can also see several predictors that show insignificant correlations but before we discount any features, we will use the transformed data to satisfy the conditions of linear regression. Our R^2 **was 0.06** and our **error was 4675** which is indicative of a very poor model.

Linear Regression Model 2 The second model built was based on the entire dataset as well but various features were transformed as discussed in the Data Preparation section of this assignment, e.g. by applying BoxCox and log transformations. A linear regression model was fit to assess the predictability of our response variable.

```
##
## Call:
## stats::lm(formula = TARGET_AMT_transform ~ ., data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.8072 -2.3239 -0.9094  1.9629 10.4832
```

```
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     3.6525701  0.5896527   6.194 6.23e-10 ***
## AGE                             0.0022763  0.0058109   0.392 0.695268
## BLUEBOOK_transform             -0.0450465  0.0135550  -3.323 0.000895 ***
## CAR_AGE_transform              -0.0436398  0.0260314  -1.676 0.093706 .
## CAR_TYPEPanel Truck             0.2696825  0.2176776   1.239 0.215427
## CAR_TYPEPickup                  0.5349225  0.1397707   3.827 0.000131 ***
## CAR_TYPESports Car              1.0611065  0.1783942   5.948 2.86e-09 ***
## CAR_TYPEVan                     0.6540534  0.1734607   3.771 0.000164 ***
## CAR_TYPEz_SUV                   0.7966993  0.1452430   5.485 4.29e-08 ***
## CAR_USEPrivate                 -0.9603331  0.1341355  -7.159 9.06e-13 ***
## CLM_FREQ                        0.1265979  0.0706179   1.793 0.073068 .
## EDUCATIONBachelors             -0.2608015  0.1689184  -1.544 0.122653
## EDUCATIONMasters               -0.0405783  0.2415543  -0.168 0.866598
## EDUCATIONPhD                    0.0610753  0.2802850   0.218 0.827511
## EDUCATIONz_High School          0.1607296  0.1416156   1.135 0.256432
## HOME_VAL_transform             -0.0004504  0.0001237  -3.642 0.000272 ***
## HOMEKIDS                        0.0768615  0.0540685   1.422 0.155206
## INCOME_transform               -0.0149842  0.0040164  -3.731 0.000193 ***
## JOBClerical                     0.7530586  0.2775453   2.713 0.006681 **
## JOBDoctor                      -0.2722778  0.3337705  -0.816 0.414667
## JOBHome Maker                   0.2315929  0.3150573   0.735 0.462318
## JOBLawyer                       0.2394928  0.2422125   0.989 0.322814
## JOBManager                     -0.4610222  0.2360851  -1.953 0.050891 .
## JOBProfessional                 0.4794468  0.2521012   1.902 0.057244 .
## JOBStudent                      0.1418251  0.3272668   0.433 0.664767
## JOBz_Blue Collar                0.7664547  0.2625805   2.919 0.003525 **
## KIDSDRIV                        0.5033165  0.0933104   5.394 7.15e-08 ***
## MSTATUSz_No                     0.5541900  0.1238162   4.476 7.75e-06 ***
## MVR_PTS                         0.1803303  0.0219374   8.220 2.46e-16 ***
## OLDCLAIM_transform              0.0152553  0.0198498   0.769 0.442198
## PARENT1Yes                      0.6159434  0.1655566   3.720 0.000201 ***
## RED_CARyes                     -0.0485750  0.1233161  -0.394 0.693664
## REVOKEDYes                      1.0131055  0.1286314   7.876 3.97e-15 ***
## SEXz_F                         -0.1151818  0.1485014  -0.776 0.437999
## TIF_transform                  -0.3134415  0.0387586  -8.087 7.32e-16 ***
## TRAVTIME_transform              0.0983533  0.0165498   5.943 2.96e-09 ***
## URBANICITYz_Highly Rural/ Rural -2.3793240  0.1150897 -20.674  < 2e-16 ***
## YOJ                            -0.0049852  0.0142641  -0.349 0.726729
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.229 on 6083 degrees of freedom
## Multiple R-squared:  0.2267, Adjusted R-squared:  0.222
## F-statistic: 48.21 on 37 and 6083 DF,  p-value: < 2.2e-16
```



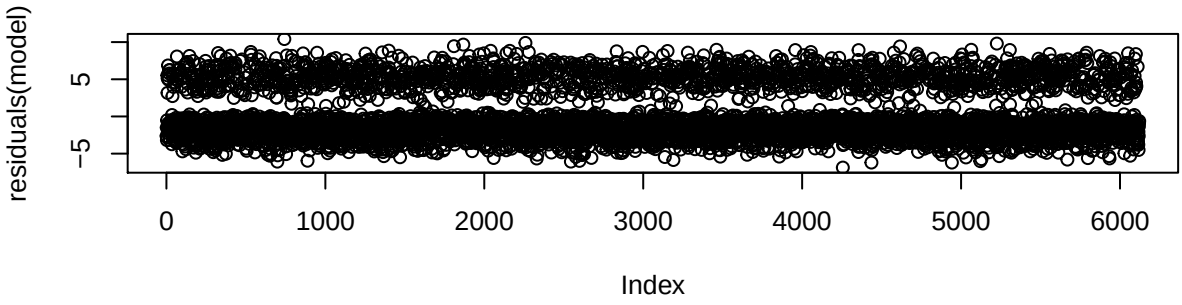


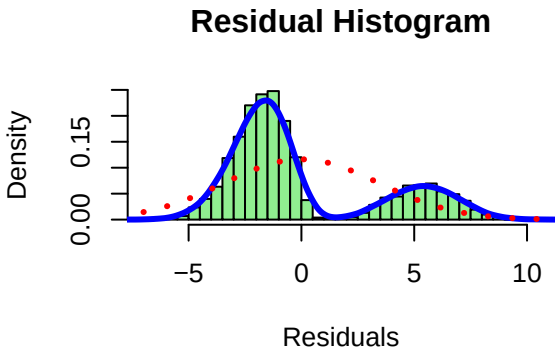
Results of this model show that we satisfy the requirements of linear regression with residuals scattered around 0 and a normal-like Q-Q plot. There are still several non-significant variables such as `OLDCLAIM`, `SEXz_F`, `YOJ`, `JOBDoctor`, `JOBLawyer`, `JOBStudent` and `CAR_TYPE`. The Model 2 R^2 **is 0.22** with a **standard error of 3.2**, which shows better performance over Model 1.

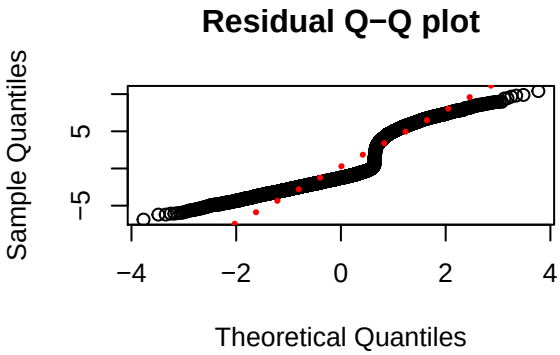
Linear Regression Model 3 This third model builds upon attempts to perform a linear regression by removing select features that are insignificant to the model created in Model 2. In addition to feature selection, using the tidymodels framework, we removed any features that are highly correlated with each other as well as any zero-variance features. Only numeric features were considered in this model.

```
##
## Call:
## stats::lm(formula = TARGET_AMT_transform ~ ., data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.875 -2.296 -1.261  2.823 10.407
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.593e+00  3.656e-01   9.826  < 2e-16 ***
## BLUEBOOK_transform  -4.961e-02  1.099e-02  -4.514 6.49e-06 ***
## CAR_AGE_transform   -1.095e-01  2.121e-02  -5.162 2.52e-07 ***
## HOME_VAL_transform  -9.218e-04  9.478e-05  -9.726  < 2e-16 ***
## HOMEKIDS             1.564e-01  4.534e-02   3.450 0.000564 ***
## KIDSDRIV             4.500e-01  9.747e-02   4.617 3.97e-06 ***
## MVR_PTS              2.262e-01  2.317e-02   9.764  < 2e-16 ***
## OLDCLAIM_transform   1.235e-01  1.151e-02  10.723  < 2e-16 ***
## TIF_transform       -3.069e-01  4.112e-02  -7.462 9.68e-14 ***
## TRAVTIME_transform   5.690e-02  1.733e-02   3.283 0.001034 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.436 on 6111 degrees of freedom
## Multiple R-squared:  0.1208, Adjusted R-squared:  0.1195
## F-statistic: 93.28 on 9 and 6111 DF,  p-value: < 2.2e-16
```








```
## Selecting by Overall
## Selecting by Overall
## Selecting by Overall
```

This model did not perform as well linear Model 2. This is likely due to the omission of features that were not numeric. Model 3 R^2 **was 0.12** with an **RMSE of 3.43**.

Logistic Regression Model

Logistic Regression Model 1 The next set of models built are logistic regression models. These predict the outcome of non-numeric variables. This is done by assessing the probability that a particular class will occur. In this case, we are doing a binary classification of the feature "TARGET_FLAG" where the outcome is 0 or 1.

Similar to the linear regression models, the first model built was based on the entire dataset. The logistic models will be attempting to predict the feature "TARGET_FLAG" using the existing features within the dataset except for "TARGET_AMT". These features are to be considered unknown.

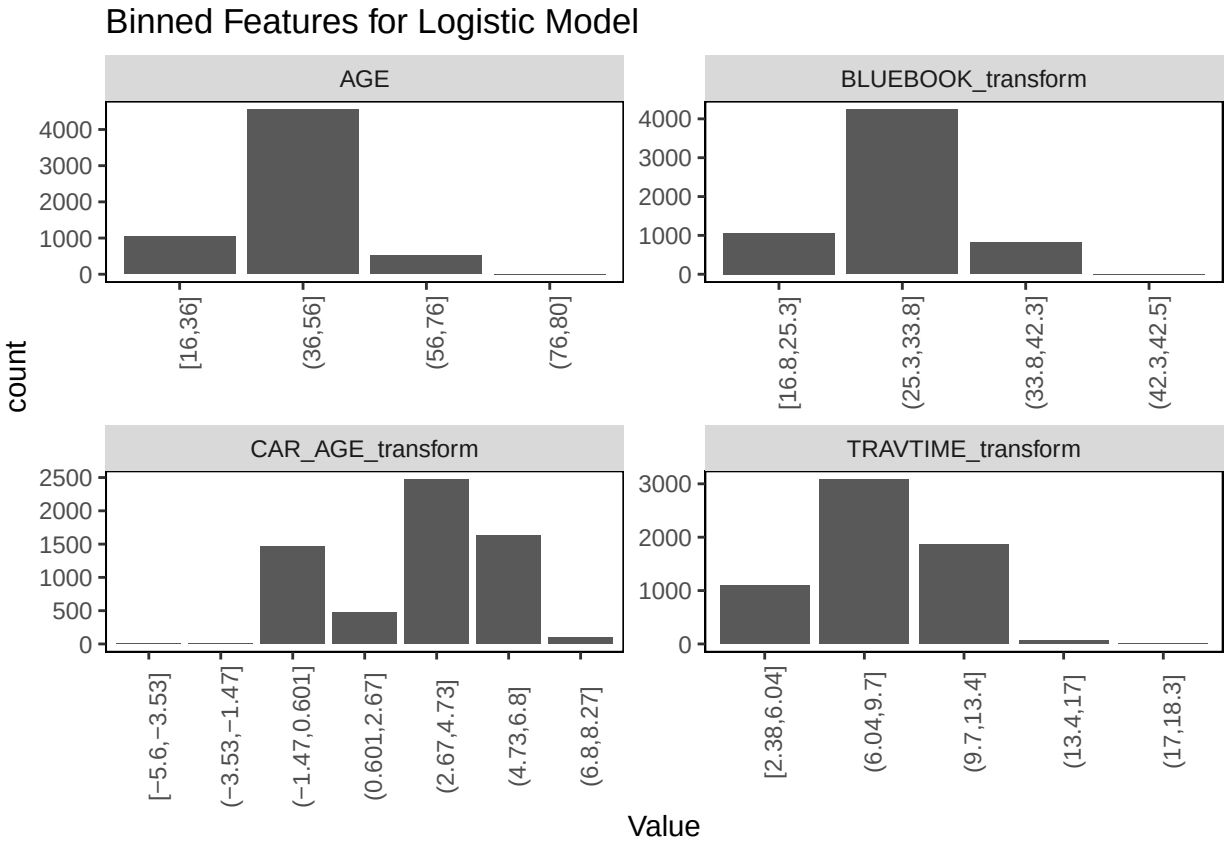
```
##
## Call:
## stats::glm(formula = TARGET_FLAG ~ BLUEBOOK + CAR_AGE + HOME_VAL +
##     INCOME + PARENT1 + HOME_VAL + MSTATUS + SEX + EDUCATION +
```

```
##     JOB + TRAVTIME + CAR_USE + KIDSDRIV + AGE + HOMEKIDS + YOJ +
##     CAR_TYPE + RED_CAR + OLDCLAIM + CLM_FREQ + REVOKED + MVR_PTS +
##     CAR_AGE + URBANICITY, family = stats::binomial, data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4579  -0.7228  -0.4096   0.6134   2.9266
##
## Coefficients:
##                                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)                     -1.365e+00  3.650e-01  -3.739 0.000184 ***
## BLUEBOOK                        -2.021e-05  6.103e-06  -3.311 0.000930 ***
## CAR_AGE                         -1.335e-02  8.635e-03  -1.546 0.122161
## HOME_VAL                        -1.462e-06  4.007e-07  -3.648 0.000265 ***
## INCOME                          -2.416e-06  1.282e-06  -1.885 0.059487 .
## PARENT1Yes                       3.827e-01  1.258e-01   3.042 0.002348 **
## MSTATUSz_No                      4.646e-01  9.715e-02   4.783 1.73e-06 ***
## SEXz_F                          -6.515e-02  1.284e-01  -0.508 0.611789
## EDUCATIONBachelors              -2.591e-01  1.318e-01  -1.967 0.049233 *
## EDUCATIONMasters                -1.289e-01  2.043e-01  -0.631 0.528268
## EDUCATIONPhD                     6.120e-02  2.420e-01   0.253 0.800332
## EDUCATIONz_High School           6.478e-02  1.092e-01   0.593 0.552861
## JOBClerical                      4.823e-01  2.264e-01   2.130 0.033133 *
## JOBDoctor                       -3.475e-01  3.005e-01  -1.157 0.247454
## JOBHome Maker                    2.940e-01  2.412e-01   1.219 0.222907
## JOBLawyer                        1.357e-01  1.983e-01   0.684 0.493786
## JOBManager                      -4.817e-01  1.989e-01  -2.422 0.015436 *
## JOBProfessional                  2.325e-01  2.062e-01   1.128 0.259475
## JOBStudent                       2.845e-01  2.477e-01   1.149 0.250735
## JOBz_Blue Collar                 4.082e-01  2.131e-01   1.916 0.055399 .
## TRAVTIME                         1.346e-02  2.154e-03   6.247 4.17e-10 ***
## CAR_USEPrivate                  -7.212e-01  1.049e-01  -6.878 6.08e-12 ***
## KIDSDRIV                         3.824e-01  7.017e-02   5.450 5.04e-08 ***
## AGE                              2.761e-03  4.594e-03   0.601 0.547887
## HOMEKIDS                         7.568e-02  4.260e-02   1.777 0.075613 .
## YOJ                             -2.135e-02  9.811e-03  -2.176 0.029533 *
## CAR_TYPEPanel Truck              4.137e-01  1.868e-01   2.214 0.026801 *
## CAR_TYPEPickup                   4.970e-01  1.150e-01   4.321 1.55e-05 ***
## CAR_TYPESports Car               9.410e-01  1.483e-01   6.344 2.24e-10 ***
## CAR_TYPEVan                      6.382e-01  1.440e-01   4.433 9.29e-06 ***
## CAR_TYPEz_SUV                    6.840e-01  1.276e-01   5.360 8.30e-08 ***
## RED_CARyes                      -2.742e-02  1.003e-01  -0.273 0.784523
## OLDCLAIM                        -1.256e-05  4.468e-06  -2.812 0.004927 **
## CLM_FREQ                         1.808e-01  3.295e-02   5.489 4.04e-08 ***
## REVOKEDYes                       8.748e-01  1.040e-01   8.409  < 2e-16 ***
## MVR_PTS                          1.241e-01  1.573e-02   7.889 3.04e-15 ***
## URBANICITYz_Highly Rural/ Rural -2.292e+00  1.280e-01 -17.912  < 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: 7037.3  on 6120  degrees of freedom
## Residual deviance: 5529.6  on 6084  degrees of freedom
```

```
## AIC: 5603.6
##
## Number of Fisher Scoring iterations: 5
```

The results of the first Logistic regression model on the untransformed data show an **AIC=5486** with several insignificant predictors. We can use this information on the P-values to assess which features can be removed from the model.

Logistic Regression Model 2 In the second logistic regression model, we remove variables with low variance, as well as variables that may exhibit collinearity. Additionally, 4 numeric features from our training set were binned into groups and used as predictors in lieu of numerically based predictors. The plot below shows the 4 features that were chosen to be binned within the analysis.




```
##
## Call:
## stats::glm(formula = TARGET_FLAG ~ ., family = stats::binomial,
##     data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3281  -0.6980  -0.3924   0.5596   2.8649
##
## Coefficients:
##                                Estimate Std. Error z value Pr(>|z|)
## (Intercept)                   1.366e+01  5.354e+02   0.026 0.979645
## AGE(36,56]                   -2.731e-01  9.812e-02  -2.783 0.005379 **
```

```
## AGE(56,76]                        6.111e-01  1.473e-01   4.147 3.36e-05 ***
## AGE(76,80]                       -1.030e+01  5.354e+02  -0.019 0.984652
## BLUEBOOK_transform(25.3,33.8]    -3.900e-01  9.400e-02  -4.149 3.35e-05 ***
## BLUEBOOK_transform(33.8,42.3]    -7.578e-01  1.612e-01  -4.701 2.60e-06 ***
## BLUEBOOK_transform(42.3,42.5]    -1.236e+01  5.354e+02  -0.023 0.981587
## CAR_AGE_transform(-3.53,-1.47]   -2.494e+01  6.127e+02  -0.041 0.967532
## CAR_AGE_transform(-1.47,0.601]   -1.373e+01  5.354e+02  -0.026 0.979539
## CAR_AGE_transform(0.601,2.67]    -1.382e+01  5.354e+02  -0.026 0.979400
## CAR_AGE_transform(2.67,4.73]     -1.382e+01  5.354e+02  -0.026 0.979406
## CAR_AGE_transform(4.73,6.8]      -1.396e+01  5.354e+02  -0.026 0.979206
## CAR_AGE_transform(6.8,8.27]      -1.458e+01  5.354e+02  -0.027 0.978268
## CAR_TYPEPanel Truck               4.107e-01  1.842e-01   2.229 0.025817 *
## CAR_TYPEPickup                    5.367e-01  1.168e-01   4.596 4.32e-06 ***
## CAR_TYPESports Car                8.421e-01  1.450e-01   5.809 6.29e-09 ***
## CAR_TYPEVan                       6.090e-01  1.420e-01   4.287 1.81e-05 ***
## CAR_TYPEz_SUV                     6.551e-01  1.224e-01   5.351 8.76e-08 ***
## CAR_USEPrivate                   -7.280e-01  1.065e-01  -6.834 8.27e-12 ***
## CLM_FREQ                          8.306e-02  5.120e-02   1.622 0.104750
## EDUCATIONBachelors               -2.377e-01  1.343e-01  -1.770 0.076775 .
## EDUCATIONMasters                 -1.111e-01  2.062e-01  -0.539 0.589842
## EDUCATIONPhD                     -1.231e-02  2.383e-01  -0.052 0.958792
## EDUCATIONz_High School            7.617e-02  1.123e-01   0.678 0.497690
## HOME_VAL_transform               -3.941e-04  9.860e-05  -3.997 6.41e-05 ***
## HOMEKIDS                          4.233e-02  4.299e-02   0.985 0.324831
## INCOME_transform                 -1.242e-02  3.261e-03  -3.810 0.000139 ***
## JOBClerical                       4.536e-01  2.283e-01   1.987 0.046921 *
## JOBDoctor                        -4.580e-01  3.031e-01  -1.511 0.130741
## JOBHome Maker                    -5.740e-02  2.617e-01  -0.219 0.826377
## JOBLawyer                         7.345e-02  2.004e-01   0.367 0.713956
## JOBManager                       -5.383e-01  2.005e-01  -2.685 0.007260 **
## JOBProfessional                   2.091e-01  2.087e-01   1.002 0.316428
## JOBStudent                       -1.693e-01  2.681e-01  -0.631 0.527756
## JOBz_Blue Collar                  4.076e-01  2.158e-01   1.889 0.058862 .
## KIDSDRIV                          4.718e-01  7.438e-02   6.343 2.25e-10 ***
## MSTATUSz_No                       4.923e-01  1.018e-01   4.836 1.33e-06 ***
## MVR_PTS                           1.019e-01  1.651e-02   6.171 6.80e-10 ***
## OLDCLAIM_transform                1.957e-02  1.460e-02   1.340 0.180259
## PARENT1Yes                        3.360e-01  1.275e-01   2.636 0.008392 **
## RED_CARyes                       -3.336e-02  1.021e-01  -0.327 0.743885
## REVOKEDYes                        6.809e-01  9.506e-02   7.163 7.92e-13 ***
## SEXz_F                           -2.551e-02  1.235e-01  -0.206 0.836440
## TIF_transform                    -2.408e-01  3.181e-02  -7.572 3.68e-14 ***
## TRAVTIME_transform(6.04,9.7]      2.715e-01  9.675e-02   2.806 0.005015 **
## TRAVTIME_transform(9.7,13.4]      6.150e-01  1.039e-01   5.917 3.28e-09 ***
## TRAVTIME_transform(13.4,17]       1.988e-01  3.811e-01   0.522 0.601996
## TRAVTIME_transform(17,18.3]      -9.893e+00  3.637e+02  -0.027 0.978298
## URBANICITYz_Highly Rural/ Rural -2.284e+00  1.289e-01 -17.721  < 2e-16 ***
## YOJ                              -4.063e-03  1.162e-02  -0.350 0.726558
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 7037.3  on 6120  degrees of freedom
```

```
## Residual deviance: 5386.6  on 6071  degrees of freedom
## AIC: 5486.6
##
## Number of Fisher Scoring iterations: 12
```

Our logistic model has an **AIC=5486**. This is lower than Model 1 and implies that this is better at predicting the classification for `TARGET_FLAG`.

Logistic Regression Model 3 One of the measures to consider when running classification models is the distribution of the binary classification. In this case, the dataset innately has a biased classification, at 0.73% of values showing 0, and 27% of values showing 1.

```
##
##         0         1
## 0.7361843 0.2638157
```

Because of this class imbalance, if we were to simply guess that all values are 0, that would technically lead to a 73% accurate model. To account for this, we will downsample the majority class in the final logistic model. This downsampling will allow the distribution of the training set to be 50/50. The testing dataset will remain as is.

```
##
##   0   1
## 0.5 0.5
```

```
##
## Call:
## stats::glm(formula = TARGET_FLAG ~ ., family = stats::binomial,
##     data = data)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.82953  -0.83526   0.00015   0.82576   2.50218
##
## Coefficients:
##                                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)                      1.378e+01  5.354e+02   0.026 0.979460
## AGE(36,56]                      -2.449e-01  1.276e-01  -1.919 0.054974 .
## AGE(56,76]                       7.372e-01  1.942e-01   3.796 0.000147 ***
## BLUEBOOK_transform(25.3,33.8]   -3.761e-01  1.208e-01  -3.115 0.001841 **
## BLUEBOOK_transform(33.8,42.3]   -6.091e-01  2.048e-01  -2.974 0.002938 **
## CAR_AGE_transform(-3.53,-1.47]  -2.436e+01  6.527e+02  -0.037 0.970231
## CAR_AGE_transform(-1.47,0.601]  -1.285e+01  5.354e+02  -0.024 0.980847
## CAR_AGE_transform(0.601,2.67]   -1.300e+01  5.354e+02  -0.024 0.980634
## CAR_AGE_transform(2.67,4.73]    -1.289e+01  5.354e+02  -0.024 0.980786
## CAR_AGE_transform(4.73,6.8]     -1.294e+01  5.354e+02  -0.024 0.980713
## CAR_AGE_transform(6.8,8.27]     -1.358e+01  5.354e+02  -0.025 0.979771
## CAR_TYPEPanel Truck              2.423e-01  2.348e-01   1.032 0.302085
## CAR_TYPEPickup                   4.567e-01  1.466e-01   3.114 0.001843 **
## CAR_TYPESports Car               9.331e-01  1.826e-01   5.111 3.21e-07 ***
## CAR_TYPEVan                      6.007e-01  1.763e-01   3.407 0.000656 ***
## CAR_TYPEz_SUV                    8.021e-01  1.522e-01   5.271 1.36e-07 ***
```

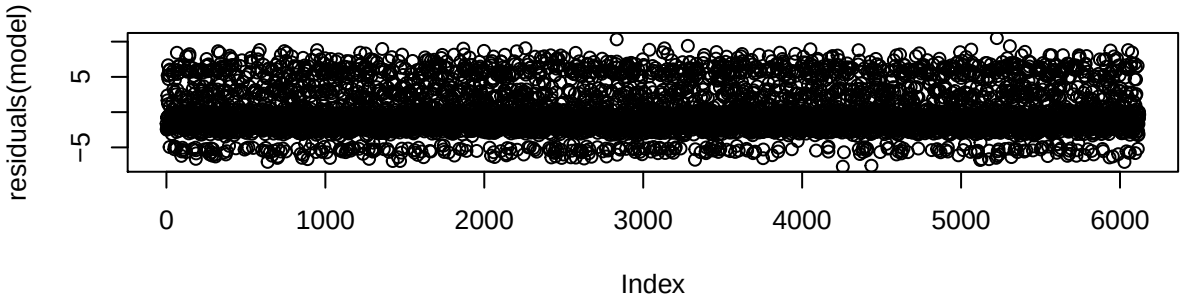
```
## CAR_USEPrivate                  -8.733e-01  1.372e-01  -6.363 1.97e-10 ***
## CLM_FREQ                         4.885e-02  6.803e-02   0.718 0.472684
## EDUCATIONBachelors              -1.693e-01  1.698e-01  -0.997 0.318597
## EDUCATIONMasters                 1.281e-01  2.597e-01   0.493 0.621944
## EDUCATIONPhD                     1.612e-01  3.065e-01   0.526 0.598818
## EDUCATIONz_High School           2.225e-01  1.431e-01   1.554 0.120126
## HOME_VAL_transform              -4.454e-04  1.260e-04  -3.536 0.000406 ***
## HOMEKIDS                         4.943e-02  5.482e-02   0.902 0.367285
## INCOME_transform                -1.497e-02  4.155e-03  -3.602 0.000316 ***
## JOBClerical                      6.519e-01  2.832e-01   2.302 0.021346 *
## JOBDoctor                       -5.432e-01  3.608e-01  -1.505 0.132216
## JOBHome Maker                    1.465e-01  3.209e-01   0.457 0.647987
## JOBLawyer                       -1.019e-02  2.452e-01  -0.042 0.966858
## JOBManager                      -3.502e-01  2.443e-01  -1.434 0.151701
## JOBProfessional                  4.409e-01  2.617e-01   1.685 0.091998 .
## JOBStudent                       7.238e-02  3.339e-01   0.217 0.828374
## JOBz_Blue Collar                 5.588e-01  2.698e-01   2.071 0.038381 *
## KIDSDRIV                         4.812e-01  1.004e-01   4.795 1.63e-06 ***
## MSTATUSz_No                      6.108e-01  1.279e-01   4.775 1.80e-06 ***
## MVR_PTS                          1.097e-01  2.192e-02   5.006 5.57e-07 ***
## OLDCLAIM_transform               3.433e-02  1.920e-02   1.788 0.073721 .
## PARENT1Yes                       3.218e-01  1.660e-01   1.938 0.052587 .
## RED_CARyes                      -4.719e-02  1.295e-01  -0.364 0.715608
## REVOKEDYes                       6.927e-01  1.284e-01   5.396 6.82e-08 ***
## SEXz_F                          -1.323e-01  1.556e-01  -0.850 0.395267
## TIF_transform                   -2.958e-01  4.089e-02  -7.235 4.67e-13 ***
## TRAVTIME_transform(6.04,9.7]     2.548e-01  1.221e-01   2.086 0.036937 *
## TRAVTIME_transform(9.7,13.4]     5.980e-01  1.318e-01   4.537 5.71e-06 ***
## TRAVTIME_transform(13.4,17]      3.339e-01  4.397e-01   0.759 0.447682
## URBANICITYz_Highly Rural/ Rural -2.310e+00  1.496e-01 -15.442  < 2e-16 ***
## YOJ                             -6.071e-03  1.495e-02  -0.406 0.684613
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4441.7  on 3203  degrees of freedom
## Residual deviance: 3256.3  on 3157  degrees of freedom
## AIC: 3350.3
##
## Number of Fisher Scoring iterations: 12

## Selecting by Overall
## Selecting by Overall
## Selecting by Overall
```

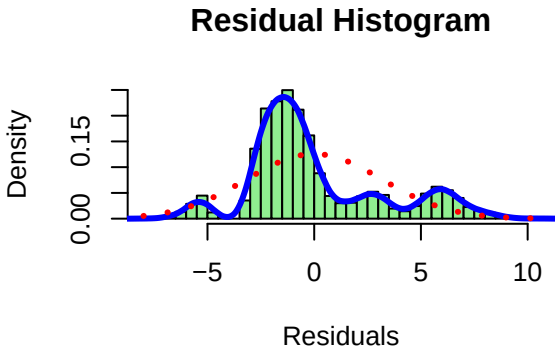
This model has an **AIC score of 3350**, significantly less than that of the first two logistic models and thus we can say that this model has performed better. Additional metrics on each model are presented in the Model Performance Summary at the end of this document.

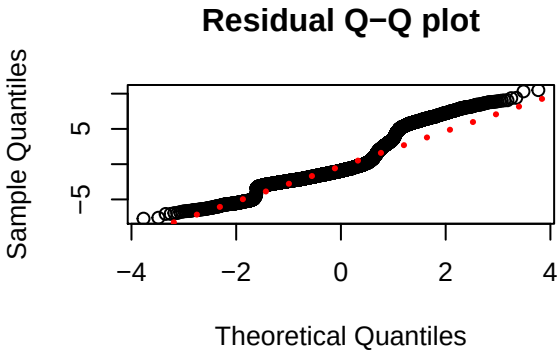
Using Logistic Results to improve Linear Regression Finally, to tune the linear regression model, we added the predictions made from our final logistic regression model to obtain a new feature `TARGET_FLAG`. We will rerun a linear regression with the added feature.

```
##
## Call:
## stats::lm(formula = TARGET_AMT_transform ~ ., data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -7.727 -1.984 -0.934  1.376 10.497
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       1.246e+00  6.105e-01   2.040 0.041354 *
## AGE                              -2.257e-03  5.743e-03  -0.393 0.694378
## BLUEBOOK_transform               -3.672e-02  1.339e-02  -2.742 0.006116 **
## CAR_AGE_transform                -3.407e-02  2.569e-02  -1.326 0.184855
## CAR_TYPEPanel Truck               2.849e-01  2.147e-01   1.326 0.184727
## CAR_TYPEPickup                    4.113e-01  1.382e-01   2.976 0.002932 **
## CAR_TYPESports Car                7.874e-01  1.772e-01   4.442 9.06e-06 ***
## CAR_TYPEVan                       5.357e-01  1.714e-01   3.126 0.001780 **
## CAR_TYPEz_SUV                     5.735e-01  1.443e-01   3.974 7.15e-05 ***
## CAR_USEPrivate                   -6.676e-01  1.342e-01  -4.973 6.76e-07 ***
## CLM_FREQ                          7.363e-02  6.978e-02   1.055 0.291406
## EDUCATIONBachelors               -2.035e-01  1.667e-01  -1.221 0.222189
## EDUCATIONMasters                 -6.280e-02  2.383e-01  -0.264 0.792147
## EDUCATIONPhD                      2.707e-02  2.765e-01   0.098 0.922005
## EDUCATIONz_High School            9.169e-02  1.398e-01   0.656 0.511972
## HOME_VAL_transform               -3.179e-04  1.224e-04  -2.597 0.009425 **
## HOMEKIDS                          5.050e-02  5.338e-02   0.946 0.344149
## INCOME_transform                 -9.809e-03  3.982e-03  -2.463 0.013799 *
## JOBClerical                       4.233e-01  2.750e-01   1.539 0.123779
## JOBDoctor                        -3.403e-01  3.293e-01  -1.033 0.301539
## JOBHome Maker                     1.046e-01  3.110e-01   0.336 0.736672
## JOBLawyer                         1.100e-01  2.392e-01   0.460 0.645415
## JOBManager                       -4.648e-01  2.329e-01  -1.996 0.045990 *
## JOBProfessional                   2.828e-01  2.492e-01   1.135 0.256480
## JOBStudent                        1.505e-03  3.230e-01   0.005 0.996283
## JOBz_Blue Collar                  4.361e-01  2.603e-01   1.675 0.093914 .
## KIDSDRIV                          3.460e-01  9.285e-02   3.727 0.000196 ***
## MSTATUSz_No                       4.229e-01  1.226e-01   3.450 0.000564 ***
## MVR_PTS                           1.088e-01  2.233e-02   4.872 1.13e-06 ***
## OLDCLAIM_transform                2.079e-02  1.959e-02   1.062 0.288441
## PARENT1Yes                        3.327e-01  1.648e-01   2.019 0.043492 *
## RED_CARyes                       -1.425e-02  1.217e-01  -0.117 0.906781
## REVOKEDYes                        6.398e-01  1.301e-01   4.917 9.00e-07 ***
## SEXz_F                           -4.570e-02  1.466e-01  -0.312 0.755234
## TIF_transform                    -2.154e-01  3.897e-02  -5.528 3.38e-08 ***
## TRAVTIME_transform                7.394e-02  1.643e-02   4.499 6.96e-06 ***
## URBANICITYz_Highly Rural/ Rural -1.861e+00  1.204e-01 -15.463  < 2e-16 ***
## YOJ                               2.166e-05  1.408e-02   0.002 0.998773
## preds                             1.954e+00  1.505e-01  12.980  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.186 on 6082 degrees of freedom
## Multiple R-squared:  0.2476, Adjusted R-squared:  0.2429
```

```
## F-statistic: 52.67 on 38 and 6082 DF,  p-value: < 2.2e-16
```








```
## Selecting by Overall
```

After using the results of the optimal logistic regression model to predict TARGET_FLAG and passing that into linear regression, the R^2 for the linear regression model has increased to **0.2476** with a **standard error of 3.186**.

Model Performance Summary

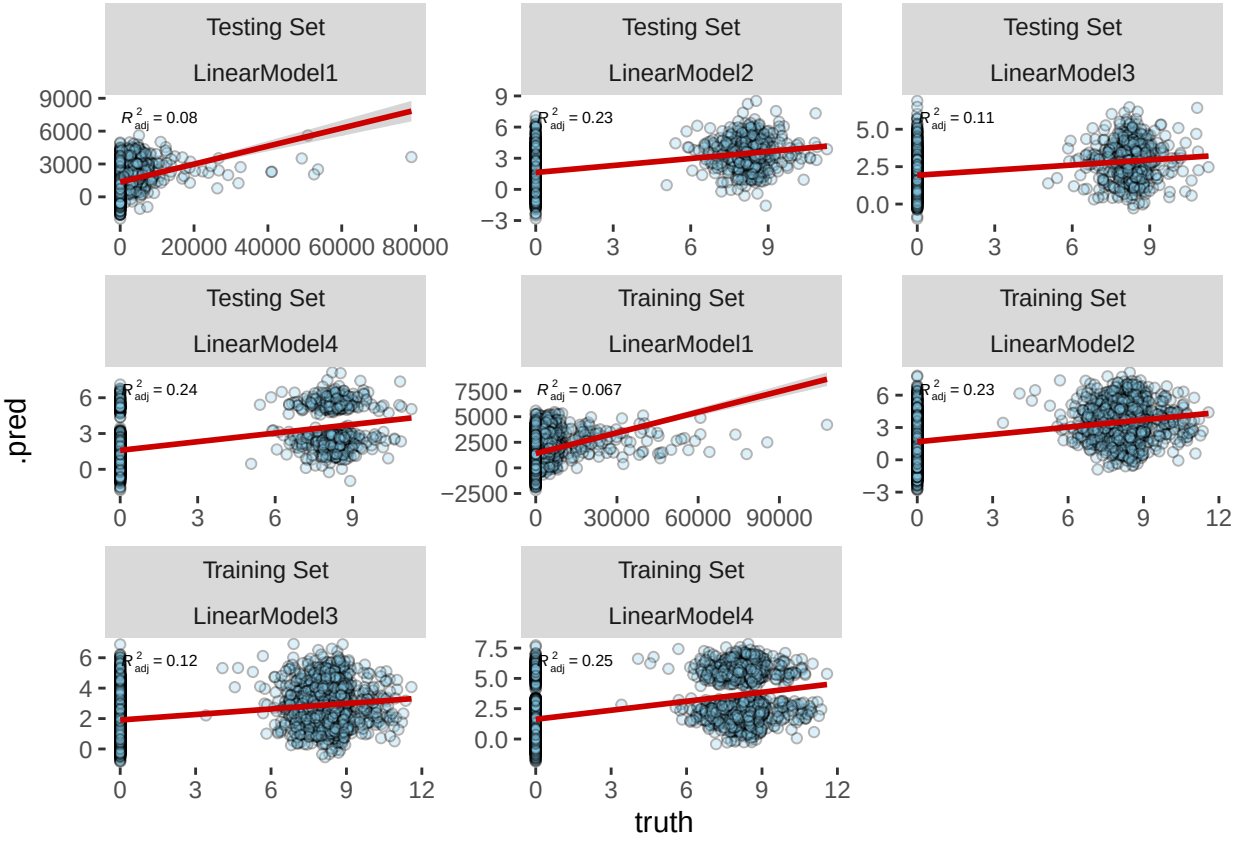
Finally, we can view the results of each of the linear regression models. The results for both the training and testing datasets are plotted below. There are many data points where `TARGET_FLAG` is 0 and thus leads to regression plots that may be weighted accordingly. Omitting points with 0 do not improve the model.

Table 1: Training Dataset Model Results

model	set	rmse	rsq	mae
LinearModel1	Training Set	4660.736206	0.0673940	1994.766874
LinearModel2	Training Set	3.219356	0.2267470	2.606465
LinearModel3	Training Set	3.432846	0.1207904	2.813728
LinearModel4	Training Set	3.175672	0.2475892	2.460059

Table 2: Testing Dataset Model Results

model	set	rmse	rsq	mae
LinearModel1	Testing Set	4138.492195	0.0803887	1919.310196
LinearModel2	Testing Set	3.243748	0.2326278	2.621742
LinearModel3	Testing Set	3.484797	0.1132622	2.874374
LinearModel4	Testing Set	3.218473	0.2448160	2.488462



The results for both the training and testing datasets on each of the models are presented below. The linear model that performed the best on this dataset to predict our response variable, `TARGET_AMT`, is linear model 4. This model uses all predictor variables with transformations done to normalize numeric variables that did not follow Gaussian distributions. Additionally, the predictor variable, `TARGET_FLAG`, was included in this analysis via prediction from the logistic regression model. Although this model may be the best of the four, it may not be sufficient in real-life use cases as the R^2 implies that the model explains about 24% of the variance within the dataset. The results were consistent between the testing and the training datasets implying that no obvious overfitting or underfitting occurred.

Results of the logistic regression models are presented below for both the training and the testing datasets.

Table 3: Training Dataset Model Results

model	set	accuracy	kap
LogisticModel1	Training Set	0.7907205	0.3775704
LogisticModel2	Training Set	0.7998693	0.4142744
LogisticModel3	Training Set	0.7506242	0.5012484

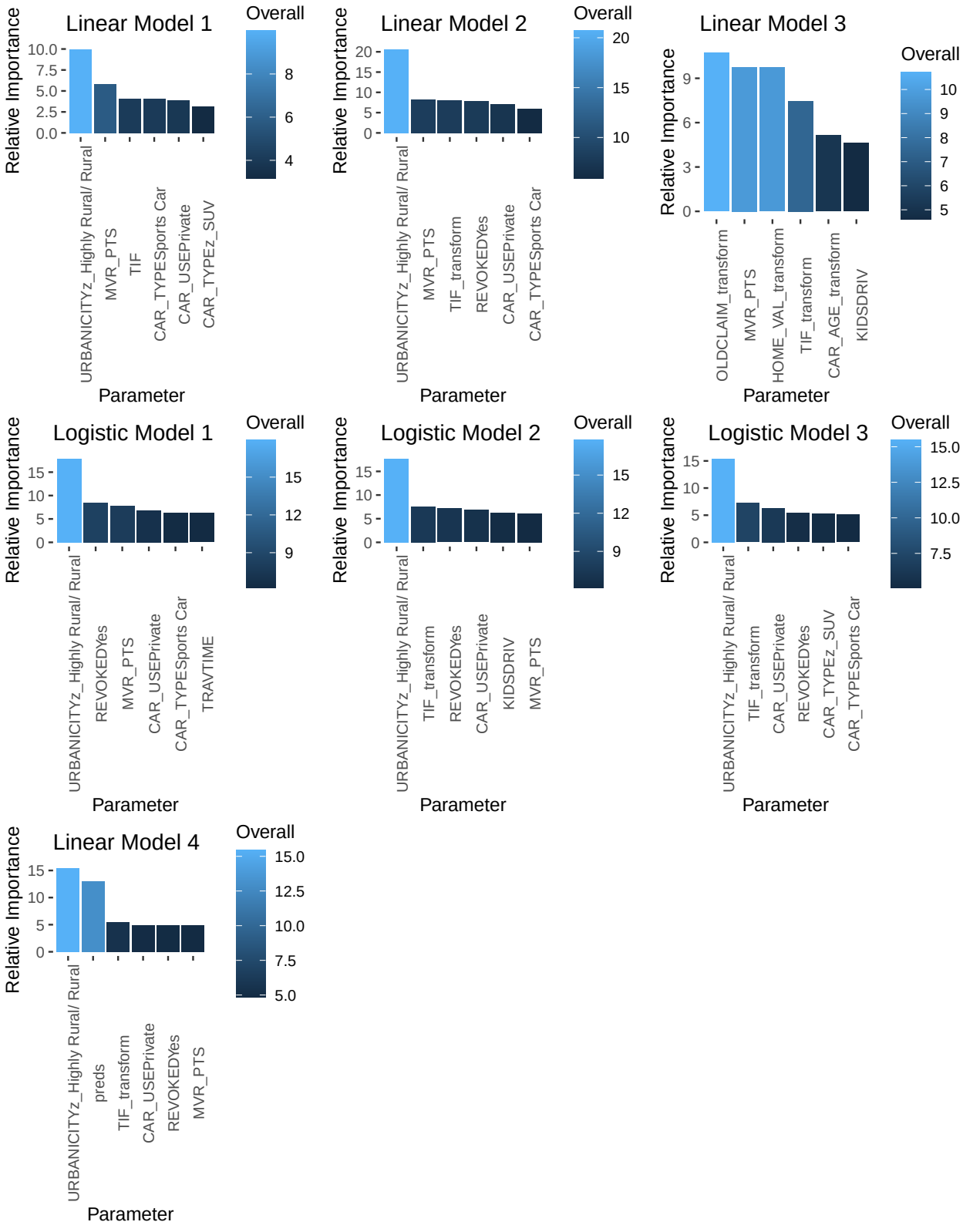
Table 4: Testing Dataset Model Results

model	set	accuracy	kap
LogisticModel1	Testing Set	0.7887255	0.3842650
LogisticModel2	Testing Set	0.7855741	0.3788016
LogisticModel3	Testing Set	0.7262022	0.4075439

The model that performed best using these classification models is **Logistic Model 3**. Although the accuracy of Logistic Models 1 and 2 are shown higher, the metric of importance here is the Kappa. The Kappa is a similar measure to accuracy however it is normalized by the accuracy that would be expected by chance alone which is useful with class imbalances presented in datasets. Due to the undersampling of the majority class for this model, the true accuracy (KAP), was best under Model 3. The results of these models were consistent between both the testing and the training datasets implying minimal overfitting and underfitting and the ability for the model to generalize well.

The final thing we will take a look at is our variable importance for each model.

The final figure presented below shows the feature importance in each model. Note that not all features are presented from the models, but only the top 6 from each. Amongst all the models with the exception of linear model 3 (which did not consider ordinal features), being in an **urban city** that was **highly rural** was best predictor of when someone was in a car crash and if they would file claims against their insurance accordingly. Motor vehicle points was also a consistent feature that shows that people who have a history of violations tend to be more likely to get into a car accident. This is consistent with how car insurance plans are priced with high premiums being charged in urban areas like NYC relative to the suburbs.



References

- A Modern Approach to Regression with R: Simon Sheather

- Linear Models with R: Julian Faraway.
- R package vignette, mixtools: An R Package for Analyzing Finite Mixture Models

R Code

```
# ============================================================================================
# Load Libraries and Define Helper functions
# ============================================================================================

include=FALSE, paged.print=FALSE}
library(MASS)
library(scales)
library(rpart.plot)
library(ggplot2)
library(ggfortify)
library(gridExtra)
library(forecast)
library(fpp2)
library(fma)
library(kableExtra)
library(e1071)
library(mlbench)
library(ggcorrplot)
library(DataExplorer)
library(timeDate)
library(caret)
library(GGally)
library(corrplot)
library(RColorBrewer)
library(tibble)
library(tidyr)
library(tidyverse)
library(tidyselect)
library(dplyr)
library(reshape2)
library(mixtools)
library(tidymodels)
library(ggpmisc)
library(regclass)
library(pROC)
library(naniar)
library(RANN)
#' Print a side-by-side Histogram and QQPlot of Residuals
#'
#' @param model A model
#' @examples
#' residPlot(myModel)
#' @return null
#' @export
residPlot <- function(model) {
  # Make sure a model was passed
  if (is.null(model)) {
    return
```

```
  }

  layout(matrix(c(1,1,2,3), 2, 2, byrow = TRUE))
  plot(residuals(model))
  hist(model[["residuals"]], freq = FALSE, breaks = "fd", main = "Residual Histogram",
       xlab = "Residuals",col="lightgreen")
  lines(density(model[["residuals"]], kernel = "ep"),col="blue", lwd=3)
  curve(dnorm(x,mean=mean(model[["residuals"]]), sd=sd(model[["residuals"]])), col="red", lwd=3, lty="d
  qqnorm(model[["residuals"]], main = "Residual Q-Q plot")
  qqline(model[["residuals"]],col="red", lwd=3, lty="dotted")
  par(mfrow = c(1, 1))
}
#' Print a Variable Importance Plot for the provided model
#'
#' @param model The model
#' @param chart_title The Title to show on the plot
#' @examples
#' variableImportancePlot(myLinearModel, 'My Title)
#' @return null
#' @export
variableImportancePlot <- function(model=NULL, chart_title='Variable Importance Plot') {
  # Make sure a model was passed
  if (is.null(model)) {
    return
  }

  # use caret and gglot to print a variable importance plot
  varImp(model) %>% as.data.frame() %>% top_n(n = 6) %>%
    ggplot(aes(x = reorder(rownames(.), desc(Overall)), y = Overall)) +
    geom_col(aes(fill = Overall)) +
    theme(panel.background = element_blank(),
          panel.grid = element_blank(),
          axis.text.x = element_text(angle = 90)) +
    scale_fill_gradient() +
    labs(title = chart_title,
         x = "Parameter",
         y = "Relative Importance")
}
#' Print a Facet Chart of histograms
#'
#' @param df Dataset
#' @param box Facet size (rows)
#' @examples
#' histbox(my_df, 3)
#' @return null
#' @export
histbox <- function(df, box) {
    par(mfrow = box)
    ndf <- dimnames(df)[[2]]

    for (i in seq_along(ndf)) {
            data <- na.omit(unlist(df[, i]))
            hist(data, breaks = "fd", main = paste("Histogram of", ndf[i]),
                 xlab = ndf[i], freq = FALSE)
```

```
            lines(density(data, kernel = "ep"), col = 'red')
    }

    par(mfrow = c(1, 1))
}
#' Extract key performance results from a model
#'
#' @param model A linear model of interest
#' @examples
#' model_performance_extraction(my_model)
#' @return data.frame
#' @export
model_performance_extraction <- function(model=NULL) {
  # Make sure a model was passed
  if (is.null(model)) {
    return
  }

  data.frame("RSE" = model$sigma,
             "Adj R2" = model$adj.r.squared,
             "F-Statistic" = model$fstatistic[1])
}
#' Initial cleaning of the dataset
#'
#' @dataset dataset being cleaned, specific to insurance datasets
#' @return data.frame
#' @export
initial_cleaning<- function(dataset){

  dataset<-dataset %>%
    dplyr::select(-INDEX) %>%
    #converting the currency based columns to numeric
    mutate_at(vars(c("INCOME","HOME_VAL","OLDCLAIM","BLUEBOOK")),~as.numeric(str_replace_all(.,c("\\$"="
    mutate_at(vars("TARGET_FLAG"),as.factor)

  return(dataset)
}

# =============================================================================
# Load Data set
# =============================================================================

# Load insurance dataset
df <- read.csv('https://raw.githubusercontent.com/djlofland/DS621_F2020_Group3/master/Homework_4/dataset
df_eval <- read.csv('https://raw.githubusercontent.com/djlofland/DS621_F2020_Group3/master/Homework_4/da

# Drop the INDEX column - this won't be useful and cleaning currency based character columns
df<-initial_cleaning(df)
df_eval <- initial_cleaning(df_eval)

# =============================================================================
# Summary Stats
# =============================================================================
```

```
**Dimensions of training dataset**
dim(df)

**Dimensions of evaluation dataset**
dim(df_eval)

# Display summary statistics
summary(df)

# ============================================================================================
# Check Class Bias
# ============================================================================================

prop.table(table(df$TARGET_FLAG)) %>% kable() %>%
    kable_styling(
      full_width = F) %>%
    add_header_above(header = c("Classification of Target Flag"=2))

# ============================================================================================
# Distributions
# ============================================================================================

DataExplorer::plot_bar(
  data = df,
         order_bar = T,
         ggtheme=theme_bw())

DataExplorer::plot_histogram(
  geom_histogram_args = list(alpha = 0.5),
   data = df,
         ggtheme=theme_bw())

# ============================================================================================
# Boxplots
# ============================================================================================

df_character_wide<-
df %>% select_if(function(col) is.numeric(col)==F | all(col==.$TARGET_AMT)) %>%
  pivot_longer(cols = -TARGET_AMT,names_to="variable",values_to="value")

df_character_wide %>%
  ggplot(mapping = aes(x = value, y = TARGET_AMT))+
  geom_boxplot()+facet_wrap(.~variable, scales="free")+
  theme_bw()+
  theme(axis.text.x = element_text(angle = 90))

df_character_wide %>%
  ggplot(mapping = aes(x = value, y = TARGET_AMT))+
  geom_boxplot()+facet_wrap(.~variable, scales="free")+
  theme_bw()+
  theme(axis.text.x = element_text(angle = 90))+
  coord_cartesian(ylim = c(NA,5000))

# ============================================================================================
```

```
# Variable Plots
# ================================================================================

DataExplorer::plot_scatterplot(
    data = dplyr::select_if(df,is.numeric),
    by = "TARGET_AMT",
         ggtheme=theme_bw(),
    theme_config = list(axis.text.x = element_text(angle = 90)))
DataExplorer::plot_correlation(data = df,type = "all",cor_args = list("use" = "pairwise.complete.obs"))

# ================================================================================
# Data Sparsity Check
# ================================================================================

DataExplorer::plot_missing(df,ggtheme = theme_bw())

# ================================================================================
# Missing Data
# ================================================================================

imputation <- preProcess(df, method = "knnImpute")
predict_imputer <- predict(imputation, df)
#compare to above chart
missing2 <- predict_imputer %>%
  miss_var_summary()
kable(missing2 %>%
  filter(pct_miss > 0)) %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed"))

table_names <- data.frame(col = names(imputation$mean), mean = imputation$mean, sd = imputation$std)
for(i in table_names$col){
  predict_imputer[i] <- predict_imputer[i]*imputation$std[i] + imputation$mean[i]
}

# ================================================================================
# Transform non-normal variables
# ================================================================================

# boxcox done on bluebook,
# performed boxcox transformation after identifying proper lambda
#
# Performed log on Target AMT, and old claim
transformative_cleaning<-function(DF, TEMP){
  return(
    #transforming dataset with boxcox
    DF %>% mutate_at(c("BLUEBOOK","CAR_AGE","HOME_VAL","INCOME","TIF","TRAVTIME"),
                     function(x) BoxCox(x, BoxCox.lambda(x))) %>%
      #transforming with log
      mutate_at(c("TARGET_AMT","OLDCLAIM"),function(x) if_else(x==0,0,log(x))) %>%
      #removing all infinite values generated from transformations
      mutate_if(is.numeric, list(~na_if(., Inf))) %>%
      mutate_if(is.numeric, list(~na_if(., -Inf))) %>%
      rename("BLUEBOOK_transform" = "BLUEBOOK", "CAR_AGE_transform" = "CAR_AGE", "HOME_VAL_transform" =
      #merge(TEMP, DF, by.x = 0, by.y = 0)
```

```
  )
}


df_temp <- predict_imputer %>% select("BLUEBOOK", "CAR_AGE", "HOME_VAL", "INCOME", "TIF", "TRAVTIME", "
# Build clean dataframe with transformation
clean_df <- transformative_cleaning(predict_imputer, df_temp)
clean_df <- clean_df[ , order(names(clean_df))]
#View new distributions of nominal values
DataExplorer::plot_histogram(
  geom_histogram_args = list(alpha = 0.5),
   data = clean_df,
         ggtheme=theme_bw())

# ============================================================================================
# Linear Regression Model 1
# ============================================================================================

#75% data test training split
set.seed(1)
df_split<-initial_split(clean_df,.75)
df_train<-training(df_split)
df_test<-testing(df_split)

#tidy models spec
lm_spec<-linear_reg() %>%
  set_engine(engine = "lm")

# Model 1 - Includes all nominal predictors and no transformations
m1train<-merge(df_temp, df_train, by.x = 0, by.y = 0)
m1test<-merge(df_temp, df_test, by.x = 0, by.y = 0)
model1 <- lm_spec %>%  fit(TARGET_AMT ~ AGE + BLUEBOOK + CAR_AGE + CLM_FREQ + HOME_VAL + HOMEKIDS + INC
(mod1_sum <- summary(model1$fit))

residPlot(mod1_sum)

linearM_results<- model1 %>%
  predict(new_data = m1train) %>%
  mutate(
    truth = m1train$TARGET_AMT,
    model = "LinearModel1",
    set = "Training Set"
  ) %>%
  bind_rows(model1 %>%
  predict(new_data = m1test) %>%
  mutate(
    truth = m1test$TARGET_AMT,
    model = "LinearModel1",
    set = "Testing Set"))


# ============================================================================================
# Linear Regression Model 2
# ============================================================================================
```

```
#recipe that removes all non numeric variables
model2recipe<-
  df_train %>% recipe(TARGET_AMT_transform~.) %>%
  #remove all non numeric based variables
  step_rm(TARGET_FLAG) %>%
  prep()

#apply recipe to both train and test sets
m2train<-bake(model2recipe, new_data =  df_train)
m2test<-bake(model2recipe,new_data= df_test)

#fit model with tidymodels syntax
model2<-lm_spec %>%
  fit(TARGET_AMT_transform ~ .,data = m2train)
(mod2_sum <- summary(model2$fit))
residPlot(mod2_sum)

#push results into table to be called later
linearM_results<- bind_rows(
  linearM_results,
  model2 %>%
  predict(new_data = m2train) %>%
  mutate(
    truth = m2train$TARGET_AMT_transform,
    model = "LinearModel2",
    set = "Training Set"
  ),model2 %>%
  predict(new_data = m2test) %>%
  mutate(
    truth = m2test$TARGET_AMT_transform,
    model = "LinearModel2",
    set = "Testing Set"))



# ============================================================================================
# Linear Regression Model 3
# ============================================================================================

#building model by removing 4 least significant numeric variables from predicting
model3recipe<-
  df_train %>% recipe(TARGET_AMT_transform~.) %>%
  #remove all non numeric based variables
  recipes::step_rm(recipes::all_nominal()) %>%
  recipes::step_rm(c(AGE,YOJ,CLM_FREQ,INCOME_transform)) %>%
  #removing any heavily correlated predictor variables
  recipes::step_corr(recipes::all_predictors()) %>%
  #removing any variables with low variance relative to response variable
  recipes::step_nzv(recipes::all_predictors()) %>%
  prep()

#apply recipe to both train and test sets
m3train<-bake(model3recipe, new_data =  df_train)
```

```
m3test<-bake(model3recipe,new_data= df_test)

model3<-lm_spec %>%
  parsnip::fit(TARGET_AMT_transform ~ .,data = m3train)
(mod3_sum <- summary(model3$fit))
residPlot(mod3_sum)

linearM_results<- bind_rows(
  linearM_results,
  model3 %>%
  predict(new_data = m3train) %>%
  mutate(
    truth = m3train$TARGET_AMT_transform,
    model = "LinearModel3",
    set = "Training Set"
  ),model3 %>%
  predict(new_data = m3test) %>%
  mutate(
    truth = m3test$TARGET_AMT_transform,
    model = "LinearModel3",
    set = "Testing Set"))


linVP1 <-variableImportancePlot(model1$fit, "Linear Model 1")
linVP2 <-variableImportancePlot(model2$fit, "Linear Model 2")
linVP3 <-variableImportancePlot(model3$fit, "Linear Model 3")

# ============================================================================================
# Logistic Regression Model 1
# ============================================================================================

logit_spec<-logistic_reg() %>%
  set_engine("glm") %>%
  set_mode("classification")

model1 <- logit_spec %>%  fit(TARGET_FLAG ~ BLUEBOOK + CAR_AGE + HOME_VAL + INCOME + PARENT1 + HOME_VAL
(mod1_sum <- summary(model1$fit))

logisticM_results<-
bind_rows(
predict(model1, new_data = m1train) %>%
  mutate(
    truth = m1train$TARGET_FLAG,
    model = "LogisticModel1",
    set = "Training Set"
  ),
predict(model1, new_data = m1test)  %>%
  mutate(
    truth = m1test$TARGET_FLAG,
    model = "LogisticModel1",
    set = "Testing Set"
  )
)
```

```
# ================================================================================
# Logistic Regression Model 2
# ================================================================================

model2recipe<-
  df_train %>% recipe(TARGET_FLAG~.) %>%
  #removing all target_AMT to prevent knowing answer to models
  step_rm(TARGET_AMT_transform) %>%
  #remove all non numeric based variables
  step_modeimpute(all_nominal()) %>%
  step_medianimpute(all_numeric()) %>%
  step_nzv(all_predictors()) %>%
  step_corr(all_numeric()) %>%
  #the following cut steps bin the data into groups of 4 or 5 for the specified variable in each step
  step_cut(AGE,breaks = seq(min(df_train$AGE,na.rm = T),max(df_train$AGE,na.rm = T),max(df_train$AGE,na
  step_cut(BLUEBOOK_transform,breaks = seq(min(df_train$BLUEBOOK_transform,na.rm = T),max(df_train$BLUE
  step_cut(CAR_AGE_transform,breaks = seq(min(df_train$CAR_AGE_transform,na.rm = T),max(df_train$CAR_AG
  step_cut(TRAVTIME_transform,breaks = seq(min(df_train$TRAVTIME_transform,na.rm = T),max(df_train$TRAV

#apply recipe to both train and test sets
m2train<-bake(model2recipe %>% prep(), new_data =  df_train)
m2test<-bake(model2recipe %>% prep(),new_data= df_test)

m2train %>%
  select(c(AGE,BLUEBOOK_transform,CAR_AGE_transform,TRAVTIME_transform)) %>%
  pivot_longer(cols=c(tidyselect::everything()),names_to = "Feature",values_to="Value") %>%
  ggplot(aes(x = Value)) +
  geom_histogram(stat = 'count')+
  facet_wrap(~Feature, scales = 'free')+
  theme(panel.background = element_blank(),
        panel.border = element_rect(color = 'black', fill = NA),
        axis.text.x = element_text(angle = 90))+
  labs(title = "Binned Features for Logistic Model")

model2 <- logit_spec %>%  fit(TARGET_FLAG ~ .,data = m2train)
(mod2_sum <- summary(model2$fit))

logisticM_results<-
bind_rows(logisticM_results,
predict(model2, new_data = m2train) %>%
  mutate(
    truth = m2train$TARGET_FLAG,
    model = "LogisticModel2",
    set = "Training Set"
  ),
predict(model2, new_data = m2test)  %>%
  mutate(
    truth = m2test$TARGET_FLAG,
    model = "LogisticModel2",
    set = "Testing Set"
  )
)

# ================================================================================
```

```
# Logistic Regression Model 3
# ================================================================================

prop.table(table(df$TARGET_FLAG))

model3recipe<-
 model2recipe %>%
  step_downsample(TARGET_FLAG)


#apply recipe to both train and test sets
m3train<-juice(model3recipe %>% prep())
prop.table(table(m3train$TARGET_FLAG))

m3test<-bake(model3recipe %>% prep(),new_data= df_test)

model3 <- logit_spec %>%  fit(TARGET_FLAG ~ .,data = m3train)
(mod3_sum <- summary(model3$fit))
model3$fit$fitted.values %>% View
logisticM_results<-
bind_rows(logisticM_results,
predict(model3, new_data = m3train) %>%
  mutate(
    truth = m3train$TARGET_FLAG,
    model = "LogisticModel3",
    set = "Training Set"
  ),
predict(model3, new_data = m3test)  %>%
  mutate(
    truth = m3test$TARGET_FLAG,
    model = "LogisticModel3",
    set = "Testing Set"
  )
)


logVP1 <-variableImportancePlot(model1$fit, "Logistic Model 1")
logVP2 <-variableImportancePlot(model2$fit, "Logistic Model 2")
logVP3 <-variableImportancePlot(model3$fit, "Logistic Model 3")


# ================================================================================
Using Logistic Results to improve Linear Regression
# ================================================================================


#building model by removing 4 least significant numeric variables from predicting

df_train['preds'] <- logisticM_results %>% filter(model == 'LogisticModel2') %>% filter(set == 'Trainin
df_test['preds'] <- logisticM_results %>% filter(model == 'LogisticModel2') %>% filter(set == 'Testing

df_train['preds'] <- df_train['preds'] %>% sapply(as.numeric)
df_test['preds'] <- df_test['preds'] %>% sapply(as.numeric)
```

```
model4recipe<-
  df_train %>% recipe(TARGET_AMT_transform~.) %>%
  #removing any heavily correlated predictor variables
  step_rm(TARGET_FLAG) %>%
  recipes::step_corr(recipes::all_numeric()) %>%
  #removing any variables with low variance relative to response variable
  recipes::step_nzv(recipes::all_numeric()) %>%
  prep()

#apply recipe to both train and test sets
m4train<-bake(model4recipe, new_data =  df_train)
m4test<-bake(model4recipe,new_data= df_test)

model4<-lm_spec %>%
  parsnip::fit(TARGET_AMT_transform ~ .,data = m4train)
(mod4_sum <- summary(model4$fit))
residPlot(mod4_sum)

linearM_results<- bind_rows(
  linearM_results,
  model4 %>%
  predict(new_data = m4train) %>%
  mutate(
    truth = m4train$TARGET_AMT_transform,
    model = "LinearModel4",
    set = "Training Set"
  ),model4 %>%
  predict(new_data = m4test) %>%
  mutate(
    truth = m4test$TARGET_AMT_transform,
    model = "LinearModel4",
    set = "Testing Set"))


linVP4 <-variableImportancePlot(model4$fit, "Linear Model 4")

# ================================================================================
Model Performance Summary
# ================================================================================

linearM_results %>%
  ggplot(aes(x = truth,y = .pred))+
  geom_point(pch =21, alpha = 0.3, fill = "skyblue") +
  geom_smooth(method = "lm", color = "red3") +
  facet_wrap(set~model, scales = 'free')+
stat_poly_eq(aes(label = paste(..adj.rr.label.., sep = "~~~")),
                   label.x.npc = "left", label.y.npc = .9,
                   formula = y~x, parse = TRUE, size = 2)+
  theme(panel.background = element_blank(),
        panel.grid = element_blank())


#getting linear results for training dataset
```

```
linearM_results %>% filter(set == "Training Set") %>%
  group_by(model,set) %>%
metrics(truth = truth, estimate = .pred) %>%
  select(-c('.estimator')) %>%
  pivot_wider(names_from = '.metric',values_from = '.estimate') %>%
#pushing to kable style table
  kable() %>%
  kable_styling(
    bootstrap_options = c("hover", "condensed", "responsive"),
    full_width = F)

#getting linear results for testing dataset
linearM_results %>% filter(set == "Testing Set") %>%
  group_by(model,set) %>%
metrics(truth = truth, estimate = .pred) %>%
  select(-c('.estimator')) %>%
  pivot_wider(names_from = '.metric',values_from = '.estimate') %>%
  #pushing to kable style table
  kable() %>%
  kable_styling(
    bootstrap_options = c("hover", "condensed", "responsive"),
    full_width = F)

logisticM_results %>% filter(set == "Training Set") %>%
  group_by(model,set) %>%
metrics(truth = truth, estimate = .pred_class) %>%
  select(-c('.estimator')) %>%
  pivot_wider(names_from = '.metric',values_from = '.estimate')%>%
  #pushing to kable style table
  kable() %>%
  kable_styling(
    bootstrap_options = c("hover", "condensed", "responsive"),
    full_width = F)

logisticM_results %>% filter(set == "Testing Set") %>%
  group_by(model,set) %>%
metrics(truth = truth, estimate = .pred_class) %>%
  select(-c('.estimator')) %>%
  pivot_wider(names_from = '.metric',values_from = '.estimate')%>%
  #pushing to kable style table
  kable() %>%
  kable_styling(
    bootstrap_options = c("hover", "condensed", "responsive"),
    full_width = F)

grid.arrange(linVP1,linVP2,linVP3,logVP1,logVP2,logVP3,linVP4,ncol = 3)

# ================================================================================
End of R Code
# ================================================================================
```