DATA 621 - Auto Insurance

Erik Nylander

April 9, 2016

1 Data Exploration

In this analysis we will be investigating an auto insurance data set that contains 8161 individuals with auto insurance. The data contains two response variables. The first (TARGET_FLAG) is a target flag for individuals whose vheicals were involved in an accident. A "1"" indicates that the person was involved in a car crash while a "0" indicates that the person was not involved in an accident. The second variable (TARGET_AMT) is the cost of repair for the vehicle if the individual was involved in an accident, the value will be zero if the person was not involved in an accident. The data set also contains an evaluation set with 2141 individuals where the TARGET_FLAG and TARGET_AMT variables have been removed. To facilitate the model selection we will be randomly dividing the insurance data into a training set that contains 70% of the data and a test set that contains 30% of the data.

1.1 Explanation of the Variables

- Target Variables
  - TARGET_FLAG: Variable indicating if the car was in an accident
  - TARGET_AMT: The cost of repairs if the car was involved in an accident
- Predictor Variables
  - KIDSDRIV: Number of children who drive in the home
  - AGE: Age of the driver
  - HOMEKIDS: Number of children in the household
  - YOJ: Number of years on the job
  - INCOME: Income
  - PARENT1: Single parent
  - HOME_VAL: Home value, 0 if no home owned
  - MSTATUS: Marital status
  - SEX: Gender
  - EDUCATION: Max education attained by the individual
  - JOB: Job Category
  - TRAVTIME: Distance traveled to work
  - CAR_USE: Vehicle use, private or commercial
  - BLUEBOOK: Value of the vehicle
  - TIF: Time insured by the company
  - CAR_TYPE: Make of the car
  - RED_CAR: Is the car red?
  - OLDCLAIM: Total claims over the last 5 years
  - CLM_FREQ: Number of claims in the last 5 years
  - REVOKED: Has the drivers licence been revoked in the last 7 years?
  - MVR_PTS: Motor vehicle record points
  - CAR_AGE: Age of the vehicle
  - URBANICITY: Is the home work area urban or rural
- User Created Variables
  - HOME_OWN: Binary variable indicating if a home is owned

- JOB_TYPE: Binary indicator if the job is white or blue collar
- HIGHER_ED: Has the individual completed college?
- KD: Are there driving Children in the household?
- KIDS: Does the insured party have kids?

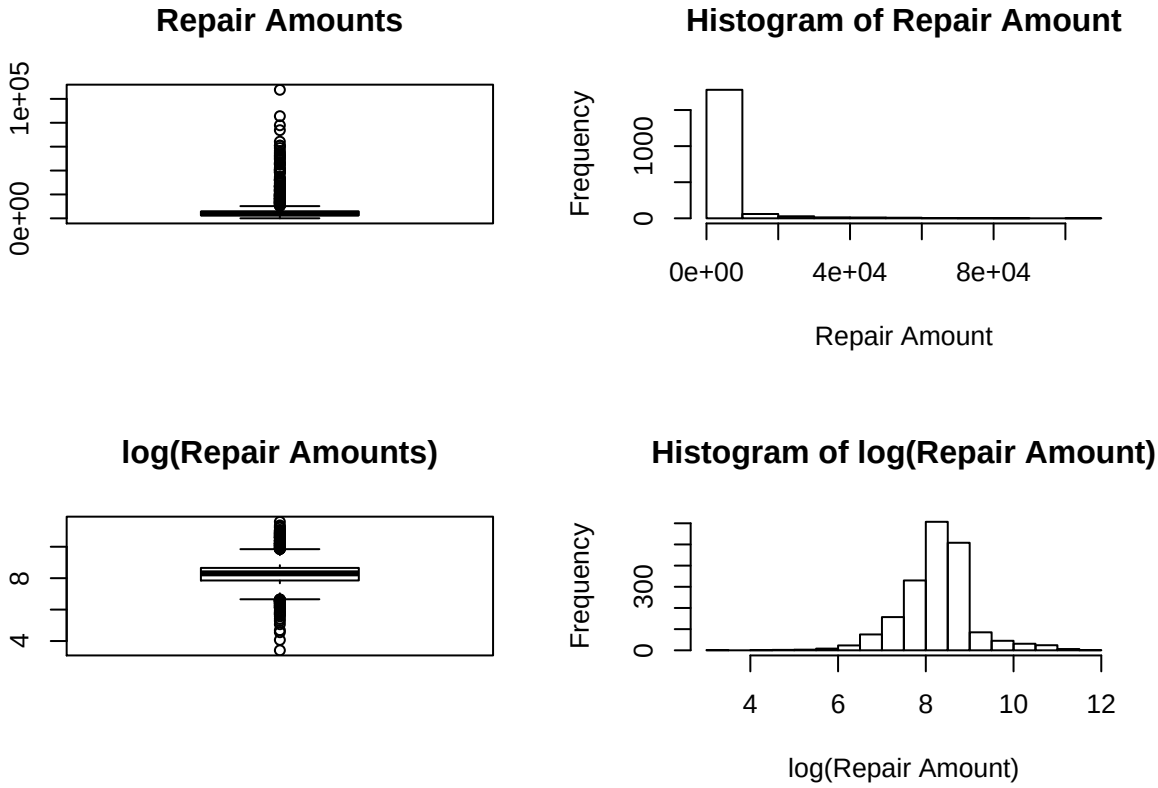
1.2 Exploring the Target Variables

We will be constructing two different models for the data set and have two different target variables that we would like to explore. The first variable is the TARGET_FLAG variable. For this data set the variable takes on a value of 1 if the vehicle has been involved in an accident and takes on a value of 0 if there has not been an accident. Table 1 contains the results. We see that about 26.38% of the vehicles in the data set have been involved in accidents.

Table 1: Prevalence of the target Variable in the Data

	No Accidents	Accident
Frequency	6008	2153
Proportion	0.7362	0.2638

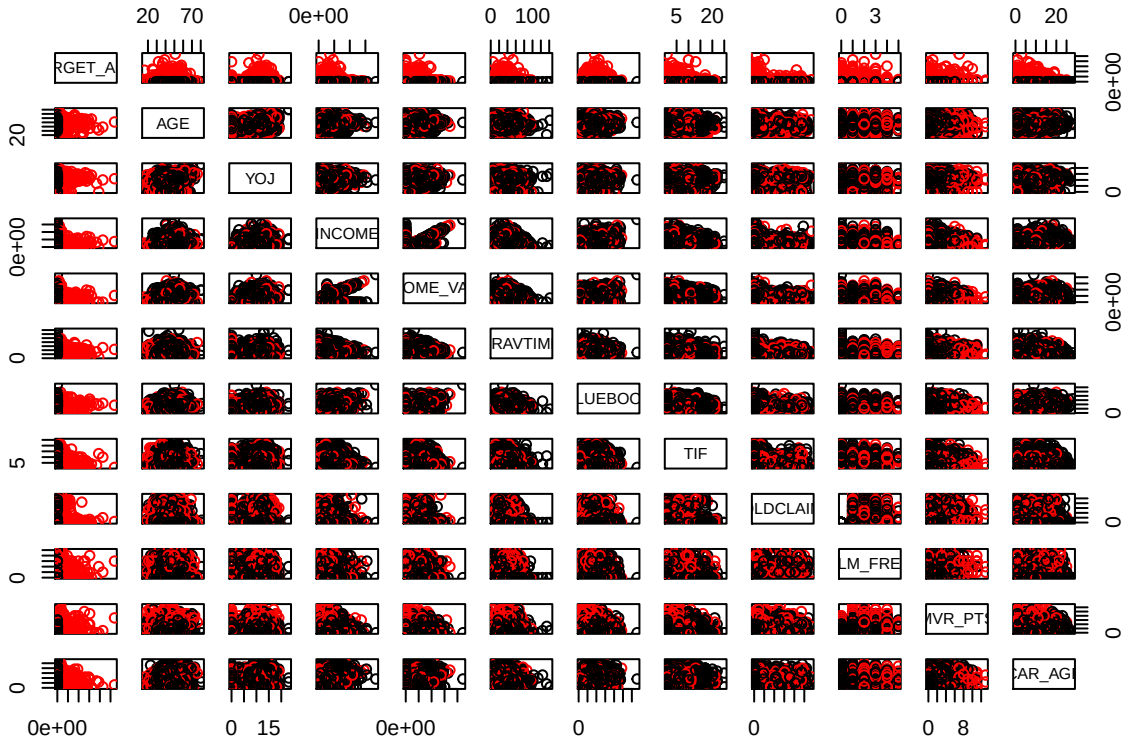
For our linear regression model we will be exploring the TARGET_AMT variable. This variable contains the cost of repair for a vehical if it has been involved in an accident. Since this variable only has a value if the vechical has been in an accident we will subset the data to only include the observations that come from vehicals that have been involved in an accident. The figure below shows the box plot and histogram for the TARGET_AMT variable. We notice that the data shows a strong right skew. We computed the the Box-Cox Transformation on the TARGET_AMT variable and end up deciding to perform a log transformation on the TARGET_AMT. While the data has a much better shape under the transformation, we do still note that there are a number of outliers. To deal with these outliers we will construct our models using a robust regression on the data. (Note that these plots were constructed after the data cleaning and removing of missing values and represent the observations that will be used in the model construction.)



1.3 Exploring the Predictor Variables

We will next look at the distribution of the predictor variables. The first issue that we discover with the predictor variables is fact that there a number of missing values in the data set. For our numerical variables we will elect to replace the missing values using the median value of the predictor variable. For the categorical variables we will simply drop these observations from the data set. After removing the missing values we will next construct a set of categorical variables that will replace some some of our predictors that contain lots of zero values. While it is not included in the final analysis we have attempted to fit a number of models without success before constructing these categorical variables. We have elected to divide up our categorical and numerical variables and have included the boxplots and histograms for each of the numerical predictor variables in Appendix A. We have also included probability tables for each of the categorical tables in Appendix B.

We do note a few things from looking at these plots. The first is that many of the predictor variables are highly skewed. While we have transformed some of the variables to categorical which will be discussed below, we also notice that a number of the different predictor variables also have outliers that will effect the model building. We will likely need to transform the variables to have a chance of constructing a valid model. We will also attempt a robust regression to try and achieve a valid model. We will also look at the pairs plot of the numeric predictor values to check for any patterns that we can find in the data. The pairs plot shows us that there are very few patterns in the data and that there seems to be little correlation between the numeric predictor variables and the response variable TARGET_AMT and we will have to attempt some transformations of the variables to achieve a correlation.



Finally we have our categorical variables. We have elected to construct a set of tables containing the probabilities of having an accident giving each of the different categories. We do notice a few things from the data. The first is that there do appear to be some differences between the each of the categorical variables and the probability of getting in an accident. This implies that we will likely be able to use these determine the likelihood of a vehicle getting in an accident. We also note an interesting results in Table 2. Even though the urban legend would indicate that women are better drivers then men, we do not see this result in our data.

Table 2: Accident Rates for Men vs. Women

Gender	no Accident	Accident
Male	0.749	0.251
Female	0.725	0.275

2 Data Preperation

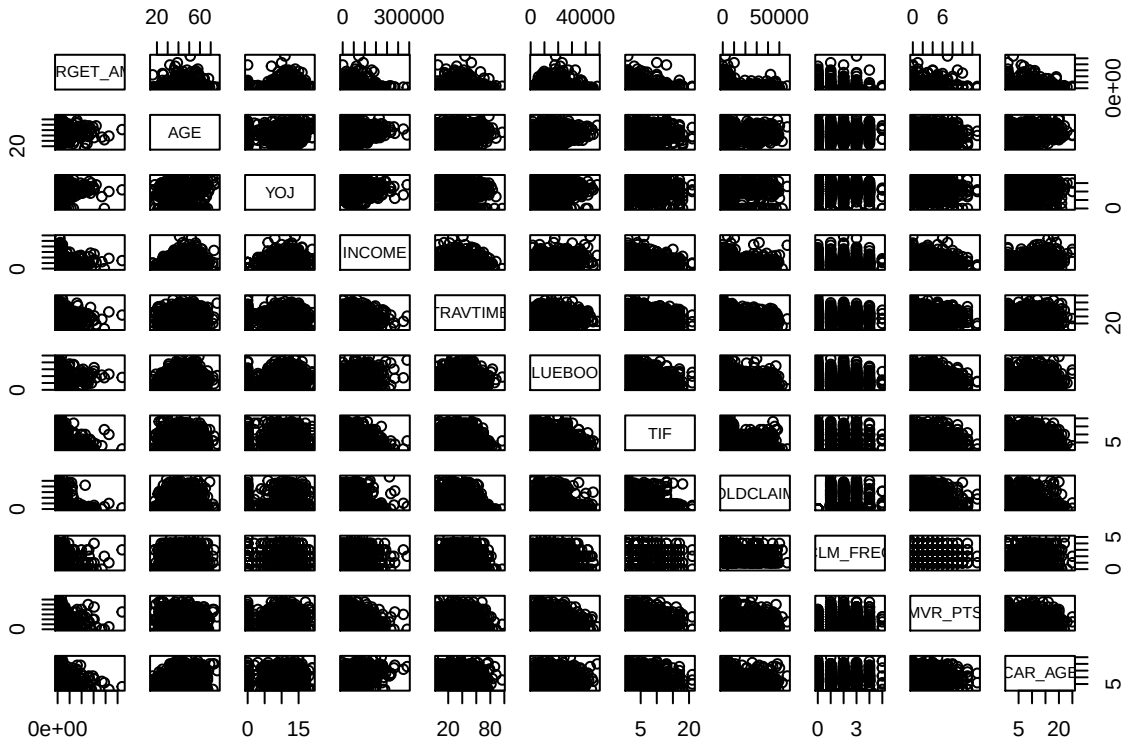
For this data set there are a number of different things that we will need to do to prepare the data for analysis. The first issue that we needed to deal with was the monetary values in the data. These were transformed into numerical values to facilitate analysis. The second issue that we ran across were the missing values in the data set. The AGE, YOJ, INCOME, HOME_VAL, and CAR_AGE predictor variables were all missing between 400 and 500 values each. We have elected to fill these values with the median values for the data.

2.1 Creating Categorical Variables

The next issue with the data that we will be addressing is transformation of some of our variables into categorical variables. We have elected to create 5 categorical variables from the given data. The HOME_VAL variable has been converted into a HOME_OWN which indicates home ownership. The JOB variable has been converted to a JOB_TYPE variable that splits the job types into blue collar as identified in the data and white collar for the other jobs this data still contains 526 NA's which we will end up dropping from the data. The next categorical data that we have created is HIGHER_ED which divides the data into two groups based on the completion of an undergraduate degree or higher. The last two variables that we have created are the KD and KIDS variables. The first indicates if there is a student driver in the household and the second informs us of the presence of kids in the household.

2.2 Transforming the Numerical Variables

The final data preparation step that we will be taking is to transform the TARGET_AMT and predictor variables to attempt to normalize the variables and hopefully reduce the outliers. We have used the Box-Cox method determining the type of transformations and in Appendix A we can see the results of these transformations. We can also see in the figure below the pairs plot that is created with these transformed variables. There still does not appear to be a pattern to these data and as we will see in the construction of the models this leads to difficulty in finding a potentially valid model.



2.3 Final Note about the Data

Even with the transformations that we have created there are still a number of variables that contain outliers that were not cleared up with our transformations. While we considered the option of dropping these outliers

they did not to contain data that would be considered extraordinary given our understanding of the problem. Therefore we have elected to leave the outliers in our analysis and will be preforming a robust regression in our model construction to help reduce the effect of these outliers.

Our final choice with our data was based on needing to subset the data for our final analysis. For the construction of the logistic regression models we have elected to leave in all of the reaming data which comes to 7208 observations that will be randomly divided into a training set (70%) and test set (30%). For the construction of the linear models predict the cost of repair of a vehicle we have elected to look only at the observations for vehicles that have actually been involved in an accident. This reduces our data set to 1907 observations which will also be broken into a training set (70%) and test set (30%).

3 Building our Models

For this analysis we will be constructing two linear models to predict the cost of repairs of a vehicle that has been in an accident and three logistic models that predict the probability of a given individual being involved in an accident. We will begin by building the models for predicting the cost of repair and then we will move on to predicting the probability of an individual being in an accident. All of the below models are tested using the test data set.

3.1 Predicting the Cost of Repairs using Ridge Regression

For the first model we will use ridge regression on the transformed variable set to find the best subset of our predictor variables to used for the model. We will be judging the models using the mean prediction error and the standard error. We start by calculating our best λ and $\log(\lambda)$ values. We can see in the figure below and our analysis that the ideal $\lambda = 0.666$ and that the ideal $\log(\lambda) = -0.406$. This subset of the variables results in a mean prediction error of 0.6132 and a standard error of 0.0587.

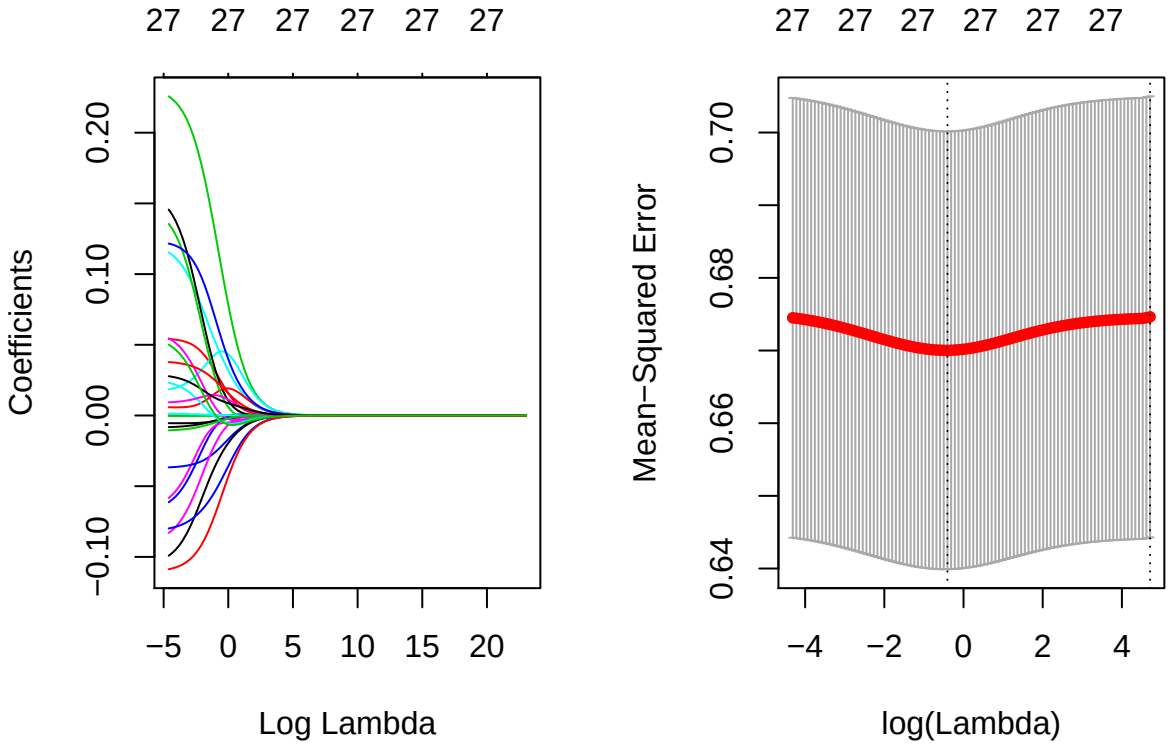
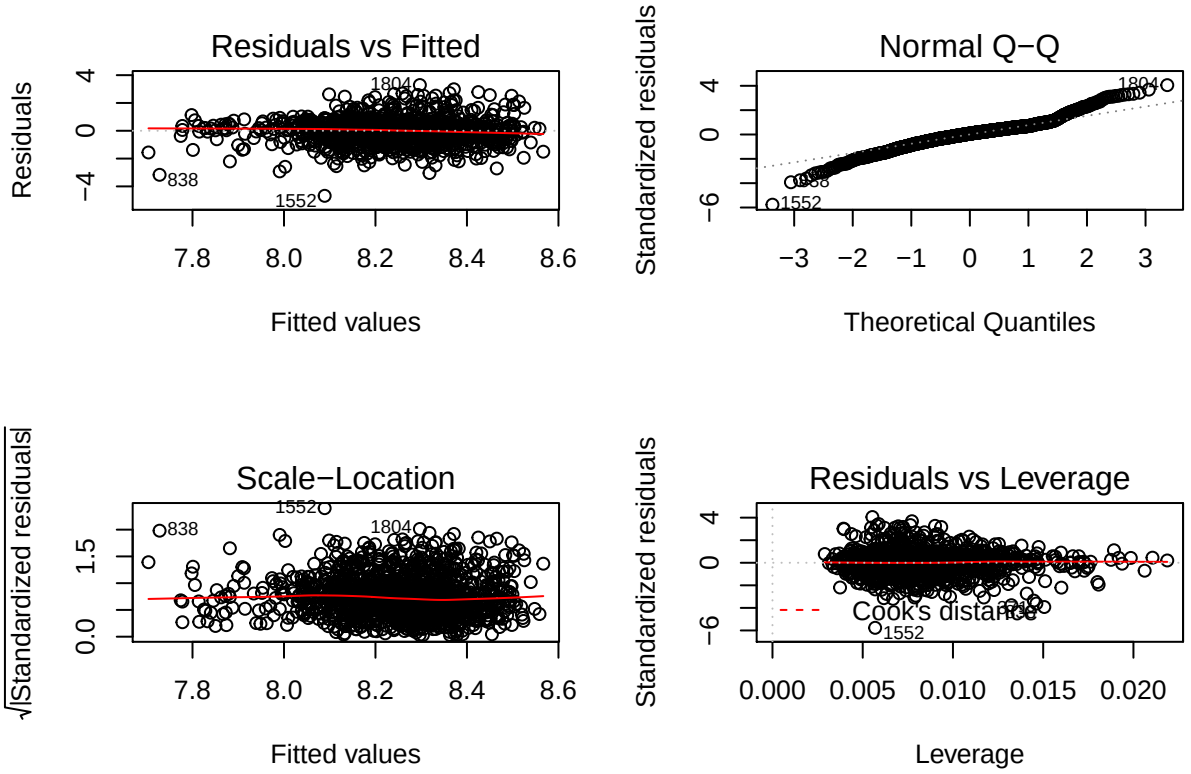


Table 3 contains the best subset of the variables as selected by the ridge regression. This subset of variables creates a model with an R-squared of 0.026. While this value is extremely small it is the best that we were able to develop. Given the series of transformations that we have done the values of the coefficients can not be interpreted but this does give us our best bet for predictive power. The figure below also contains the diagnostic plots of the residuals. We do note that the residuals are well behaved and there appear to be no issues with heteroskedasticity. We do however see that there are number of high leverage points that we will attempt to fix with our next regression.

Table 3: Ridge Regression Model (* = significant at $\alpha = 0.05$)

Variable	Coefficient
Intercept	6.61*
AGE	-0.021
YOJ	0.00006
INCOME	-0.0004
PARENT1Yes	-0.068
MSTATUSNo	-1.061
SEXF	-0.044
TRAVTIME	-0.006
CAR_USEPrivate	0.056
BLUEBOOK	0.201*
TIF	-0.0395

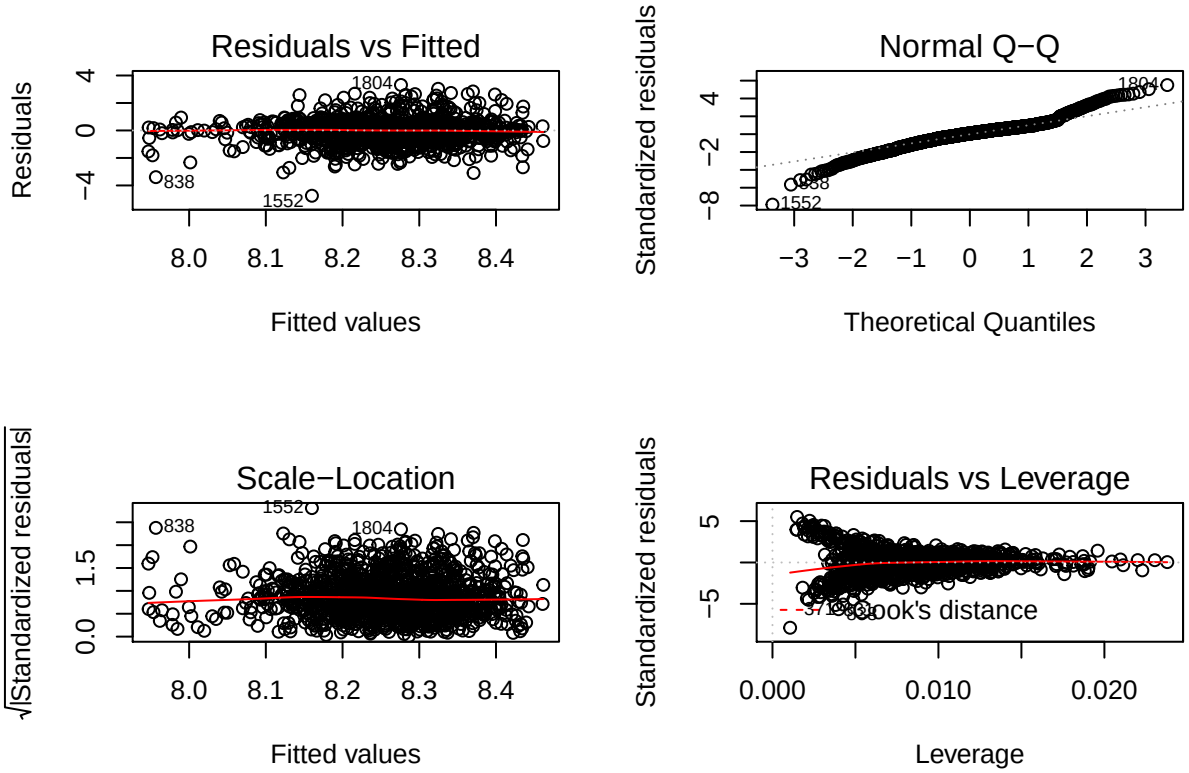


3.2 Predicting the Cost of Repairs using Robust Regression

For our second linear model we will use robust regression and after some trial and error we have elected to use the same best subset as indicated by the ridge regression. Our model generates a mean prediction error of 0.619 and a standard error of 0.059. Table 4 contains the values for the coefficients predicted by this model. We do see that this model generates no significant predictors but it does indicate similar values for the coefficient as the model above. We have also included the diagnostic plots for the residuals below. We see that the residuals are well behaved with no real issues.

Table 4: Ridge Regression Model (* = significant at $\alpha = 0.05$)

Variable	Coefficient
Intercept	7.4373
AGE	-0.0332
YOJ	0.0003
INCOME	-0.0003
PARENT1Yes	-0.0779
MSTATUSNo	0.0746
SEXF	-0.0265
TRAVTIME	-0.0057
CAR_USEPrivate	0.0224
BLUEBOOK	0.1191
TIF	-0.0253



3.3 Predicting the Likliehood of an Accident Using the Full data

For our first logistic model we will be fitting a logistic model to the full data set without transformations. While this model may not end up being the best one we were curious to see if we could generate a descent model out of our messy data. The coefficients for the full model are in Table 5. The fitting of the full model and running it against the test data results in a accuracy of 0.7826 and an area under the curve of 0.8086. Table 6 contains the confusion matrix of the predicted values in the test data set.

Table 5: Full Logistic Model with Original Data (* = significant at $\alpha = 0.05$)

Variable	Coefficient	Factor of Odds
Intercept	-0.149	0.861
AGE	-0.004	0.996
YOJ	-0.004	0.995
INCOME	-0.000007*	0.999
PARENT1Yes	0.265	1.303
MSTATUSNo	0.501*	1.651
SEXF	-0.22	0.8
TRAVTIME	0.013*	1.013
CAR_USEPrivate	-0.801*	0.449
BLUEBOOK	-0.00002*	0.999
TIF	-0.049*	0.952
CAR_TYPEPanel	0.65*	1.916
CAR_TYPEpickup	0.565*	1.758
CAR_TYPESports	1.046*	2.845
CAR_TYPEVan	0.492*	1.635
CAR_TYPESUV	0.879*	2.407
RED_CARYes	-0.178	0.837
OLDCLAIM	-0.00005*	0.999
CLM_FREQ	0.178*	1.195
REVOKEDYes	0.855*	2.352
MVR_PTS	0.122*	1.129
CAR_AGE	-0.004	0.997
URBANICITYRural	-2.372*	0.093
HOME_OWNYes	-0.315*	0.73
JOB_TYPEWC	-0.176	0.839
HIGHER_EDYes	-0.436*	0.646
KDYes	0.415*	1.514
KIDSYes	0.261*	1.298

From Table 5 we note some interesting results. It appears that we having a sports car, an SUV, or a revoked licence have the largest effect on increasing the odds of being in an accident. At the same time living in a rural area, owning your own home, and having a higher degree decreases ones odds of being in an accident.

From the confusion matrix below we do see that this model seems to over-predict the odds of an individual being in an accident when they are actually a lower risk of accident. This could be problematic in choosing this model.

Table 6: Confusion Matrix of Predicted Values (Full Data)

Class	Pred. Low	Pred. High
True Low	1445	362

Class	Pred. Low	Pred. High
True High	108	247

3.4 Predicting the Likliehood of an Accident using the Transformed Data

Four our second logistic model we will be using the full transformed data set to build a logistic model to calculate the odds of being in an accident. The coefficients for the full model are in Table 7. The fitting of the full model and running it against the test data results in a accuracy of 0.7798 and an area under the curve of 0.8106. Table 8 contains the confusion matrix of the predicted values in the test data set.

Table 7: Full Logistic Model with Transformed Data (* = significant at $\alpha = 0.05$)

Variable	Coefficient	Factor of Odds
Intercept	2.265*	9.634
AGE	-0.104	0.9
YOJ	0.0008	1
INCOME	-0.003*	0.999
PARENT1Yes	0.284	1.328
MSTATUSNo	0.537*	1.711
SEXF	-0.279*	0.757
TRAVTIME	0.155*	1.167
CAR_USEPrivate	-0.79*	0.454
BLUEBOOK	-0.267*	0.766
TIF	-0.213*	0.808
CAR_TYPEPanel	0.535*	1.708
CAR_TYPEpickup	0.586*	1.797
CAR_TYPESports	1.079*	2.941
CAR_TYPEVan	0.494*	1.64
CAR_TYPESUV	0.933*	2.543
RED_CARYes	-0.188	0.829
OLDCLAIM	-0.003*	0.997
CLM_FREQ	0.443*	1.558
REVOKEDYes	0.829*	2.292
MVR_PTS	0.243*	1.128
CAR_AGE	-0.024	0.976
URBANICITYRural	-2.374*	0.093
HOME_OWNYes	-0.284*	0.753
JOB_TYPEWC	-0.234*	0.791
HIGHER_EDYes	-0.42*	0.657
KDYes	0.43*	1.538
KIDSYes	0.196	1.216

The results of this model are very similar to what we found under the model without transformations however the transformations have made the interpretation of the variables difficult. Therefore we will look at the confusion matrix to see if we have done a better job of predicting the data, the results are in Table 8 below. We do note that this model seems to preform slightly worse then the previous model.

Table 8: Confusion Matrix of Predicted Values (Transformed Data)

Class	Pred. Low	Pred. High
True Low	1443	366
True High	110	243

3.5 Predicting Using a Reduced Model

For our last Model we will use the transformed data and a reduced predictor set to fit the model. The coefficients are found in Table 9 below. The model has an accuracy of 0.7798 and an area under the curve of 0.8105.

Table 9: Full Logistic Model with Transformed Data (* = significant at $\alpha = 0.05$)

Variable	Coefficient	Factor of Odds
Intercept	2.265*	9.634
AGE	-0.128*	0.9
INCOME	-0.003*	0.999
PARENT1Yes	0.439*	1.328
MSTATUSNo	0.446*	1.711
TRAVTIME	0.153*	1.167
CAR_USEPrivate	-0.781	0.454
BLUEBOOK	-0.299*	0.766
TIF	-0.209*	0.808
CAR_TYPEPanel	0.596*	1.708
CAR_TYPEpickup	0.587*	1.797
CAR_TYPESports	0.959*	2.941
CAR_TYPEVan	0.541*	1.64
CAR_TYPESUV	0.83*	2.543
OLDCLAIM	-0.003*	0.997
CLM_FREQ	0.442*	1.558
REVOKEDYes	0.836*	2.292
MVR_PTS	0.243*	1.128
URBANICITYRural	-2.372*	0.093
HOME_OWNYes	-0.28*	0.753
JOB_TYPEWC	-0.246*	0.791
HIGHER_EDYes	-0.473*	0.657
KDYes	0.522*	1.538

We do notice that the coefficients are very similar to what we see in the previous model and we get very similar results. Finally in Table 10 we can see the confusion matrix for our last model. We notice that this model performs in a very similar way to the other models.

Table 10: Confusion Matrix of Predicted Values (Reduced Transformed Data)

Class	Pred. Low	Pred. High
True Low	1443	366
True High	110	243

4 Selecting the Models

To select our models we will look at our various measures of performance for the models.

4.1 Predicting Repair Costs

We will start with the linear models. We have found that both sets of linear models have reasonable residuals even though both of them fail to have a high predictive power. We will therefore use the mean prediction error and standard error. Table 11 contains the comparison for the two models.

Table 11: Comparison of the Linear Models

	Ridge Model	Robust Model
MPE	0.6132	0.619
SE	0.0587	0.59

From this we see that even though it has a very low R-squared value we will elect to go with the model chosen by ridge regression in section 3.1 we will use this model to predict the cost of repairs. We do note that we have very little confidence in this model and ultimately these predictor variables seem to tell us very little about the actual cost of repairs. Table 12 below contains the first 6 predicted repair costs using our chosen model.

Table 12: First 6 Predicted Repair Costs

Observation	1	2	3	4	5	6
Repair Cost	5046.49	4339.61	303.36	3881.92	4475	4531.91

4.2 Predicting the Likelihood of an Accident

Finally we will pick from our three logistic models to determine which one we will use to predict the probability of a person being in an accident. Table 13 below shows the various metric for each of our models.

Table 13: Comparison of our Logistic Models

Metric	Full Data	Transformed Data	Reduced Model
Accuracy	0.7826	0.7798	0.7798
CER	0.2174	0.2202	0.2202
Precision	0.6958	0.6884	0.6884
Sensitivity	0.4056	0.399	0.399
Specificity	0.9292	0.9292	0.9292
F1 Score	0.5125	0.5051	0.5051
AUC	0.8086	0.8106	0.8106
Log Likelihood	-2216.408	-2212.882	-2217.936

We see that the three models are virtually indistinguishable and therefore we feel that we are free to choose a model. Since we will need to transform the data to perform the linear regression we will elect to choose the reduced model given that it is more parsimonious and will help us to avoid over-fitting. Table 14 gives the results of running this model on our evaluation data. We see that this model predicts accident probabilities that are slightly less then what we would have expected from the training data.

Table 14: Prediction of the Target Variable

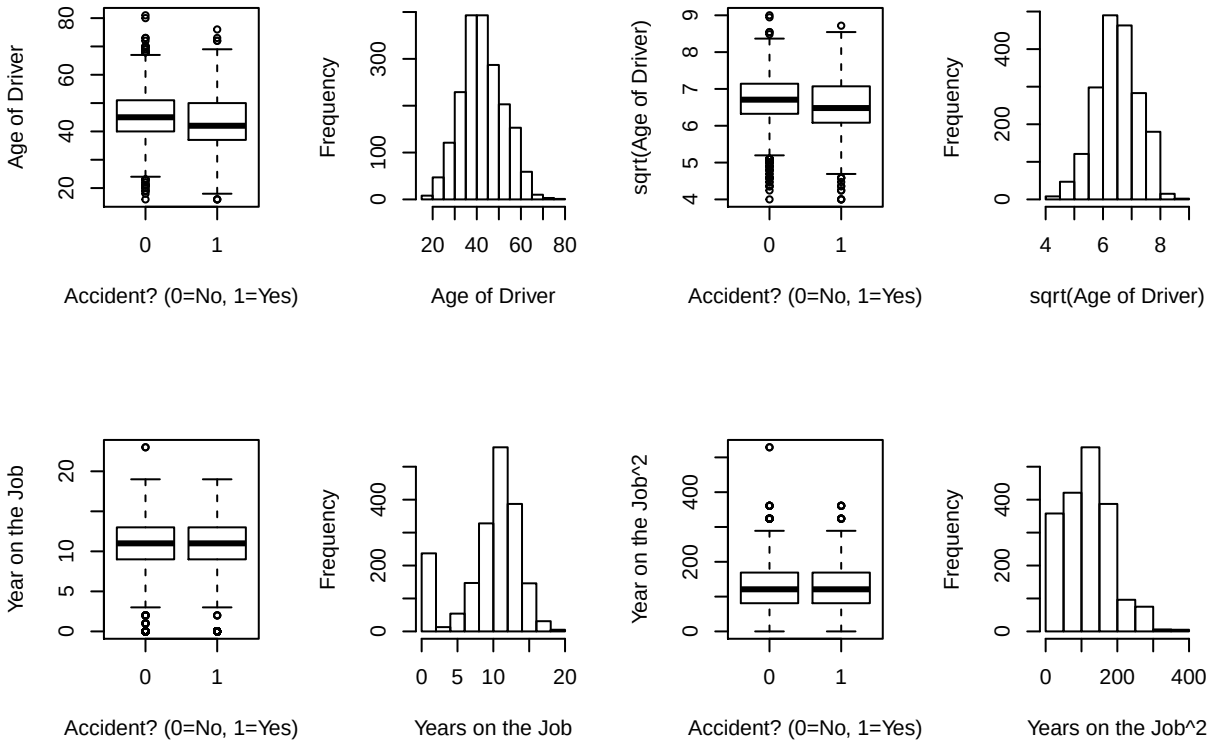
	No Accident	Accident
Frequency	1586	316
Proportion	0.834	0.166

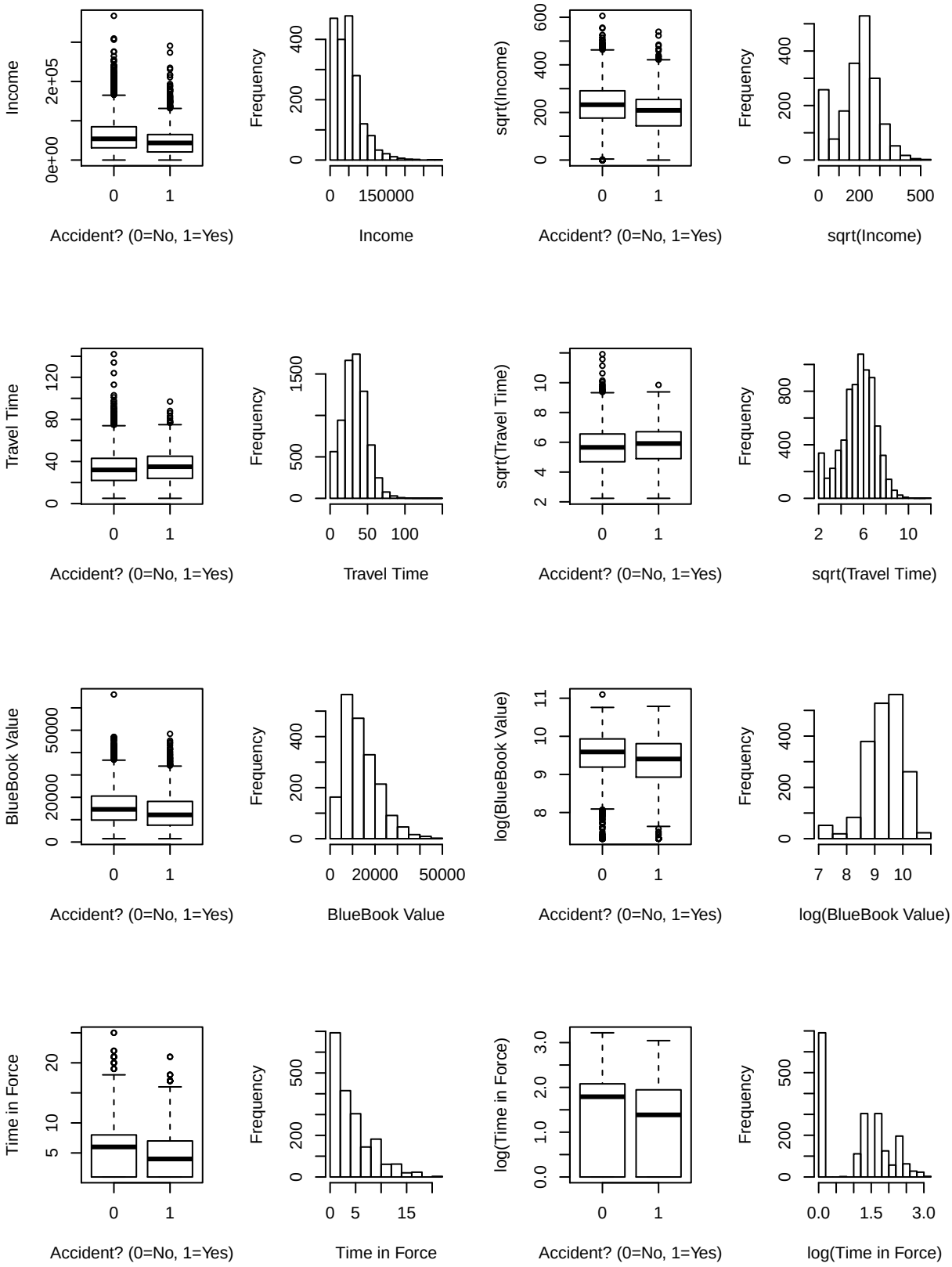
5 Concluding Results

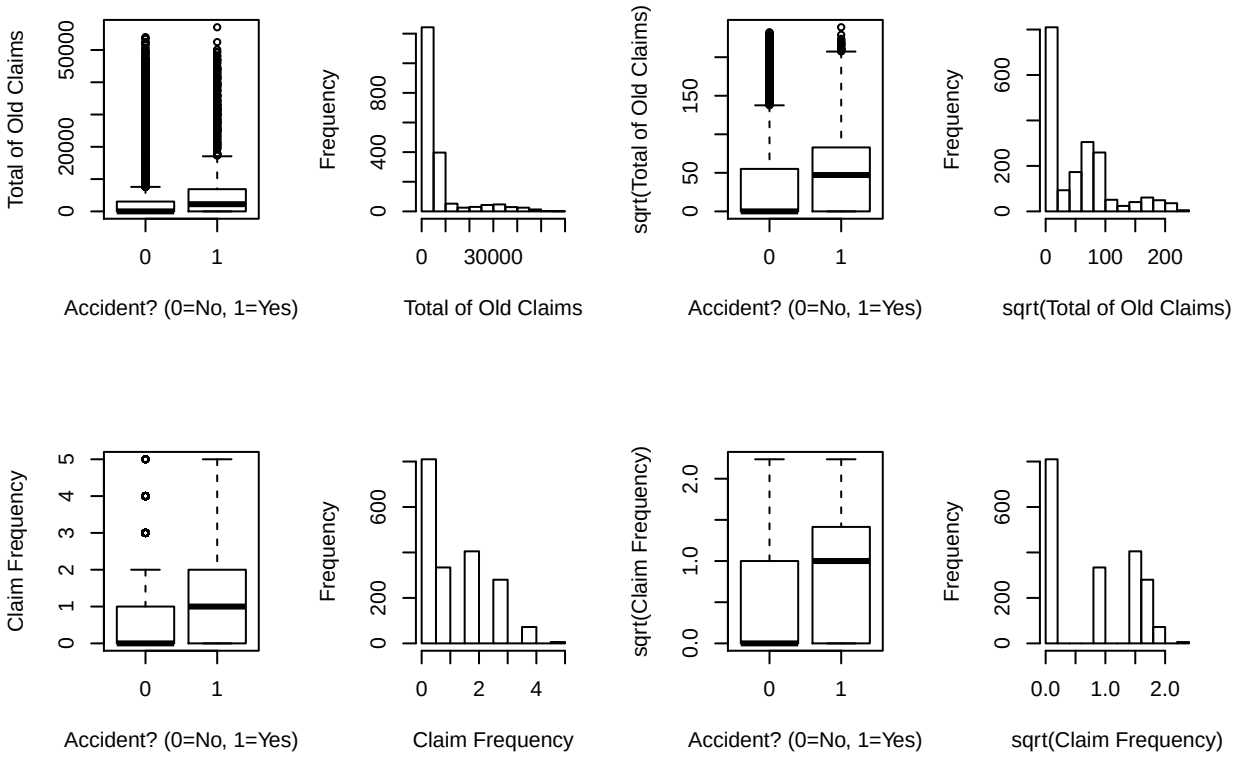
Given the results of all of our tests and measurements of these three models we conclude that we have a fairly trustworthy model that appears to do a good job predicting if a given driver will get into an accident. We see that the number of accidents predicted are slight less then what we would expect so we should be carefully with individuals that are close to a .5 probability. On the other hand our model for predicting the cost of repair is not a trustworthy model. It appears that this set of predictor variables sis not give us a good model. Thinking about the situation though, this does make sense. Ultimately the cost of repair is a mixture of the cost of the car's and severity of the accident and we do not have data on these values.

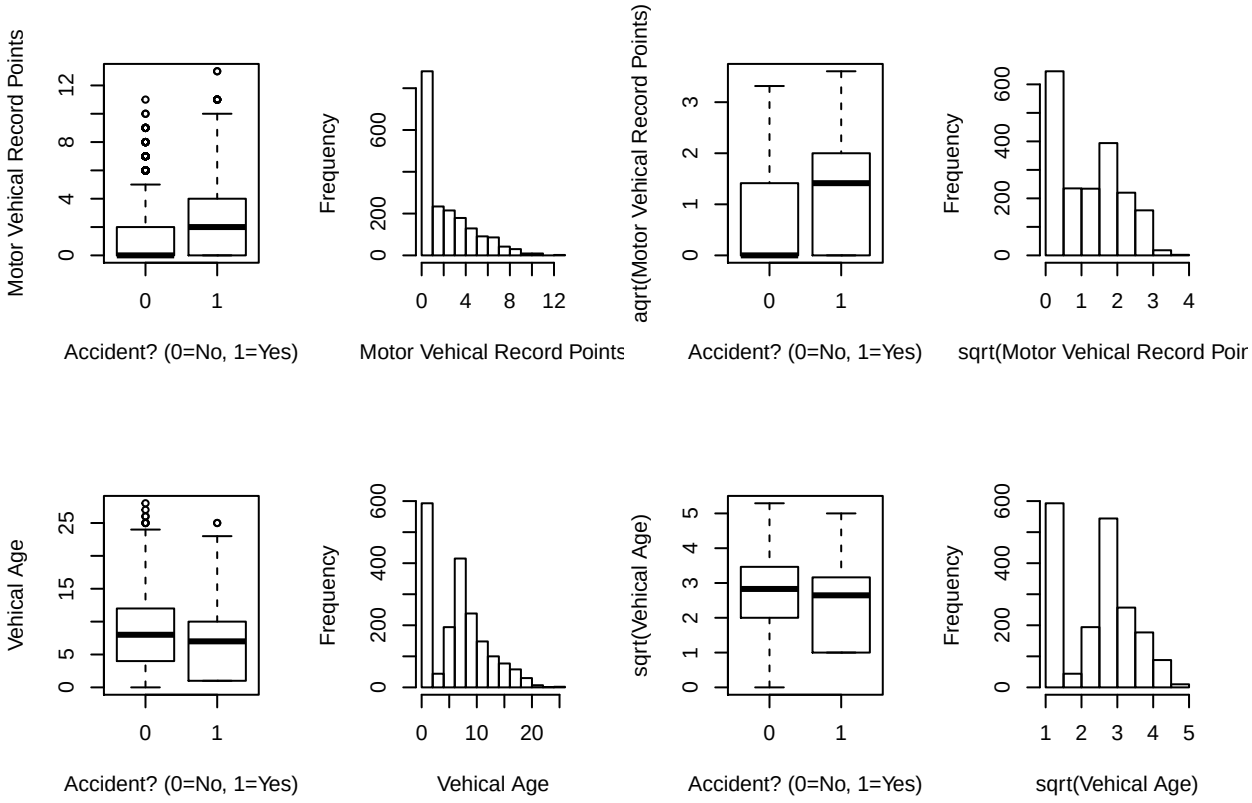
Apendix A: Plots of the Numerical Predictor Variables

For the plots below the first two plots are the original data and the second two are under a Box-Cox Transformation.




Income
0e+00
2e+05
0
1
Accident? (0=No, 1=Yes)
Frequency
0
200
400
0
150000
Income
sqrt(Income)
0
200
400
600
0
1
Accident? (0=No, 1=Yes)
Frequency
0
200
400
0
200
500
sqrt(Income)
Travel Time
0
40
80
120
0
1
Accident? (0=No, 1=Yes)
Frequency
0
500
1500
0
50
100
Travel Time
sqrt(Travel Time)
2
4
6
8
10
0
1
Accident? (0=No, 1=Yes)
Frequency
0
400
800
2
6
10
sqrt(Travel Time)
BlueBook Value
0
20000
50000
0
1
Accident? (0=No, 1=Yes)
Frequency
0
200
400
0
20000
50000
BlueBook Value
log(BlueBook Value)
8
9
10
11
0
1
Accident? (0=No, 1=Yes)
Frequency
0
200
400
7
8
9
10
log(BlueBook Value)
Time in Force
5
10
20
0
1
Accident? (0=No, 1=Yes)
Frequency
0
200
500
0
5
15
Time in Force
log(Time in Force)
0.0
1.0
2.0
3.0
0
1
Accident? (0=No, 1=Yes)
Frequency
0
200
500
0.0
1.5
3.0
log(Time in Force)


Total of Old Claims
0
20000
50000
0
1
Accident? (0=No, 1=Yes)
Frequency
0
400
800
0
30000
Total of Old Claims
sqrt(Total of Old Claims)
0
50
150
0
1
Accident? (0=No, 1=Yes)
Frequency
0
200
600
0
100
200
sqrt(Total of Old Claims)
Claim Frequency
0
1
2
3
4
5
0
1
Accident? (0=No, 1=Yes)
Frequency
0
200
600
0
2
4
Claim Frequency
sqrt(Claim Frequency)
0.0
1.0
2.0
0
1
Accident? (0=No, 1=Yes)
Frequency
0
200
600
0.0
1.0
2.0
sqrt(Claim Frequency)


Motor Vehical Record Points
0
4
8
12
0
1
Accident? (0=No, 1=Yes)
Frequency
0
200
600
0
4
8
12
Motor Vehical Record Points
aqrt(Motor Vehical Record Points)
0
1
2
3
0
1
Accident? (0=No, 1=Yes)
Frequency
0
200
400
600
0
1
2
3
4
sqrt(Motor Vehical Record Poir
Vehical Age
0
5
15
25
0
1
Accident? (0=No, 1=Yes)
Frequency
0
200
400
600
0
10
20
Vehical Age
sqrt(Vehical Age)
0
1
2
3
4
5
0
1
Accident? (0=No, 1=Yes)
Frequency
0
200
400
600
1
2
3
4
5
sqrt(Vehical Age)

Apendix B: Tables for our Categorical Data

The following tables contain the probabilities of the different categories with respect to Target variable indicating if the car has been in an accident.

One Parent	no Accident	Accident
No	0.764	0.236
Yes	0.557	0.443

Married	no Accident	Accident
No	0.662	0.338
Yes	0.784	0.216

Gender	no Accident	Accident
Male	0.749	0.251
Female	0.725	0.275

Higher Ed	no Accident	Accident
No	0.664	0.336
Yes	0.797	0.203

Car Use	no Accident	Accident
Commercial	0.639	0.361
Private	0.784	0.216

Gender	no Accident	Accident
Male	0.749	0.251
Female	0.725	0.275

Car Type	no Accident	Accident
Minivan	0.838	0.162
Panel Truck	0.715	0.285
Pickup	0.678	0.322
Sports Car	0.661	0.339
Van	0.748	0.252
SUV	0.701	0.299

Red Car	no Accident	Accident
No	0.73	0.27
Yes	0.749	0.251

Revoked License	no Accident	Accident
No	0.76	0.24
Yes	0.557	0.443

Environment	no Accident	Accident
Urban	0.682	0.318
Rural	0.929	0.071

Home Ownership	no Accident	Accident
No	0.631	0.369
Yes	0.78	0.22

Job Type	no Accident	Accident
Blue Collar	0.652	0.348
White Collar	0.761	0.238

Young Driver	no Accident	Accident
No	0.751	0.249
Yes	0.626	0.374

Children	no Accident	Accident
No	0.778	0.222
Yes	0.66	0.34

Apendix C: Code for this analysis

The code below was used to conduct the above analysis.

```
#############################################################################################################
#### Loading Libraries ####
library(plyr)
library(dplyr)


#############################################################################################################
#### Data Loading and Cleaning ####
# Loading the data
auto <- read.csv("DATA621/insurance_training_data.csv")
auto.eval <- read.csv("DATA621/insurance-evaluation-data.csv")

# Summary of data
summary(auto)
dim(auto)

summary(auto.eval)
dim(auto.eval)

# Fix monetary values variable
# Income
auto$INCOME <- as.numeric(gsub("[$,]","", as.character(auto$INCOME)))
auto.eval$INCOME <- as.numeric(gsub("[$,]","", as.character(auto.eval$INCOME)))
# Home Value
auto$HOME_VAL <- as.numeric(gsub("[$,]","", as.character(auto$HOME_VAL)))
auto.eval$HOME_VAL <- as.numeric(gsub("[$,]","", as.character(auto.eval$HOME_VAL)))
# Blue Book Value
auto$BLUEBOOK <- as.numeric(gsub("[$,]","", as.character(auto$BLUEBOOK)))
auto.eval$BLUEBOOK <- as.numeric(gsub("[$,]","", as.character(auto.eval$BLUEBOOK)))
# Old Claims
auto$OLDCLAIM <- as.numeric(gsub("[$,]","", as.character(auto$OLDCLAIM)))
auto.eval$OLDCLAIM <- as.numeric(gsub("[$,]","", as.character(auto.eval$OLDCLAIM)))

# Home Value, given that we are representing home owner ship we will create a home owner variable
auto$HOME_OWN[auto$HOME_VAL == 0] <- "No"
auto$HOME_OWN[auto$HOME_VAL != 0] <- "Yes"
auto$HOME_OWN <- as.factor(auto$HOME_OWN)
# Auto Eval
auto.eval$HOME_OWN[auto.eval$HOME_VAL == 0] <- "No"
auto.eval$HOME_OWN[auto.eval$HOME_VAL != 0] <- "Yes"
auto.eval$HOME_OWN <- as.factor(auto.eval$HOME_OWN)

# Creating a Job Type variable to distinguish white collar vs blue collar jobs
auto$JOB_TYPE[auto$JOB == "z_Blue Collar"] <- "BC"
auto$JOB_TYPE[auto$JOB != "z_Blue Collar" & auto$JOB != ""] <- "WC"
auto$JOB_TYPE <- as.factor(auto$JOB_TYPE)
# Auto Eval
auto.eval$JOB_TYPE[auto.eval$JOB == "z_Blue Collar"] <- "BC"
auto.eval$JOB_TYPE[auto.eval$JOB != "z_Blue Collar" & auto.eval$JOB != ""] <- "WC"
auto.eval$JOB_TYPE <- as.factor(auto.eval$JOB_TYPE)
```

```r
# Creating Educaiton Level Variable
auto$HIGHER_ED[auto$EDUCATION == "<High School" |
                  auto$EDUCATION == "z_High School"] <- "No"
auto$HIGHER_ED[auto$EDUCATION == "Bachelors" | auto$EDUCATION == "Masters" |
                  auto$EDUCATION == "PhD"] <- "Yes"
auto$HIGHER_ED <- as.factor(auto$HIGHER_ED)
# Auto Eval
auto.eval$HIGHER_ED[auto.eval$EDUCATION == "<High School" |
                      auto.eval$EDUCATION == "z_High School"] <- "No"
auto.eval$HIGHER_ED[auto.eval$EDUCATION == "Bachelors" | auto.eval$EDUCATION == "Masters" |
                  auto.eval$EDUCATION == "PhD"] <- "Yes"
auto.eval$HIGHER_ED <- as.factor(auto.eval$HIGHER_ED)

# Creating the Driving Kids Variable
auto$KD[auto$KIDSDRIV == 0] <- "No"
auto$KD[auto$KIDSDRIV != 0] <- "Yes"
auto$KD <- as.factor(auto$KD)
# Auto Eval
auto.eval$KD[auto.eval$KIDSDRIV == 0] <- "No"
auto.eval$KD[auto.eval$KIDSDRIV != 0] <- "Yes"
auto.eval$KD <- as.factor(auto.eval$KD)

# Creating the Kids at Home Variable
auto$KIDS[auto$HOMEKIDS == 0] <- "No"
auto$KIDS[auto$HOMEKIDS != 0] <- "Yes"
auto$KIDS <- as.factor(auto$KIDS)
# Auto Eval
auto.eval$KIDS[auto.eval$HOMEKIDS == 0] <- "No"
auto.eval$KIDS[auto.eval$HOMEKIDS != 0] <- "Yes"
auto.eval$KIDS <- as.factor(auto.eval$KIDS)

summary(auto)

# Renaming the Urbanicity
auto$URBANICITY <- revalue(auto$URBANICITY,c("Highly Urban/ Urban"= "Urban",
                                             "z_Highly Rural/ Rural" = "Rural"))
auto.eval$URBANICITY <- revalue(auto.eval$URBANICITY,c("Highly Urban/ Urban"= "Urban",
                                                  "z_Highly Rural/ Rural" = "Rural"))

# Renaming Male Female
auto$SEX <- revalue(auto$SEX,c("z_F"= "F"))
auto.eval$SEX <- revalue(auto.eval$SEX,c("z_F"= "F"))

# Renaming Marital Status
auto$MSTATUS <- revalue(auto$MSTATUS, c("z_No" = "No"))
auto.eval$MSTATUS <- revalue(auto.eval$MSTATUS, c("z_No" = "No"))

# Renaming SUV's
auto$CAR_TYPE <- revalue(auto$CAR_TYPE, c("z_SUV" = "SUV"))
auto.eval$CAR_TYPE <- revalue(auto.eval$CAR_TYPE, c("z_SUV" = "SUV"))

# Fixing missing values using the median value
auto.data <- auto
```

```r
# Age
auto.data$AGE[is.na(auto.data$AGE)] <- median(auto.data$AGE, na.rm = TRUE)
auto.eval$AGE[is.na(auto.eval$AGE)] <- median(auto.eval$AGE, na.rm = TRUE)
# Income
auto.data$INCOME[is.na(auto.data$INCOME)] <- median(auto.data$INCOME, na.rm = TRUE)
auto.eval$INCOME[is.na(auto.eval$INCOME)] <- median(auto.eval$INCOME, na.rm = TRUE)
# Years on Job
auto.data$YOJ[is.na(auto.data$YOJ)] <- median(auto.data$YOJ, na.rm = TRUE)
auto.eval$YOJ[is.na(auto.eval$YOJ)] <- median(auto.eval$YOJ, na.rm = TRUE)
# Home Value
auto.data$HOME_VAL[is.na(auto.data$HOME_VAL)] <- median(auto.data$HOME_VAL, na.rm = TRUE)
auto.eval$HOME_VAL[is.na(auto.eval$HOME_VAL)] <- median(auto.eval$HOME_VAL, na.rm = TRUE)
# Car Age
auto.data$CAR_AGE[is.na(auto.data$CAR_AGE)] <- median(auto.data$CAR_AGE, na.rm = TRUE)
auto.eval$CAR_AGE[is.na(auto.eval$CAR_AGE)] <- median(auto.eval$CAR_AGE, na.rm = TRUE)

# Home Ownership can't be NA must remove
auto.data <- auto.data %>%
  filter(!is.na(HOME_OWN))
auto.eval <- auto.eval %>%
  filter(!is.na(HOME_OWN))

# Car Age can't be less than 0
auto.data <- auto.data%>%
  filter(CAR_AGE >= 0 )

# Job Type Can't be NA must remove
auto.data <- auto.data %>%
  filter(!is.na(JOB_TYPE))
auto.eval <- auto.eval %>%
  filter(!is.na(JOB_TYPE))

# Now that we have cleaned these data up lets select the data for our two model types
# Logistic Drop Index, Target Amount, Home Value (replaced by Home Own), Job (replaced by Job Type)
auto.logistic <- auto.data %>%
  select(-INDEX, -TARGET_AMT, -HOME_VAL, -JOB, -EDUCATION, -KIDSDRIV, -HOMEKIDS)
# Linear Model Drop Index, Target Flag, Home Value (replaced by home own), Job (replaced by Job Types)
auto.lm <- auto.data %>%
  filter(TARGET_FLAG != 0) %>%
  select(-INDEX, -TARGET_FLAG, -HOME_VAL, -JOB, -EDUCATION, -KIDSDRIV, -HOMEKIDS)

#########################################################################################################
#### Investigating the Data ####
# Investigating the target data for the logistic model
table(auto$TARGET_FLAG)
table(auto$TARGET_FLAG)/sum(table(auto$TARGET_FLAG))

# Investigating the claim data for the linear model
par(mfrow = c(2,2))
boxplot(auto.lm$TARGET_AMT, main = "Repair Amounts")
hist(auto.lm$TARGET_AMT, main = "Histogram of Repair Amount",
     xlab = "Repair Amount")
# Highly right skewd lets look at the log(TARGET_AMT)
```

```r
boxplot(log(auto.lm$TARGET_AMT), main = "log(Repair Amounts)")
BoxCox.lambda(auto.lm$TARGET_AMT) # Indicates log transform
hist(log(auto.lm$TARGET_AMT), main = "Histogram of log(Repair Amount)",
     xlab = "log(Repair Amount)")

# log(TARGET_AMT) seems to work well the data is better centered with heavier tails.


# Building the boxplots and histogram
# Kids Driving
par(mfrow = c(2,3))
boxplot(KIDSDRIV ~ TARGET_FLAG, data = auto.logistic, xlab = "Accident? (0=No, 1=Yes)",
        ylab = "# of Driving Children") # No difference between the two
hist(auto.lm$KIDSDRIV, main = "", xlab = "# of Driving Children")
BoxCox.lambda(auto.lm$KIDSDRIV) #Indicates we should log transform
hist(log(auto.lm$KIDSDRIV), main = "", xlab = "log(# of Driving Children)") # skewed clustered at 1

# Age
boxplot(AGE ~ TARGET_FLAG, data = auto.logistic, xlab = "Accident? (0=No, 1=Yes)",
        ylab = "Age of Driver") # No accident older driver
hist(auto.lm$AGE, main = "", xlab = "Age of Driver")
BoxCox.lambda(auto.lm$AGE) #Indicates we should sqrt transform
hist(sqrt(auto.lm$AGE), main = "", xlab = "sqrt(Age of Driver)") # Fairly Normal

# Children at Home
par(mfrow = c(2,3))
boxplot(HOMEKIDS ~ TARGET_FLAG, data = auto.logistic, xlab = "Accident? (0=No, 1=Yes)",
        ylab = "# of Children") # No real difference
hist(auto.lm$HOMEKIDS, main = "", xlab = "# of Children")
BoxCox.lambda(auto.lm$HOMEKIDS) #Indicates we should log transform
hist(sqrt(auto.lm$HOMEKIDS), main = "", xlab = "sqrt(# of Children)") # Cluster at 0 normal otherwise

# Years on Job
boxplot(YOJ ~ TARGET_FLAG, data = auto.logistic, xlab = "Accident? (0=No, 1=Yes)",
        ylab = "Year on the Job") # No difference between the two
hist(auto.lm$YOJ, main = "", xlab = "Years on the Job")
BoxCox.lambda(auto.lm$YOJ) #Indicates we should square transform
hist((auto.lm$YOJ)^2, main = "", xlab = "Years on the Job^2") # Normal with second cluster at 0

# Income
par(mfrow = c(2,3))
boxplot(INCOME ~ TARGET_FLAG, data = auto.logistic, xlab = "Accident? (0=No, 1=Yes)",
        ylab = "Income") # Higher income, less accidents
hist(auto.lm$INCOME, main = "", xlab = "Income")
BoxCox.lambda(auto.lm$INCOME) #Indicates we should sqrt transform
hist(sqrt(auto.lm$INCOME), main = "", xlab = "sqrt(Income)") #Right skew, gamma or Erlang

# Travel Time
boxplot(TRAVTIME ~ TARGET_FLAG, data = auto.logistic, xlab = "Accident? (0=No, 1=Yes)",
        ylab = "Travel Time") # May Not be a difference
hist(auto.logistic$TRAVTIME, main = "", xlab = "Travel Time")
BoxCox.lambda(auto.lm$TRAVTIME) #Indicates we should sqrt transform
hist(sqrt(auto.logistic$TRAVTIME), main = "", xlab = "sqrt(Travel Time)") #Right skew
```

```r
# Bluebook Value
par(mfrow = c(2,3))
boxplot(BLUEBOOK ~ TARGET_FLAG, data = auto.logistic, xlab = "Accident? (0=No, 1=Yes)",
        ylab = "BlueBook Value") # More expensive car less accidents
hist(auto.lm$BLUEBOOK, main = "", xlab = "BlueBook Value")
BoxCox.lambda(auto.lm$BLUEBOOK) #Indicates we should log transform
hist(log(auto.lm$BLUEBOOK), main = "", xlab = "log(BlueBook Value)") #Right skew

# Time in Force (Customer Time)
boxplot(TIF ~ TARGET_FLAG, data = auto.logistic, xlab = "Accident? (0=No, 1=Yes)",
        ylab = "Time in Force") # Longer in force less accidents
hist(auto.lm$TIF, main = "", xlab = "Time in Force")
BoxCox.lambda(auto.lm$TIF) #Indicates we should log transform
hist(log(auto.lm$TIF), main = "", xlab = "log(Time in Force)") #Right skew

# Old Claims
par(mfrow = c(2,3))
boxplot(OLDCLAIM ~ TARGET_FLAG, data = auto.logistic, xlab = "Accident? (0=No, 1=Yes)",
        ylab = "Total of Old Claims") # Slightly higher old claims in accident cars
hist(auto.lm$OLDCLAIM, main = "", xlab = "Total of Old Claims")
BoxCox.lambda(auto.lm$OLDCLAIM) #Indicates we should log transform
hist(sqrt(auto.lm$OLDCLAIM), main = "", xlab = "sqrt(Total of Old Claims)") #Right skew

# Claim Frequency
boxplot(CLM_FREQ ~ TARGET_FLAG, data = auto.logistic, xlab = "Accident? (0=No, 1=Yes)",
        ylab = "Claim Frequency") # Slightly higher old claims in accident cars
hist(auto.lm$CLM_FREQ, main = "", xlab = "Claim Frequency")
BoxCox.lambda(auto.lm$CLM_FREQ) #Indicates we should sqrt transform because of inf
hist(sqrt(auto.lm$CLM_FREQ), main = "", xlab = "sqrt(Claim Frequency)") #Cluster at 0, normal otherwise

#Motor Vehical Record Points
par(mfrow = c(2,3))
boxplot(MVR_PTS ~ TARGET_FLAG, data = auto.logistic, xlab = "Accident? (0=No, 1=Yes)",
        ylab = "Motor Vehical Record Points") # Higher points more accidents
hist(auto.lm$MVR_PTS, main = "", xlab = "Motor Vehical Record Points")
BoxCox.lambda(auto.lm$MVR_PTS) #Indicates we should log transform
hist(sqrt(auto.lm$MVR_PTS), main = "", xlab = "sqrt(Motor Vehical Record Points)") #right skewed

# Car Age
boxplot(CAR_AGE ~ TARGET_FLAG, data = auto.logistic, xlab = "Accident? (0=No, 1=Yes)",
        ylab = "Vehical Age") # less accidents on older vehicals
hist(auto.lm$CAR_AGE, main = "", xlab = "Vehical Age")
BoxCox.lambda(auto.lm$CAR_AGE) #Indicates we should log transform
hist(sqrt(auto.lm$CAR_AGE), main = "", xlab = "sqrt(Vehical Age)") #Cluster at 0, right skew otherwise

summary(auto.logistic)

# Categorical Variables
# Single Parent
table(auto.logistic$PARENT1, auto.logistic$TARGET_FLAG)
table(auto.logistic$PARENT1, auto.logistic$TARGET_FLAG)[1,]/
  sum(table(auto.logistic$PARENT1, auto.logistic$TARGET_FLAG)[1,])
table(auto.logistic$PARENT1, auto.logistic$TARGET_FLAG)[2,]/
```

```
  sum(table(auto.logistic$PARENT1, auto.logistic$TARGET_FLAG)[2,])
# Increased probability of being in an accident if a single parent 0.44 vs 0.23

# Marital Status
table(auto.logistic$MSTATUS, auto.logistic$TARGET_FLAG)
table(auto.logistic$MSTATUS, auto.logistic$TARGET_FLAG)[1,]/
  sum(table(auto.logistic$MSTATUS, auto.logistic$TARGET_FLAG)[1,])
table(auto.logistic$MSTATUS, auto.logistic$TARGET_FLAG)[2,]/
  sum(table(auto.logistic$MSTATUS, auto.logistic$TARGET_FLAG)[2,])
# Slight increase in probability of being in an accident if unmarried

# Gender
table(auto.logistic$SEX, auto.logistic$TARGET_FLAG)
table(auto.logistic$SEX, auto.logistic$TARGET_FLAG)[1,]/
  sum(table(auto.logistic$SEX, auto.logistic$TARGET_FLAG)[1,])
table(auto.logistic$SEX, auto.logistic$TARGET_FLAG)[2,]/
  sum(table(auto.logistic$SEX, auto.logistic$TARGET_FLAG)[2,])
# Females slightly more likely to get into accidents probably not significant

# Education Level
table(auto.logistic$HIGHER_ED, auto.logistic$TARGET_FLAG)
table(auto.logistic$HIGHER_ED, auto.logistic$TARGET_FLAG)[1,]/
  sum(table(auto.logistic$HIGHER_ED, auto.logistic$TARGET_FLAG)[1,])
table(auto.logistic$HIGHER_ED, auto.logistic$TARGET_FLAG)[2,]/
  sum(table(auto.logistic$HIGHER_ED, auto.logistic$TARGET_FLAG)[2,])
# Higher Education reduces the accident risk

# Car Use
table(auto.logistic$CAR_USE, auto.logistic$TARGET_FLAG)
table(auto.logistic$CAR_USE, auto.logistic$TARGET_FLAG)[1,]/
  sum(table(auto.logistic$CAR_USE, auto.logistic$TARGET_FLAG)[1,])
table(auto.logistic$CAR_USE, auto.logistic$TARGET_FLAG)[2,]/
  sum(table(auto.logistic$CAR_USE, auto.logistic$TARGET_FLAG)[2,])
# Private Car use has less chance of accidents

# Car Type
table(auto.logistic$CAR_TYPE, auto.logistic$TARGET_FLAG)
table(auto.logistic$CAR_TYPE, auto.logistic$TARGET_FLAG)[1,]/
  sum(table(auto.logistic$CAR_TYPE, auto.logistic$TARGET_FLAG)[1,])
table(auto.logistic$CAR_TYPE, auto.logistic$TARGET_FLAG)[2,]/
  sum(table(auto.logistic$CAR_TYPE, auto.logistic$TARGET_FLAG)[2,])
table(auto.logistic$CAR_TYPE, auto.logistic$TARGET_FLAG)[3,]/
  sum(table(auto.logistic$CAR_TYPE, auto.logistic$TARGET_FLAG)[3,])
table(auto.logistic$CAR_TYPE, auto.logistic$TARGET_FLAG)[4,]/
  sum(table(auto.logistic$CAR_TYPE, auto.logistic$TARGET_FLAG)[4,])
table(auto.logistic$CAR_TYPE, auto.logistic$TARGET_FLAG)[5,]/
  sum(table(auto.logistic$CAR_TYPE, auto.logistic$TARGET_FLAG)[5,])
table(auto.logistic$CAR_TYPE, auto.logistic$TARGET_FLAG)[6,]/
  sum(table(auto.logistic$CAR_TYPE, auto.logistic$TARGET_FLAG)[6,])
# Minivan 0.162, Panel Truck 0.285, Pickup 0.322, Sports Car 0.339, Van 0.252, SUV 0.299

# Red Car
table(auto.logistic$RED_CAR, auto.logistic$TARGET_FLAG)
```

```
table(auto.logistic$RED_CAR, auto.logistic$TARGET_FLAG)[1,]/
  sum(table(auto.logistic$RED_CAR, auto.logistic$TARGET_FLAG)[1,])
table(auto.logistic$RED_CAR, auto.logistic$TARGET_FLAG)[2,]/
  sum(table(auto.logistic$RED_CAR, auto.logistic$TARGET_FLAG)[2,])
# P(A|Not Red) = 0.27, P(A|Red) = 0.251

# Lisence Revoked
table(auto.logistic$REVOKED, auto.logistic$TARGET_FLAG)
table(auto.logistic$REVOKED, auto.logistic$TARGET_FLAG)[1,]/
  sum(table(auto.logistic$REVOKED, auto.logistic$TARGET_FLAG)[1,])
table(auto.logistic$REVOKED, auto.logistic$TARGET_FLAG)[2,]/
  sum(table(auto.logistic$REVOKED, auto.logistic$TARGET_FLAG)[2,])
#P(A|non-Revoked) = 0.24, P(A|Revoked) = 0.443

# Urban or Rural
table(auto.logistic$URBANICITY, auto.logistic$TARGET_FLAG)
table(auto.logistic$URBANICITY, auto.logistic$TARGET_FLAG)[1,]/
  sum(table(auto.logistic$URBANICITY, auto.logistic$TARGET_FLAG)[1,])
table(auto.logistic$URBANICITY, auto.logistic$TARGET_FLAG)[2,]/
  sum(table(auto.logistic$URBANICITY, auto.logistic$TARGET_FLAG)[2,])
# P(A|Urban) = 0.318, P(A|Rural) =0.071

# Home Ownership
table(auto.logistic$HOME_OWN, auto.logistic$TARGET_FLAG)
table(auto.logistic$HOME_OWN, auto.logistic$TARGET_FLAG)[1,]/
  sum(table(auto.logistic$HOME_OWN, auto.logistic$TARGET_FLAG)[1,])
table(auto.logistic$HOME_OWN, auto.logistic$TARGET_FLAG)[2,]/
  sum(table(auto.logistic$HOME_OWN, auto.logistic$TARGET_FLAG)[2,])
#P(A|Rent) = 0.369, P(A|Own) = 0.22

# Job Type
table(auto.logistic$JOB_TYPE, auto.logistic$TARGET_FLAG)
table(auto.logistic$JOB_TYPE, auto.logistic$TARGET_FLAG)[1,]/
  sum(table(auto.logistic$JOB_TYPE, auto.logistic$TARGET_FLAG)[1,])
table(auto.logistic$JOB_TYPE, auto.logistic$TARGET_FLAG)[2,]/
  sum(table(auto.logistic$JOB_TYPE, auto.logistic$TARGET_FLAG)[2,])
#P(A|BC) = 0.348, P(A|WC) = 0.238

# KD Kids Driving
table(auto.logistic$KD, auto.logistic$TARGET_FLAG)
table(auto.logistic$KD, auto.logistic$TARGET_FLAG)[1,]/
  sum(table(auto.logistic$KD, auto.logistic$TARGET_FLAG)[1,])
table(auto.logistic$KD, auto.logistic$TARGET_FLAG)[2,]/
  sum(table(auto.logistic$KD, auto.logistic$TARGET_FLAG)[2,])
#P(A|NO KD) = 0.249, P(A|KD) = 0.374

# Kids?
table(auto.logistic$KIDS, auto.logistic$TARGET_FLAG)
table(auto.logistic$KIDS, auto.logistic$TARGET_FLAG)[1,]/
  sum(table(auto.logistic$KIDS, auto.logistic$TARGET_FLAG)[1,])
table(auto.logistic$KIDS, auto.logistic$TARGET_FLAG)[2,]/
  sum(table(auto.logistic$KIDS, auto.logistic$TARGET_FLAG)[2,])
#P(A|NO KD) = 0.222, P(A|KD) = .34
```

```r
par(mfrow = c(1,1))
pairs(auto.data[c(3, 4, 5, 6, 7, 14, 16, 17, 20, 21, 22, 23)], col=as.factor(auto.data$TARGET_FLAG))

summary(auto.data)
#########################################################################################################
#### Building the Linear Model to Predict cost given an Accident ####
library(forecast)
library(glmnet)
library(car)
library(MASS)
# 1907 Accident rows.
# Subset the data into training(70%) and test(30%)
n_lm <- dim(auto.lm)[1]
set.seed(2001)

test <- sample(n_lm, round(n_lm * .3))
lm.train <- auto.lm[-test,]
lm.test <- auto.lm[test,]

#### Linear Model with BoxCox Transformations #####
# Under BoxCox Transformation of predictor and response variables
auto.BC <- auto.lm
auto.BC$TARGET_AMT <- log(auto.BC$TARGET_AMT)
auto.BC$AGE <- sqrt(auto.BC$AGE)
auto.BC$YOJ <- (auto.BC$YOJ)^2
auto.BC$INCOME <- sqrt(auto.BC$INCOME)
auto.BC$TRAVTIME <- sqrt(auto.BC$TRAVTIME)
auto.BC$BLUEBOOK <- log(auto.BC$BLUEBOOK)
auto.BC$TIF <- log(auto.BC$TIF)
auto.BC$OLDCLAIM <- sqrt(auto.BC$OLDCLAIM)
auto.BC$CLM_FREQ <- sqrt(auto.BC$CLM_FREQ)
auto.BC$MVR_PTS <- sqrt(auto.BC$MVR_PTS)
auto.BC$CAR_AGE <- sqrt(auto.BC$CAR_AGE)
summary(auto.BC)

# Ridge Regression
autoBC.train <- auto.BC[-test,]
autoBC.test <- auto.BC[test,]
x <- model.matrix(TARGET_AMT ~ ., data = auto.BC)[,-1] # define predictor matrix
# excl intercept col of 1s
x.train <- x[-test,]                          # define training predictor matrix
x.test <- x[test,]                            # define test predictor matrix
y <- auto.BC$TARGET_AMT                       # define response variable
y.train <- y[-test]                           # define training response variable
y.test <- y[test]                             # define test response variable
n.train <- dim(autoBC.train)[1]                 # training sample size = 332
n.test <- dim(autoBC.test)[1]                   # test sample size = 110

# Selecting a model using Ridge Regression
par(mfrow = c(1,2))
grid <- 10^seq(10, -2, length = 100)
ridge.mod <- glmnet(x.train, y.train, alpha = 0, lambda = grid, thresh = 1e-12)
plot(ridge.mod, xvar = "lambda", label = TRUE)
```

```r
set.seed(1306)
cv.out <- cv.glmnet(x.train, y.train, alpha = 0)
plot(cv.out)
bestlam <- cv.out$lambda.min
bestlam                                    # Lambda = 4.904021 (leads to smallest CV error)
log(bestlam)

ridge.mod <- glmnet(x.train, y.train, alpha = 0, lambda = bestlam)
ridge.pred <- predict(ridge.mod, s = bestlam, newx = x.test)
mean((ridge.pred - y.test)^2)              # Mean Prediction Error = 3074.378
sd((ridge.pred - y.test)^2)/sqrt(n.test)   # Standard Error = 357.9628

largelam <- cv.out$lambda.1se
largelam                                   # Lambda = 41.67209 (largest lambda w/in 1 SE)
ridge.mod <- glmnet(x.train, y.train, alpha = 0, lambda = largelam)
ridge.pred <- predict(ridge.mod, s = largelam, newx = x.test)
mean((ridge.pred - y.test)^2)              # Mean Prediction Error = 3070.87
sd((ridge.pred - y.test)^2)/sqrt(n.test)   # Standard Error = 350.5467

# Here are the estimated coefficients
predict(ridge.mod, type = "coefficients", s = largelam)[1:11,]

autoBC_ridgefit <- lm(TARGET_AMT ~ AGE + YOJ + INCOME +
                      PARENT1 + MSTATUS + SEX + TRAVTIME + CAR_USE +
                      BLUEBOOK + TIF, data = autoBC.train)
summary(autoBC_ridgefit)
plot(autoBC_ridgefit)

#### Using Robust Regression ####
rr.autoBC <- rlm(TARGET_AMT ~ ., data = autoBC.train)
summary(rr.autoBC)
plot(rr.autoBC)

rr2.autoBC <- lm(TARGET_AMT ~ AGE + YOJ + INCOME +
        PARENT1 + MSTATUS + SEX + TRAVTIME + CAR_USE +
        BLUEBOOK + TIF, data = autoBC.train)
summary(rr2.autoBC)
plot(rr2.autoBC)

# None of the coefficients are significant

mean((lm.test$TARGET_AMT - predict(auto_ridgefit, lm.test))^2)  # Mean Predictor Error (test MSE) = 55,
sd((lm.test$TARGET_AMT - predict(auto_ridgefit, lm.test))^2)/sqrt(n.test) # Standard Error = 13,736,151
# Even under normalizing Tranformations we have no better predictions with the only significant
# predictor of Marital Status at 0.1 level. No Homoskedasiticy in the Residuals.


#########################################################################################################
#### Logistic Regression ####
library(caret)
library(pROC)
library(knitr)
#Building the training and test data. 70/30
```

```r
n_log <- dim(auto.logistic)[1]
set.seed(4321)

test <- sample(n_log, round(n_log * .3))

#### Logistic Model 1 - No Transformations ####
log.train <- auto.logistic[-test,]
log.test <- auto.logistic[test,]

# Fitting all of the variables
auto.log <- glm(TARGET_FLAG ~ ., data = log.train,
                family = binomial(link = "logit"))

summary(auto.log) # AIC 4488.
logLik(auto.log) # -2216.408 on 28 df

# Logit model odds ratios
coefficients(auto.log)
exp(auto.log$coefficients)

# Assessing the model
log.probs <- predict(auto.log, newdata=log.test, type="response")
log.pred <- ifelse(log.probs > 0.5, 1, 0)
table(log.pred, log.test$TARGET_FLAG)
mean(log.pred == log.test$TARGET_FLAG)

# Generating the Confusion Matrix and metrics
confusionMatrix(data = as.factor(log.pred),
                reference = as.factor(log.test$TARGET_FLAG),
                positive = "1")
# Classification Error Rate
1-.7826 # 0.2174
# F1 Score 2*Percision*Sensitivity/percision + sensitivity
(2*0.6958*0.4056)/(0.6958 + 0.4056) #0.5125
# Generating the ROC curve and the AUC
roc_log <- roc(response = log.test$TARGET_FLAG,
               predictor = log.probs)
auc(roc_log) #0.8086
plot(roc_log, legacy.axes = TRUE) # Not included in the report

#### Logistic Model Two - BoxCox Transformations ####
# Second Model under the transformations used in the second linear model.
BC.logistic <- auto.logistic
BC.logistic$AGE <- sqrt(BC.logistic$AGE)
BC.logistic$YOJ <- (BC.logistic$YOJ)^2
BC.logistic$INCOME <- sqrt(BC.logistic$INCOME)
BC.logistic$TRAVTIME <- sqrt(BC.logistic$TRAVTIME)
BC.logistic$BLUEBOOK <- log(BC.logistic$BLUEBOOK)
BC.logistic$TIF <- log(BC.logistic$TIF)
BC.logistic$OLDCLAIM <- sqrt(BC.logistic$OLDCLAIM)
BC.logistic$CLM_FREQ <- sqrt(BC.logistic$CLM_FREQ)
BC.logistic$MVR_PTS <- sqrt(BC.logistic$MVR_PTS)
BC.logistic$CAR_AGE <- sqrt(BC.logistic$CAR_AGE)
```

```r
BClog.train <- BC.logistic[-test,]
BClog.test <- BC.logistic[test,]

# Fitting all of the transformed variables
auto.BClog <- glm(TARGET_FLAG ~ ., data = BClog.train,
                  family = binomial(link = "logit"))

summary(auto.BClog)
logLik(auto.BClog) # -2212.882 on 28 df

# Logit model odds ratios
coefficients(auto.BClog)
exp(auto.BClog$coefficients)

# Assessing the model
log.probs <- predict(auto.BClog, newdata=BClog.test, type="response")
log.pred <- ifelse(log.probs > 0.5, 1, 0)
table(log.pred, BClog.test$TARGET_FLAG)
mean(log.pred == BClog.test$TARGET_FLAG)

# Generating the Confusion Matrix and metrics
confusionMatrix(data = as.factor(log.pred),
                reference = as.factor(BClog.test$TARGET_FLAG),
                positive = "1")
# Classification Error Rate
1-.7798 # 0.2202
# F1 Score 2*Percision*Sensitivity/percision + sensitivity
(2*0.6884*0.399)/(0.6884 + 0.399) #0.5051
# Generating the ROC curve and the AUC
roc_log <- roc(response = BClog.test$TARGET_FLAG,
               predictor = log.probs)
auc(roc_log) #0.8106
plot(roc_log, legacy.axes = TRUE) # Not included in the report

# This model is slightly better to what we found without the transformation of the variables and we
# will elect to transform the original model.

#### Logistic Model 3 - Reduced from the first Model ####
# Fitting the model after removing the less significant variables.
# -YOJ, -KIDS, -AGE, -carage, -sex,
auto2.log <- glm(TARGET_FLAG ~ . - SEX - CAR_AGE - KIDS - RED_CAR - YOJ,
                 data = BClog.train,
                 family = binomial(link = "logit"))

summary(auto2.log)
logLik(auto2.log) # -2217.936 on 24 df

# Logit model odds ratios
coefficients(auto2.log)
exp(auto2.log$coefficients)

# Assessing the model
```

```r
log.probs <- predict(auto2.log, newdata=BClog.test, type="response")
log.pred <- ifelse(log.probs > 0.5, 1, 0)
table(log.pred, log.test$TARGET_FLAG)
mean(log.pred == log.test$TARGET_FLAG)

# Generating the Confusion Matrix and metrics
confusionMatrix(data = as.factor(log.pred),
                reference = as.factor(BClog.test$TARGET_FLAG),
                positive = "1")
# Classification Error Rate
1-.7798 # 0.2202
# F1 Score 2*Percision*Sensitivity/percision + sensitivity
(2*0.6884*0.399)/(0.6884 + 0.399) #0.5052
# Generating the ROC curve and the AUC
roc_log <- roc(response = BClog.test$TARGET_FLAG,
               predictor = log.probs)
roc_log <- roc(response = BClog.test$TARGET_FLAG,
               predictor = log.probs)
auc(roc_log) #0.8111
plot(roc_log, legacy.axes = TRUE) # Not included in the report

# The reduced model performs in a very similar fashion to the full model and can be prefered for
# simplicity of the model.

#############################################################################################
#### Predicting the costs #####
auto.eval_lm <- auto.eval
auto.eval_lm$TARGET_AMT <- log(auto.eval_lm$TARGET_AMT)
auto.eval_lm$AGE <- sqrt(auto.eval_lm$AGE)
auto.eval_lm$YOJ <- (auto.eval_lm$YOJ)^2
auto.eval_lm$INCOME <- sqrt(auto.eval_lm$INCOME)
auto.eval_lm$TRAVTIME <- sqrt(auto.eval_lm$TRAVTIME)
auto.eval_lm$BLUEBOOK <- log(auto.eval_lm$BLUEBOOK)
auto.eval_lm$TIF <- log(auto.eval_lm$TIF)
auto.eval_lm$OLDCLAIM <- sqrt(auto.eval_lm$OLDCLAIM)
auto.eval_lm$CLM_FREQ <- sqrt(auto.eval_lm$CLM_FREQ)
auto.eval_lm$MVR_PTS <- sqrt(auto.eval_lm$MVR_PTS)
auto.eval_lm$CAR_AGE <- sqrt(auto.eval_lm$CAR_AGE)

repairs <- predict(autoBC_ridgefit, auto.eval_lm)
head(exp(repairs))

#### Predicting the Probability of an accident ####

auto.probs <- predict(auto2.log, newdata=auto.eval_lm, type = "response")
auto.pred  <- ifelse(auto.probs > 0.5, 1, 0)
head(auto.probs, 10)
head(auto.pred)

# Generating the table of predicted values.
table(auto.pred)
table(auto.pred)/sum(table(auto.pred))
```