Predict 411 Unit2 Insurance

Jennifer Wanat

Contents

Introduction 1
Data 1
Section 1: Data Exploration 1
Section 2: Data Preparation 5
Section 2.1: Data Transformations 6
Section 2.2: Outlier Transformation 6
Section 3: Models 6
Section 3.1: Model 1 for TARGET_FLAG 6
Section 3.2: Model 2 for TARGET_FLAG 9
Section 3.3: Model 3 for TARGET_FLAG 11
Section 3.4: Model for TARGET_AMT 13
Section 4: Model Selection 14
Conclusion 15
References 15

Introduction

In this project our objective is to be able to provide a generalized linear regression model (GLM) to predict a car crash for a customer at an auto insurance company. The data set contains records of customers including aspects related to income, car information, education and job. In order to build a model to provide a prediction of a car crash, an exploratory data analysis (EDA) of the data set was completed prior to this analysis, which allowed for an evaluation and selection of the most promising predictor variables.

The generated model summaries and parameters were provided, and assessed for predictive accuracy by examing model selection criteria. Then a model was selected, and used to score new data to predict the outcome of a car crash. The analysis was conducted with the R programming language.

Data

The data was entered in an Excel spreadsheet and saved as a comma separated value file (.csv). The .csv file was opened in R and saved into a data frame called `data`. The data frame contains 8161 observations of 26 variables for customers at an auto insurance company. Refer to the data dictionary for variable definition and theoretical effect (positive or negative impact) on car crashes. The variable *INDEX* is an identification variable and was not used for modeling purposes. An EDA was conducted on the remaining 25 variables.

A second data set was entered in an Excel spreadsheet and saved as a comma separated value file (.csv). The .csv file was opened in R and saved into a data frame called `test`. The `test` data frame contains 2141 observations of the same 26 variables for customers at an auto insurance company. The `test` set does not contain values for two variables, TARGET_FLAG and TARGET_AMT. The best model selected from the `data` set will be used to score the `test` data.

Section 1: Data Exploration

Some of the variables in the data set were viewed as integers instead of factors. Prior to EDA, instructions in R were specified for categorical variables to ensure the variables were understood correctly. Variables that

Table 1: Statistical Summary of Variables.

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
TARGET_AMT	1	8161	1504.3	4704.0	0	593.7	0.0	0	107586.1	107586.1	8.7	112.3	52.1
AGE	2	8155	44.8	8.6	45	44.8	8.9	16	81.0	65.0	0.0	-0.1	0.1
YOJ	3	7707	10.5	4.1	11	11.1	3.0	0	23.0	23.0	-1.2	1.2	0.0
INCOME	4	7716	61898.1	47572.7	54028	56841.0	41792.3	0	367030.0	367030.0	1.2	2.1	541.6
HOME_VAL	5	7697	154867.3	129123.8	161160	144032.1	147867.1	0	885282.0	885282.0	0.5	0.0	1471.8
TRAVTIME	6	8161	33.5	15.9	33	33.0	16.3	5	142.0	137.0	0.4	0.7	0.2
BLUEBOOK	7	8161	15709.9	8419.7	14440	15036.9	8450.8	1500	69740.0	68240.0	0.8	0.8	93.2
TIF	8	8161	5.4	4.1	4	4.8	4.4	1	25.0	24.0	0.9	0.4	0.0
OLDCLAIM	9	8161	4037.1	8777.1	0	1719.3	0.0	0	57037.0	57037.0	3.1	9.9	97.2
CLM_FREQ	10	8161	0.8	1.2	0	0.6	0.0	0	5.0	5.0	1.2	0.3	0.0
MVR_PTS	11	8161	1.7	2.1	1	1.3	1.5	0	13.0	13.0	1.3	1.4	0.0
CAR_AGE	12	7651	8.3	5.7	8	8.0	7.4	-3	28.0	31.0	0.3	-0.7	0.1

recorded dollar amounts were specified to keep only the numerical information (dollar signs and commas were removed). The factor level labels for URBANICITY were shortened from “Highly Urban/ Urban” and “z_Highly Rural/ Rural” to “Urban” and “Rural”, respectively.

A basic statistical summary of the variables was prepared. The descriptive summary includes the number of observations (n), mean, standard deviation (sd), median, trimmed mean (trimmed), median absolute deviation from the mean (mad), minimum value (min), maximum value (max), the difference between the minimum and maximum values (range), skewness (skew), kurtosis, and standard error (se). Five variables were noted to have missing values (NA), as determined by a n value less than 8161, and were examined further. These variables were: AGE, YOJ (years on job), INCOME, HOME_VAL (home value), and CAR_AGE (vehicle age). Categorical or binary (yes/no) were not included for this analysis.

An indicator variable for missing values in the INCOME (NA_INCOME) and JOB (NA_JOB) variables was created for use as a possible predictor for modeling purposes. An indicator variable was also created to signify if there were any children driving in a household (DO_KIDS_DRIVE).

The number and percentage of missing values for these variables is examined in the table below. All variables have 6.25% or less missing values. The missing values will need to be imputed in order to build a logistic regression model. This will be discussed in Section 2.

The quantiles were examined for variables that contained missing values, and for TARGET_AMT. There is a large difference between the 99th and 100th percent quantiles.

Table 2: Percentage of Missing Values for Data Set Variables.

	Number of missing values	Percent of missing values
AGE	6	0.07
YOJ	454	5.56
CAR_AGE	510	6.25
HOME_VAL	464	5.69
INCOME	445	5.45

Table 3: Percentage of Missing Values for Test Set Variables.

	Number of missing values	Percent of missing values
AGE	1	0.05
YOJ	94	4.39
CAR_AGE	129	6.03
HOME_VAL	111	5.18
INCOME	125	5.84

Table 4: Quantiles of Variables of Interest.

	0%	1%	5%	10%	25%	50%	75%	90%	95%	99%	100%
AGE	16	25	30	34	39	45	51	56	59	64.00	81.0
YOJ	0	0	0	5	9	11	13	15	15	17.00	23.0
TRAVTIME	5	5	7	13	22	33	44	54	60	75.00	142.0
TIF	1	1	1	1	1	4	7	11	13	17.00	25.0
CAR_AGE	-3	1	1	1	1	8	12	16	18	21.00	28.0
TARGET_AMT	0	0	0	0	0	0	1036	4904	6452	19831.02	107586.1

[a] 0% = minimum, 25% = Q1, 50% = median, 75% = Q3, 100% = maximum

The variables TRAVTIME and BLUEBOOk were examined by histogram and boxplot. Both of these variables had a right skewed distribution and contained values that were zero. A transformation would help normalize the distribution, although a `log()` tranformation would not be appropriate due to the values of zero.

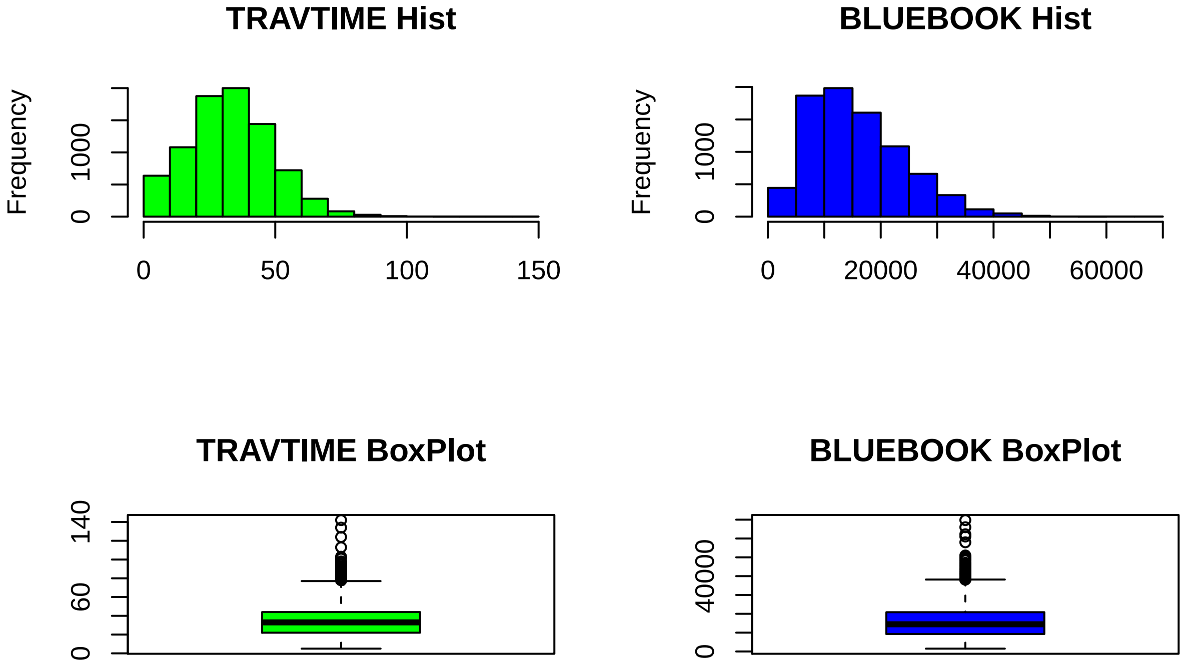


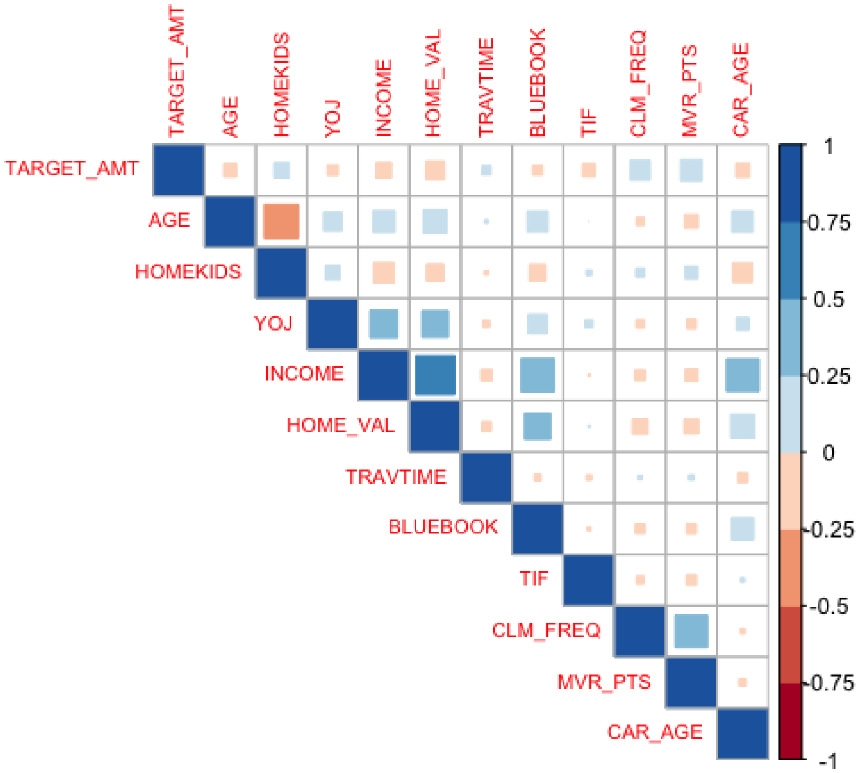
Figure 1: Visual Examination of TRAVTIME and BLUEBOOK.

The correlation matrix was constructed into a graphical display. The correlation coefficient is proportional to the color and size of the square. Positive correlation is blue and negative correlation is red. The large dark blue square visualized diagonally across the plot represents a correlation of 1, and it is the correlation of the TARGET_AMT against itself.

It was noted that HOME_VAL and INCOME had a strong positive correlation with each other. BLUEBOOK and CAR_AGE also had a positive correlation with INCOME. HOMEKIDS (the number of children at home) had a strong negative correlation with AGE. Models generated with these variable will be monitored for multicollinearity.

The variance inflation factor (VIF) estimates how much the variance of a predictor is inflated and a high value reflects multicollinearity. A predictor with high correlation to many predictors will have a high VIF value and conversely, a predictor with low correlation to many predictors will have a low VIF value.

Predictors with strong correlations with each other would result in a high variance inflation factor (VIF) and contribute to model instability. If any of the predictors have more than one degree of freedom (Df), then the generalized variance inflation factor (GVIF) is calculated. Small values of VIF/GVIF are preferred. The VIF/GVIF values will be examined with each model output.



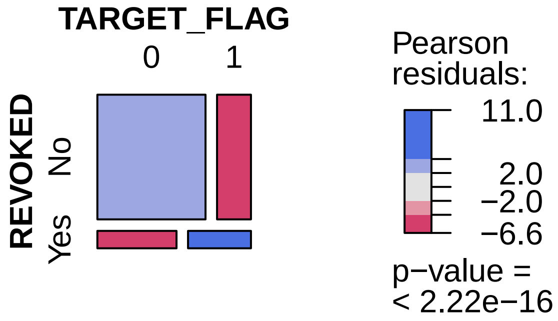
A mosaic plot is a graphical representation of contigency table, and plots were generated with various predictors and the TARGET_FLAG variable. The mosaic plots are composed of bars, where the width and height of the bars represent the relative frequency of the compaired variables. Color is added to the bars to represent the sign and absolute size of each residual from a fitted model. A residual value colored blue indicates that there are a higher frequency of observations than expected with the assumption of an independent model, while a residual value colored red indicates that there are a lower frequency of observations than expected.

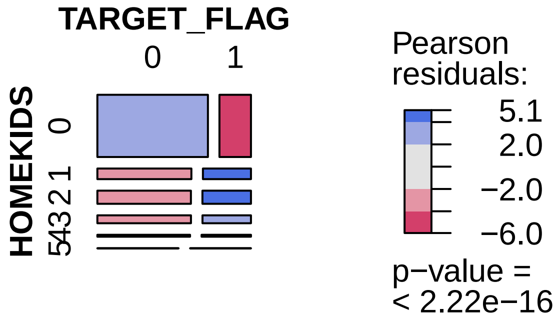
The mosaic plots (data not shown, available in code) for SEX (gender) and RED_CAR indicated that these variables had no impact and did not contribute positively or negatively towards the TARGET_FLAG response for a car crash. Models would not be created with these two variables.

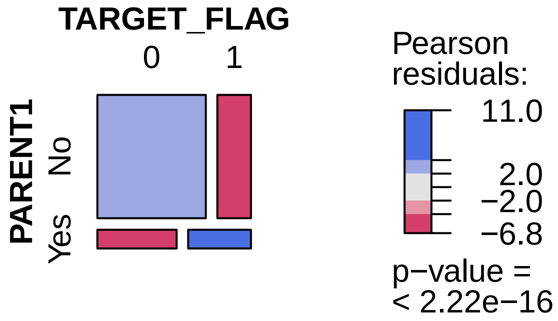
Variables that contributed to a car crash included: a higher number of motor vehicle record points (MVR_PTS), less time as an insurance customer (TIF, time in force), a higher number of claims in past five years (CLM_FREQ), a revoked licence in the past seven years (REVOKED), children at home (HOMEKIDS), being a single parent (PARENT1), living in an urban area (URBANICITY), commercial vice private use of car (CAR_USE), and not being married (MSTATUS).

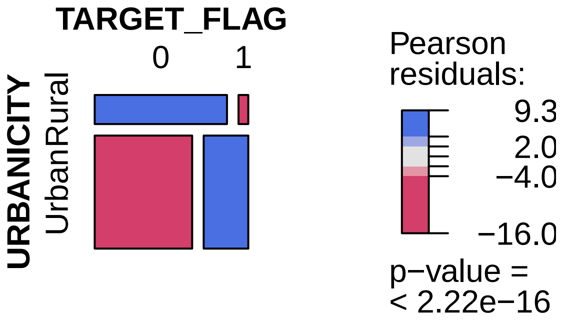
Blue collar and student job categories (JOB), and high school or less education categories (EDUCATION) also contributed to a car crash. A minivan was less likely to be in a car crash but a sports car was more likely to be in a car crash (CAR_TYPE).

Mosaic plots for REVOKED, HOMEKIDS, PARENT1 and URBANICITY are included.









Section 2: Data Preparation

Before creating a predictive model, the data needed to be prepared. Missing values would need to be imputed, and variables with outliers would need a type of transformation in order to reduce the effect of the outlier on the model accuracy. After data preparation has been completed, then predictive models may be generated.

Section 2.1: Data Transformations

Variables with missing values were imputed with the `na.aggregate()` function in R. The `na.aggregate` function will fill in the missing values utilizing the mean value for the variables. The mean value was calculated at the aggregate level based upon categorical levels of another variable. Missing values were calculated for YOJ by JOB, INCOME by JOB, HOME_VAL by JOB, and CAR_AGE by CAR_TYPE. The exception was for the variable AGE, as the mean value was calculated considering all observations within the variable. There were only six missing values for AGE, and the mean (44.8) and median (45) are nearly identical, which indicated a normal distribution.

Based upon the EDA, additional indicator variables were created to reflect whether an individual was a home owner (HOME_OWNER), had a college eduation (EDUCATION_COLLEGE), an old claim within the past five years greater than $15,000 (OLDCLAIM_GR15000), the presence of an old claim within the past five years (OLDCLAIM_YES), any claim frequency within the past five years (CLM_FREQ_NOT_ZERO), the car type was a minivan (CAR_TYPE_MINIVAN), a blue collar job (JOB_BLUE_COLLAR), or a student (JOB_STUDENT). Due to the presence of zero values and the observed skewed distribution for both TRAVTIME and BLUEBOOK variables, a square root transformation of these variables was completed (SQRT_TRAVTIME and SQRT_BLUEBOOK).

INCOME was binned into five different levels based upon the provided amount. Missing INCOME would be specified as "NA" and a zero INCOME would be specified as "Zero". INCOME between $1 and $30,000 would be specified as "Low", between $30,000 and $80,000 would be specified as "Medium", and greater than $80,000 would be specified as "High".

All data transformations applied to the `data` set were also applied to the `test` set.

Section 2.2: Outlier Transformation

During the EDA, it was observed that CAR_AGE had negative values for the vehicle age. All negative values were changed to zero. The TARGET_AMT variable was selected to undergo a truncation transformation. This variable was selected based upon the data distribution observed in the EDA, and was truncated to the 99% quantile. All outlier transformations applied to the `data` set were also applied to the `test` set.

Section 3: Models

Utilizing the data set, three models were created with predictor variables selected from EDA. The three models generated were by manual selection. Models were assessed with various metrics in Section 4.

Section 3.1: Model 1 for TARGET_FLAG

A logistic regression model was created using selected variables from the `data` set. The following model was generated:

$$\begin{aligned}E(TARGETFLAG) = -2.4690792 - 0.0039001(AGE) - 0.0059574(SQRTBLUEBOOK)+\\ 0.0149101(TRAVTIME) + 0.2106650(KIDSDRIV) + 2.4069289(URBANICITY)+\\ 0.3848345(DOKIDSDRIVE) + 0.1482239(CLMFREQ) + 0.7335438(REVOKEDYes)+\\ 0.1080602(MVRPTS) - 0.0011960(CARAGE) - 0.0552656(TIF) - 0.3555840(EDUCATIONBachelors)-\\ 0.2674665(EDUCATIONMasters) - 0.2081238(EDUCATIONPhD) + 0.0268848(EDUCATIONHighSchool)+\\ 0.5287490(MSTATUSNo) + 0.3913958(PARENT1Yes) - 0.7607992(CARUSEPrivate)+\\ 0.6079957(CARTYPEPanelTruck) + 0.5600394(CARTYPEPickup) + 0.9496163(CARTYPESportsCar)+\\ 0.6684223(CARTYPEVan) + 0.7216991(CARTYPESUV) + 0.0183420(YOJ) + 0.3794691(JOBClerical)-\\ 0.3836033(JOBDoctor) + 0.0098874(JOBHomemaker) + 0.1278668(JOBLawyer)-\\ 0.5456012(JOBManager) + 0.1595035(JOBProfessional) + 0.0342360(JOBStudent)+\\ 0.3029168(JOBBlueCollar) - 0.7640751(INCOMEbinLow) - 0.8667903(INCOMEbinMedium)-\\ 1.2195809(INCOMEbinHigh) - 0.0000012(HOMEVAL)\end{aligned}$$

Table 5: Model 1 Summary.

	Min	1Q	Median	3Q	Max
Deviance Residuals	-2.4858	-0.7165	-0.3994	0.6228	3.1437

term	estimate	std.error	statistic	p.value
(Intercept)	-2.4690792	0.3537335	-6.9800543	0.0000000
AGE	-0.0039001	0.0037414	-1.0424257	0.2972144
SQRT_BLUEBOOK	-0.0059574	0.0011075	-5.3790108	0.0000001
TRAVTIME	0.0149101	0.0018870	7.9016461	0.0000000
KIDSDRIV	0.2106650	0.1218891	1.7283324	0.0839286
URBANICITYUrban	2.4069289	0.1131070	21.2801046	0.0000000
DO_KIDS_DRIVE1	0.3848345	0.1951836	1.9716540	0.0486491
CLM_FREQ	0.1482239	0.0255508	5.8011487	0.0000000
REVOKEDYes	0.7335438	0.0804995	9.1124041	0.0000000
MVR_PTS	0.1080602	0.0135973	7.9471863	0.0000000
CAR_AGE	-0.0011960	0.0075597	-0.1582006	0.8742987
TIF	-0.0552656	0.0073497	-7.5194300	0.0000000
EDUCATIONBachelors	-0.3555840	0.1181387	-3.0098869	0.0026135
EDUCATIONMasters	-0.2674665	0.1803131	-1.4833451	0.1379828
EDUCATIONPhD	-0.2081238	0.2109474	-0.9866147	0.3238316
EDUCATIONz_High School	0.0268848	0.0973661	0.2761206	0.7824555
MSTATUSz_No	0.5287490	0.0813572	6.4991039	0.0000000
PARENT1Yes	0.3913958	0.1005263	3.8934661	0.0000988
CAR_USEPrivate	-0.7607992	0.0919927	-8.2702095	0.0000000
CAR_TYPEPanel Truck	0.6079957	0.1480734	4.1060422	0.0000402
CAR_TYPEPickup	0.5600394	0.1008465	5.5533822	0.0000000
CAR_TYPESports Car	0.9496163	0.1083098	8.7675970	0.0000000
CAR_TYPEVan	0.6684223	0.1224264	5.4597907	0.0000000
CAR_TYPEz_SUV	0.7216991	0.0861080	8.3813257	0.0000000
YOJ	0.0183420	0.0111316	1.6477433	0.0994054
JOBClerical	0.3794691	0.1970881	1.9253777	0.0541821
JOBDoctor	-0.3836033	0.2672054	-1.4356123	0.1511127
JOBHome Maker	0.0098874	0.2210313	0.0447332	0.9643199

term	estimate	std.error	statistic	p.value
JOBLawyer	0.1278668	0.1693562	0.7550172	0.4502387
JOBManager	-0.5456012	0.1712082	-3.1867703	0.0014387
JOBProfessional	0.1595035	0.1785102	0.8935259	0.3715756
JOBStudent	0.0342360	0.2257281	0.1516690	0.8794480
JOBz_Blue Collar	0.3029168	0.1857545	1.6307374	0.1029457
INCOME_binLow	-0.7640751	0.1779292	-4.2942658	0.0000175
INCOME_binMedium	-0.8667903	0.1992994	-4.3491871	0.0000137
INCOME_binHigh	-1.2195809	0.2145170	-5.6852424	0.0000000
HOME_VAL	-0.0000012	0.0000003	-3.6279594	0.0002857

```
Null Deviance: 9418 on 8160 degrees of freedom
```

```
Residual Deviance: 7274.8 on 8124 degrees of freedom
```

	GVIF	Df	GVIF^(1/(2*Df))
AGE	1.26	1	1.12
SQRT_BLUEBOOK	1.67	1	1.29
TRAVTIME	1.04	1	1.02
KIDSDRIV	5.36	1	2.31
URBANICITY	1.15	1	1.07
DO_KIDS_DRIVE	5.46	1	2.34
CLM_FREQ	1.16	1	1.08
REVOKED	1.01	1	1.00
MVR_PTS	1.15	1	1.07
CAR_AGE	2.03	1	1.42
TIF	1.01	1	1.00
EDUCATION	10.80	4	1.35
MSTATUS	1.94	1	1.39
PARENT1	1.62	1	1.27
CAR_USE	2.46	1	1.57
CAR_TYPE	2.54	5	1.10
YOJ	2.49	1	1.58
JOB	35.12	8	1.25
INCOME_bin	8.05	3	1.42
HOME_VAL	1.88	1	1.37

The sign in front of the coefficient indicates whether it has a positive (+) or negative (-) effect on TARGET_FLAG.

Twenty predictors were included in the model and after all the factor levels were taken into account there were 36 predictors used in the model. All variables were significant at the 95th significance level (p-value) except for AGE, KIDSDRIV, CAR_AGE, all of the EDUCATION factors except for Bachelors, YOJ, and all of the JOB factors except for manager.

The variance inflation factor (GVIF) for each predictor variable was calculated. The GVIF is a measurement of collinearity between predictors. Small values of GVIF are preferred. There were high values of GVIF (greater than 3) for KIDSDRIV (5.36), DO_KIDS_DRIVE (5.36), EDUCATION (10.80), JOB (35.12), and INCOME_bin (8.05). The another model would be generated that would try to avoid these multicollinearity issues.

Section 3.2: Model 2 for TARGET_FLAG

A second logistic regression model was created using selected variables from the `data` set. The following model was generated:

$$E(TARGETFLAG) = -2.9930989 - 0.0073755(SQRTBLUEBOOK) + 0.1715590(SQRTTRAVTIME) + 0.6757376(DOKIDSDRIVEYes) + 2.2813270(URBANICITYUrban) + 0.1477370(CLMFREQ) + 0.7457224(REVOKEDYes) + 0.1180524(MVRPTS) - 0.0539494(TIF) - 0.6522806(EDUCATIONCOLLEGEYes) + 0.3737162(MSTATUSNo) + 0.4622121(PARENT1) - 0.8323661(CARUSEPrivate) + 0.5215327(CARTYPEPanelTruck) + 0.4987342(CARTYPEPickup) + 0.9425059(CARTYPESportsCar) + 0.6000715(CARTYPEVan) + 0.7221387(CARTYPESUV) - 0.0000020(HOMEVAL)$$

Table 8: Model 2 Summary.

	Min	1Q	Median	3Q	Max
Deviance Residuals	-2.4868	-0.7232	-0.4179	0.644	3.1631

term	estimate	std.error	statistic	p.value
(Intercept)	-2.9930989	0.2242267	-13.348541	0.00e+00
SQRT_BLUEBOOK	-0.0073755	0.0010655	-6.922356	0.00e+00
SQRT_TRAVTIME	0.1715590	0.0206407	8.311671	0.00e+00
DO_KIDS_DRIVE1	0.6757376	0.0873738	7.733872	0.00e+00
URBANICITYUrban	2.2813270	0.1112176	20.512281	0.00e+00
CLM_FREQ	0.1477370	0.0253081	5.837537	0.00e+00
REVOKEDYes	0.7457224	0.0795724	9.371626	0.00e+00
MVR_PTS	0.1180524	0.0134348	8.787086	0.00e+00
TIF	-0.0539494	0.0072771	-7.413543	0.00e+00
EDUCATION_COLLEGE1	-0.6522806	0.0630070	-10.352505	0.00e+00
MSTATUSz_No	0.3737162	0.0762084	4.903872	9.00e-07
PARENT1Yes	0.4622121	0.0939469	4.919932	9.00e-07
CAR_USEPrivate	-0.8323661	0.0695955	-11.960059	0.00e+00
CAR_TYPEPanel Truck	0.5215327	0.1367567	3.813579	1.37e-04
CAR_TYPEPickup	0.4987342	0.0972085	5.130560	3.00e-07
CAR_TYPESports Car	0.9425059	0.1059095	8.899162	0.00e+00
CAR_TYPEVan	0.6000715	0.1188640	5.048386	4.00e-07
CAR_TYPEz_SUV	0.7221387	0.0845727	8.538675	0.00e+00
HOME_VAL	-0.0000020	0.0000003	-6.932515	0.00e+00

```
Null Deviance: 9418 on 8160 degrees of freedom
```

```
Residual Deviance: 7389 on 8142 degrees of freedom
```

	GVIF	Df	GVIF^(1/(2*Df))
SQRT_BLUEBOOK	1.58	1	1.26
SQRT_TRAVTIME	1.03	1	1.01
DO_KIDS_DRIVE	1.11	1	1.05
URBANICITY	1.11	1	1.06

	GVIF	Df	GVIF^(1/(2*Df))
CLM_FREQ	1.16	1	1.08
REVOKED	1.00	1	1.00
MVR_PTS	1.15	1	1.07
TIF	1.01	1	1.00
EDUCATION_COLLEGE	1.19	1	1.09
MSTATUS	1.73	1	1.32
PARENT1	1.44	1	1.20
CAR_USE	1.43	1	1.19
CAR_TYPE	2.02	5	1.07
HOME_VAL	1.44	1	1.20

The sign in front of the coefficient indicates whether it has a positive (+) or negative (-) effect on TARGET_FLAG.

Fourteen predictors were included in the model and after all the factor levels were taken into account there were 18 predictors used in the model. All variables were significant at the 99.9th significance level.

The variance inflation factor (GVIF) for each predictor variable was calculated. The GVIF is a measurement of collinearity between predictors. Small values of GVIF are preferred. There were no high values of GVIF (greater than 3) for Model 2. The highest observed GVIF was for MSTATUS (1.32).

Section 3.3: Model 3 for TARGET_FLAG

A third logistic regression model was created using selected variables from the `data` set. The following model was generated:

$$
\begin{aligned}
E(TARGETFLAG) = -1.9721759 + 0.5705656(DOKIDSDRIVE) - 0.3626029(KIDSNo) - \\
0.0000010(HOMEVAL) + 0.6780504(MSTATUS) - 0.5088680(EDUCATIONCOLLEGE) - \\
0.7476867(CARUSEPrivate) - 0.0540670(TIF) - 0.6635262(CARTYPEMINIVAN) + \\
0.1459388(CLMFREQ) + 0.7529170(REVOKEDYes) + 0.1146596(MVRPTS) + \\
2.3244532(URBANICITYUrban) + 0.1744749(SQRTTRAVTIME) - \\
0.0065280(SQRTBLUEBOOK) - 0.4765991(INCOMEbinLow) - \\
0.5426896(INCOMEbinMedium) - 0.9856646(INCOMEbinHigh)
\end{aligned}
$$

Table 11: Model 3 Summary.

	Min	1Q	Median	3Q	Max
Deviance Residuals	-2.472	-0.7226	-0.4073	0.6366	3.1491

term	estimate	std.error	statistic	p.value
(Intercept)	-1.9721759	0.2173091	-9.075442	0.0000000
DO_KIDS_DRIVE1	0.5705656	0.0951154	5.998666	0.0000000
KIDSZ_No	-0.3626029	0.0689822	-5.256473	0.0000001
HOME_VAL	-0.0000010	0.0000003	-3.191520	0.0014153
MSTATUSz_No	0.6780504	0.0697759	9.717547	0.0000000
EDUCATION_COLLEGE1	-0.5088680	0.0680933	-7.473095	0.0000000
CAR_USEPrivate	-0.7476867	0.0624601	-11.970628	0.0000000
TIF	-0.0540670	0.0072964	-7.410133	0.0000000
CAR_TYPE_MINIVAN1	-0.6635262	0.0737413	-8.998026	0.0000000
CLM_FREQ	0.1459388	0.0253439	5.758336	0.0000000
REVOKEDYes	0.7529170	0.0798137	9.433427	0.0000000
MVR_PTS	0.1146596	0.0134899	8.499682	0.0000000
URBANICITYUrban	2.3244532	0.1117981	20.791526	0.0000000
SQRT_TRAVTIME	0.1744749	0.0206869	8.434064	0.0000000
SQRT_BLUEBOOK	-0.0065280	0.0009429	-6.923470	0.0000000
INCOME_binLow	-0.4765991	0.1192523	-3.996560	0.0000643
INCOME_binMedium	-0.5426896	0.1149616	-4.720618	0.0000024
INCOME_binHigh	-0.9856646	0.1388045	-7.101102	0.0000000

```
Null Deviance: 9418 on 8160 degrees of freedom

Residual Deviance: 7349.8 on 8143 degrees of freedom
```

	GVIF	Df	GVIF^(1/(2*Df))
DO_KIDS_DRIVE	1.32	1	1.15
KIDS	1.36	1	1.17
HOME_VAL	1.76	1	1.33
MSTATUS	1.45	1	1.20
EDUCATION_COLLEGE	1.39	1	1.18

	GVIF	Df	GVIF^(1/(2*Df))
CAR_USE	1.14	1	1.07
TIF	1.01	1	1.00
CAR_TYPE_MINIVAN	1.04	1	1.02
CLM_FREQ	1.16	1	1.08
REVOKED	1.00	1	1.00
MVR_PTS	1.15	1	1.07
URBANICITY	1.13	1	1.06
SQRT_TRAVTIME	1.03	1	1.01
SQRT_BLUEBOOK	1.23	1	1.11
INCOME_bin	1.88	3	1.11

The sign in front of the coefficient indicates whether it has a positive (+) or negative (-) effect on TARGET_FLAG.

Fifteen predictors were included in the model and after all the factor levels were taken into account there were 17 predictors used in the model. All variables were significant at the 99.9th significance level exept for HOME_VAL, which was significant at the 95th significance level.

The variance inflation factor (GVIF) for each predictor variable was calculated. The GVIF is a measurement of collinearity between predictors. Small values of GVIF are preferred. There were no high values of GVIF (greater than 3) for Model 3. The highest observed GVIF was for HOME_VAL (1.33).

Section 3.4: Model for TARGET_AMT

A single regression model was prepared for TARGET_AMT, which is dollar amount paid if an insurance customer was in a car crash. The ordinary least squares (OLS) regression model was created using selected variables from the `data` set. The following model was generated:

$$\begin{aligned} E(TARGETAMT) = -506.3266 + 5020.7681(TARGETFLAG1) + \\ 3.8390(SQRTBLUEBOOK) + 24.4535(MVRPTS) \end{aligned}$$

Table 14: TARGET_AMT model summary.

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
Residuals	-5077	-164	-17	0	111	15042

term	estimate	std.error	statistic	p.value
(Intercept)	-506.326634	90.0899979	-5.620231	0.0000000
TARGET_FLAG1	5020.768097	54.3097510	92.446900	0.0000000
SQRT_BLUEBOOK	3.839009	0.6856149	5.599366	0.0000000
MVR_PTS	24.453456	11.0871602	2.205565	0.0274422

Residual Standard Error	Adjusted R-squared	MultipleR-squared	F-Statistic
2097.9	0.5269	0.527	3029.79

Term	VIF
TARGET_FLAG	1.06
SQRT_BLUEBOOK	1.01
MVR_PTS	1.05

The sign in front of the coefficient indicates whether it has a positive (+) or negative (-) effect on TARGET_AMT. Variable effects were as expected, as the presence of a TARGET_FLAG or an increasing SQRT_BLUEBOOK amount would increase the amount paid in a car crash.

Three predictors were included in the model and all variables were significant at the 99.9th significance level, except for MVR_PTS which was significant at the 95th significance level. The multiple R-squared value is 0.527. This indicates that 52.7% of the variation in carryover can be explained by the independent variables (predictors) in the regression model. R-squared has a monotonic relationship with model parameters. To account for this relationship, the adjusted R-squared will penalize models that have too many unimportant predictor variables and will allow models of different sizes to be compared. The adjusted R-squared value is 0.5269. The adjusted R-square value will always be lower than the multiple R-squared value.

The residual standard error of 2097.9 is the average error in predicting TARGET_AMT from the predictors in this model. The global F-statistic is 3029.79, and tests whether the predictors as a group predict the TARGET_AMT above chance levels.

The variance inflation factor (VIF) for each predictor variable was calculated. The VIF is a measurement of collinearity between predictors. Small values of VIF are preferred. The VIF values were low (1.06 or less) for all predictors.

Section 4: Model Selection

For each of the logistic models the deviance, log likelihood, Akaike's Information Criterion (AIC), and Bayesian Information Criterion (BIC) were calculated. These metrics are used to evaluate the model fit. A small value of AIC and BIC is desired. The deviance is equal to -2 times the log likelihood of the model. Smaller values of deviance are preferred.

	null.deviance	df.null	logLik	AIC	BIC	deviance	df.residual
Model 1	9417.962	8160	-3637.391	7348.782	7608.046	7274.782	8124
Model 2	9417.962	8160	-3694.500	7427.001	7560.136	7389.001	8142
Model 3	9417.962	8160	-3674.904	7385.808	7511.936	7349.808	8143

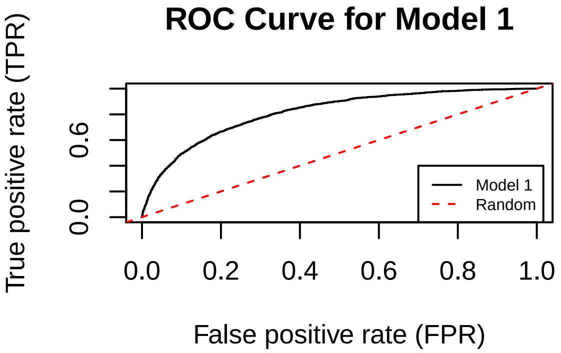
The MSE measures the average of the squares of the residuals, and the MAE measures the average of the absolute value of the residuals. A smaller value of the MSE and MAE is preferred.

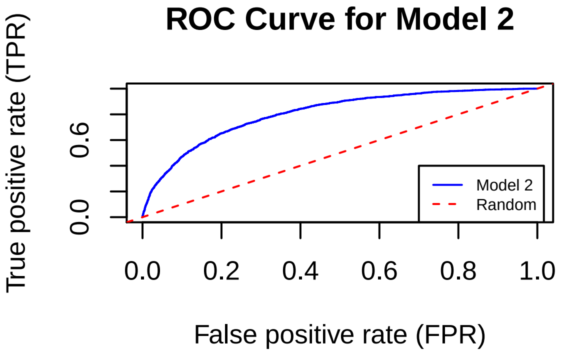
The Kolmogorov-Smirnov (KS) statisitic is the maximum difference between the cumulative true positve rate and the cumulative false positive rate. The KS statisitic can range from zero to 1. Larger values of the KS statistic are preferred.

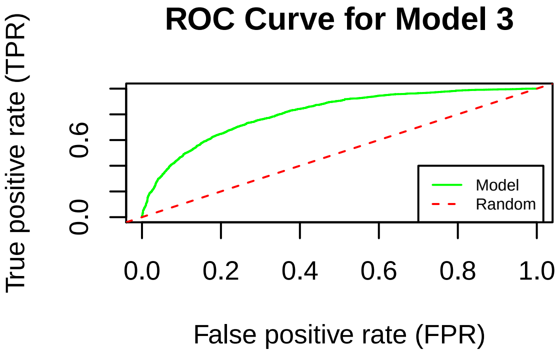
The Somer's D statistic has values that can range from -1 to 1, and is a measure of concordant (C), discordant (D), and tie (T) pairs. A value of 1 indicates perfect accuracy and a value of -1 indicates that there are no pair matches. Larger values of the Somer's D are preferred.

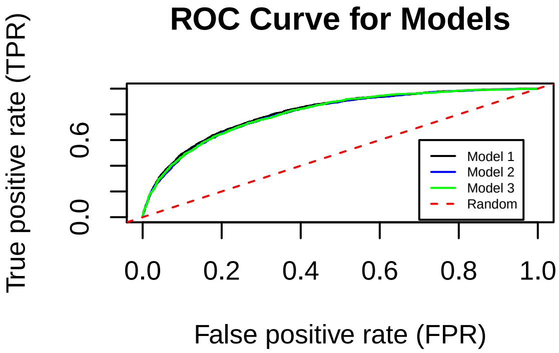
The Receiver Operating Characteristic (ROC) curve graphically demonstrates model performance for different probability cutoffs. The area under the ROC curve (AUROC) indicates the sensitivity of the model to distinguish true positives (sensitivity) from false positives (1-specificity). The AUROC has values that can range from zero to 1. A larger value of AUROC is preferred.

	MSE	MAE	KS Stat	Somers D	AUROC
Model 1	15.02239	2.012712	0.4706	0.6300	0.8147
Model 2	14.96228	2.013328	0.4586	0.6146	0.8069
Model 3	15.80288	2.014858	0.4605	0.6195	0.8095









Conclusion

Model performance was nearly identical for all models. Model 1 criteria for logLik, AIC, deviance, MAE, KS stat, Somer's D, and AUROC were the best out of all three models. Model 2 had the best MSE. Model 3 had the best BIC. Model 3 performed better than Model 2 for logLik, AIC, BIC, deviance, KS stat, Somer's D, and AUROC. Model 3 had fewer predictors than Model 1.

Model 3 would be used to predict the TARGET_FLAG probability in the `test` set. Model 4 would be used to predict the TARGET_AMT in the `test` set.

References

Fox, J. and Weisberg, S. (2011). An R Companion to Applied Regression, Second Edition. Thousand Oaks, CA: Sage Publications, Inc.

Kabacoff, R. (2015). R in Action, Second Edition. Shelter Island, NY: Manning Publications Co.

Lander, J. (2014). R for Everyone. Upper Saddle River, NJ: Addison-Wesley.

Pardoe, I. (2012). Applied Regression Modeling, Second Edition. Hoboken, NJ: John Wiley & Sons, Inc.

Stowell, S. (2014). Using R for Statistics. New York, NY: Apress.

Zeileis, A., Meyer, D., Hornik, K. (2007). Residual-Based Shadings for Visualizing (Conditional) Independence. Journal of Computational and Graphical Statistics, 16(3), 507 - 525.

##Code

```
knitr::opts_chunk$set(echo = FALSE, fig.pos = 'h')
knitr::opts_chunk$set(dev = 'pdf')
library(knitr)

# Note, some of these libraries are not needed for this template code.
library(readr)
library(dplyr)
library(zoo)
library(psych)
library(ROCR)
library(corrplot)
library(car)
library(InformationValue)
library(rJava)
library(pbkrtest)
library(leaps)
library(MASS)
library(glm2)
library(aod)
library(vcd)
library(gridExtra)
library(mice)
library(RColorBrewer)
library(kableExtra)
library(broom)
library(xtable)
library(xlsxjars)
library(xlsx)



# Data Import and Variable Type Changes
setwd("~/Desktop/R/")
data <- read.csv("logit_insurance.csv")
test <- read.csv("logit_insurance_test.csv")

summary(data)
summary(test)

str(data)
str(test)

######################## Data Exploration ############################
### Need to make sure our data is understood correctly by R,
###since we have a mix of numerical and categorical

data$INDEX <- as.factor(data$INDEX)
data$TARGET_FLAG <- as.factor(data$TARGET_FLAG)
data$SEX <- as.factor(data$SEX)
data$EDUCATION <- as.factor(data$EDUCATION)
data$PARENT1 <- as.factor(data$PARENT1)
data$INCOME <- suppressWarnings(as.numeric(gsub("[^0-9.]", "", data$INCOME)))
```

```
#removes $ and , keeps numbers and . adds NA for empty, and keeps zeros
data$HOME_VAL <- suppressWarnings(as.numeric(gsub("[^0-9.]", "", data$HOME_VAL)))
#removes $ and , keeps numbers and . adds NA for empty, and keeps zeros
data$MSTATUS <- as.factor(data$MSTATUS)
data$REVOKED <- as.factor(data$REVOKED)
data$RED_CAR <- as.factor(ifelse(data$RED_CAR=="yes", 1, 0))
data$KIDS <- as.factor(ifelse(data$HOMEKIDS==0, "Z_No", "Yes"))
data$URBANICITY <- ifelse(data$URBANICITY == "Highly Urban/ Urban", "Urban", "Rural")
data$URBANICITY <- as.factor(data$URBANICITY)
#definitely need the second line for URBANICITY as factor
data$JOB <- as.factor(data$JOB)
data$CAR_USE <- as.factor(data$CAR_USE)
data$CAR_TYPE <- as.factor(data$CAR_TYPE)
data$DO_KIDS_DRIVE <- as.factor(ifelse(data$KIDSDRIV > 0, 1, 0 ))
#need this otherwise it is viewed as integer
data$OLDCLAIM <- suppressWarnings(as.numeric(gsub("[^0-9.]", "", data$OLDCLAIM)))
#removes $ and , keeps numbers and . adds NA for empty, and keeps zeros
data$BLUEBOOK <- suppressWarnings(as.numeric(gsub("[^0-9.]", "", data$BLUEBOOK)))
#removes $ and , keeps numbers and . adds NA for empty, and keeps zeros
data$NA_INCOME <- ifelse(is.na(data$INCOME), 1, 0)
data$NA_JOB <- ifelse(data$JOB=="", 1, 0)

summary(data)

######## Same treatment on test data set ###########################
### Need to make sure our data is understood correctly by R,
### since we have a mix of numerical and categorical

test$INDEX <- as.factor(test$INDEX)
test$TARGET_FLAG <- as.factor(test$TARGET_FLAG)
test$SEX <- as.factor(test$SEX)
test$EDUCATION <- as.factor(test$EDUCATION)
test$PARENT1 <- as.factor(test$PARENT1)
test$INCOME <- suppressWarnings(as.numeric(gsub("[^0-9.]", "", test$INCOME)))
test$HOME_VAL <- suppressWarnings(as.numeric(gsub("[^0-9.]", "", test$HOME_VAL)))
test$MSTATUS <- as.factor(test$MSTATUS)
test$REVOKED <- as.factor(test$REVOKED)
test$RED_CAR <- as.factor(ifelse(test$RED_CAR=="yes", 1, 0))
test$KIDS <- as.factor(ifelse(test$HOMEKIDS==0, "Z_No", "Yes"))
test$URBANICITY <- ifelse(test$URBANICITY == "Highly Urban/ Urban", "Urban", "Rural")
test$URBANICITY <- as.factor(test$URBANICITY)
test$JOB <- as.factor(test$JOB)
test$CAR_USE <- as.factor(test$CAR_USE)
test$CAR_TYPE <- as.factor(test$CAR_TYPE)
test$DO_KIDS_DRIVE <- as.factor(ifelse(test$KIDSDRIV > 0, 1, 0 ))
test$OLDCLAIM <- suppressWarnings(as.numeric(gsub("[^0-9.]", "", test$OLDCLAIM)))
test$BLUEBOOK <- suppressWarnings(as.numeric(gsub("[^0-9.]", "", test$BLUEBOOK)))
test$NA_INCOME <- ifelse(is.na(test$INCOME), 1, 0)
test$NA_JOB <- ifelse(test$JOB=="", 1, 0)

summary(test)
```

```
###################################
#table of n, mean, sd, median, trimmed, mad, min, max, range, skew, kurtosis, se
#from psych package
#all data
#kable(round(describe(data[5,7,8,10,15,17,18,21,25]), 1),

data.describe <- data[-c(1,2,4,6,9,11,12,13,14,16,19,20,23,26:30)]
kable(round(describe(data.describe), 1),
      caption = "Statistical Summary of Variables.",
      format = "latex", booktabs = T ) %>%
  kable_styling(latex_options = c("striped", "scale_down"))


#grid.table(round(describe(data.describe), 1))

##### Summary statistics of variables by TARGET_FLAG classification
#subset data by Target Flag value 0 or 1
data_target1 <- subset(data, TARGET_FLAG==1)
data_target0 <- subset(data, TARGET_FLAG==0)

#table of n, mean, sd, median, trimmed, mad, min, max, range, skew, kurtosis, se
#from psych package
#data with Target Flag value equal to 0
data0 <- data_target0[,c(5,7,15,18,25)]
kable(round(describe(data0), 1),
      caption = "Statistical Summary of Variables.",
      format = "latex", booktabs = T ) %>%
  kable_styling(latex_options = c("striped", "scale_down"))

#table of n, mean, sd, median, trimmed, mad, min, max, range, skew, kurtosis, se
#from psych package
#data with Target Flag value equal to 1
data1 <- data_target1[,c(5,7,15,18,25)]
kable(round(describe(data1), 1),
      caption = "Statistical Summary of Variables.",
      format = "latex", booktabs = T ) %>%
  kable_styling(latex_options = c("striped", "scale_down"))


#function to describe variables with missing data
myNA.variable.summary <- function(variable){
  NAsum.summary <- sum(is.na(variable))
  NAmean.summary <- (round(mean(is.na(variable)), 4)*100)
  return(c(NAsum.summary, NAmean.summary))
}

#table of data variables with missing data
NA.age <- myNA.variable.summary(data$AGE)
NA.yoj <- myNA.variable.summary(data$YOJ)
NA.car_age <- myNA.variable.summary(data$CAR_AGE)
NA.home_val <- myNA.variable.summary(data$HOME_VAL)
NA.income <- myNA.variable.summary(data$INCOME)
```

```r
overview.NA <- rbind(NA.age, NA.yoj, NA.car_age, NA.home_val, NA.income)
colnames(overview.NA) <- c("Number of missing values", "Percent of missing values")
rownames(overview.NA) <- c("AGE", "YOJ", "CAR_AGE", "HOME_VAL", "INCOME")
kable(overview.NA, caption = 'Percentage of Missing Values for Data Set Variables.')

#table of test variables with missing data
NA.tage <- myNA.variable.summary(test$AGE)
NA.tyoj <- myNA.variable.summary(test$YOJ)
NA.tcar_age <- myNA.variable.summary(test$CAR_AGE)
NA.thome_val <- myNA.variable.summary(test$HOME_VAL)
NA.tincome <- myNA.variable.summary(test$INCOME)

test.overview.NA <- rbind(NA.tage, NA.tyoj, NA.tcar_age, NA.thome_val, NA.tincome)
colnames(test.overview.NA) <- c("Number of missing values", "Percent of missing values")
rownames(test.overview.NA) <- c("AGE", "YOJ", "CAR_AGE", "HOME_VAL", "INCOME")
kable(test.overview.NA, caption = 'Percentage of Missing Values for Test Set Variables.')


# Histograms for Numeric Variables

par(mfrow=c(2,2))
hist(data$AGE, col = "green", xlab = "AGE", main = "AGE Hist")
hist(data$YOJ, col = "blue", xlab = "YOJ", main = "YOJ Hist")
boxplot(data$AGE, col = "green", main = "AGE BoxPlot")
boxplot(data$YOJ, col = "blue", main = "YOJ BoxPlot")
par(mfrow=c(1,1))

par(mfrow=c(2,2))
hist(data$TARGET_AMT, col = "green", xlab = "TARGET_AMT", main = "TARGET_AMT Hist")
hist((data$TIF), col = "blue", xlab = "TIF", main = "TIF Hist")
boxplot(data$TARGET_AMT, col = "green", main = "TARGET_AMT BoxPlot")
boxplot(data$TIF, col = "blue", main = "TIF BoxPlot")
par(mfrow=c(1,1))
#Target amount has a large number of zero values

par(mfrow=c(2,2))
hist(data$MVR_PTS, col = "red", xlab = "MVR_PTS", main = "MVR_PTS Hist")
hist(data$CAR_AGE, col = "blue", xlab = "CAR_AGE", main = "CAR_AGE Hist")
boxplot(data$MVR_PTS, col = "red", main = "MVR_PTS BoxPlot")
boxplot(data$CAR_AGE, col = "blue",main = "CAR_AGE BoxPlot")
par(mfrow=c(1,1))

par(mfrow=c(2,2))
hist(data$CLM_FREQ, col = "red", xlab = "CLM_FREQ", main = "CLM_FREQ Hist")
hist(data$OLDCLAIM, col = "blue", xlab = "OLDCLAIM", main = "OLDCLAIM Hist")
boxplot(data$CLM_FREQ, col = "red", main = "CLM_FREQ BoxPlot")
boxplot(data$OLDCLAIM, col = "blue",main = "OLDCLAIM BoxPlot")
par(mfrow=c(1,1))

par(mfrow=c(2,2))
hist(data$HOME_VAL, col = "red", xlab = "HOME_VAL", main = "HOME_VAL Hist")
hist(data$INCOME, col = "blue", xlab = "INCOME", main = "INCOME Hist")
boxplot(data$HOME_VAL, col = "red", main = "HOME_VAL BoxPlot")
boxplot(data$INCOME, col = "blue",main = "INCOME BoxPlot")
par(mfrow=c(1,1))

#This is better with the sqrt of INCOME
#cant do a straight natural log of home value due to zero values, would need a constant added first
#took the natural log of HOME_VAL and  sqrt of INCOME
#I don't like natural log of home value
#home value includes $0 which indicates renting

#Exploring data transformations conducted below
plot(data$CAR_AGE, data$OLDCLAIM)
boxplot(data$AGE~data$TARGET_FLAG)
plot(data$AGE, data$OLDCLAIM)
abline(a=15000, b=0, col = c("red"), lty = 2)
boxplot(data$CAR_AGE~data$CAR_TYPE)
boxplot(data$HOME_VAL~data$JOB, las=2)
boxplot(data$INCOME~data$JOB, las=2)
boxplot(data$YOJ~data$JOB, las=2)

#histogram and boxplot of TRAVTIME and BLUEBOOK
par(mfrow=c(2,2))
hist(data$TRAVTIME, col = "green", xlab = "", main = "TRAVTIME Hist")
hist(data$BLUEBOOK, col = "blue", xlab = "", main = "BLUEBOOK Hist")
boxplot(data$TRAVTIME, col = "green", main = "TRAVTIME BoxPlot")
boxplot(data$BLUEBOOK, col = "blue", main = "BLUEBOOK BoxPlot")
par(mfrow=c(1,1))
#due to the zero values and skewed distribution,
#these variable would benefit from a sqrt transformation


#quantile calculations
#quantile(x,  probs = c(0.1, 0.5, 1, 2, 5, 10, 50, NA)/100)
quantile.age <- quantile(data$AGE,
                         probs = c(0,1,5,10,25,50,75,90,95,99,100)/100, na.rm=TRUE)
quantile.yoj <- quantile(data$YOJ,
                        probs = c(0,1,5,10,25,50,75,90,95,99,100)/100, na.rm=TRUE)
quantile.travtime <- quantile(data$TRAVTIME,
                        probs = c(0,1,5,10,25,50,75,90,95,99,100)/100, na.rm=TRUE)
quantile.tif <- quantile(data$TIF,
                         probs = c(0,1,5,10,25,50,75,90,95,99,100)/100, na.rm=TRUE)
quantile.car_age <- quantile(data$CAR_AGE,
                         probs = c(0,1,5,10,25,50,75,90,95,99,100)/100, na.rm=TRUE)
quantile.target_amt <- quantile(data$TARGET_AMT,
                         probs = c(0,1,5,10,25,50,75,90,95,99,100)/100, na.rm=TRUE)
overview.quantile <- rbind(quantile.age,quantile.yoj,quantile.travtime,quantile.tif,
                           quantile.car_age, quantile.target_amt)
colnames(overview.quantile) <- c("0%","1%","5%","10%","25%","50%",
                                 "75%", "90%", "95%", "99%", "100%")
rownames(overview.quantile) <- c("AGE", "YOJ", "TRAVTIME",
                                 "TIF", "CAR_AGE", "TARGET_AMT")

kable(overview.quantile, caption = 'Quantiles of Variables of Interest.',
      format = "latex", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "scale_down")) %>%
  add_footnote(c("0% = minimum, 25% = Q1, 50% = median, 75% = Q3, 100% = maximum"))

#consider travtime to 99% (75 minutes)
#car_age minimum is less than zero, correct this to reflect 1% value (1)


#list 10 maximum values of variables
max.age <- head(sort(data$AGE, decreasing = TRUE), 10)
max.yoj <- head(sort(data$YOJ, decreasing = TRUE), 10)
max.travtime <- head(sort(data$TRAVTIME, decreasing = TRUE), 10)
max.tif <- head(sort(data$TIF, decreasing = TRUE), 10)
max.car_age <- head(sort(data$CAR_AGE, decreasing = TRUE), 10)
max.target_amt <- head(sort(data$TARGET_AMT, decreasing = TRUE), 10)
overview.max <- rbind(max.age, max.yoj, max.travtime, max.tif, max.car_age, max.target_amt)
rownames(overview.max) <- c("AGE", "YOJ", "TRAVTIME",
                            "TIF", "CAR_AGE", "TARGET_AMT")
kable(overview.max, caption = 'List of the Ten Highest Maximum Values.')

#list 10 minimum values of variables
min.age <- tail(sort(data$AGE, decreasing = TRUE), 10)
min.yoj <- tail(sort(data$YOJ, decreasing = TRUE), 10)
min.travtime <- tail(sort(data$TRAVTIME, decreasing = TRUE), 10)
min.tif <- tail(sort(data$TIF, decreasing = TRUE), 10)
min.car_age <- tail(sort(data$CAR_AGE, decreasing = TRUE), 10)
min.target_amt <- tail(sort(data$TARGET_AMT, decreasing = TRUE), 10)
overview.min <- rbind(min.age, min.yoj, min.travtime, min.tif, min.car_age, max.target_amt)
rownames(overview.min) <- c("AGE", "YOJ", "TRAVTIME",
                            "TIF", "CAR_AGE", "TARGET_AMT")
kable(overview.min, caption = 'List of the Ten Lowest Minimum Values.')


numeric <- subset(data, select = c(TARGET_AMT, AGE, HOMEKIDS, YOJ,
                                   INCOME, HOME_VAL, TRAVTIME, BLUEBOOK, TIF,
                                   CLM_FREQ, MVR_PTS, CAR_AGE), na.rm = TRUE)
c <- cor(numeric)

corrplot::corrplot(c, method = "square",
         col=brewer.pal(n=8, name="RdBu"),
         diag = TRUE, tl.cex = 0.7, number.cex = 0.8,
         type= "upper")

#income, home value, bluebook, and car age are positively correlated with each other

################## Mosaic Plots ###############################
# These plots indicated that the variable did not contribute to TARGET_FLAG
mosaic(~SEX+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE)
#Sex (MorF) does not make a difference in crash
mosaic(~RED_CAR+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE)
#red car does not make a difference in crash

# These plots indicated that the variable did contibute to TARGET_FLAG
mosaic(~JOB+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE,
       labeling_arg = list(abbreviate = c(JOB = TRUE)))
#blue collar and student more likely to crash
mosaic(~MVR_PTS+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE)
#the more tickets more likely to crash
mosaic(~TIF+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE)
#the less time a customer (1 year) more likely to crash
mosaic(~CLM_FREQ+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE)
#greater than zero more likely to crash
mosaic(~EDUCATION+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE,
       labeling_arg = list(abbreviate = c(EDUCATION = TRUE)))
#high school or less education more likely to crash
mosaic(~CAR_TYPE+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE)
#minivan less likely to crash
#sports car more likely to crash
mosaic(~CAR_USE+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE)
#most use car for private use. Commerical use more likely crash
mosaic(~MSTATUS+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE)
#not married more likely in crash

# Mosaic plots with multiple variables
mosaic(~EDUCATION+PARENT1+URBANICITY+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE)
mosaic(~PARENT1+URBANICITY+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE)
mosaic(~CAR_TYPE+TARGET_FLAG+PARENT1, data=data, shade=TRUE, legend=TRUE)

# Example mosaic plots
par(mfrow=c(2,2))
mosaic(~REVOKED+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE)
#revoked more likely to crash
mosaic(~HOMEKIDS+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE)
#no kids less likely to crash
par(mfrow=c(1,1))

par(mfrow=c(2,2))
mosaic(~PARENT1+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE)
#single parent more likely to crash
mosaic(~URBANICITY+TARGET_FLAG, data=data, shade=TRUE, legend=TRUE)
#urban is more likely crash
par(mfrow=c(1,1))


########### Part 2: Data Transformation ##################
# Fix NA's, note car age
#There are NAs in age, YOJ, income, home_val, car_age

data$AGE[is.na(data$AGE)] <- mean(data$AGE, na.rm = "TRUE")
#there are only 6 NA's in age, so mean imputation is OK
data$YOJ <- na.aggregate(data$YOJ, data$JOB, mean, na.rm = TRUE)
#Homemaker and Student mean are much different than other jobs
data$INCOME <- na.aggregate(data$INCOME, data$JOB, mean, na.rm = TRUE)
data$HOME_VAL <- na.aggregate(data$HOME_VAL, data$JOB, mean, na.rm = TRUE )
data$CAR_AGE <- na.aggregate(data$CAR_AGE, data$CAR_TYPE, mean, na.rm = TRUE)
data$CAR_AGE[data$CAR_AGE < 0 ] <- 0
#fixes car age values less than zero
```

```
data$HOME_OWNER <- ifelse(data$HOME_VAL == 0, 0, 1)
data$HOME_OWNER <- as.factor(data$HOME_OWNER)
data$EDUCATION_COLLEGE <- ifelse((data$EDUCATION == c('z_High School')|
                                    (data$EDUCATION == c('<High School'))), 0,1)
data$EDUCATION_COLLEGE <- as.factor(data$EDUCATION_COLLEGE)
data$OLDCLAIM_GR15000 <- ifelse(data$OLDCLAIM < 15000, 0,1)
data$OLDCLAIM_GR15000 <- as.factor(data$OLDCLAIM_GR15000)
data$OLDCLAIM_YES <- ifelse(data$OLDCLAIM > 0, 1,0)
data$OLDCLAIM_YES <- as.factor(data$OLDCLAIM_YES)
data$CLM_FREQ_NOT_ZERO <- ifelse(data$CLM_FREQ > 0, 1,0)
data$CLM_FREQ_NOT_ZERO <- as.factor(data$CLM_FREQ_NOT_ZERO)
data$JOB_BLUE_COLLAR <- ifelse(data$JOB == c('z_Blue Collar'), 1,0)
data$JOB_BLUE_COLLAR <- as.factor(data$JOB_BLUE_COLLAR)
data$JOB_STUDENT <- ifelse(data$JOB == c('Student'), 1,0)
data$JOB_STUDENT <- as.factor(data$JOB_STUDENT)
data$CAR_TYPE_MINIVAN <- ifelse(data$CAR_TYPE == c('Minivan'), 1,0)
data$CAR_TYPE_MINIVAN <- as.factor(data$CAR_TYPE_MINIVAN)

data$SQRT_TRAVTIME <- sqrt(data$TRAVTIME)
data$SQRT_BLUEBOOK <- sqrt(data$BLUEBOOK)
data$TARGET_AMT[(data$TARGET_AMT > 19831)] = 19831

#cant do natural log of home value due to zero values, would need to add a constant first
#cant do natural log of income due to zero values, would need to add a constant first
#I dont think income and home value will be used in the model
#these variables are reflected in other categories

# Bin Income
data$INCOME_bin[is.na(data$INCOME)] <- "NA"
data$INCOME_bin[data$INCOME == 0] <- "Zero"
data$INCOME_bin[data$INCOME >= 1 & data$INCOME < 30000] <- "Low"
data$INCOME_bin[data$INCOME >= 30000 & data$INCOME < 80000] <- "Medium"
data$INCOME_bin[data$INCOME >= 80000] <- "High"
data$INCOME_bin <- factor(data$INCOME_bin)
data$INCOME_bin <- factor(data$INCOME_bin, levels=c("NA","Zero","Low","Medium","High"))

##### Must conduct the same tranformations to the test data set #################

test$AGE[is.na(test$AGE)] <- mean(test$AGE, na.rm = "TRUE")
test$YOJ <- na.aggregate(test$YOJ, test$JOB, mean, na.rm = TRUE)
test$INCOME <- na.aggregate(test$INCOME, test$JOB, mean, na.rm = TRUE)
test$HOME_VAL <- na.aggregate(test$HOME_VAL, test$JOB, mean, na.rm = TRUE )
test$CAR_AGE <- na.aggregate(test$CAR_AGE, test$CAR_TYPE, mean, na.rm = TRUE)
test$CAR_AGE[test$CAR_AGE < 0 ] <- 0
test$HOME_OWNER <- ifelse(test$HOME_VAL == 0, 0, 1)
test$HOME_OWNER <- as.factor(test$HOME_OWNER)
test$EDUCATION_COLLEGE <- ifelse((test$EDUCATION == c('z_High School')|
                                    (test$EDUCATION == c('<High School'))), 0,1)
test$EDUCATION_COLLEGE <- as.factor(test$EDUCATION_COLLEGE)
test$OLDCLAIM_GR15000 <- ifelse(test$OLDCLAIM < 15000, 0,1)
test$OLDCLAIM_GR15000 <- as.factor(test$OLDCLAIM_GR15000)
test$OLDCLAIM_YES <- ifelse(test$OLDCLAIM > 0, 1,0)
test$OLDCLAIM_YES <- as.factor(test$OLDCLAIM_YES)
```

```
test$CLM_FREQ_NOT_ZERO <- ifelse(test$CLM_FREQ > 0, 1,0)
test$CLM_FREQ_NOT_ZERO <- as.factor(test$CLM_FREQ_NOT_ZERO)
test$JOB_BLUE_COLLAR <- ifelse(test$JOB == c('z_Blue Collar'), 1,0)
test$JOB_BLUE_COLLAR <- as.factor(test$JOB_BLUE_COLLAR)
test$JOB_STUDENT <- ifelse(test$JOB == c('Student'), 1,0)
test$JOB_STUDENT <- as.factor(test$JOB_STUDENT)
test$CAR_TYPE_MINIVAN <- ifelse(test$CAR_TYPE == c('Minivan'), 1,0)
test$CAR_TYPE_MINIVAN <- as.factor(test$CAR_TYPE_MINIVAN)

test$SQRT_TRAVTIME <- sqrt(test$TRAVTIME)
test$SQRT_BLUEBOOK <- sqrt(test$BLUEBOOK)
data$TARGET_AMT[(data$TARGET_AMT > 19831)] = 19831

# Bin Income
test$INCOME_bin[is.na(test$INCOME)] <- "NA"
test$INCOME_bin[test$INCOME == 0] <- "Zero"
test$INCOME_bin[test$INCOME >= 1 & test$INCOME < 30000] <- "Low"
test$INCOME_bin[test$INCOME >= 30000 & test$INCOME < 80000] <- "Medium"
test$INCOME_bin[test$INCOME >= 80000] <- "High"
test$INCOME_bin <- factor(test$INCOME_bin)
test$INCOME_bin <- factor(test$INCOME_bin, levels=c("NA","Zero","Low","Medium","High"))



#################### Part 3: Model Creation ############################################
#Function for Mean Square Error Calculation
mse <- function(sm)
  mean(sm$residuals^2)

#Function for Mean Absolute Error Calculation
mae <- function(sm)
  mean(abs(sm$residuals))

############# Part 3: TARGET_FLAG ######################
############## Model 1 #################
#Model Development for TARGET_FLAG
# USE THIS ONE
Model1 <- glm(TARGET_FLAG ~ AGE + SQRT_BLUEBOOK + TRAVTIME +
                KIDSDRIV + URBANICITY + DO_KIDS_DRIVE +
                CLM_FREQ + REVOKED + MVR_PTS + CAR_AGE +
                TIF + EDUCATION + MSTATUS + PARENT1 +
                CAR_USE + CAR_TYPE + YOJ + JOB + INCOME_bin + HOME_VAL,
              data = data, family = binomial())
summary(Model1)
data$Model1Prediction <- predict(Model1, type = "response")
#for the most part, JOB does not help the model
#education, age, car_age does not help; consider removing from next model
vif(Model1)



########################### Model 1: Statistics #######################################
```

```
#Function for model summary statistics
model.summary <- function(model){
  nulldeviance.summary <- round(summary(model)$null.deviance,1)
  nd_df.summary <- round(summary(model)$df.null,0)
  residualdeviance.summary <- round(summary(model)$deviance,1)
  rd_df.summary <- round(summary(model)$df.residual,0)
  return(c(nulldeviance.summary, nd_df.summary, residualdeviance.summary, rd_df.summary))
}

#Model residual summary table
Model1.residuals <- t(round(fivenum(resid(Model1)), 4))
rownames(Model1.residuals) <- c("Deviance Residuals")
colnames(Model1.residuals) <- c("Min", "1Q", "Median", "3Q", "Max")
kable(Model1.residuals, caption = 'Model 1 Summary.')

#Model Summary table of coefficients
Model1.alternate <- tidy(Model1)
kable(Model1.alternate)

#Model Deviance information
Model1.summary <- t(model.summary(Model1))
cat("Null Deviance:", Model1.summary[1], "on", Model1.summary[2], "degrees of freedom")
cat("Residual Deviance:", Model1.summary[3], "on", Model1.summary[4], "degrees of freedom")


#Table of model VIF
Model1.vif <- round(vif(Model1), 2)
#Model1.vif.term <- names(Model1.vif)
#Model1.vif.numbers <- unname(Model1.vif)
#Model1.vif.final <- cbind(Model1.vif.term, Model1.vif.numbers)
#colnames(Model1.vif.final) <- c("Term", "GVIF", "Df", "GVIF^(1/(2*DF))")
kable(Model1.vif)


############## Model 2 #################
Model2 <- glm(TARGET_FLAG ~ SQRT_BLUEBOOK + SQRT_TRAVTIME +
                DO_KIDS_DRIVE +  URBANICITY +
                CLM_FREQ + REVOKED + MVR_PTS + TIF +
                EDUCATION_COLLEGE + MSTATUS + PARENT1 +
                CAR_USE + CAR_TYPE + HOME_VAL,
                data = data, family = binomial())
summary(Model2)
data$Model2Prediction <- predict(Model2, type = "response")
vif(Model2)


########################### Model 2: Statistics #######################################

#Model residual summary table
Model2.residuals <- t(round(fivenum(resid(Model2)), 4))
rownames(Model2.residuals) <- c("Deviance Residuals")
colnames(Model2.residuals) <- c("Min", "1Q", "Median", "3Q", "Max")
kable(Model2.residuals, caption = 'Model 2 Summary.')
```

```r
#Model Summary table of coefficients
Model2.alternate <- tidy(Model2)
kable(Model2.alternate)

#Model Deviance information
Model2.summary <- t(model.summary(Model2))
cat("Null Deviance:", Model2.summary[1], "on", Model2.summary[2], "degrees of freedom")
cat("Residual Deviance:", Model2.summary[3], "on", Model2.summary[4], "degrees of freedom")


#Table of model VIF
Model2.vif <- round(vif(Model2), 2)
kable(Model2.vif)


############## Model 3 #################
Model3 <- glm(TARGET_FLAG ~ DO_KIDS_DRIVE + KIDS +
                    HOME_VAL + MSTATUS + EDUCATION_COLLEGE +
                    CAR_USE + TIF + CAR_TYPE_MINIVAN +
                    CLM_FREQ + REVOKED +
                    MVR_PTS  + URBANICITY  + SQRT_TRAVTIME +
                    SQRT_BLUEBOOK + INCOME_bin,
                  data=data, family = binomial())
summary(Model3)
data$Model3Prediction <- predict(Model3, type = "response")
vif(Model3)


########################### Model 3: Statistics #######################################

#Model residual summary table
Model3.residuals <- t(round(fivenum(resid(Model3)), 4))
rownames(Model3.residuals) <- c("Deviance Residuals")
colnames(Model3.residuals) <- c("Min", "1Q", "Median", "3Q", "Max")
kable(Model3.residuals, caption = 'Model 3 Summary.')

#Model Summary table of coefficients
Model3.alternate <- tidy(Model3)
kable(Model3.alternate)

#Model Deviance information
Model3.summary <- t(model.summary(Model3))
cat("Null Deviance:", Model3.summary[1], "on", Model3.summary[2], "degrees of freedom")
cat("Residual Deviance:", Model3.summary[3], "on", Model3.summary[4], "degrees of freedom")


#Table of model VIF
Model3.vif <- round(vif(Model3), 2)
kable(Model3.vif)

summary(data)
############## Model for TARGET_AMT #################
Model4 <- lm(TARGET_AMT~TARGET_FLAG + SQRT_BLUEBOOK + MVR_PTS, data = data)
```

```r
summary(Model4)
vif(Model4)


############################ Model 4: Statistics ########################################

#Function for model summary statistics
model.summary <- function(model){
  residualse.summary <- round(summary(lm(model))$sigma,1)
  adjrs.summary <- round(summary(lm(model))$adj.r.squared,4)
  multrs.summary <- round(summary(lm(model))$r.squared,4)
  fstat.summary <- round(unname(summary(lm(model))$fstatistic)[1], 2)
  return(c(residualse.summary, adjrs.summary, multrs.summary, fstat.summary))
}

#Model residual summary table
Model4.residuals <- t(round(summary(Model4$residuals), 0))
rownames(Model4.residuals) <- c("Residuals")
kable(Model4.residuals, caption = 'TARGET_AMT model summary.')

#Model Summary table of coefficients
Model4.alternate <- tidy(Model4)
kable(Model4.alternate)

#Table of model ANOVA
Model4.summary <- t(model.summary(Model4))
colnames(Model4.summary) <- c('Residual Standard Error','Adjusted R-squared',
                              'MultipleR-squared','F-Statistic')
kable(Model4.summary)


#Table of model VIF
Model4.vif <- round(vif(Model4), 2)
Model4.vif.term <- names(Model4.vif)
Model4.vif.numbers <- unname(Model4.vif)
Model4.vif.final <- cbind(Model4.vif.term, Model4.vif.numbers)
colnames(Model4.vif.final) <- c("Term", "VIF")
kable(Model4.vif.final)


Model1.glance <- glance(Model1)
Model2.glance <- glance(Model2)
Model3.glance <- glance(Model3)
Model.glance <- rbind(Model1.glance, Model2.glance, Model3.glance)
rownames(Model.glance) <- c("Model 1", "Model 2", "Model 3")
kable(Model.glance)


Model1.mse <- mse(Model1)
Model2.mse <- mse(Model2)
Model3.mse <- mse(Model3)

Model1.mae <- mae(Model1)
```

```r
Model2.mae <- mae(Model2)
Model3.mae <- mae(Model3)

Model1.ks_stat <- ks_stat(actuals=data$TARGET_FLAG,
                          predictedScores=data$Model1Prediction)
Model2.ks_stat <- ks_stat(actuals=data$TARGET_FLAG,
                          predictedScores=data$Model2Prediction)
Model3.ks_stat <- ks_stat(actuals=data$TARGET_FLAG,
                          predictedScores=data$Model3Prediction)

Model1.somersD <- round(somersD(actuals=data$TARGET_FLAG,
                                predictedScores=data$Model1Prediction), 4)
Model2.somersD <- round(somersD(actuals=data$TARGET_FLAG,
                                predictedScores=data$Model2Prediction), 4)
Model3.somersD <- round(somersD(actuals=data$TARGET_FLAG,
                                predictedScores=data$Model3Prediction), 4)

Model1.AUROC <- round(InformationValue::AUROC(actuals=data$TARGET_FLAG,
                                              predictedScores=data$Model1Prediction), 4)
Model2.AUROC <- round(InformationValue::AUROC(actuals=data$TARGET_FLAG,
                                              predictedScores=data$Model2Prediction), 4)
Model3.AUROC <- round(InformationValue::AUROC(actuals=data$TARGET_FLAG,
                                              predictedScores=data$Model3Prediction), 4)

Model1.review <- cbind(Model1.mse, Model1.mae, Model1.ks_stat, Model1.somersD, Model1.AUROC)
Model2.review <- cbind(Model2.mse, Model2.mae, Model2.ks_stat, Model2.somersD, Model2.AUROC)
Model3.review <- cbind(Model3.mse, Model3.mae, Model3.ks_stat, Model3.somersD, Model3.AUROC)

Model.review <- rbind(Model1.review, Model2.review, Model3.review)
rownames(Model.review) <- c("Model 1", "Model 2", "Model 3")
colnames(Model.review) <- c("MSE", "MAE", "KS Stat", "Somers D", "AUROC")
kable(Model.review)


par(mfrow=c(2,2))
#ROC of Model 1
pred1 <- prediction(data$Model1Prediction, data$TARGET_FLAG)
perf1 <- performance(pred1, "tpr", "fpr")
plot(perf1, xlab = "False positive rate (FPR)", ylab = "True positive rate (TPR)",
     main = "ROC Curve for Model 1")
legend(0.7, 0.4, legend = c("Model 1", "Random"),
       lty = c("solid", "dashed"), col = c("black", "red"),
       cex = 0.6)
abline(a=0, b=1, col = c("red"), lty = 2)

#ROC of Model 2
pred2 <- prediction(data$Model2Prediction, data$TARGET_FLAG)
perf2 <- performance(pred2, "tpr", "fpr")
plot(perf2, xlab = "False positive rate (FPR)", ylab = "True positive rate (TPR)",
     main = "ROC Curve for Model 2", col= "blue")
legend(0.7, 0.4, legend = c("Model 2", "Random"),
       lty = c("solid", "dashed"), col = c("blue", "red"),
       cex = 0.6)
```

```
abline(a=0, b=1, col = c("red"), lty = 2)
par(mfrow=c(1,1))


par(mfrow=c(2,2))
#ROC of Model 3
pred3 <- prediction(data$Model3Prediction, data$TARGET_FLAG)
perf3 <- performance(pred3, "tpr", "fpr")
plot(perf3, xlab = "False positive rate (FPR)", ylab = "True positive rate (TPR)",
     main = "ROC Curve for Model 3", col = "green")
legend(0.7, 0.4, legend = c("Model", "Random"),
       lty = c("solid", "dashed"), col = c("green", "red"),
       cex = 0.6)
abline(a=0, b=1, col = c("red"), lty = 2)

#ROC of all models
plot(perf1@x.values[[1]], perf1@y.values[[1]], type='s', col = "black",
     xlab = "False positive rate (FPR)", ylab = "True positive rate (TPR)",
     main = "ROC Curve for Models")
lines(perf2@x.values[[1]], perf2@y.values[[1]], type='s', col = "blue")
lines(perf3@x.values[[1]], perf3@y.values[[1]], type='s', col = "green")
legend(0.7, 0.6, legend = c("Model 1", "Model 2", "Model 3", "Random"),
       lty = c("solid", "solid", "solid", "dashed"),
       col = c("black", "blue", "green", "red"),
       cex = 0.5)
abline(a=0, b=1, col = c("red"), lty = 2)
par(mfrow=c(1,1))


#### Part 5:  Score Model on Test Data set and output csv file

# Again, double-checking to make sure we don't have any NA's in our Test Data Set
summary(test)
#################### Score Test Data ##########################
##### Model coefficients used to create P_TARGET_FLAG,

test$P_TARGET_FLAG <- predict(Model3, newdata = test, type = "response")

test$P_TARGET_AMT <- -506.326634 +
  5020.768097*test$P_TARGET_FLAG +
  3.839009*test$SQRT_BLUEBOOK +
  24.453456*test$MVR_PTS

test$P_TARGET_AMT[test$P_TARGET_AMT < 0 ] <- 0

#Scored Data File
scores <- test[c("INDEX","P_TARGET_FLAG", "P_TARGET_AMT")]
write.xlsx(scores, file = "logit_insurance_test.xlsx",
           sheetName = "Scored Data File", col.names = TRUE)
```