AUTO INSURANCE CLAIMS

Predicting Accidents & Coverage Amounts

Rafal Decowski

CUNY | DATA MINING

Objective

The objective is to build a multiple and binary logistic regression models to predict whether a car insurance customer will be in a traffic incident and the monetary amount of the claim.

Contents

DATA EXPLORATION

Dataset

The dataset contains car and home insurance customer information. It has 8161 cases across 23 predictor variables and two response variables - TARGET_FLAG & TARGET_AMT. The predictor variables are of mixed types – from character to integers. There are 7213 complete cases which means we may need to do additional cleansing to either fill out the gaps or drop rows with missing data. The first response variable is a binary indication of whether the insured car was in an accident. The second variable is the claim amount – how much money the insurance company pay out to the customer.

VARIABLE NAME	*DEFINITION*	*THEORETICAL EFFECT*
INDEX	Identification Variable (do not use)	None
TARGET_FLAG	Was Car in a crash?	1=Yes O=No
TARGET_AMT	If car was in a crash, what was the cost	None
AGE	Age of Driver	Very young people tend to be risky. Maybe very old people also.
BLUEBOOK	Value of Vehicle	Unknown effect on probability of collision, but probably effect
CAR_AGE	Vehicle Age	Unknown effect on probability of collision, but probably effect the payout if there is a crash
CAR_TYPE	Type of Car	Unknown effect on probability of collision, but probably effect the payout if there is a crash
CAR_USE	Vehicle Use	Commercial vehicles are driven more, so might increase probability of collision
CLM_FREQ #	Claims (Past 5 Years)	The more claims you filed in the past, the more you are likely to file in the future
EDUCATION	Max Education Level	Unknown effect, but in theory more educated people tend to drive more safely
HOMEKIDS	# Children at Home	Unknown effect
HOME_VAL	Home Value	In theory, home owners tend to drive more responsibly
INCOME	Income	In theory, rich people tend to get into fewer crashes
JOB	Job Category	In theory, white collar jobs tend to be safer
KIDSDRIV	# Driving Children	When teenagers drive your car, you are more likely to get into crashes
MSTATUS	Marital Status	In theory, married people drive more safely
MVR_PTS	Motor Vehicle Record Points	If you get lots of traffic tickets, you tend to get into more crashes
OLDCLAIM	Total Claims (Past 5 Years)	If your total payout over the past five years was high, this suggests future payouts will be high
PARENT1	Single Parent	Unknown effect
RED_CAR	A Red Car	Urban legend says that red cars (especially red sports cars) are more risky. Is that true?
REVOKED	License Revoked (Past 7 Years)	If your license was revoked in the past 7 years, you probably are a more risky driver.
SEX	Gender	Urban legend says that women have less crashes then men. Is that true?
TIF	Time in Force	People who have been customers for a long time are usually more safe.
TRAVTIME	Distance to Work	Long drives to work usually suggest greater risk
URBANICITY	Home/Work Area	Unknown
YOJ	Years on Job	People who stay at a job for a long time are usually more safe

Descriptive Statistics

Descriptive statistics help us identify variations, ranges, distributions, missing values and more with a simple summary table. This will later help us drive decisions on transformations, normalizations and general data cleansing.

The table below tells me that there are some missing values as the *n* column contains different numbers. There seem to be some data errors such as *car_age* marked as negative 3.

	n	*mean*	*sd*	*min*	*max*	*range*	*se*	*IQR*
KIDSDRIV	8161	0.17	0.51	0	4	4	0.01	0
AGE	8155	44.79	8.63	16	81	65	0.1	12
HOMEKIDS	8161	0.72	1.12	0	5	5	0.01	1
YOJ	7707	10.5	4.09	0	23	23	0.05	4
TRAVTIME	8161	33.49	15.91	5	142	137	0.18	22
TIF	8161	5.35	4.15	1	25	24	0.05	6
CLM_FREQ	8161	0.8	1.16	0	5	5	0.01	2
MVR_PTS	8161	1.7	2.15	0	13	13	0.02	3
CAR_AGE	7651	8.33	5.7	-3	28	31	0.07	11

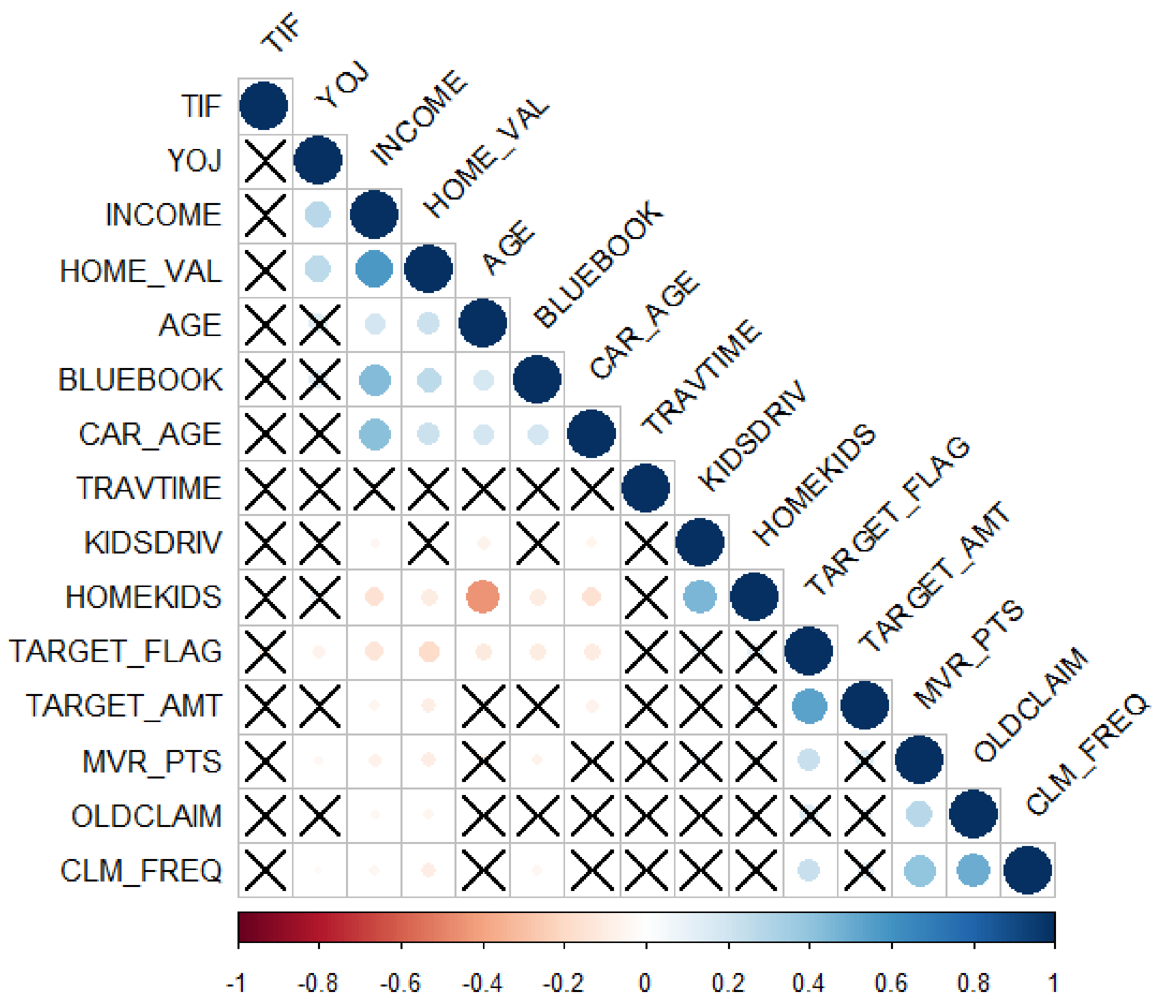
Correlation

The correlation helps us highlight predictor variables that have a strong relationship with the target variable. It helps us narrow down the important ones and discard the ones that do not significantly affect the prediction results.

TARGET_FLAG	
TARGET_AMT	0.836771
KIDSDRIV	0.092513
AGE	-0.11294
HOMEKIDS	0.114991
YOJ	-0.06575
INCOME	-0.13724
HOME_VAL	-0.18014
TRAVTIME	0.053116
BLUEBOOK	-0.10581
TIF	-0.07896
OLDCLAIM	0.140237
CLM_FREQ	0.222084
MVR_PTS	0.225479
CAR_AGE	-0.10625

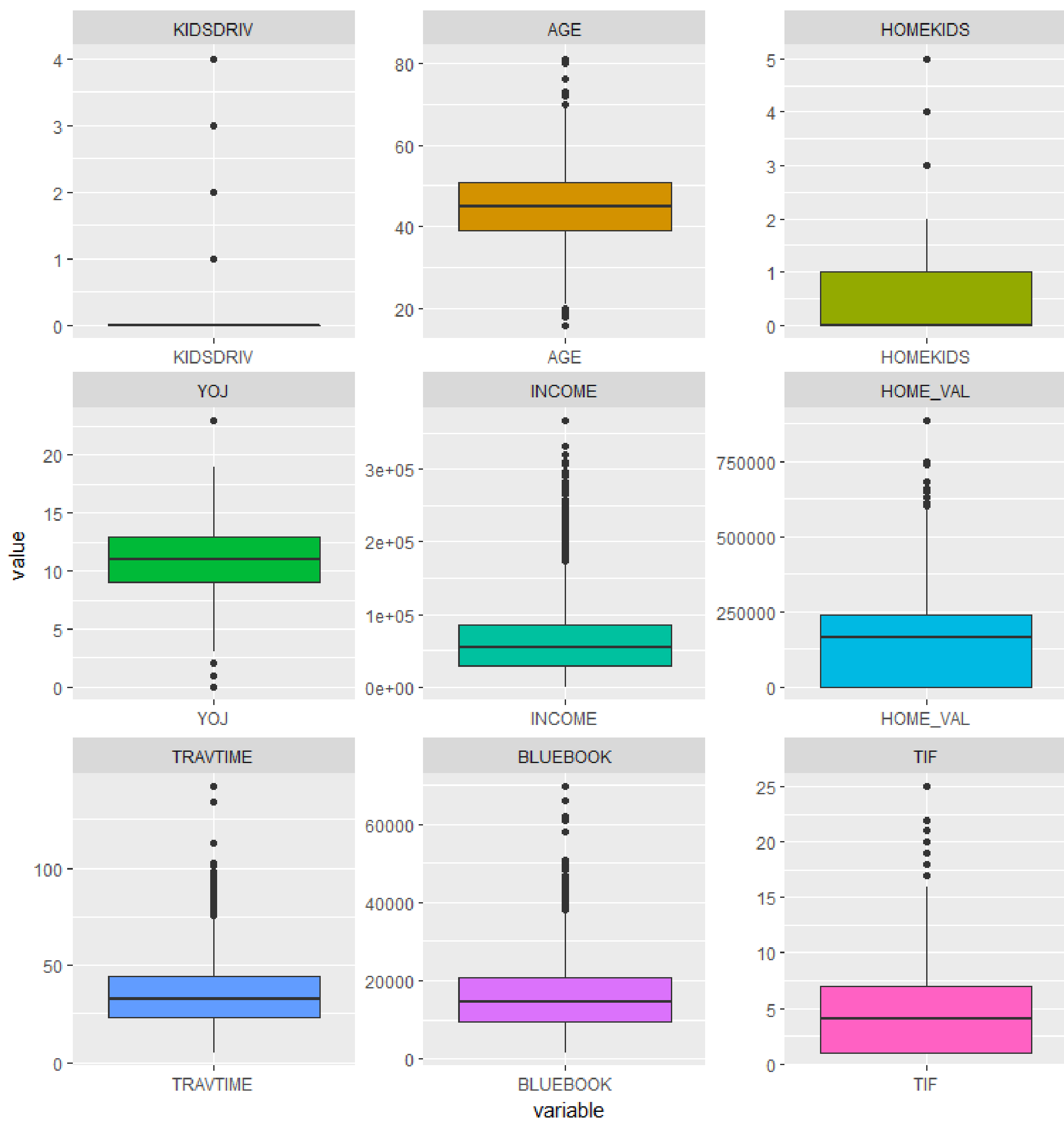
TARGET_AMT	
TARGET_FLAG	0.836771
KIDSDRIV	0.078971
AGE	-0.10123
HOMEKIDS	0.10973
YOJ	-0.05518
INCOME	-0.11989
HOME_VAL	-0.14136
TRAVTIME	0.044852
BLUEBOOK	-0.09564
TIF	-0.06494
OLDCLAIM	0.106371
CLM_FREQ	0.188975
MVR_PTS	0.201151
CAR_AGE	-0.09178

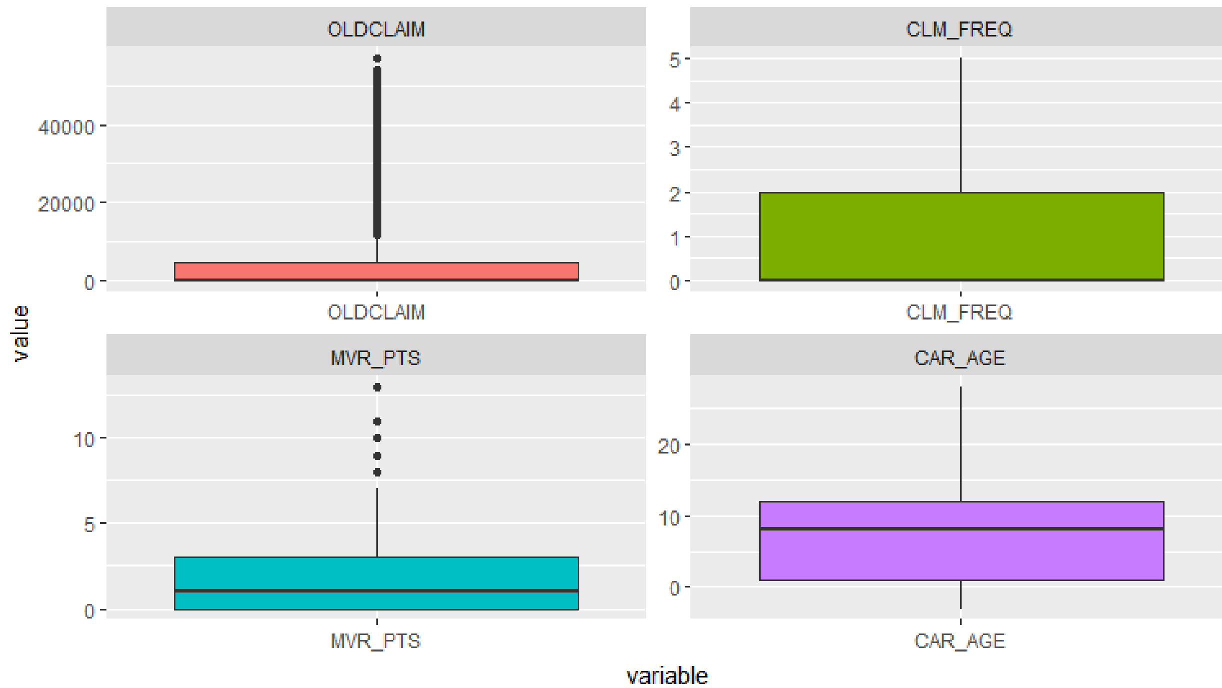
The image below shows positive (blue) and negative(red) correlation between all variables. The crossed-out fields are rejected by a 95% confidence level. Domain knowledge makes this chart more significant as it helps form more advanced hypotheses and see how variables are related. The confidence level marked multiple fields as statistically insignificant which may help us reduce number of variables included in the models.



Boxplots

The boxplots below help us bring the descriptive statistics from the previous section into neat visuals. We can easily determine ranges, medians and outliers. Variables with a high number of outliers may need additional cleansing and transformations which may help with improving accuracy of models. It seems there are many outliers for *OLDCLAIM* and *INCOME* variables and they are only visible for some variables. It suggests that proper handling of them may increase the accuracy of out model.




OLDCLAIM
CLM_FREQ
MVR_PTS
CAR_AGE
value
variable
40000
20000
0
5
4
3
2
1
0
10
5
0
20
10
0
OLDCLAIM
CLM_FREQ
MVR_PTS
CAR_AGE

DATA PREPARATION

The insurance customer data contains a mixture of different variable types. For model building we need to convert them to numerical values or factor class. Finally, we can clean up obvious error and test various transformations. In the previous section we noted that there are some outliers as well as missing values which also may need to be treated.

At this point domain knowledge often is the most powerful as it helps with deriving new features, grouping or partitioning existing features into more informative categories or 'buckets.'

Cleanup:

- Converted 5 monetary variables into numeric but removing the dollar sign and commas. One of the variables is *TARGET*_AMT
- Some of the variables contained prefix '*z*_' in their values. Removed for improved esthetics.
- Converted categorical variables to factors and the integers into numerical
- Dropped any rows with missing data leaving 6448 complete cases

Transformations applied:

- Replaced all outliers from the response variable - *TARGET*_AMT with the feature's mean.

- The set was split into two – one with TARGET_FLAG 0 as insurance_not_claimed and the other 1 insurance_claimed. Models for predicting *TARGET*_AMT variable will utilize only the cases where the claim was made. This should reduce skewness.

- Finally, divided insurance_claimed dataframe into training and testing with a ratio of 75:25.

Model Building

Binary Model 1

This model uses the original dataset with all available variables and no other transformation besides the type conversion to numerical. The reason why I decided to do this because the dataset contains a variety of features from education through profession, kids to the car color and value. This seems to be wide range of topics and I did not want to introduce any personal bias at this stage.

```
Call:
lm(formula = TARGET_FLAG ~ ., data = insurance, family = binomial())

Residuals:
     Min       1Q   Median       3Q      Max 
-0.56422 -0.11784 -0.05857  0.01183  0.96508 

Coefficients:
                         Estimate Std. Error t value Pr(>|t|)    
(Intercept)             1.915e-02  3.284e-02   0.583 0.559744    
TARGET_AMT              6.549e-04  6.112e-06 107.158  < 2e-16 ***
KIDSDRIV                2.284e-02  6.533e-03   3.496 0.000475 ***
AGE                    -1.562e-04  4.074e-04  -0.383 0.701414    
HOMEKIDS               -5.898e-03  3.758e-03  -1.569 0.116656    
YOJ                    -5.869e-04  8.590e-04  -0.683 0.494468    
INCOME                 -2.442e-08  1.093e-07  -0.223 0.823243    
PARENT1Yes              4.911e-02  1.155e-02   4.253 2.13e-05 ***
HOME_VAL               -1.244e-07  3.487e-08  -3.569 0.000361 ***
MSTATUSYes             -1.300e-02  8.541e-03  -1.522 0.128000    
SEXM                    5.229e-03  1.044e-02   0.501 0.616591    
EDUCATIONBachelors     -1.378e-02  1.187e-02  -1.161 0.245599    
EDUCATIONMasters       -8.588e-03  1.746e-02  -0.492 0.622762    
EDUCATIONPhD            2.722e-03  2.091e-02   0.130 0.896443    
EDUCATIONz_High School  7.534e-03  9.865e-03   0.764 0.445020    
JOBClerical             3.577e-02  1.972e-02   1.814 0.069712 .  
JOBDoctor              -1.740e-02  2.352e-02  -0.740 0.459411    
JOBHome Maker           2.404e-02  2.129e-02   1.129 0.258896    
JOBLawyer               2.255e-02  1.709e-02   1.320 0.186842    
JOBManager             -3.303e-02  1.678e-02  -1.969 0.049047 *  
JOBProfessional         1.624e-02  1.781e-02   0.912 0.361919    
JOBStudent              1.112e-02  2.185e-02   0.509 0.610868    
JOBz_Blue Collar        2.512e-02  1.859e-02   1.351 0.176677    
TRAVTIME                7.393e-04  1.860e-04   3.973 7.16e-05 ***
CAR_USEPrivate         -4.313e-02  9.522e-03  -4.530 6.02e-06 ***
BLUEBOOK               -6.499e-07  4.926e-07  -1.319 0.187082    
TIF                    -2.843e-03  7.013e-04  -4.054 5.10e-05 ***
CAR_TYPEPanel Truck     2.254e-02  1.599e-02   1.410 0.158713    
CAR_TYPEPickup          2.554e-02  9.781e-03   2.611 0.009037 ** 
CAR_TYPESports Car      6.360e-02  1.243e-02   5.115 3.23e-07 ***
CAR_TYPESUV             4.098e-02  1.020e-02   4.016 5.99e-05 ***
CAR_TYPEVan             2.709e-02  1.229e-02   2.203 0.027603 *  
RED_CARyes              3.664e-03  8.565e-03   0.428 0.668846    
OLDCLAIM               -3.539e-08  4.262e-07  -0.083 0.933813    
CLM_FREQ                9.855e-03  3.157e-03   3.122 0.001805 ** 
REVOKEDYes              5.218e-02  1.006e-02   5.188 2.19e-07 ***
MVR_PTS                 5.350e-03  1.495e-03   3.578 0.000349 ***
CAR_AGE                -7.074e-04  7.320e-04  -0.966 0.333931    
URBANICITYurban         9.827e-02  8.245e-03  11.919  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2322 on 6409 degrees of freedom
Multiple R-squared:  0.7244,	Adjusted R-squared:  0.7227 
F-statistic: 443.2 on 38 and 6409 DF,  p-value: < 2.2e-16
```

Binary Model 2

This model is an extension of the first one. I applied a stepwise approach using the built in function *stepAIC()* in both directions. This helped me to reduce the number of variables from 26 to 20. No additional transformations were applied.

```
Call:
glm(formula = TARGET_FLAG ~ KIDSDRIV + HOMEKIDS + PARENT1 + HOME_VAL +
    MSTATUS + JOB + TRAVTIME + CAR_USE + BLUEBOOK + TIF + CAR_TYPE +
    CLM_FREQ + REVOKED + MVR_PTS + CAR_AGE + URBANICITY, family = binomial(),
    data = insurance)

Deviance Residuals:
   Min       1Q   Median       3Q      Max
-2.575   -0.718   -0.407    0.645    3.125

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)          -3.23e+00   2.67e-01  -12.12  < 2e-16 ***
KIDSDRIV              3.33e-01   6.74e-02    4.94  7.7e-07 ***
HOMEKIDS              3.38e-02   3.78e-02    0.89  0.37109
PARENT1Yes            4.63e-01   1.21e-01    3.82  0.00013 ***
HOME_VAL             -1.84e-06   3.53e-07   -5.23  1.7e-07 ***
MSTATUSYes           -3.62e-01   9.30e-02   -3.89  1.0e-04 ***
JOBClerical           7.26e-01   1.76e-01    4.14  3.5e-05 ***
JOBDoctor            -8.46e-02   2.66e-01   -0.32  0.75038
JOBHome Maker         4.71e-01   1.97e-01    2.39  0.01691 *
JOBLawyer             2.74e-01   1.83e-01    1.50  0.13411
JOBManager           -5.89e-01   1.78e-01   -3.31  0.00094 ***
JOBProfessional       1.72e-01   1.66e-01    1.04  0.29900
JOBStudent            4.75e-01   1.96e-01    2.43  0.01525 *
JOBz_Blue Collar      4.64e-01   1.62e-01    2.87  0.00407 **
TRAVTIME              1.55e-02   2.11e-03    7.35  2.0e-13 ***
CAR_USEPrivate       -7.84e-01   9.81e-02   -7.99  1.4e-15 ***
BLUEBOOK             -2.72e-05   5.21e-06   -5.22  1.8e-07 ***
TIF                  -5.29e-02   8.20e-03   -6.44  1.2e-10 ***
CAR_TYPEPanel Truck   7.38e-01   1.67e-01    4.43  9.5e-06 ***
CAR_TYPEPickup        5.49e-01   1.12e-01    4.90  9.6e-07 ***
CAR_TYPESports Car    1.03e+00   1.20e-01    8.60  < 2e-16 ***
CAR_TYPESUV           7.68e-01   9.63e-02    7.98  1.5e-15 ***
CAR_TYPEVan           6.78e-01   1.37e-01    4.96  7.0e-07 ***
CLM_FREQ              1.56e-01   2.85e-02    5.49  3.9e-08 ***
REVOKEDYes            7.28e-01   9.04e-02    8.05  8.2e-16 ***
MVR_PTS               1.12e-01   1.51e-02    7.39  1.5e-13 ***
CAR_AGE              -2.21e-02   7.16e-03   -3.09  0.00202 **
URBANICITYurban       2.29e+00   1.23e-01   18.53  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 7445.1  on 6447  degrees of freedom
Residual deviance: 5801.2  on 6420  degrees of freedom
AIC: 5857

Number of Fisher Scoring iterations: 5
```

Binary Model 3

This is the only model that is populated with handpicked variables based on the p-values of the first model as well as personal intuition.

```
Call:
glm(formula = TARGET_FLAG ~ AGE + CLM_FREQ + PARENT1 + MSTATUS +
    REVOKED + URBANICITY + MVR_PTS, family = binomial(), data = insurance)

Deviance Residuals:
   Min       1Q   Median       3Q      Max
-1.919   -0.766   -0.580    0.860    2.656

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)     -1.95992    0.20721   -9.46  < 2e-16 ***
AGE             -0.02005    0.00371   -5.41  6.3e-08 ***
CLM_FREQ         0.21309    0.02659    8.01  1.1e-15 ***
PARENT1Yes       0.60310    0.09922    6.08  1.2e-09 ***
MSTATUSYes      -0.33520    0.07041   -4.76  1.9e-06 ***
REVOKEDYes       0.74127    0.08353    8.87  < 2e-16 ***
URBANICITYurban  1.55948    0.11465   13.60  < 2e-16 ***
MVR_PTS          0.13820    0.01418    9.74  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 7445.1  on 6447  degrees of freedom
Residual deviance: 6492.2  on 6440  degrees of freedom
AIC: 6508

Number of Fisher Scoring iterations: 5
```

Multiple linear regression - Model 4

This model was trained on data selected in the preparation phase. It includes a random sample (75%) of cases where car was in a crash (TARGET_FLAG = 1). This data was also treated with the replacement of outliers with the variable's mean. The goal was to reduce unusually high claims. The model uses all variables available variables.

```
Call:
lm(formula = TARGET_AMT ~ ., data = insurance_claimed_train)

Residuals:
   Min     1Q Median     3Q    Max 
 -3790  -1423    -70   1163   6082 

Coefficients: (1 not defined because of singularities)
                        Estimate Std. Error t value Pr(>|t|)    
(Intercept)             3.32e+03   6.65e+02    5.00  6.5e-07 ***
TARGET_FLAG                   NA         NA      NA       NA    
KIDSDRIV               -6.27e+01   1.08e+02   -0.58    0.563    
AGE                     8.65e+00   7.09e+00    1.22    0.222    
HOMEKIDS                8.53e+01   6.92e+01    1.23    0.218    
YOJ                    -2.13e+01   1.65e+01   -1.30    0.195    
INCOME                 -3.57e-03   2.33e-03   -1.54    0.125    
PARENT1Yes             -7.13e+01   1.96e+02   -0.36    0.716    
HOME_VAL                1.12e-03   6.76e-04    1.66    0.098 .  
MSTATUSYes             -8.23e+01   1.70e+02   -0.49    0.627    
SEXM                   -1.90e+02   2.22e+02   -0.86    0.391    
EDUCATIONBachelors     -3.46e+02   2.14e+02   -1.62    0.105    
EDUCATIONMasters        3.21e+02   3.74e+02    0.86    0.390    
EDUCATIONPhD            6.86e+02   4.55e+02    1.51    0.132    
EDUCATIONz_High School -9.36e+00   1.71e+02   -0.05    0.956    
JOBClerical             4.83e+02   4.08e+02    1.19    0.236    
JOBDoctor               1.23e+02   5.44e+02    0.23    0.821    
JOBHome Maker           1.36e+02   4.33e+02    0.31    0.753    
JOBLawyer              -1.23e+02   3.51e+02   -0.35    0.726    
JOBManager              6.17e+02   3.65e+02    1.69    0.091 .  
JOBProfessional         6.14e+02   3.80e+02    1.62    0.106    
JOBStudent              4.96e+02   4.35e+02    1.14    0.255    
JOBz_Blue Collar        3.27e+02   3.92e+02    0.84    0.404    
TRAVTIME               -9.07e-01   3.73e+00   -0.24    0.808    
CAR_USEPrivate         -9.64e+01   1.75e+02   -0.55    0.581    
BLUEBOOK               -4.08e-04   1.00e-02   -0.04    0.968    
TIF                     1.46e+01   1.39e+01    1.05    0.295    
CAR_TYPEPanel Truck     1.67e+02   3.16e+02    0.53    0.596    
CAR_TYPEPickup          3.77e+01   2.01e+02    0.19    0.851    
CAR_TYPESports Car      1.63e+01   2.48e+02    0.07    0.948    
CAR_TYPESUV            -8.41e+00   2.21e+02   -0.04    0.970    
CAR_TYPEVan             1.73e+02   2.64e+02    0.65    0.513    
RED_CARyes              5.51e+01   1.68e+02    0.33    0.743    
OLDCLAIM                2.78e-03   7.51e-03    0.37    0.711    
CLM_FREQ               -4.70e+01   5.27e+01   -0.89    0.373    
REVOKEDYes              7.57e+00   1.76e+02    0.04    0.966    
MVR_PTS                 3.33e+01   2.29e+01    1.45    0.147    
CAR_AGE                 5.72e+00   1.45e+01    0.39    0.694    
URBANICITYurban         2.04e+02   2.51e+02    0.81    0.417    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1950 on 1224 degrees of freedom
  (3574 observations deleted due to missingness)
Multiple R-squared:  0.0255,	Adjusted R-squared:  -0.00397 
F-statistic: 0.865 on 37 and 1224 DF,  p-value: 0.7
```

Multiple linear regression - Model 5

This model was designed purely on car-related variables and not the customer information. It was driven by the value, age and type of the car as well as previous claim to add variability. The dataset used was the same as for model 4.

```
Call:
glm(formula = TARGET_AMT ~ BLUEBOOK + OLDCLAIM + CAR_AGE + CLM_FREQ +
    CAR_TYPE, data = insurance_claimed_train)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
 -3944   -1426     -39    1250    6389

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)          3.90e+03   1.92e+02   20.32   <2e-16 ***
BLUEBOOK             3.10e-04   8.50e-03    0.04     0.97
OLDCLAIM             3.76e-03   5.91e-03    0.64     0.52
CAR_AGE              1.87e+00   1.02e+01    0.18     0.85
CLM_FREQ            -2.34e+01   4.82e+01   -0.49     0.63
CAR_TYPEPanel Truck  2.36e+02   2.68e+02    0.88     0.38
CAR_TYPEPickup       1.14e+02   1.87e+02    0.61     0.54
CAR_TYPESports Car   1.71e+02   2.02e+02    0.85     0.40
CAR_TYPESUV          1.17e+02   1.70e+02    0.69     0.49
CAR_TYPEVan          1.61e+02   2.39e+02    0.67     0.50
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 3820733)

    Null deviance: 4790905337  on 1261  degrees of freedom
Residual deviance: 4783557361  on 1252  degrees of freedom
  (3574 observations deleted due to missingness)
AIC: 22720

Number of Fisher Scoring iterations: 2
```

Model Selection

Binary Linear Regression Models Review

Performance of models can be measured in many ways. I an external package called *caret* to tap into metrics that will help me identify the best performing model.

By running confusionMatrix() function on each of the models we can classify outcomes of our predictions into 4 buckets – True Positive, True Negative, False Positive and False Negative and at the same time calculate multiple metrics.

I extracted the data from the function above and put it into a new dataframe for easier model comparison. The table below shows overall accuracies and their ranges. We can easily determine that all 3 models scored high in accuracy but placed model 1 in the lead.

	model1	*model2*	*model3*
Accuracy	0.94	0.9	0.88
Kappa	0.82	0.71	0.65
AccuracyLower	0.93	0.89	0.87
AccuracyUpper	0.94	0.9	0.88
AccuracyNull	0.74	0.74	0.74
AccuracyPValue	0	0	0
McnemarPValue	0	0	0

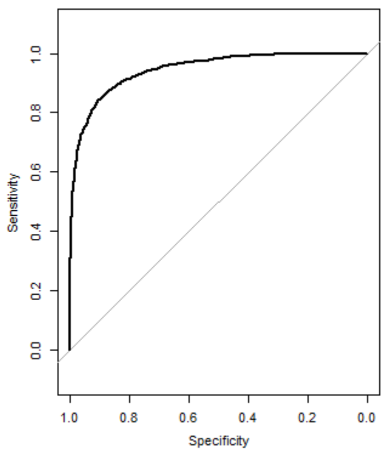
There are several additional metrics that can be extracted from the function and they are the following:

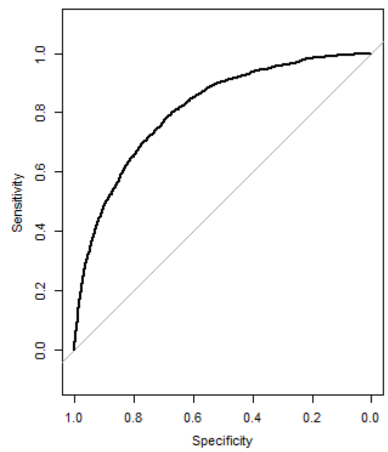
	model1	*model2*	*model3*
Sensitivity	0.77	0.71	0.61
Specificity	1	0.96	0.97
Pos Pred Value	0.98	0.87	0.89
Neg Pred Value	0.92	0.9	0.88
Precision	0.98	0.87	0.89
Recall	0.77	0.71	0.61
F1	0.86	0.78	0.73
Prevalence	0.26	0.26	0.26
Detection Rate	0.2	0.19	0.16
Detection Prevalence	0.21	0.21	0.18
Balanced Accuracy	0.88	0.84	0.79

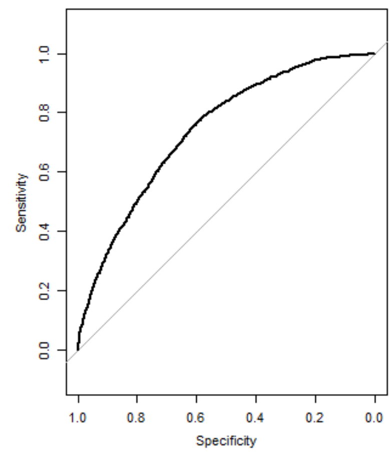
Once again, model 1 is in the lead with all of the metrics.

Another great way to compare models is to determine their *Receiver Operating Characteristic* (ROC) and the Area Under the Curve (AUC). Package pROC provides a function that quickly calculated the AUC and plotted the results.

	MODEL1	MODEL2	MODEL3
AUC	0.945	0.811	0.741







The faster the line approaches to 1 on the Y axis the better the model is performing. We can note a huge difference in model 1 versus model 2 & 3.

Binary Model Summary

The dataset we worked with has proven to be great for building a binary linear regression model for predicting whether or not a customer will be in a traffic collision. It contained multiple variables with strong relationships to the target variable. We built 3 models which included all variables, only selected ones based on the sideAIC technique and finally one that focused on the insured car and not the customer. The performance metrics outlined above indicated that all 3 scored high but with notable differences. **Model 1** is the winner. Further tuning of the model would include dropping at least one more variable, as well as outlier handling, other transformations to derive more advanced features. With accuracy score of 94% and other metric scores just as high, we can trust this model will help us predict whether the customer will be in a car crash and making a claim.

Multiple Linear Regression Models Review

The performances of multiple linear regression models were determined by running a prediction on the testing data set aside in the previous section. With the predicted values, were used to create two accuracy-measuring metrics – '*Min Max Accuracy*' and '*Mean Absolute Percentage Error*'.

$$MinMaxAccuracy = mean\left(\frac{min\,(actuals, predicteds)}{max\,(actuals, predicteds)}\right)$$

$$MeanAbsolutePercentageError\;(MAPE) = mean\left(\frac{abs\,(predicteds - actuals)}{actuals}\right)$$

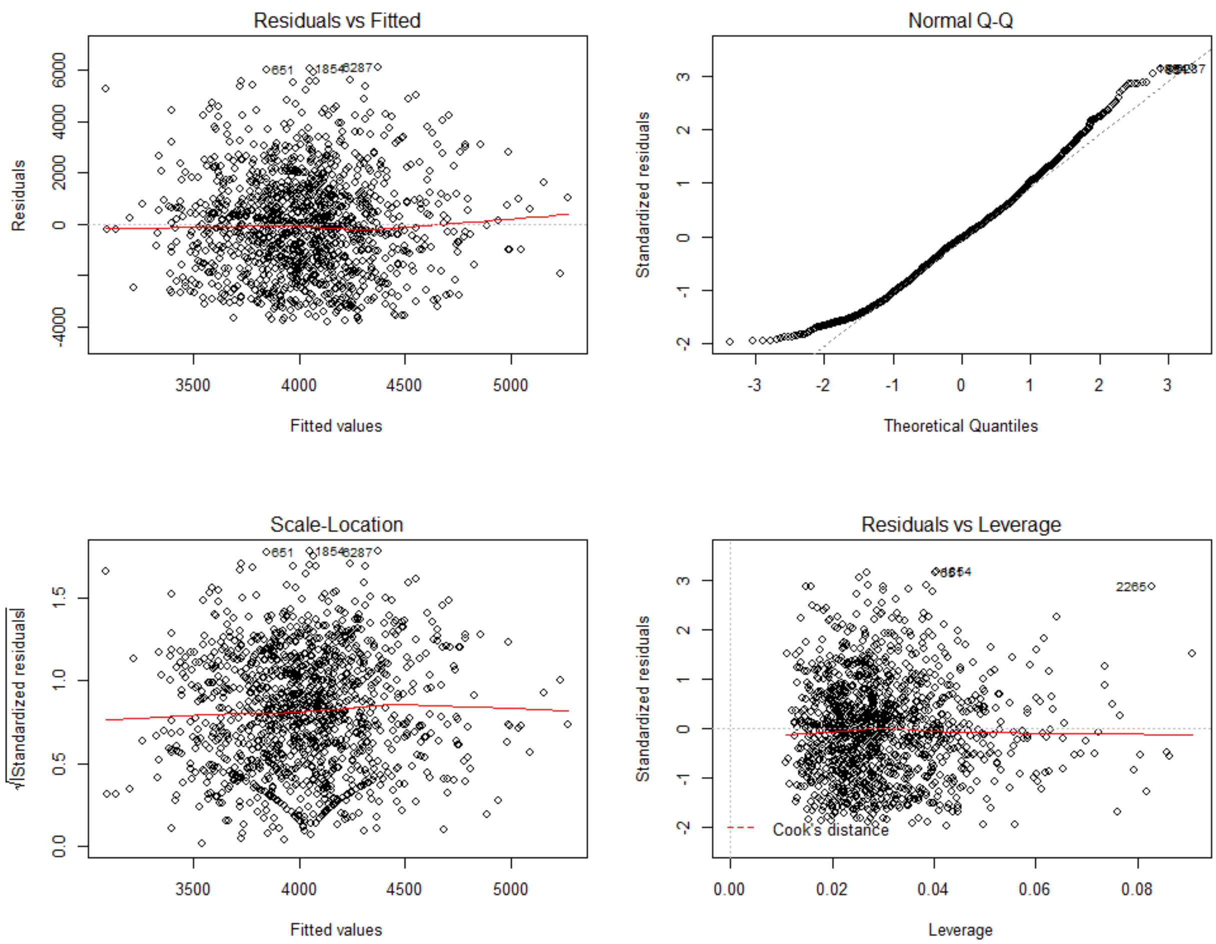
The following are the scores our two models:

	model4	*model5*
MinMax	0.689	0.697
MAPE	0.787	0.763

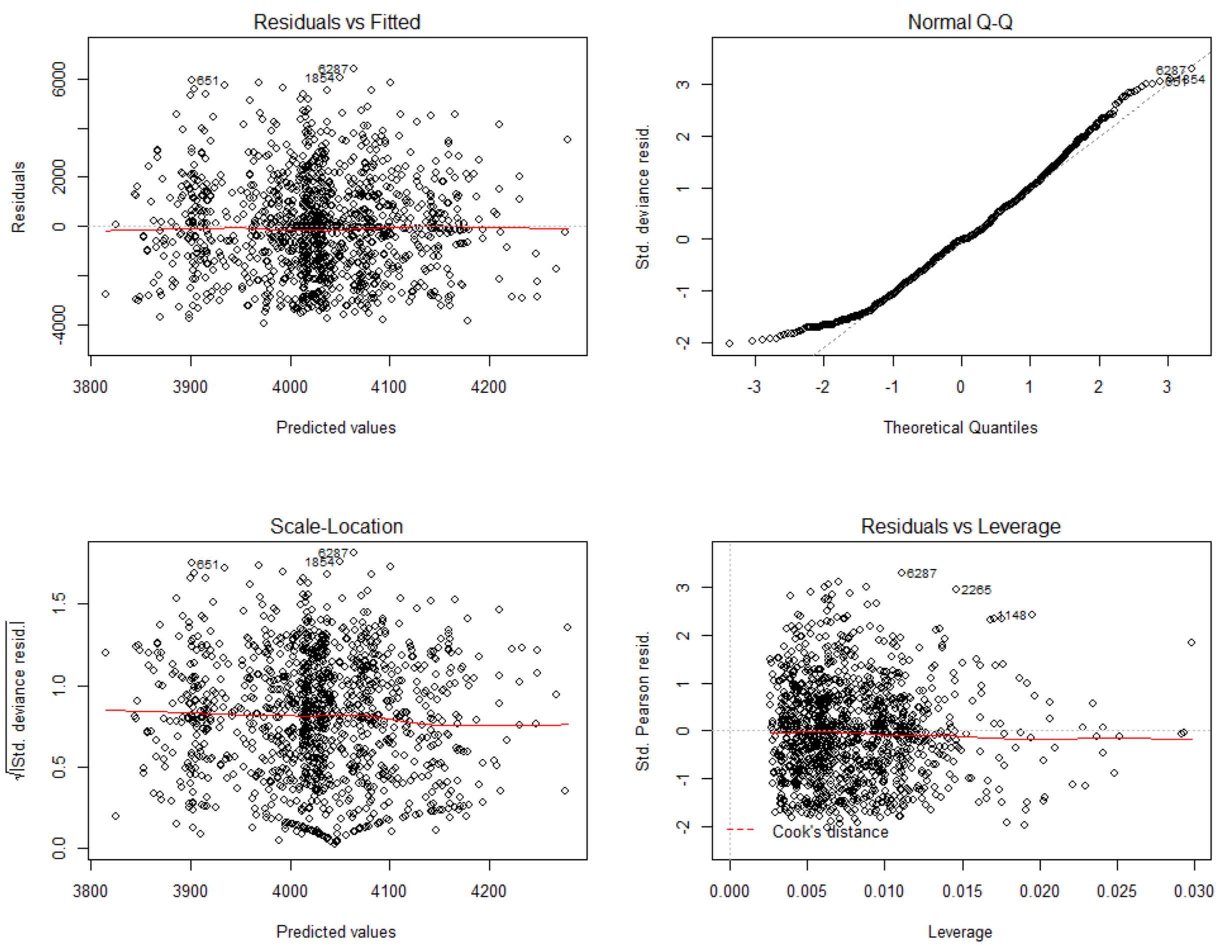
Residual Analysis

The residual plots (Normal Q-Q specifically) show a potential problem with a non-normal distribution. ‘Heavy tails’ usually mean that the data have more extreme values than would be expected if they truly came from a Normal distribution. The same behavior is visible in both models.

Model 4



Model 5


Residuals vs Fitted
Residuals
Predicted values
Normal Q-Q
Std. deviance resid.
Theoretical Quantiles
Scale-Location
√|Std. deviance resid.|
Predicted values
Residuals vs Leverage
Std. Pearson resid.
Cook's distance
Leverage

Multiple Linear Regression Model Summary

The predictor variables did not seem to be heavily correlated to the response variables. The skewness of data might have negatively impacted the overall performance of our models. Most of the predicted claim amounts averaged around $4,000 with about $500 variation range.

With almost exact MinMax and MAPE scores for both models **Model 5** wins as it requires only 5 car-describing variables. The model performs with around 70% accuracy.

Car Insurance - Customer Analytics

Rafal Decowski

```
library(dplyr)
library(tidyr)
library(knitr)
library(stringr)
library(reshape2)

library(ggplot2)
library(corrplot)


########################################################
# Loading data and simple transformations
########################################################
insurance <- read.csv2('D:\\Rafal\\CUNY\\621\\hw\\hw4\\insurance_training_data.csv', sep=',', row.names=


# Convert Currencies to numbers
insurance$INCOME <- as.numeric(gsub('[$,]', '', insurance$INCOME))
insurance$BLUEBOOK <- as.numeric(gsub('[$,]', '', insurance$BLUEBOOK))
insurance$OLDCLAIM <- as.numeric(gsub('[$,]', '', insurance$OLDCLAIM))
insurance$HOME_VAL <- as.numeric(gsub('[$,]', '', insurance$HOME_VAL))

# Convert female indicator, mstatus,car_type
insurance$SEX <- gsub('z_F', 'F', insurance$SEX)
insurance$MSTATUS <- gsub('z_No', 'No', insurance$MSTATUS)
insurance$CAR_TYPE <- gsub('z_SUV', 'SUV', insurance$CAR_TYPE)

# Convert URBANICITY
insurance$URBANICITY <- gsub('Highly Urban/ Urban', 'urban', insurance$URBANICITY)
insurance$URBANICITY <- gsub('z_Highly Rural/ Rural', 'rural', insurance$URBANICITY)

# Convert character class columns to factors
insurance$CAR_TYPE <- as.factor(insurance$CAR_TYPE)
insurance$MSTATUS <- as.factor(insurance$MSTATUS)
insurance$SEX <- as.factor(insurance$SEX)
insurance$URBANICITY <- as.factor(insurance$URBANICITY)
insurance$CAR_TYPE <- as.factor(insurance$CAR_TYPE)

# Convert Integers to numeric
insurance$TARGET_FLAG <- as.numeric(insurance$TARGET_FLAG)
insurance$KIDSDRIV <- as.numeric(insurance$KIDSDRIV)
insurance$AGE <- as.numeric(insurance$AGE)
insurance$HOMEKIDS <- as.numeric(insurance$HOMEKIDS)
insurance$YOJ <- as.numeric(insurance$YOJ)
insurance$TRAVTIME <- as.numeric(insurance$TRAVTIME)
insurance$TIF <- as.numeric(insurance$TIF)
insurance$CLM_FREQ <- as.numeric(insurance$CLM_FREQ)
insurance$CAR_AGE <- as.numeric(insurance$CAR_AGE)
```

```r
#insurance$TARGET_AMT <- as.numeric(insurance$TARGET_AMT) turned out to be incorrect
insurance$TARGET_AMT <- as.numeric(as.character(insurance$TARGET_AMT))

# Drop rows with missing values
insurance <- insurance[complete.cases(insurance), ]
rownames(insurance) <- 1:nrow(insurance)

# Select only numeric variables
insurance_only_numeric <- select_if(insurance, is.numeric)




insurance_claimed <- insurance[which(insurance$TARGET_FLAG==1), ]
# Removing outliers by creating a minimum and maximum 'benchmark' value with interquartile values
remove_outliers <- function(x, na.rm = TRUE) {
  qnt <- quantile(x, probs=c(.25, .75), na.rm = na.rm)
  H <- 1.5 * IQR(x, na.rm = na.rm)
  y <- x
  y[x < (qnt[1] - H)] <- NA
  y[x > (qnt[2] + H)] <- NA
  return(y)
}

# Copy the dataframe with 14 variables
insurance_no_outliers <- cbind(insurance_claimed)

# Iterating through every varaible and replacing outliers with NA's
for(i in c(2)){
  insurance_no_outliers[,i] <- remove_outliers(insurance_claimed[,i])
}
insurance_no_outliers[is.na(insurance_no_outliers[,2]), 2] <- mean(insurance_no_outliers[,2], na.rm = T

insurance = subset(insurance, select = -c(INDEX, TARGET_AMT) )


# Select only cases with TARGET_FLAG = 1
smp_size <- floor(0.75 * nrow(insurance))

## set the seed to make your partition reproductible
set.seed(123)
train_ind <- sample(seq_len(nrow(insurance)), size = smp_size)

insurance_claimed_train <- insurance_no_outliers[train_ind, ]
insurance_claimed_test <- insurance_no_outliers[-train_ind, ]







insurance_not_claimed <- insurance[which(insurance$TARGET_FLAG==0), ]
```

```
########################################################
# Descriptive statistics
########################################################
library(psych)
stats <- round(describe(insurance, omit=TRUE, skew = FALSE, IQR = TRUE),2)
kable(stats)

# Other attempt to capture descriptive statistics
stats2 <- as.data.frame(summary(insurance_only_numeric)) %>% separate(Freq, c("metric", "value"), ":")
stats2$metric <- trimws(stats2$metric, which = c("both", "left", "right"))
stats2$value <- trimws(stats2$value, which = c("both", "left", "right"))
stats2 <- stats2[,2:4]
stats2 <- dcast(stats2, Var2  ~ metric, value.var="value")


########################################################
# Boxplots
########################################################


ggplot(data = melt(as.data.frame(insurance[,3:17])), aes(x=variable, y=value)) +
  geom_boxplot(aes(fill=variable)) +
  theme(legend.position="none") +
  facet_wrap( ~ variable, scales="free")

ggplot(data = melt(as.data.frame(insurance[,18:24])), aes(x=variable, y=value)) +
  geom_boxplot(aes(fill=variable)) +
  theme(legend.position="none") +
  facet_wrap( ~ variable, scales="free")




########################################################
# Correlation
########################################################

cormat <- cor(insurance_only_numeric)
res1 <- cor.mtest(cormat, conf.level = .95)
corrplot(cormat, type = "lower", order = "hclust", tl.col = "black", tl.srt = 45, p.mat = res1$p, sig.l

# Target Correlation
target_flag_corr <- cor(insurance_only_numeric)['TARGET_FLAG',]
#target_amt_corr <- cor(insurance_only_numeric)['TARGET_AMT',]



########################################################
# Binary Logistic regression models
########################################################
library(MASS)
library(scales)
```

```r
# Model 1 contains all variables and is performed on the original dataset with no transformations
model1 <- lm(TARGET_FLAG ~ ., data = insurance, family = binomial())

# Adjusting model 1 based on the stepwise suggestions to create model 2
step_m1 <- stepAIC(model1, direction="both")
step_m1$anova
model2 <- glm(TARGET_FLAG ~ KIDSDRIV + HOMEKIDS + PARENT1 + HOME_VAL +
    MSTATUS + JOB + TRAVTIME + CAR_USE + BLUEBOOK + TIF + CAR_TYPE +
    CLM_FREQ + REVOKED + MVR_PTS + CAR_AGE + URBANICITY, data = insurance,   family=binomial())


# Hand-picked variables

model3 <- glm(TARGET_FLAG ~ AGE + CLM_FREQ + PARENT1 + MSTATUS + REVOKED + URBANICITY + MVR_PTS, data =


summary(model1)
summary(model2)
summary(model3)


predict1 <- predict(model1, type = 'response')
predict2 <- predict(model2, type = 'response')
predict3 <- predict(model3, type = 'response')


########################################################
# Measuring Performance
########################################################
library(caret)

# Create Vectors for the predicted values
c1 <- c(insurance$TARGET_FLAG, predict1 > 0.5)
c2 <- c(insurance$TARGET_FLAG, predict2 > 0.5)
c3 <- c(insurance$TARGET_FLAG, predict3 > 0.5)
pred_df <- data.frame(insurance$TARGET_FLAG, c1, c2, c3)

# Measuring Performance
cm1 <- confusionMatrix(factor(pred_df$c1),factor(pred_df$insurance.TARGET_FLAG), positive = '1')
cm2 <- confusionMatrix(factor(pred_df$c2),factor(pred_df$insurance.TARGET_FLAG), positive = '1')
cm3 <- confusionMatrix(factor(pred_df$c3),factor(pred_df$insurance.TARGET_FLAG), positive = '1')

performance_measures1 <- round(data.frame(cm1$overall, cm2$overall, cm3$overall),2)
names(performance_measures1) <- c('model1', 'model2', 'model3')
performance_measures2 <- round(data.frame(cm1$byClass, cm2$byClass, cm3$byClass),2)
names(performance_measures2) <- c('model1', 'model2', 'model3')


########################################################
# ROC
########################################################
library(pROC)
par(mfrow=c(1, 3))
```

```
roc(insurance$TARGET_FLAG ~ predict1, insurance, plot=TRUE)
roc(insurance$TARGET_FLAG ~ predict2, insurance, plot=TRUE)
roc(insurance$TARGET_FLAG ~ predict3, insurance, plot=TRUE)


########################################################
# Multiple Logistic regression models
########################################################
library(MASS)
library(scales)

# Model 1 contains all variables and is performed on the original dataset with no transformations
model4 <- lm(TARGET_AMT ~ ., data = insurance_claimed_train)

# Adjusting model 1 based on the stepwise suggestions to create model 2
step_m4 <- stepAIC(model4, direction="both")
step_m4$anova
model5 <- glm(TARGET_AMT ~ BLUEBOOK + OLDCLAIM + CAR_AGE + CLM_FREQ + CAR_TYPE, data = insurance_claimed


summary(model4)
summary(model5)


predict4 <- predict(model4, insurance_claimed_test)
predict5 <- predict(model5, insurance_claimed_test)


actuals_preds4 <- data.frame(cbind(actuals=insurance_claimed_test$TARGET_AMT, predicteds=predict4))
actuals_preds5 <- data.frame(cbind(actuals=insurance_claimed_test$TARGET_AMT, predicteds=predict5))

min_max_accuracy4 <- mean(apply(actuals_preds4, 1, min) / apply(actuals_preds4, 1, max))
mape4 <- mean(abs((actuals_preds4$predicteds - actuals_preds4$actuals))/actuals_preds4$actuals)

min_max_accuracy5 <- mean(apply(actuals_preds5, 1, min) / apply(actuals_preds5, 1, max))
mape5 <- mean(abs((actuals_preds5$predicteds - actuals_preds5$actuals))/actuals_preds5$actuals)

mlr <- data.frame(row.names = c('MinMax', 'MAPE'), model4=c(0.689,0.787), model5=c(0.697,0.763))



########################################################
# Predictions
########################################################

insurance_eval <- read.csv2('D:\\Rafal\\CUNY\\621\\hw\\hw4\\insurance-evaluation-data.csv', sep=',', ro


# Convert Currencies to numbers
insurance_eval$INCOME <- as.numeric(gsub('[$,]', '', insurance_eval$INCOME))
insurance_eval$BLUEBOOK <- as.numeric(gsub('[$,]', '', insurance_eval$BLUEBOOK))
insurance_eval$OLDCLAIM <- as.numeric(gsub('[$,]', '', insurance_eval$OLDCLAIM))
insurance_eval$HOME_VAL <- as.numeric(gsub('[$,]', '', insurance_eval$HOME_VAL))
```

```r
# Convert female indicator, mstatus,car_type
insurance_eval$SEX <- gsub('z_F', 'F', insurance_eval$SEX)
insurance_eval$MSTATUS <- gsub('z_No', 'No', insurance_eval$MSTATUS)
insurance_eval$CAR_TYPE <- gsub('z_SUV', 'SUV', insurance_eval$CAR_TYPE)

# Convert URBANICITY
insurance_eval$URBANICITY <- gsub('Highly Urban/ Urban', 'urban', insurance_eval$URBANICITY)
insurance_eval$URBANICITY <- gsub('z_Highly Rural/ Rural', 'rural', insurance_eval$URBANICITY)

# Convert character class columns to factors
insurance_eval$CAR_TYPE <- as.factor(insurance_eval$CAR_TYPE)
insurance_eval$MSTATUS <- as.factor(insurance_eval$MSTATUS)
insurance_eval$SEX <- as.factor(insurance_eval$SEX)
insurance_eval$URBANICITY <- as.factor(insurance_eval$URBANICITY)
insurance_eval$CAR_TYPE <- as.factor(insurance_eval$CAR_TYPE)

# Convert Integers to numeric
insurance_eval$KIDSDRIV <- as.numeric(insurance_eval$KIDSDRIV)
insurance_eval$AGE <- as.numeric(insurance_eval$AGE)
insurance_eval$HOMEKIDS <- as.numeric(insurance_eval$HOMEKIDS)
insurance_eval$YOJ <- as.numeric(insurance_eval$YOJ)
insurance_eval$TRAVTIME <- as.numeric(insurance_eval$TRAVTIME)
insurance_eval$TIF <- as.numeric(insurance_eval$TIF)
insurance_eval$CLM_FREQ <- as.numeric(insurance_eval$CLM_FREQ)
insurance_eval$CAR_AGE <- as.numeric(insurance_eval$CAR_AGE)




predictions <- predict(object = model1, insurance_eval, type = 'response')
target <- c(predictions > 0.5)


predict5 <- predict(model5, insurance_eval, type = 'response')

insurance_eval$predicted_prob <- round(predictions,2)
insurance_eval$target <- target
insurance_eval$claim_amt <- round(predict5,2)
write.csv(insurance_eval, 'D:\\Rafal\\CUNY\\621\\hw\\hw4\\insurance-predicted.csv')
```