Data621-Hw4

Joseph Elikishvili

Overview

In this project, we will explore, analyze and model a data set containing approximately 8000 records representing a customer at an auto insurance company. Each record has two response variables. The first response variable, TARGET_FLAG, is a 1 or a 0. A "1" means that the person was in a car crash. A zero means that the person was not in a car crash. The second response variable is TARGET_AMT. This value is zero if the person did not crash their car. But if they did crash their car, this number will be a value greater than zero. Our objective is to build multiple linear regression and binary logistic regression models on the training data to predict the probability that a person will crash their car and also the amount of money it will cost if the person does crash their car. We can only use the variables given to us (or variables that you derive from the variables provided).

1. Data Exploration

We will get started by loading the data and exploring the dimensions of the dataset and getting to know the variables

```
## [1] 8161   26
```

It appears we have a total of 26 variables and 8161 records. The first variable is an index so we will remove it right away as it provides no value to us, so we are dealing with 25 variables total, 2 target variables and 23 predictor variables.

Next we will preview datatypes in each of the columns

```
## 'data.frame':    8161 obs. of  25 variables:
##  $ TARGET_FLAG: int  0 0 0 0 0 1 0 1 1 0 ...
##  $ TARGET_AMT : num  0 0 0 0 0 ...
##  $ KIDSDRIV   : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ AGE        : int  60 43 35 51 50 34 54 37 34 50 ...
##  $ HOMEKIDS   : int  0 0 1 0 0 1 0 2 0 0 ...
##  $ YOJ        : int  11 11 10 14 NA 12 NA NA 10 7 ...
##  $ INCOME     : chr  "$67,349" "$91,449" "$16,039" "" ...
##  $ PARENT1    : chr  "No" "No" "No" "No" ...
##  $ HOME_VAL   : chr  "$0" "$257,252" "$124,191" "$306,251" ...
##  $ MSTATUS    : chr  "z_No" "z_No" "Yes" "Yes" ...
##  $ SEX        : chr  "M" "M" "z_F" "M" ...
##  $ EDUCATION  : chr  "PhD" "z_High School" "z_High School" "<High School"
...
##  $ JOB        : chr  "Professional" "z_Blue Collar" "Clerical" "z_Blue
Collar" ...
##  $ TRAVTIME   : int  14 22 5 32 36 46 33 44 34 48 ...
##  $ CAR_USE    : chr  "Private" "Commercial" "Private" "Private" ...
##  $ BLUEBOOK   : chr  "$14,230" "$14,940" "$4,010" "$15,440" ...
##  $ TIF        : int  11 1 4 7 1 1 1 1 1 7 ...
```

```
##  $ CAR_TYPE   : chr  "Minivan" "Minivan" "z_SUV" "Minivan" ...
##  $ RED_CAR    : chr  "yes" "yes" "no" "yes" ...
##  $ OLDCLAIM   : chr  "$4,461" "$0" "$38,690" "$0" ...
##  $ CLM_FREQ   : int  2 0 2 0 2 0 0 1 0 0 ...
##  $ REVOKED    : chr  "No" "No" "No" "No" ...
##  $ MVR_PTS    : int  3 0 3 0 3 0 0 10 0 1 ...
##  $ CAR_AGE    : int  18 1 10 6 17 7 1 7 1 17 ...
##  $ URBANICITY : chr  "Highly Urban/ Urban" "Highly Urban/ Urban" "Highly
Urban/ Urban" "Highly Urban/ Urban" ...
```

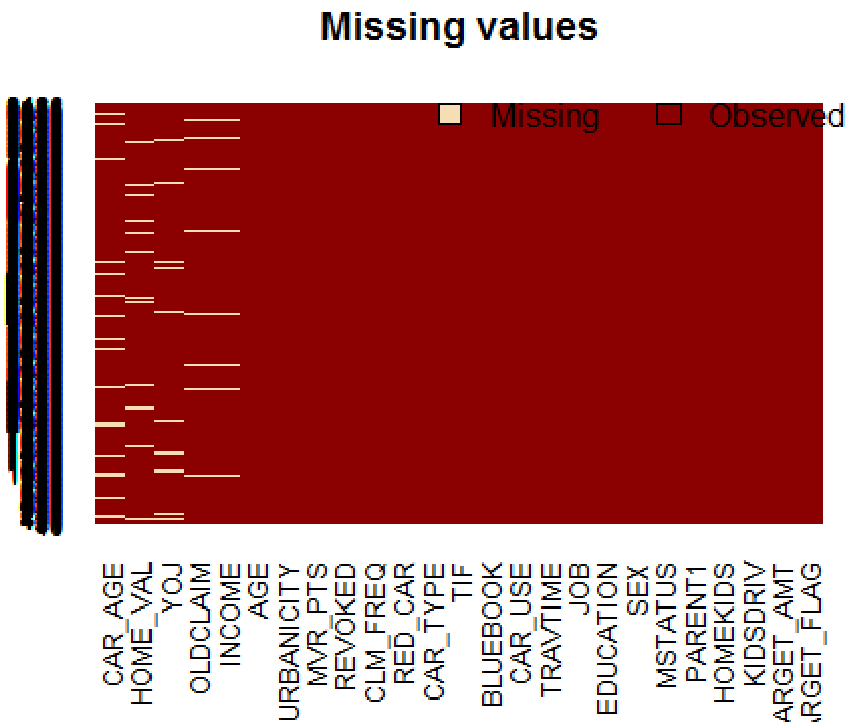
It appears there are several things we need to do in order to clean our data. 1. First we need to convert all USD from string to numeric and get rid of $ signs and , signs. This affects 4 columns. 2. Next we will need to fix some of the '_z' and '<' that made it into the set affecting some of the categorical variables and may disrupt the model later on.

Lets go ahead and fix the $ amounts first in columns: INCOME, HOME_VAL, BLUEBOOK, OLDCLAIM And then remove the '<' character and '_z' character from columns: MSTATUS, SEX, EDUCATION, JOB, CAR_TYPE, URBANCITY

Now our data is clean and we can move to the next step and check data for any missing values and develop a strategy to deal with those if we find any

```
## TARGET_FLAG  TARGET_AMT    KIDSDRIV         AGE    HOMEKIDS         YOJ 
##           0           0           0           6           0         454 
##      INCOME     PARENT1    HOME_VAL     MSTATUS         SEX   EDUCATION 
##         445           0         464           0           0           0 
##         JOB    TRAVTIME     CAR_USE    BLUEBOOK         TIF    CAR_TYPE 
##           0           0           0           0           0           0 
##     RED_CAR    OLDCLAIM    CLM_FREQ     REVOKED     MVR_PTS     CAR_AGE 
##           0         445           0           0           0         510 
##  URBANICITY 
##           0
```

Lets also visually check to see if there are any missing values, we will use Amelia library to do that and review results.



We can see that we have a total of 6 columns with missing data, 5 columns with roughly 450-500 missing values and Age is only missing 6 values.

Next we will review the predictors given to us to better understand the data we are dealing with. The table gives us a basic overview of the data

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
TARGET_FLAG	1	8161	0.26	0.44	0	0.20	0.00	0	1.0	1.0	1.07	-0.85	0.00
TARGET_AMT	2	8161	1504.32	4704.03	0	593.71	0.00	0	107586.1	107586.1	8.71	112.29	52.07
KIDSDRIV	3	8161	0.17	0.51	0	0.03	0.00	0	4.0	4.0	3.35	11.78	0.01
AGE	4	8155	44.79	8.63	45	44.83	8.90	16	81.0	65.0	-0.03	-0.06	0.10
HOMEKIDS	5	8161	0.72	1.12	0	0.50	0.00	0	5.0	5.0	1.34	0.65	0.01
YOJ	6	7707	10.50	4.09	11	11.07	2.97	0	23.0	23.0	-1.20	1.18	0.05
INCOM	7	77	6189	4757	540	5684	4179	0	367	367	1.	2.1	541

E		16	8.09	2.68	28	0.98	2.27		030.0	030.0	19	3	.58
PARENT1*	8	8161	NaN	NA	NA	NaN	NA	Inf	-Inf	-Inf	NA	NA	NA
HOME_VAL	9	7697	154867.29	129123.77	161160	144032.07	147867.11	0	885282.0	885282.0	0.49	-0.02	1471.79
MSTATUS*	10	8161	NaN	NA	NA	NaN	NA	Inf	-Inf	-Inf	NA	NA	NA
SEX*	11	8161	NaN	NA	NA	NaN	NA	Inf	-Inf	-Inf	NA	NA	NA
EDUCATION*	12	8161	NaN	NA	NA	NaN	NA	Inf	-Inf	-Inf	NA	NA	NA
JOB*	13	8161	NaN	NA	NA	NaN	NA	Inf	-Inf	-Inf	NA	NA	NA
TRAVTIME	14	8161	33.49	15.91	33	33.00	16.31	5	142.0	137.0	0.45	0.66	0.18
CAR_USE*	15	8161	NaN	NA	NA	NaN	NA	Inf	-Inf	-Inf	NA	NA	NA
BLUEBOOK	16	8161	15709.90	8419.73	14440	15036.89	8450.82	1500	69740.0	68240.0	0.79	0.79	93.20
TIF	17	8161	5.35	4.15	4	4.84	4.45	1	25.0	24.0	0.89	0.42	0.05
CAR_TYPE*	18	8161	NaN	NA	NA	NaN	NA	Inf	-Inf	-Inf	NA	NA	NA
RED_CAR*	19	8161	NaN	NA	NA	NaN	NA	Inf	-Inf	-Inf	NA	NA	NA
OLDCLAIM	20	7716	61898.09	47572.68	54028	56840.98	41792.27	0	367030.0	367030.0	1.19	2.13	541.58
CLM_FREQ	21	8161	0.80	1.16	0	0.59	0.00	0	5.0	5.0	1.21	0.28	0.01
REVOKED*	22	8161	NaN	NA	NA	NaN	NA	Inf	-Inf	-Inf	NA	NA	NA
MVR_PTS	23	8161	1.70	2.15	1	1.31	1.48	0	13.0	13.0	1.35	1.38	0.02
CAR_AGE	24	7651	8.33	5.70	8	7.96	7.41	-3	28.0	31.0	0.28	-0.75	0.07
URBANICITY*	25	8161	NaN	NA	NA	NaN	NA	Inf	-Inf	-Inf	NA	NA	NA

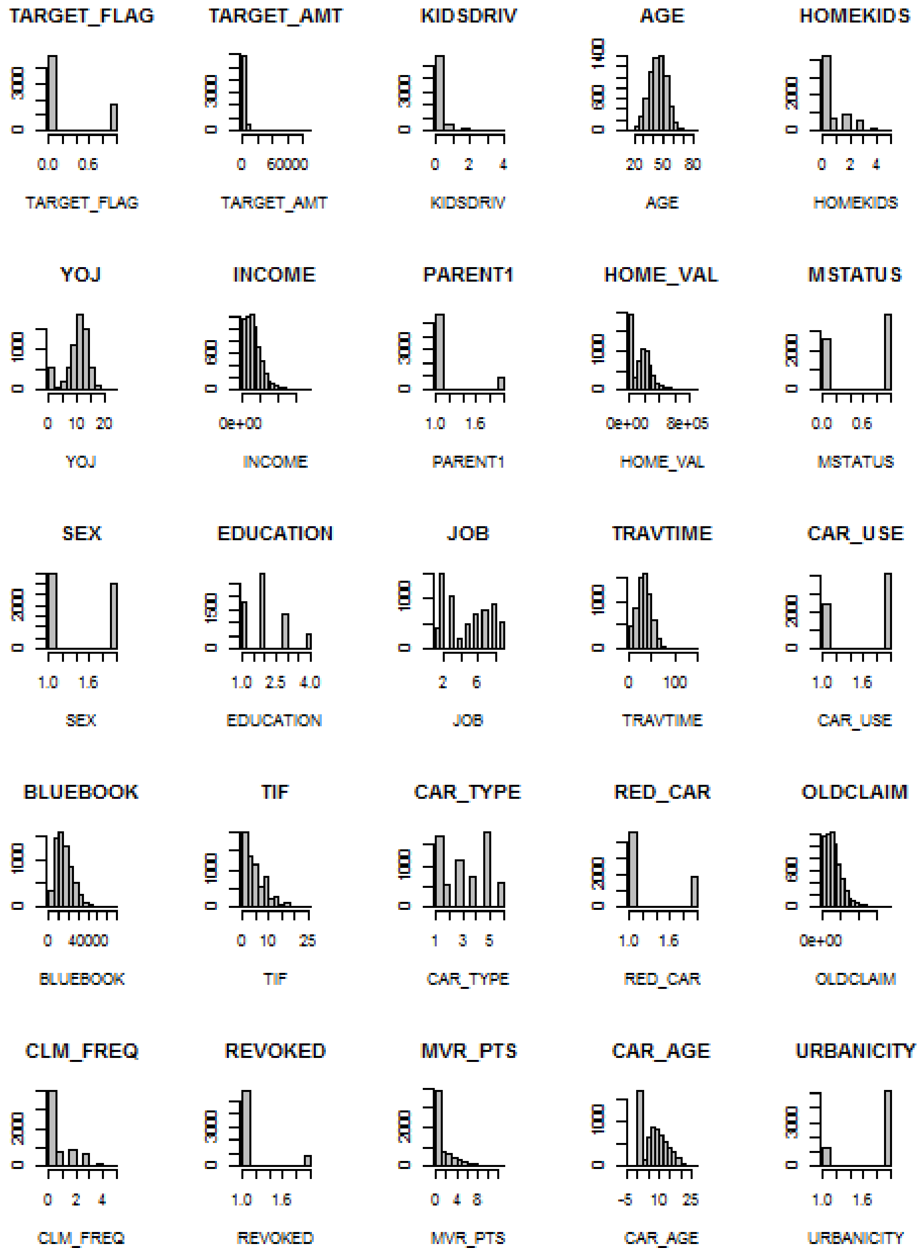
Next we will deal with the issue of missing variables. We have 2 options, either impute the missing variables or remove them. Normally we impute the variables as we want to have as many records as we can get to better our models. But in this case, I feel that it will be better to remove the rows with missing data. Since we have over 8000 records and will be left with about 6000, removing the records won't affect the accuracy. Also it is not clear whether some of the missing data actually means something, so by imputing that data, we will be in fact disrupting it. So we will remove records containing any missing data and proceed.

We will separate continuous variables to take a closer look at them separately, also we will remove factors from categorical variables and convert them to numerical values as we are having difficulty working with data in current format, also it appears that MSTATUS variables has character strings No and Yes, we will replace those with numerical 0 and 1 so that we can use this values in our models.

```
## 'data.frame':    6448 obs. of  25 variables:
##  $ TARGET_FLAG: int  0 0 0 1 1 0 1 0 0 1 ...
##  $ TARGET_AMT : num  0 0 0 2946 2501 ...
##  $ KIDSDRIV   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ AGE        : int  60 43 35 34 34 50 53 43 55 53 ...
##  $ HOMEKIDS   : int  0 0 1 1 0 0 0 0 0 0 ...
##  $ YOJ        : int  11 11 10 12 10 7 14 5 11 11 ...
##  $ INCOME     : num  67349 91449 16039 125301 62978 ...
##  $ PARENT1    : num  1 1 1 2 1 1 1 1 1 1 ...
##  $ HOME_VAL   : num  0 257252 124191 0 0 ...
##  $ MSTATUS    : num  0 0 1 0 0 0 0 1 1 0 ...
##  $ SEX        : num  2 2 1 1 1 2 1 1 2 2 ...
##  $ EDUCATION  : num  4 2 2 1 1 1 3 3 1 4 ...
##  $ JOB        : num  8 2 3 2 3 8 6 8 7 1 ...
##  $ TRAVTIME   : int  14 22 5 46 34 48 15 36 25 64 ...
##  $ CAR_USE    : num  2 1 2 1 2 1 2 2 1 1 ...
##  $ BLUEBOOK   : num  14230 14940 4010 17430 11200 ...
##  $ TIF        : int  11 1 4 1 1 7 1 7 7 6 ...
##  $ CAR_TYPE   : num  1 1 5 4 5 6 4 1 6 2 ...
##  $ RED_CAR    : num  2 2 1 1 1 1 1 1 2 2 ...
##  $ OLDCLAIM   : num  67349 91449 16039 125301 62978 ...
##  $ CLM_FREQ   : int  2 0 2 0 0 0 0 0 2 0 ...
##  $ REVOKED    : num  1 1 1 1 1 1 1 1 2 1 ...
##  $ MVR_PTS    : int  3 0 3 0 0 1 0 0 3 3 ...
##  $ CAR_AGE    : int  18 1 10 7 1 17 11 1 9 10 ...
##  $ URBANICITY : num  2 2 2 2 2 1 2 1 2 2 ...
##  - attr(*, "na.action")=Class 'omit'  Named int [1:1713] 4 5 7 8 21 29 45
46 49 54 ...
##   .. ..- attr(*, "names")= chr [1:1713] "4" "5" "7" "8" ...
```

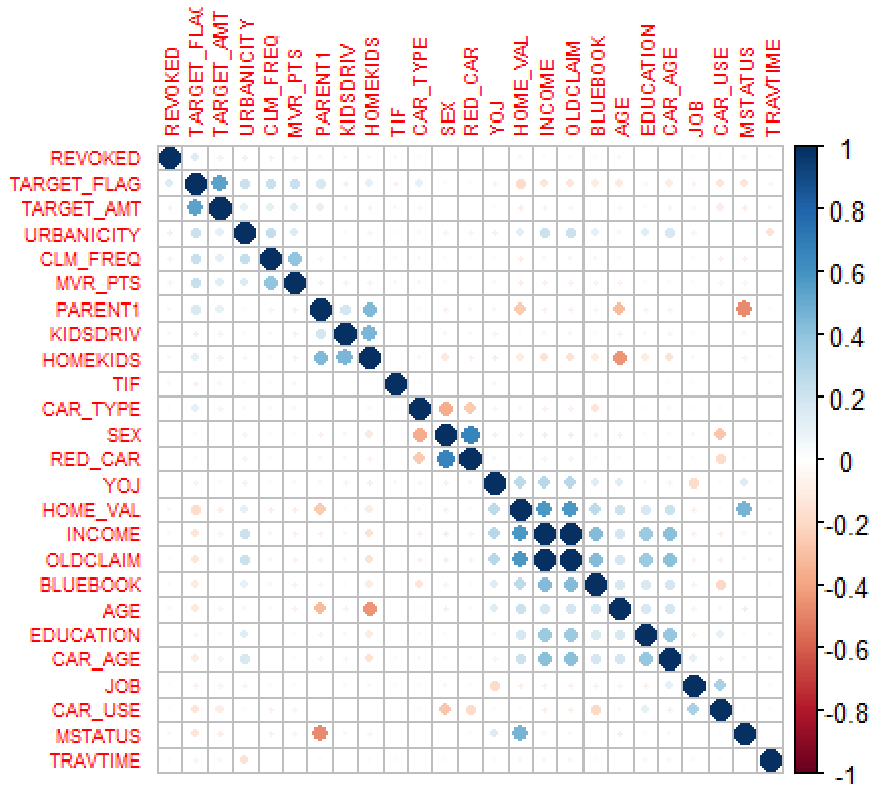
Now our data is clean, has all the data types we can use in our models and we can start working with it.

We will first plot the histograms and review how data is distributed



It appears that some of the continuous variables are skewed, but since we separated them we will work on transformation later on.

Next we will create the correlation plot to visually identify correlated predictors.

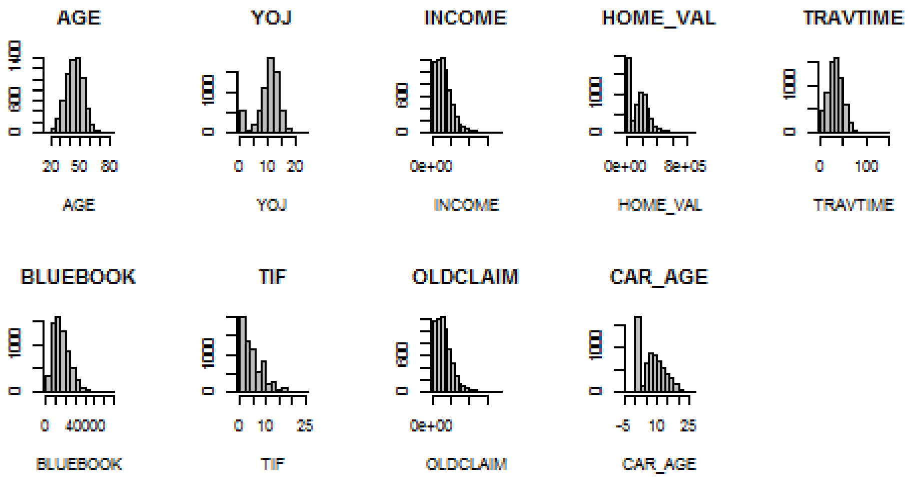


Overall looks like variables are not heavily correlated. We have couple of instances of high correlations like INCOME and OLDCLAIM, but other then that correlation should not be a big issue for us.

2 DATA PREPARATION

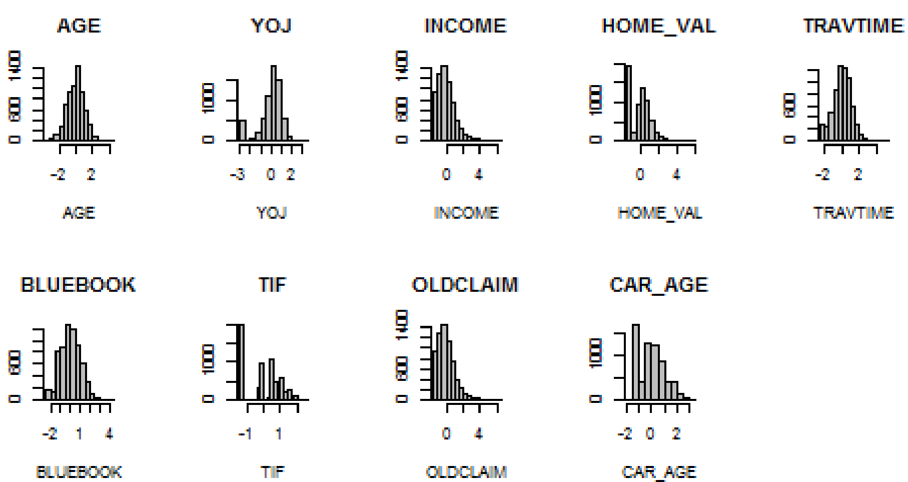
Raw Continuous variables

Since we detected some skewness in data, we separated some of the continues variables, lets view them as histograms



Box Cox Transformed

Now lets use Box-cox transformation and review the histograms to the raw data above



It appears that we have not fully removed skew, but we have certainly improved the distribution in our continuous variables. So we will use the transformed continuous predictor variables so we will merge them with our categorical variables.

Next we will split our data and assign 70% for training and 30% for validation.

```
## [1] 6448   25

## [1] 4513   25

## [1] 1935   25
```

3. Build Model

At this point we are ready to start building our models. We need to build 2 models. The first model will be logistic regression model and will be tasked with predicting if this person was in the car crash. Once we have that, our next model will be tasked with predicting the target amount as a result. We will be using a linear regression model for that.

Model 1.1

For our first model we will use all available to use variables and see how they perform. This model will act as a benchmark for the rest and we will see if we cab improve results from here.

```
##
## Call:
## glm(formula = TARGET_FLAG ~ . - TARGET_AMT, family = binomial(link =
"logit"),
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5295  -0.7385  -0.4484   0.7940   2.8196
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.51292    0.66861  -8.245  < 2e-16 ***
## KIDSDRIV     0.40879    0.11256   3.632 0.000281 ***
## AGE         -0.07391    0.06758  -1.094 0.274080
## HOMEKIDS    -0.02753    0.07034  -0.391 0.695505
## YOJ         -0.03875    0.06369  -0.608 0.542924
## INCOME      -0.27034    0.09316  -2.902 0.003709 **
## PARENT1      0.60767    0.21764   2.792 0.005237 **
## HOME_VAL    -0.21222    0.08986  -2.362 0.018192 *
## MSTATUS     -0.12091    0.17516  -0.690 0.490032
## SEX         -0.22734    0.16831  -1.351 0.176791
## EDUCATION    0.10216    0.07665   1.333 0.182560
## JOB         -0.05999    0.02415  -2.484 0.012983 *
## TRAVTIME     0.26526    0.06052   4.383 1.17e-05 ***
## CAR_USE     -0.78131    0.13376  -5.841 5.19e-09 ***
## BLUEBOOK    -0.22460    0.06772  -3.317 0.000911 ***
## TIF         -0.19573    0.05773  -3.391 0.000697 ***
## CAR_TYPE     0.13370    0.03671   3.642 0.000271 ***
## RED_CAR     -0.13802    0.17375  -0.794 0.426976
## OLDCLAIM          NA         NA      NA       NA
## CLM_FREQ     0.11095    0.05073   2.187 0.028757 *
## REVOKED      0.69872    0.15735   4.441 8.97e-06 ***
```

```
## MVR_PTS      0.09000    0.02648   3.399 0.000677 ***
## CAR_AGE     -0.14965    0.06786  -2.205 0.027427 *
## URBANICITY   2.17401    0.21753   9.994  < 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: 2285.6  on 1934  degrees of freedom
## Residual deviance: 1829.3  on 1912  degrees of freedom
## AIC: 1875.3
##
## Number of Fisher Scoring iterations: 5
```

Model 1.2

For the second model we will remove OLD_MODEL variable, it was highly correlated with INCOME and also it does not provide any benefit as we can clearly see from above, we will see if removing it will improve our results.

```
##
## Call:
## glm(formula = TARGET_FLAG ~ . - TARGET_AMT, family = binomial(link =
"logit"),
##     data = train1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5295  -0.7385  -0.4484   0.7940   2.8196
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.51292    0.66861  -8.245  < 2e-16 ***
## KIDSDRIV     0.40879    0.11256   3.632 0.000281 ***
## AGE         -0.07391    0.06758  -1.094 0.274080
## HOMEKIDS    -0.02753    0.07034  -0.391 0.695505
## YOJ         -0.03875    0.06369  -0.608 0.542924
## INCOME      -0.27034    0.09316  -2.902 0.003709 **
## PARENT1      0.60767    0.21764   2.792 0.005237 **
## HOME_VAL    -0.21222    0.08986  -2.362 0.018192 *
## MSTATUS     -0.12091    0.17516  -0.690 0.490032
## SEX         -0.22734    0.16831  -1.351 0.176791
## EDUCATION    0.10216    0.07665   1.333 0.182560
## JOB         -0.05999    0.02415  -2.484 0.012983 *
## TRAVTIME     0.26526    0.06052   4.383 1.17e-05 ***
## CAR_USE     -0.78131    0.13376  -5.841 5.19e-09 ***
## BLUEBOOK    -0.22460    0.06772  -3.317 0.000911 ***
## TIF         -0.19573    0.05773  -3.391 0.000697 ***
## CAR_TYPE     0.13370    0.03671   3.642 0.000271 ***
## RED_CAR     -0.13802    0.17375  -0.794 0.426976
## CLM_FREQ     0.11095    0.05073   2.187 0.028757 *
## REVOKED      0.69872    0.15735   4.441 8.97e-06 ***
## MVR_PTS      0.09000    0.02648   3.399 0.000677 ***
```

```
## CAR_AGE     -0.14965    0.06786  -2.205 0.027427 *
## URBANICITY   2.17401    0.21753   9.994  < 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: 2285.6  on 1934  degrees of freedom
## Residual deviance: 1829.3  on 1912  degrees of freedom
## AIC: 1875.3
##
## Number of Fisher Scoring iterations: 5
```

Model 1.3

For the 3rd model, we will try to remove all variables that have p value higher then 0.05 as they are not significant. We will see if we get better results

```
##
## Call:
## glm(formula = TARGET_FLAG ~ . - TARGET_AMT, family = binomial(link =
"logit"),
##     data = train1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5481  -0.7519  -0.4539   0.8000   2.7838
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.24858    0.60942 -10.253  < 2e-16 ***
## KIDSDRIV     0.37498    0.11090   3.381 0.000721 ***
## HOMEKIDS     0.02610    0.06400   0.408 0.683425
## YOJ         -0.06372    0.06236  -1.022 0.306926
## INCOME      -0.34756    0.08994  -3.864 0.000111 ***
## PARENT1      0.63055    0.21494   2.934 0.003350 **
## HOME_VAL    -0.21580    0.08940  -2.414 0.015788 *
## MSTATUS     -0.12189    0.17430  -0.699 0.484353
## EDUCATION    0.07552    0.07576   0.997 0.318849
## JOB         -0.06540    0.02383  -2.744 0.006062 **
## TRAVTIME     0.25435    0.06005   4.236 2.28e-05 ***
## CAR_USE     -0.62080    0.12401  -5.006 5.56e-07 ***
## TIF         -0.20163    0.05727  -3.520 0.000431 ***
## CAR_TYPE     0.17373    0.03388   5.127 2.94e-07 ***
## CLM_FREQ     0.10864    0.05052   2.150 0.031535 *
## REVOKED      0.68247    0.15654   4.360 1.30e-05 ***
## MVR_PTS      0.09499    0.02635   3.605 0.000313 ***
## CAR_AGE     -0.15239    0.06736  -2.262 0.023675 *
## URBANICITY   2.10513    0.21514   9.785  < 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: 2285.6  on 1934  degrees of freedom
## Residual deviance: 1847.7  on 1916  degrees of freedom
## AIC: 1885.7
##
## Number of Fisher Scoring iterations: 5
```

Linear Model

Here we will recreate train and test set but based only on the subset we are interested in which is where person was involved in a car crash. We will then create train and test data set similar to how we did it above.

```
## [1] 1192   25

## [1] 511  25
```

Model 2.1

We will start with including of every possible predictor to establish a benchmark model

```
##
## Call:
## lm(formula = TARGET_AMT ~ . - TARGET_FLAG, data = train_lm)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -10169  -3853  -1783    668  68907
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10589.06    5123.66   2.067   0.0393 *
## KIDSDRIV      201.79     810.47   0.249   0.8035
## AGE           387.66     463.28   0.837   0.4031
## HOMEKIDS      765.65     528.68   1.448   0.1482
## YOJ            53.30     454.86   0.117   0.9068
## INCOME       -212.92     710.50  -0.300   0.7646
## PARENT1       -58.37    1447.60  -0.040   0.9679
## HOME_VAL     1036.82     653.66   1.586   0.1134
## MSTATUS     -1131.40    1299.68  -0.871   0.3844
## SEX          2542.01    1180.81   2.153   0.0318 *
## EDUCATION    -228.59     608.70  -0.376   0.7074
## JOB           106.81     178.79   0.597   0.5505
## TRAVTIME      161.93     446.28   0.363   0.7169
## CAR_USE      -673.50     969.02  -0.695   0.4874
## BLUEBOOK     1129.01     472.18   2.391   0.0172 *
## TIF           263.84     432.64   0.610   0.5423
## CAR_TYPE     -203.21     282.01  -0.721   0.4715
## RED_CAR     -2660.12    1225.14  -2.171   0.0304 *
## OLDCLAIM          NA         NA      NA       NA
## CLM_FREQ       93.35     352.05   0.265   0.7910
## REVOKED      -797.85    1037.15  -0.769   0.4421
## MVR_PTS       164.93     172.51   0.956   0.3395
## CAR_AGE      -643.98     506.39  -1.272   0.2041
```

```
## URBANICITY  -1200.44    1763.07  -0.681   0.4963
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9441 on 488 degrees of freedom
## Multiple R-squared:  0.05753,    Adjusted R-squared:  0.01504
## F-statistic: 1.354 on 22 and 488 DF,  p-value: 0.1311
```

Model 2.2

Next we will use Stepwise selection to see if we can come up with better results, we will try all 3 methods.

Full stepwise

```
##
## Call:
## lm(formula = TARGET_AMT ~ HOMEKIDS + HOME_VAL + SEX + BLUEBOOK +
##     RED_CAR + CAR_AGE, data = train_lm)
##
## Residuals:
##    Min      1Q Median      3Q     Max
##  -9155  -3649  -1752     224   68562
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5173.6     1484.5   3.485 0.000535 ***
## HOMEKIDS       627.5      356.3   1.761 0.078773 .
## HOME_VAL       651.3      459.2   1.418 0.156723
## SEX           2870.5     1084.1   2.648 0.008356 **
## BLUEBOOK      1224.0      422.6   2.896 0.003941 **
## RED_CAR      -2740.7     1184.1  -2.315 0.021036 *
## CAR_AGE       -642.7      439.2  -1.463 0.144010
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9350 on 504 degrees of freedom
## Multiple R-squared:  0.04514,    Adjusted R-squared:  0.03378
## F-statistic: 3.971 on 6 and 504 DF,  p-value: 0.0006804
```

Forward stepwise

```
##
## Call:
## lm(formula = TARGET_AMT ~ (TARGET_FLAG + KIDSDRIV + AGE + HOMEKIDS +
##     YOJ + INCOME + PARENT1 + HOME_VAL + MSTATUS + SEX + EDUCATION +
##     JOB + TRAVTIME + CAR_USE + BLUEBOOK + TIF + CAR_TYPE + RED_CAR +
##     OLDCLAIM + CLM_FREQ + REVOKED + MVR_PTS + CAR_AGE + URBANICITY) -
##     TARGET_FLAG, data = train_lm)
##
## Residuals:
##    Min      1Q Median      3Q     Max
## -10169  -3853  -1783     668   68907
##
## Coefficients: (1 not defined because of singularities)
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10589.06    5123.66   2.067   0.0393 *
## KIDSDRIV      201.79     810.47   0.249   0.8035
## AGE           387.66     463.28   0.837   0.4031
## HOMEKIDS      765.65     528.68   1.448   0.1482
## YOJ            53.30     454.86   0.117   0.9068
## INCOME       -212.92     710.50  -0.300   0.7646
## PARENT1       -58.37    1447.60  -0.040   0.9679
## HOME_VAL     1036.82     653.66   1.586   0.1134
## MSTATUS     -1131.40    1299.68  -0.871   0.3844
## SEX          2542.01    1180.81   2.153   0.0318 *
## EDUCATION    -228.59     608.70  -0.376   0.7074
## JOB           106.81     178.79   0.597   0.5505
## TRAVTIME      161.93     446.28   0.363   0.7169
## CAR_USE      -673.50     969.02  -0.695   0.4874
## BLUEBOOK     1129.01     472.18   2.391   0.0172 *
## TIF           263.84     432.64   0.610   0.5423
## CAR_TYPE     -203.21     282.01  -0.721   0.4715
## RED_CAR     -2660.12    1225.14  -2.171   0.0304 *
## OLDCLAIM          NA         NA      NA       NA
## CLM_FREQ       93.35     352.05   0.265   0.7910
## REVOKED      -797.85    1037.15  -0.769   0.4421
## MVR_PTS       164.93     172.51   0.956   0.3395
## CAR_AGE      -643.98     506.39  -1.272   0.2041
## URBANICITY  -1200.44    1763.07  -0.681   0.4963
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9441 on 488 degrees of freedom
## Multiple R-squared:  0.05753,    Adjusted R-squared:  0.01504
## F-statistic: 1.354 on 22 and 488 DF,  p-value: 0.1311
```

Backward Stepwise

```
##
## Call:
## lm(formula = TARGET_AMT ~ HOMEKIDS + HOME_VAL + SEX + BLUEBOOK +
##     RED_CAR + CAR_AGE, data = train_lm)
##
## Residuals:
##    Min      1Q Median      3Q     Max
##  -9155  -3649  -1752     224   68562
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5173.6     1484.5   3.485 0.000535 ***
## HOMEKIDS       627.5      356.3   1.761 0.078773 .
## HOME_VAL       651.3      459.2   1.418 0.156723
## SEX           2870.5     1084.1   2.648 0.008356 **
## BLUEBOOK      1224.0      422.6   2.896 0.003941 **
## RED_CAR      -2740.7     1184.1  -2.315 0.021036 *
## CAR_AGE       -642.7      439.2  -1.463 0.144010
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 9350 on 504 degrees of freedom
## Multiple R-squared:  0.04514,    Adjusted R-squared:  0.03378
## F-statistic: 3.971 on 6 and 504 DF,  p-value: 0.0006804
```

Model 2.3

Since the Backward model selected only 2 predictors Sex and Bluebook, but Sex actually has a p value higher then 0.05, we can try to remove it and run the model based on just Bluebook.

```
##
## Call:
## lm(formula = TARGET_AMT ~ BLUEBOOK, data = train_lm)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -7651  -3552  -1971   -184  71114
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6358.1      425.3  14.949  < 2e-16 ***
## BLUEBOOK      1213.9      407.3   2.981  0.00301 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9440 on 509 degrees of freedom
## Multiple R-squared:  0.01715,    Adjusted R-squared:  0.01522
## F-statistic: 8.884 on 1 and 509 DF,  p-value: 0.003015
```

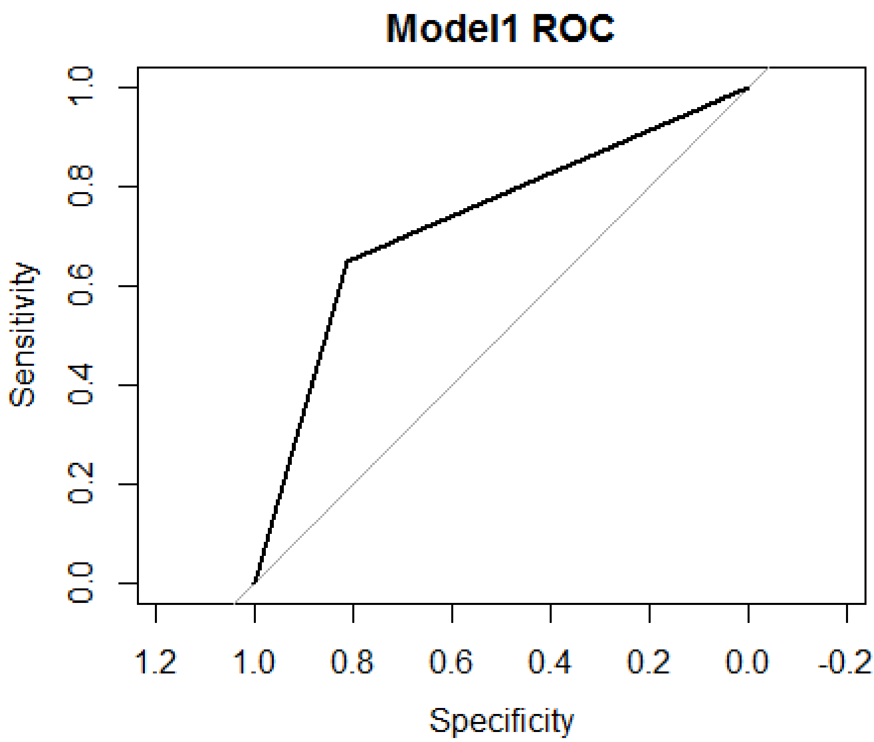
4. SELECT MODELS

In this section we will closely review all the test models and select 2 models we will use to complete our forecast. We will use a number of various metrics to do that.

Model1.1

```
##           Reference
## Prediction    0    1
##          0 3104  720
##          1  243  446

##          Sensitivity          Specificity       Pos Pred Value
##           0.38250429           0.92739767           0.64731495
##       Neg Pred Value            Precision               Recall
##           0.81171548           0.64731495           0.38250429
##                   F1           Prevalence       Detection Rate
##           0.48086253           0.25836472           0.09882561
## Detection Prevalence    Balanced Accuracy
##           0.15267006           0.65495098
```




```
## Area under the curve: 0.7295
```

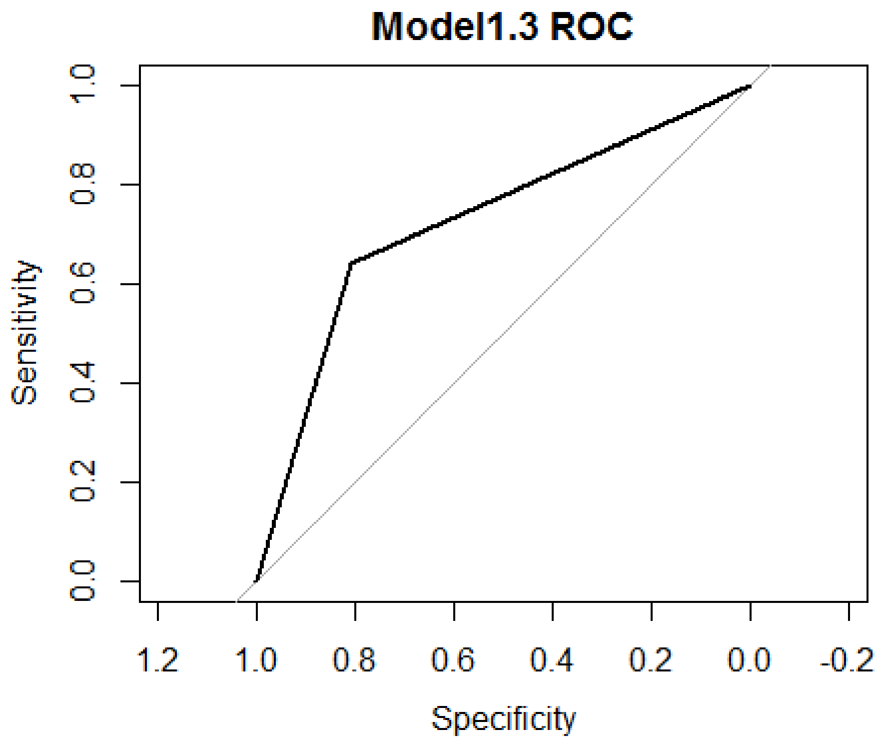
Model1.2

Since model12 has not improved our results and seems to have same results, we will skip it and move to model 1.3

Model1.3

```
##           Reference
## Prediction    0    1
##          0 3110  743
##          1  237  423

##          Sensitivity          Specificity       Pos Pred Value
##           0.36277873           0.92919032           0.64090909
##       Neg Pred Value            Precision               Recall
##           0.80716325           0.64090909           0.36277873
##                   F1           Prevalence       Detection Rate
##           0.46330778           0.25836472           0.09372923
## Detection Prevalence    Balanced Accuracy
##           0.14624418           0.64598453
```




```
## Area under the curve: 0.724
```

Final Models

For the final Logistic Regression model we will choose model1 for Logistic regression model. It has better AUC, sensitivity and pretty much every other score, so we will use model1.

For the final linear regression model we will go with the model generated by backward stepwiseas it has the lowest RMSE.

Predicting on testdata

Since we have altered the training set we will have to do the same for the test set in order to make the predictions. Additionally we will need to deal with missing data since we cannot predict on the records that is missing data. We could omit the records with missing data, but since we do not know whether this is acceptable, we will go ahead and impute the missing data in test set and predict based on that.

Previewing first 30 Records

TARGET_FLAG	TARGET_AMT
0	0.000
0	0.000
0	0.000
0	0.000
0	0.000
0	0.000
0	0.000
0	0.000
0	0.000
0	0.000
0	0.000
1	4629.324
1	5356.743
0	0.000
0	0.000
1	5525.491
1	4058.642
0	0.000
1	4882.546
0	0.000
0	0.000
0	0.000
0	0.000
0	0.000
0	0.000
0	0.000
0	0.000
0	0.000
0	0.000
0	0.000

Appendix A

R markdown file with code along with full predictions csv file available at:
https://github.com/jelikish/Cuny1/tree/master/Spring2018/621/hw4