Auto Insurance Assignment #2 Derek Hughes
PRED 411 – Sec 55
Winter 2015

INTRODUCTION

Auto insurance is a fickle game played between the insurer and those seeking insurance. On the one hand, we have the insurer who is providing a required service but doesn't want to lose money insuring the wrong people for the too small of an insurance rate. If they didn't evaluate customers using some type of risk criteria, they wouldn't have any clue how to differentiate between good, low risk drivers and high risk drivers who are likely to cost the insurance company money.

Sure, the insurance company could just save some time and charge everyone the same rate – a rate that will cover the worst case scenario (i.e. a rate that will cover the highest risk drivers). However, some of the free market powers still remain and insurance companies know they must balance between protecting their interests from the high risk drivers and providing the best rate possible to customers. This requires the creation of a model that evaluates the risk of each individual driver and determines the probability that they will crash their car. In addition, this probability is then multiplied against a severity model to determine the total expected loss for insuring each driver.

The following paper describes the processes used to create three of these models, their results, and the metrics used to evaluate how well the models performed against each other.

DATA EXPLORATION

Step 1:

Step one in model creation is usually to evaluate the data we are utilizing. Here we have approximately 8000 records of individual customers. Each customer is described in 25 variables and has two "target" variables. One target variable represents whether or not the customer had and accident and, if so, the other target variable shows how much money it cost to cover the accident costs.

Alphabetic List of Variables and Attributes					
#	**Variable**	**Type**	**Len**	**Format**	**Label**
5	AGE	Num	8	4.	Age
17	BLUEBOOK	Num	8	DOLLAR10.	Value of Vehicle
25	CAR_AGE	Num	8	4.	Vehicle Age
19	CAR_TYPE	Char	11		Type of Car
16	CAR_USE	Char	10		Vehicle Use
22	CLM_FREQ	Num	8		#Claims(Past 5 Years)
13	EDUCATION	Char	13		Max Education Level
6	HOMEKIDS	Num	8	4.	#Children @Home
10	HOME_VAL	Num	8	DOLLAR10.	Home Value
8	INCOME	Num	8	DOLLAR10.	Income
1	INDEX	Num	8		
14	JOB	Char	13		Job Category
4	KIDSDRIV	Num	8	4.	#Driving Children
11	MSTATUS	Char	5		Marital Status
24	MVR_PTS	Num	8	5.	Motor Vehicle Record Points
21	OLDCLAIM	Num	8	DOLLAR12.	Total Claims(Past 5 Years)
9	PARENT1	Char	3		Single Parent
20	RED_CAR	Char	3		A Red Car
23	REVOKED	Char	3		License Revoked (Past 7 Years)
12	SEX	Char	3		Gender
3	TARGET_AMT	Num	8		

Alphabetic List of Variables and Attributes					
#	Variable	Type	Len	Format	Label
2	TARGET_FLAG	Num	8		
18	TIF	Num	8		Time in Force
15	TRAVTIME	Num	8	4.	Distance to Work
26	URBANICITY	Char	21		Home/Work Area
7	YOJ	Num	8	4.	Years on Job

Step 2 & Step 3:

Step two and three we quickly look over the variables and remove any variables that are clearly excessive or unnecessary. Since this first model is about determining the probability that a customer will crash their car, we do not need to know how much it cost to repair their car. Thus, we remove the variable TARGET_AMT. Furthermore, we can see that the number of observations directly coincides with the INDEX number. This is redundant so we remove the INDEX variable as well.

We check to confirm that the new dataset has those variables removed. As we can see below, both INDEX and TARGET_AMT have been removed from the dataset.

Obs	TARGET_FLAG	KIDSDRIV	AGE	HOMEKIDS	YOJ	INCOME	PARENT1	HOME_VAL
1	0	0	60	0	11	$67,349	No	$0
2	0	0	43	0	11	$91,449	No	$257,252
3	0	0	35	1	10	$16,039	No	$124,191

Step 4:

Step four we explore the data more thoroughly to identify the type of variables, the mean, median, and number of missing values. To evaluate all variables, we must use procedures to handle both character and numeric variables separately. This is because we can't find the mean of a character variable. Thus, first we evaluate the numeric variables with PROC MEANS and then the character variables with PROC FREQ.

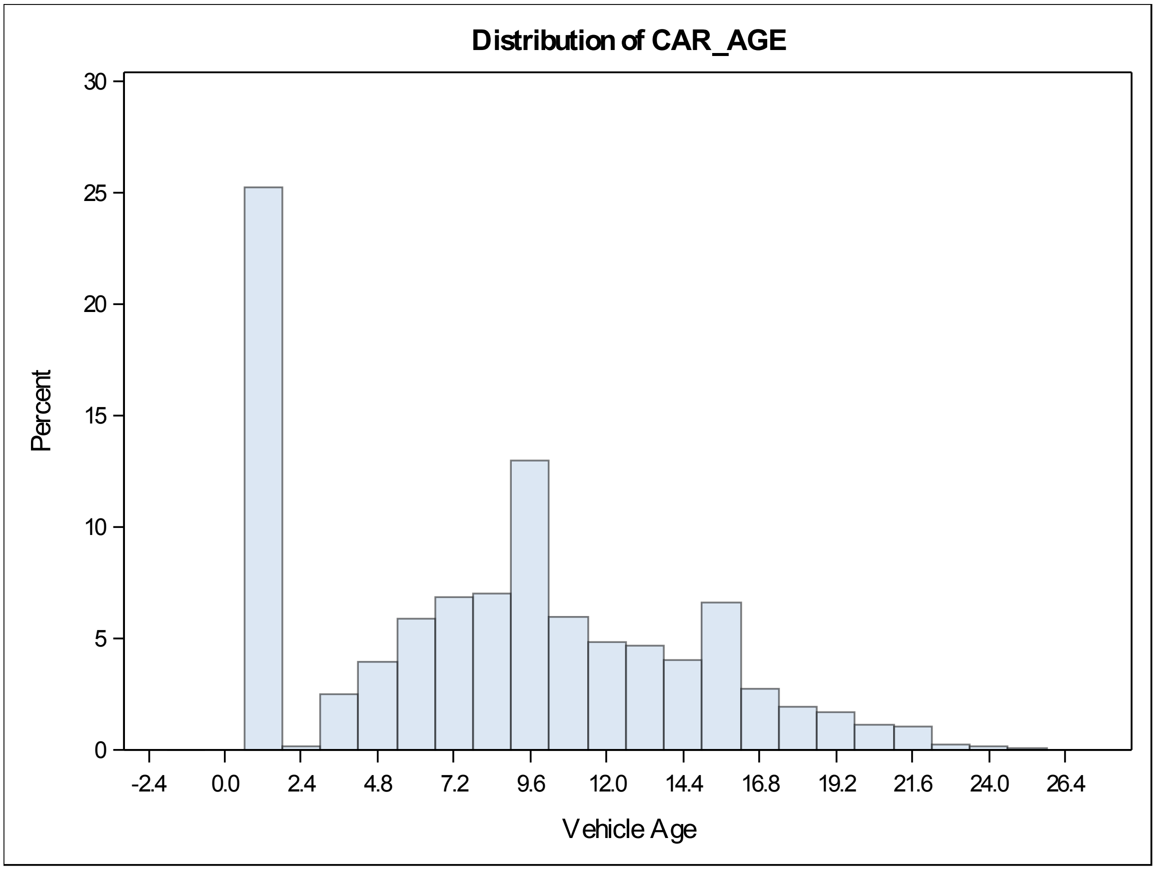
Variable	Label	N	N Miss	Mean	Median	1st Pctl	99th Pctl
TARGET_FLAG		8161	0	0.2638157	0	0	1.0000000
KIDSDRIV	#Driving Children	8161	0	0.1710575	0	0	2.0000000
AGE	Age	8155	6	44.7903127	45.0000000	25.0000000	64.0000000
HOMEKIDS	#Children @Home	8161	0	0.7212351	0	0	4.0000000
YOJ	Years on Job	7707	454	10.4992864	11.0000000	0	17.0000000
INCOME	Income	7716	445	61898.10	54028.17	0	215536.28
HOME_VAL	Home Value	7697	464	154867.29	161159.53	0	500309.15
TRAVTIME	Distance to Work	8161	0	33.4887972	32.8709696	5.0000000	75.1443301
BLUEBOOK	Value of Vehicle	8161	0	15709.90	14440.00	1500.00	39090.00
TIF	Time in Force	8161	0	5.3513050	4.0000000	1.0000000	17.0000000
OLDCLAIM	Total Claims(Past 5	8161	0	4037.08	0	0	42820.00
CLM_FREQ	Years)	8161	0	0.7985541	0	0	4.0000000
MVR_PTS	#Claims(Past 5 Years)	8161	0	1.6955030	1.0000000	0	8.0000000
CAR_AGE	Motor Vehicle Record	7651	510	8.3283231	8.0000000	1.0000000	21.0000000
	Points						
	Vehicle Age						

From the PROC MEANS statement we can see that five numeric variables (AGE, YOJ, INCOME, HOME_VAL, CAR_AGE) are missing values.

AGE is only missing 6 values so we will simply use the mean value to replace those missing values. Also, since only 6 out of 8161 values are missing for AGE we will not go to the trouble to create a new variable to identify the driver had a missing value (M_AGE) for AGE.

The remaining four numeric variables with missing values will be replaced with the mean or median. Also, to determine if any predictive power is found based on a variable having a missing value, we will create missing value flag variables.

Home values can be zero if the driver is a home renter instead of a home owner. So, to determine if a missing home value should be set to zero to indicate a renter or the median value to indicate a home buyer, we look at variables that are strong and positively correlated with the home value variable. We find that years on job, income, and car value have the strongest correlation to home value so we set home value to the median if all of those variables reach predetermined values.



The same process is used for imputing missing car age values. Income is the most highly, positively correlated variable to car age so we will use it to decide how to fill in the missing values.

We run a quick PROC UNIVARIATE for a CAR_AGE histogram and see a fairly normal distribution for 75% of the observations, while approximately 25% of CAR_AGE values are at one year old. Thus, we will set the CAR_AGE to one if the driver's income is in the highest 25% bracket; otherwise, we will set it to the highest frequency value in the histogram which is 9.6 years old.

Job Category				
JOB	**Frequency**	**Percent**	**Cumulative Frequency**	**Cumulative Percent**
	526	6.45	526	6.45
Clerical	1271	15.57	1797	22.02
Doctor	246	3.01	2043	25.03
Home Maker	641	7.85	2684	32.89
Lawyer	835	10.23	3519	43.12
Manager	988	12.11	4507	55.23
Professional	1117	13.69	5624	68.91
Student	712	8.72	6336	77.64
z_Blue Collar	1825	22.36	8161	100.00

From the PROC FREQ statement we can see that 526 missing values are found within the JOB variables. Since this is a variable with categories, we need to decide how to determine which category to use for each missing value. We will determine the category based on arbitrary income values.

Step 6:

Step six is to determine which variables are predictive. Since we are trying to predict the probability a driver will have a crash (TARGET_FLAG = 1), for category variables, we compare the percentage of each value in each category variable that are associated with a driver who had a crash to the average percent who had a crash in that category variable.

If a particular value in a category variable has a much higher percentage of drivers crashing compared to the average percentage of drivers crashing for that variable, then it's reasonable to assume that particular variable will have predictive power and to consider it for inclusion into the model. We check this for each category variable and only keep the predictive variables.

Table of SEX by TARGET_FLAG			
SEX(Gender)	**TARGET_FLAG**		
Frequency **Percent** **Row Pct** **Col Pct**	**0**	**1**	**Total**
M	2825 34.62 74.62 47.02	961 11.78 25.38 44.64	3786 46.39
z_F	3183 39.00 72.75 52.98	1192 14.61 27.25 55.36	4375 53.61
Total	6008 73.62	2153 26.38	8161 100.00

For example, SEX has the values of M and F at 25.38% and 27.25% respectively that crash their car. The average for the variable is 26.38%. The crash percentage for M or F is not much different than the average so we conclude that SEX isn't likely to have much predictive power and we drop it from the model. The same follows with the category variable RED_CAR.

We run the same process with the numeric variables. Only here we compare the variable values from drivers who crashed their car to those who did not crash. If we see a relatively significant difference between the two values then we assume there is a strong chance that variable will be predictive.

TARGET_FLAG	N Obs	Variable	Label	Mean	Median
0	6008	TARGET_FLAG		0	0
		KIDSDRIV	#Driving Children	0.1393142	0
		HOMEKIDS	#Children @Home	0.6439747	0
		TRAVTIME	Distance to Work	33.0303446	32.3028412
		BLUEBOOK	Value of Vehicle	16230.95	15000.00
		TIF	Time in Force	5.5557590	6.0000000
		OLDCLAIM	Total Claims(Past 5 Years)	3311.59	0
		CLM_FREQ	#Claims(Past 5 Years)	0.6486352	0
		MVR_PTS	Motor Vehicle Record Points	1.4137816	1.0000000
		IMP_AGE		45.3227015	46.0000000
		IMP_YOJ		10.6623275	11.0000000
		M_YOJ		0.0550932	0
		IMP_INCOME		65725.93	61898.10
		M_INCOME		0.0557590	0
		IMP_HOME_VAL		163124.54	165640.54
		M_HOME_VAL		0.0570905	0
		IMP_CAR_AGE		8.5760985	9.0000000
		M_CAR_AGE		0.0612517	0
		M_JOB		0	0
1	2153	TARGET_FLAG		1.0000000	1.0000000
		KIDSDRIV	#Driving Children	0.2596377	0
		HOMEKIDS	#Children @Home	0.9368323	0
		TRAVTIME	Distance to Work	34.7681203	34.4417857
		BLUEBOOK	Value of Vehicle	14255.90	12600.00
		TIF	Time in Force	4.7807710	4.0000000
		OLDCLAIM	Total Claims(Past 5 Years)	6061.55	2448.00
		CLM_FREQ	#Claims(Past 5 Years)	1.2169066	1.0000000
		MVR_PTS	Motor Vehicle Record Points	2.4816535	2.0000000
		IMP_AGE		43.3046686	43.0000000
		IMP_YOJ		10.0443159	11.0000000
		M_YOJ		0.0571296	0
		IMP_INCOME		51216.43	46604.18
		M_INCOME		0.0510915	0
		IMP_HOME_VAL		111398.93	108287.40
		M_HOME_VAL		0.0562007	0
		IMP_CAR_AGE		7.4667905	8.0000000
		M_CAR_AGE		0.0659545	0
		M_JOB		0	0

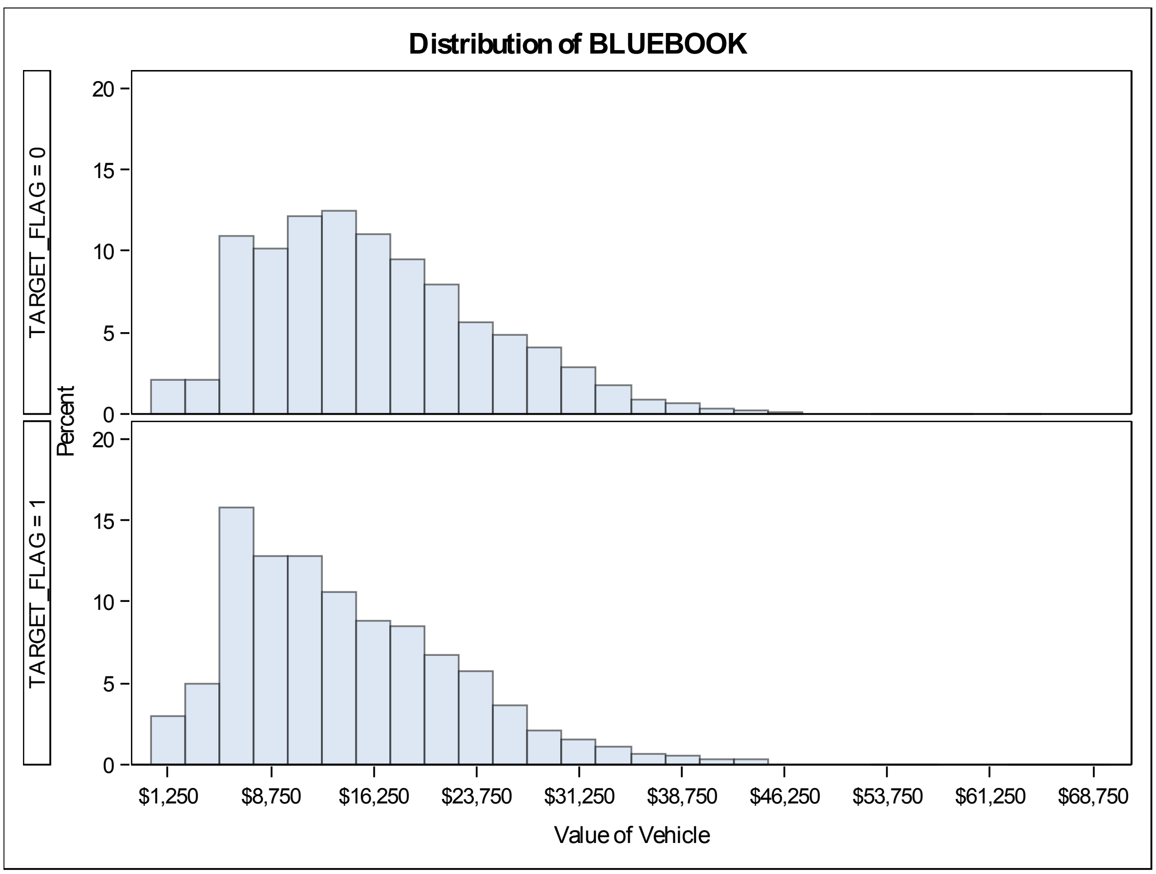
For example, KIDSDRIV has the value .13 for drivers who do not crash but .26, approximately twice as large, for drivers who do have a crash. This is a significant difference and therefore should be included in the model.

On the other hand, distance to work for those who don't have a crash is 33 minutes while it is 34.8 for those who do have a crash. This isn't that much of a difference so we will likely remove this variable, TRAVTIME, from the model.

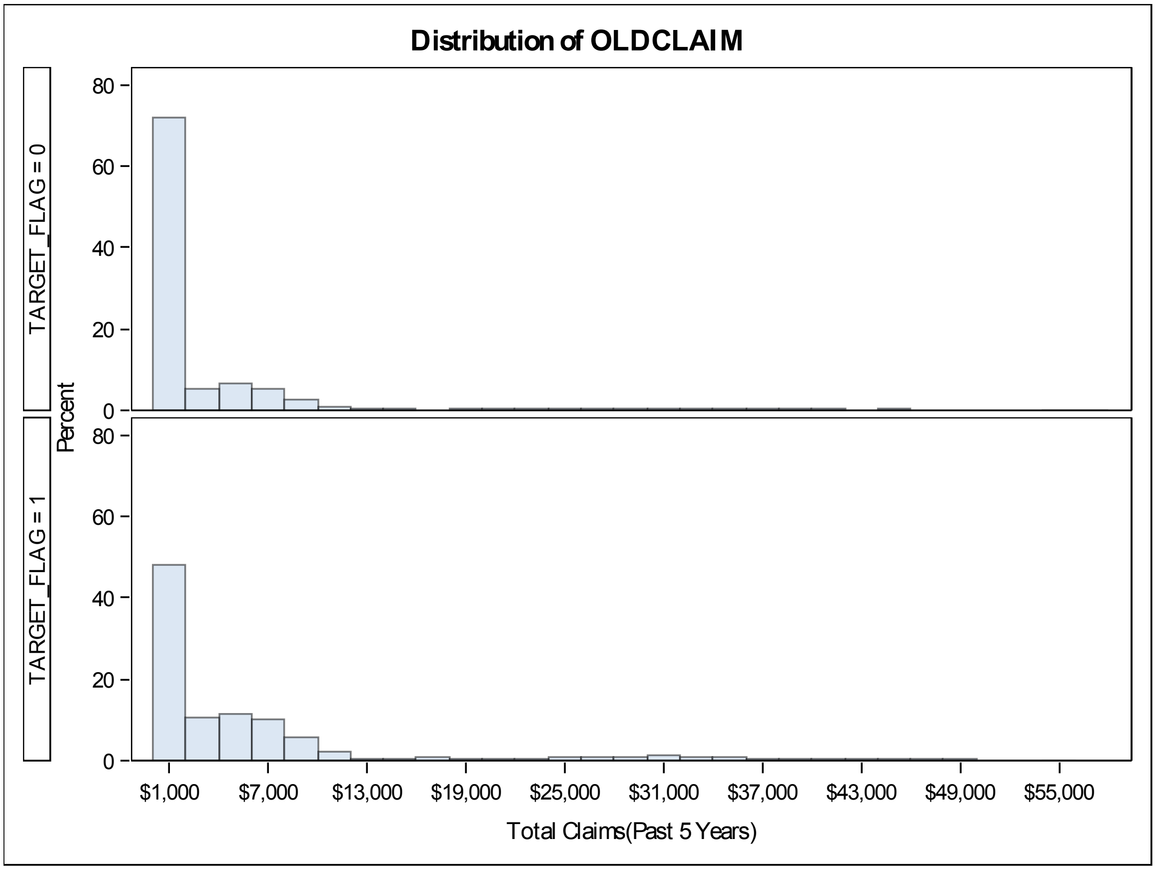
Ultimately, using the same approach, these numeric variables were removed from the model: TRAVTIME, IMP_AGE, IMP_YOJ.

Step 7:

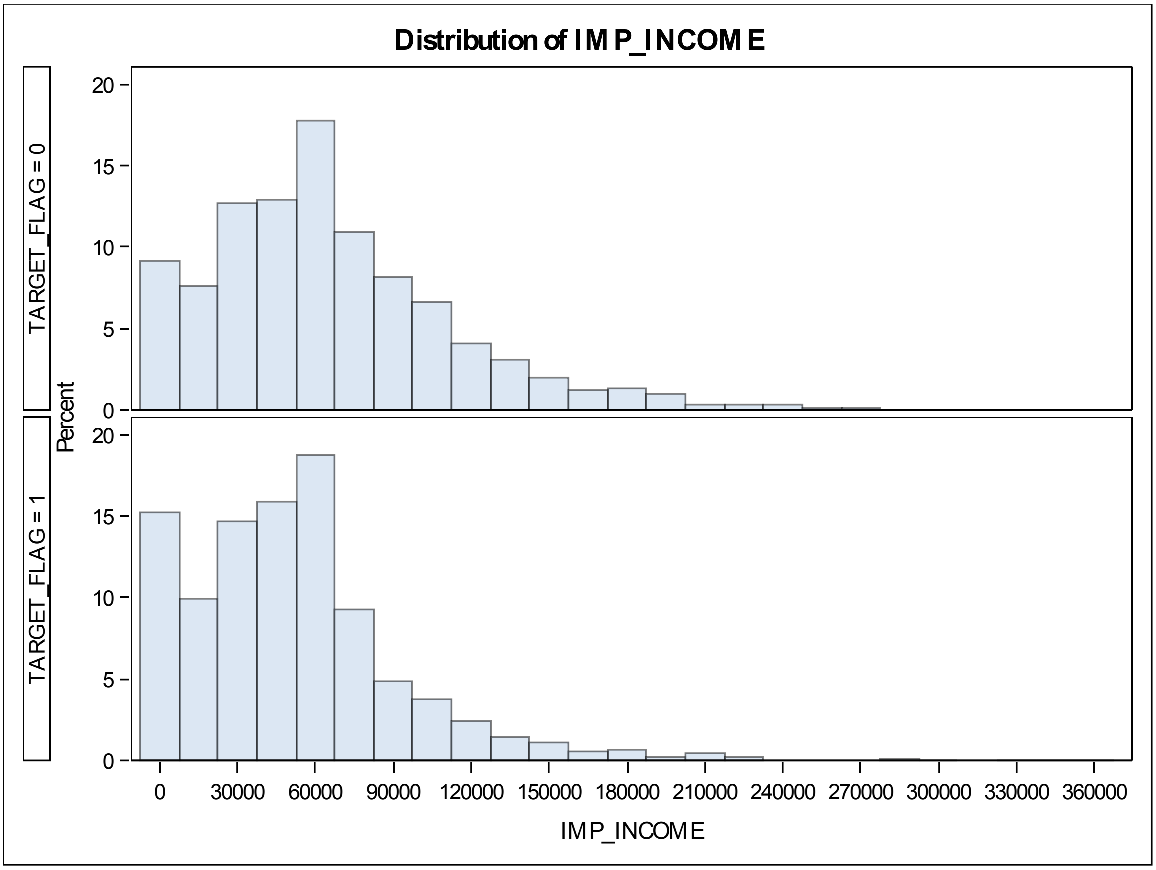
Step seven is the search and investigation of outliers. We use PROC UNIVARIATE to create histograms for the variables to check for outliers and if the variable can be trimmed to create a better distribution.



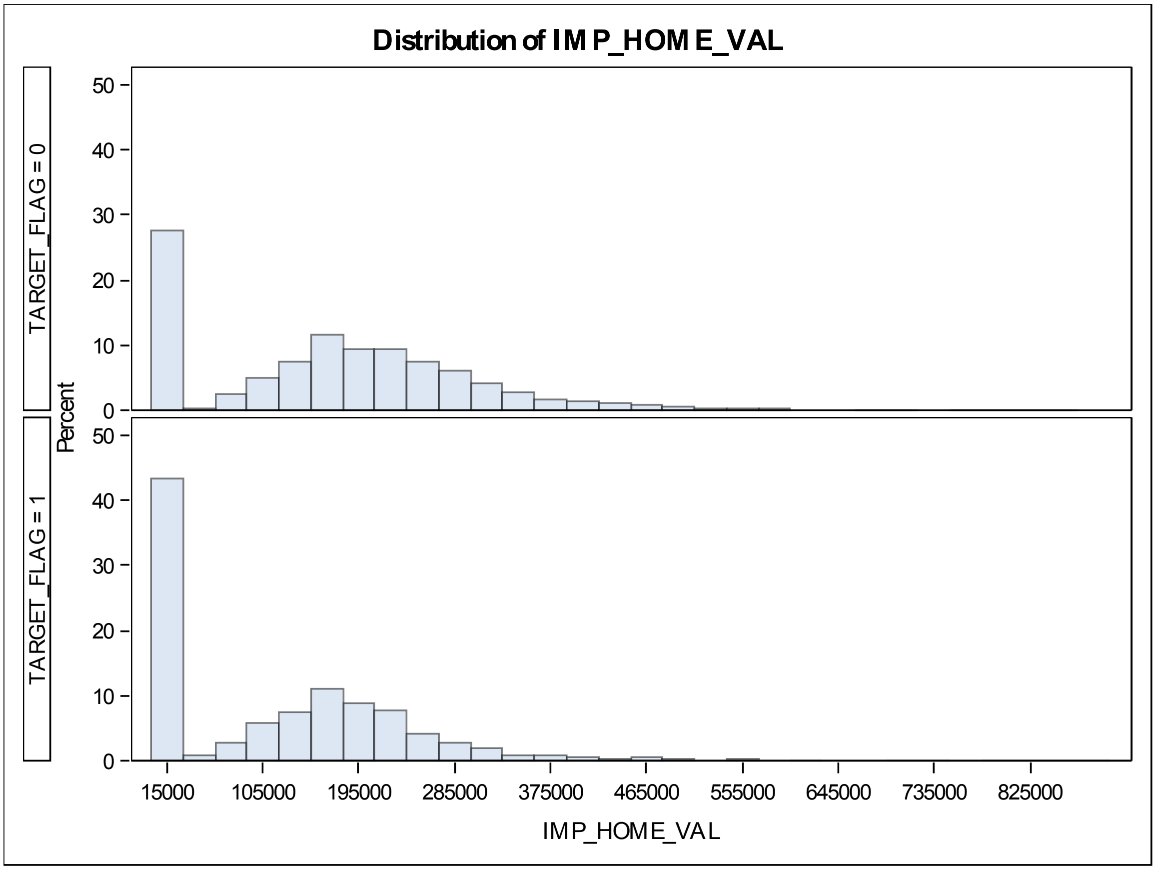
For bluebook variable, most of the values are below $46,250 so we trim and set the max value of this variable to $46,250.



For total claims in the past five years there is only a very small percentage above $20,000 so we will set that as our max value for this variable.



Income is another variable with outliers. Here we cap the max value at $210,000 to help the distribution and remove the small number of outliers.



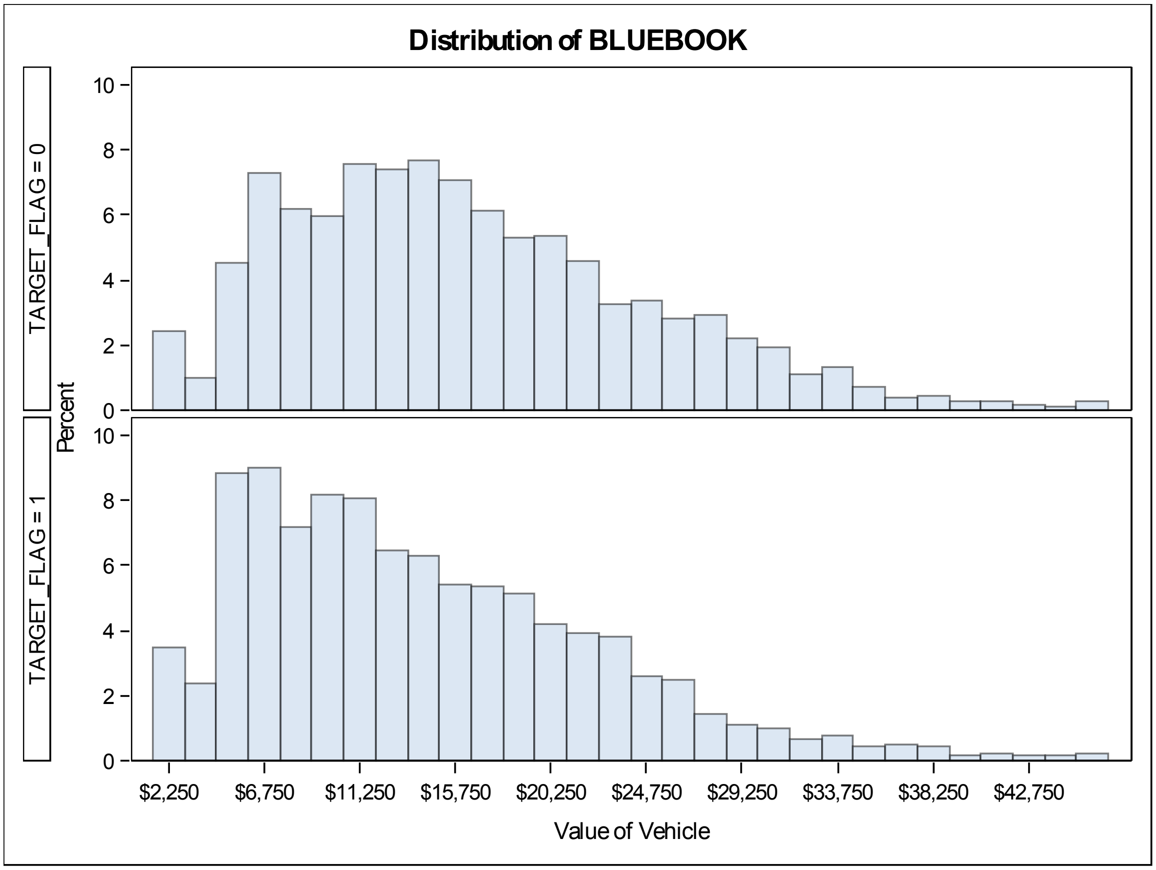
Home value is similar to income only that there are a large number of zero values indicating renters instead of home owners. Here we will max the values at $600,000 and consider creating two variables (one for renters and one for home owners) out of this variable in step eight which is creating and transforming variables.

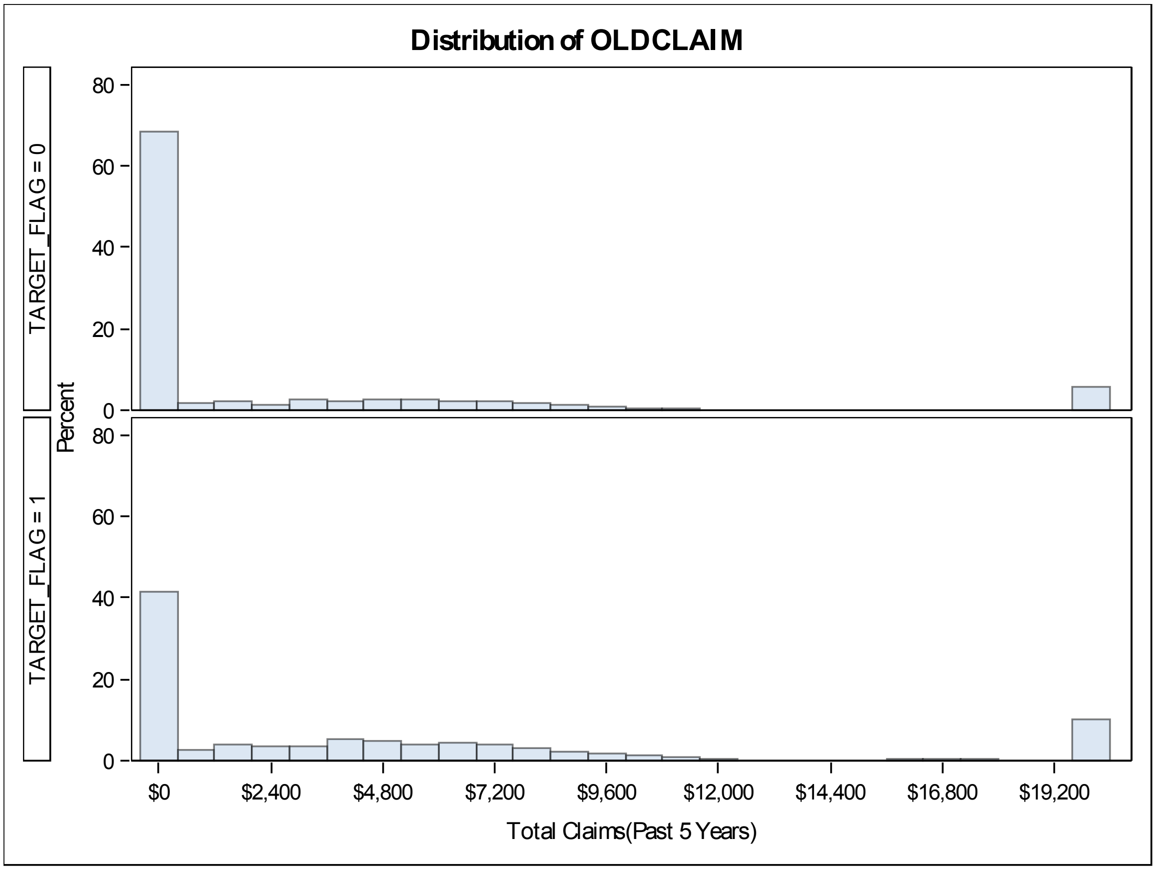
Step 8:

Step eight consists of checking each variable for the ability to transform closer to a normal distribution, combining it with other variables, or simply creating new variables from the data. For category variables, we consider creating new variables that only include values that are more predictive than the average value for that particular variable. For numeric variables, we look specifically for variables that may have a normal distribution, but also have a large spike in values at some point outside of the normal distribution. Furthermore with numeric variables, we look to correct skewed distributions.

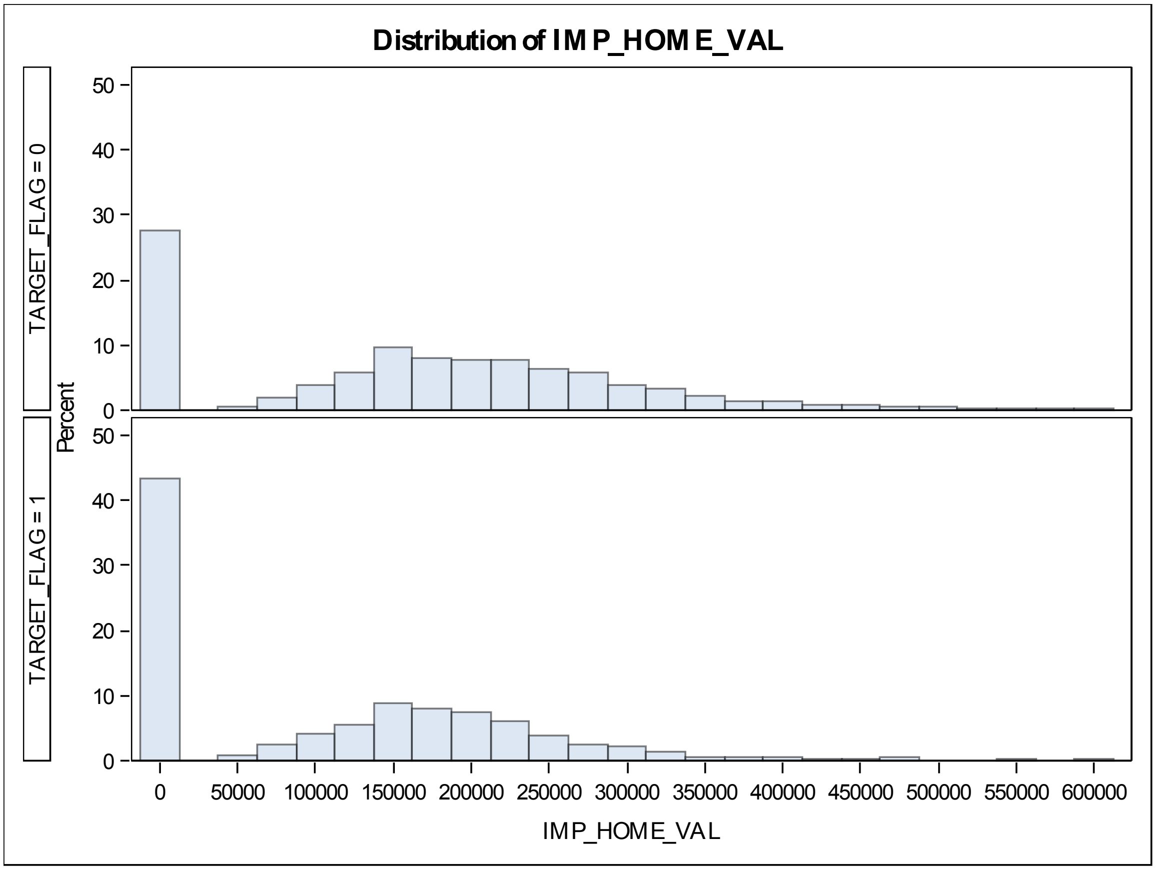
For numeric variable analysis, we run PROC UNIVARIATE again to show the histograms now that the outliers have been removed.

For category variables, we review PROC FREQ against our class variable TARGET_FLAG.

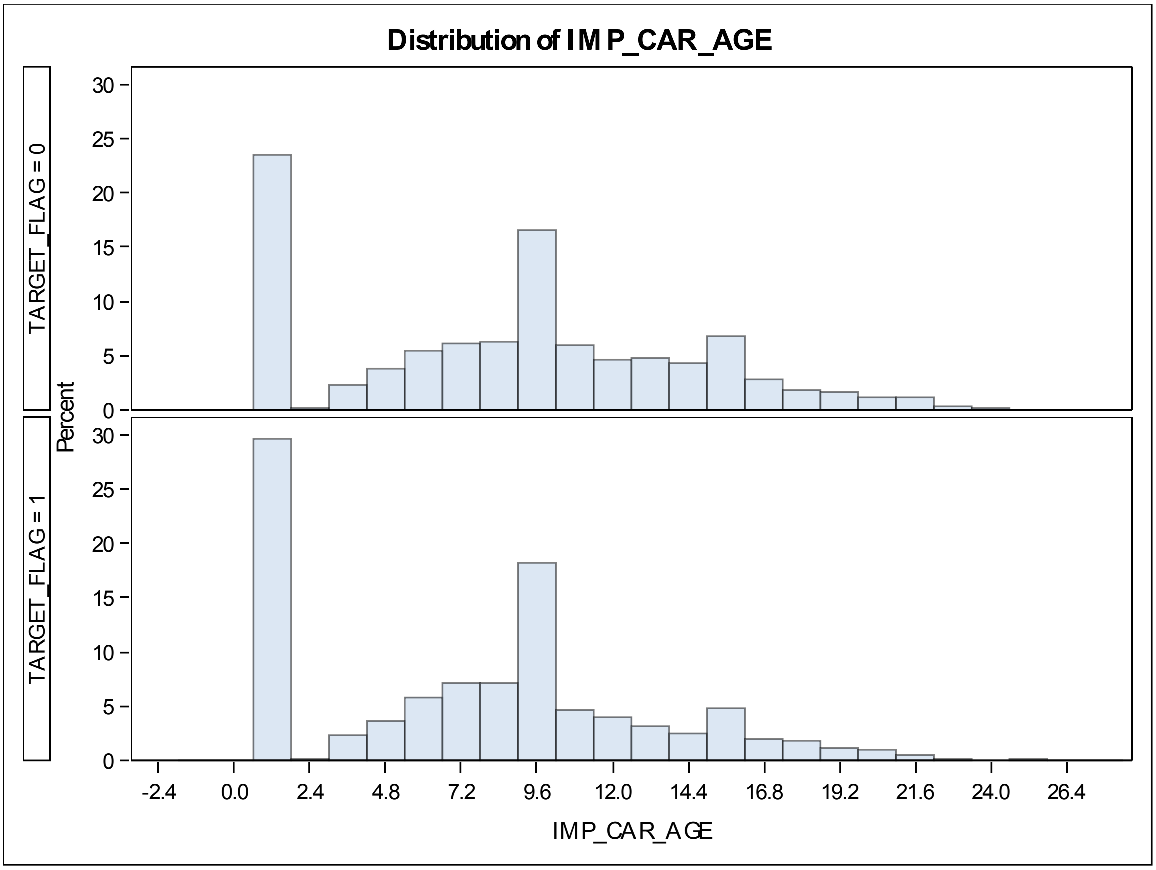

Distribution of BLUEBOOK
TARGET_FLAG = 0
TARGET_FLAG = 1
Percent
10
8
6
4
2
0
$2,250
$6,750
$11,250
$15,750
$20,250
$24,750
$29,250
$33,750
$38,250
$42,750
Value of Vehicle



With OLDCLAIM we observe a very high percentage of claims in the past five years at zero. This really affects the distribution of the values so we will try separating the zero values from the rest of the values by creating two variables. One, OLDCLAIM_ZERO will be a binary variable with 1 = yes and 0 = no. All remaining values will become part of IMP_OLDCLAIM. We will see if either has more affect on the model's predictive power.



We see a similar situation here and home values of zero indicate the driver does not own a home and rents. Thus, we will create two variables out of this variable by separating the zero values from the remaining values. IMP2_HOME_VAL will equal all values above zero and HOME_VAL_ZERO will be binary with 1=renter and 0=home owner.



With IMP_CAR_AGE we see a disproportionate amount of drivers in new cars (less than 1 year old). Here we will try to separate new car drivers from the rest of the drivers to, again, create two new variables. CAR_AGE_NEW as binary with 1 = new car and 0 = non-new car. IMP2_CAR_AGE will be used with all cars more than one year old.

Table of EDUCATION by TARGET_FLAG			
EDUCATION(Max Education Level)	TARGET_FLAG		
Frequency Percent Row Pct Col Pct	0	1	Total
<High School	818 10.02 68.00 13.62	385 4.72 32.00 17.88	1203 14.74
Bachelors	1719 21.06 76.67 28.61	523 6.41 23.33 24.29	2242 27.47
Masters	1331 16.31 80.28 22.15	327 4.01 19.72 15.19	1658 20.32
PhD	603 7.39 82.83 10.04	125 1.53 17.17 5.81	728 8.92
z_High School	1537 18.83 65.97 25.58	793 9.72 34.03 36.83	2330 28.55
Total	6008 73.62	2153 26.38	8161 100.00

With this category variable we see that education level of high school or less are both higher than the average for drivers who crash their cars. We will put these two groups into their own category variable named EDUCATION_HS to see if we can improve the model.

Table of IMP_JOB by TARGET_FLAG			
IMP_JOB	**TARGET_FLAG**		
Frequency **Percent** **Row Pct** **Col Pct**	**0**	**1**	**Total**
Clerical	910 11.15 70.98 15.15	372 4.56 29.02 17.28	1282 15.71
Doctor	347 4.25 84.02 5.78	66 0.81 15.98 3.07	413 5.06
Home Maker	461 5.65 71.81 7.67	181 2.22 28.19 8.41	642 7.87
Lawyer	757 9.28 81.14 12.60	176 2.16 18.86 8.17	933 11.43
Manager	889 10.89 85.48 14.80	151 1.85 14.52 7.01	1040 12.74
Professional	912 11.18 77.88 15.18	259 3.17 22.12 12.03	1171 14.35
Student	446 5.47 62.64 7.42	266 3.26 37.36 12.35	712 8.72

Table of IMP_JOB by TARGET_FLAG			
IMP_JOB	**TARGET_FLAG**		
Frequency Percent Row Pct Col Pct	**0**	**1**	**Total**
z_Blue Collar	1286 15.76 65.35 21.40	682 8.36 34.65 31.68	1968 24.11
Total	6008 73.62	2153 26.38	8161 100.00

While illegal in real life, we will try to find some relevance between income levels. We will separate the traditionally lower income level jobs from the higher income levels. We will create two new variables called JOB_LOW including blue collar, home maker, student, and clerical and JOB_HIGH including doctor, lawyer, manager, and professional.

Table of CAR_TYPE by TARGET_FLAG			
CAR_TYPE(Type of Car)	TARGET_FLAG		
Frequency Percent Row Pct Col Pct	0	1	Total
Minivan	1796 22.01 83.73 29.89	349 4.28 16.27 16.21	2145 26.28
Panel Truck	498 6.10 73.67 8.29	178 2.18 26.33 8.27	676 8.28
Pickup	946 11.59 68.11 15.75	443 5.43 31.89 20.58	1389 17.02
Sports Car	603 7.39 66.48 10.04	304 3.73 33.52 14.12	907 11.11
Van	549 6.73 73.20 9.14	201 2.46 26.80 9.34	750 9.19
z_SUV	1616 19.80 70.44 26.90	678 8.31 29.56 31.49	2294 28.11
Total	6008 73.62	2153 26.38	8161 100.00

For this category variable we will put z_SUV, Sports Car, and Pickup into their own category variable named CAR_TYPE_HRISK because each car type has a higher than average percentage of drivers who crash their vehicle.

BUILD AND SELECT MODELS

Step 9:

Step nine is the processing of a logistic regression and the comparison of three different models.

The first model will include all variables that had their missing values imputed with averages and the removal of those variables that did not indicate predictive properties. This includes all steps described up to Step 7and represented by the dataset SCRUBFILE1.

The second model includes everything in model 1 but also includes the removal/adjustments made for outliers. This includes all steps described up to Step 8 and represented by the dataset SCRUBFILE2.

The third model includes both previous models and includes all steps described up to Step 9. This includes all newly created variables and any variables that have been transformed. It is represented by the dataset SCRUBFILE3.

Model #1

Model Fit Statistics		
Criterion	**Intercept Only**	**Intercept and Covariates**
AIC	9419.962	7431.194
SC	9426.969	7641.407
-2 Log L	9417.962	7371.194

The model fit statistics can be used to assess how well the model fits. In this case we are looking for the lowest value of the three, which is -2LogL. -2LogL is also called the Deviance of the model and is used in the next table to calculate the Likelihood Ratio. AIC is the same metric we described before in previous assignments, and we already know that SC is a form of AIC but has a higher penalty for more parameters added to the model.

Testing Global Null Hypothesis: BETA=0			
Test	**Chi-Square**	**DF**	**Pr > ChiSq**
Likelihood Ratio	2046.7686	29	<.0001
Score	1816.4770	29	<.0001
Wald	1374.7117	29	<.0001

When testing if the model is significant, we must check if each of the three values (Likelihood ratio, score, and Wald) is significant. If so, then we can say that the model has more statistically significant predictive power with the variable(s) than without the variables. The Likelihood Ratio is especially interesting because it is used to compare the Deviance of the reduced model to the Deviance of the full model. As we can see, each metric has a probability of less than .0001 which says that the model is statistically significant.

Analysis of Maximum Likelihood Estimates						
Parameter		**DF**	**Estimate**	**Standard Error**	**Wald Chi-Square**	**Pr > ChiSq**
Intercept		1	-0.8260	0.1908	18.7386	<.0001
KIDSDRIV		1	0.4145	0.0550	56.8887	<.0001
BLUEBOOK		1	-0.00002	4.685E-6	23.3568	<.0001
TIF		1	-0.0551	0.00730	57.0614	<.0001
OLDCLAIM		1	-0.00001	3.89E-6	14.2159	0.0002
CLM_FREQ		1	0.2023	0.0284	50.8215	<.0001
MVR_PTS		1	0.1161	0.0135	73.5468	<.0001
IMP_INCOME		1	-3.07E-6	1.124E-6	7.4497	0.0063
IMP_HOME_VAL		1	-1.26E-6	3.274E-7	14.8832	0.0001
EDUCATION	**<High School**	1	0.00519	0.0937	0.0031	0.9558
EDUCATION	**Bachelors**	1	-0.4023	0.0829	23.5602	<.0001
EDUCATION	**Masters**	1	-0.3912	0.1193	10.7563	0.0010
EDUCATION	**PhD**	1	-0.3955	0.1619	5.9689	0.0146
CAR_TYPE	**Minivan**	1	-0.7093	0.0855	68.8153	<.0001

Analysis of Maximum Likelihood Estimates						
Parameter		DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
CAR_TYPE	Panel Truck	1	-0.1004	0.1497	0.4496	0.5025
CAR_TYPE	Pickup	1	-0.1789	0.0927	3.7227	0.0537
CAR_TYPE	Sports Car	1	0.2502	0.0974	6.5991	0.0102
CAR_TYPE	Van	1	-0.0934	0.1197	0.6093	0.4351
CAR_USE	Commercial	1	0.7730	0.0881	76.9401	<.0001
IMP_JOB	Clerical	1	0.0967	0.1028	0.8852	0.3468
IMP_JOB	Doctor	1	-0.3327	0.2160	2.3735	0.1234
IMP_JOB	Home Maker	1	0.0291	0.1365	0.0454	0.8313
IMP_JOB	Lawyer	1	-0.1334	0.1562	0.7298	0.3929
IMP_JOB	Manager	1	-0.8061	0.1266	40.5319	<.0001
IMP_JOB	Professional	1	-0.1462	0.1119	1.7088	0.1911
IMP_JOB	Student	1	-0.0188	0.1208	0.0243	0.8762
MSTATUS	Yes	1	-0.4736	0.0781	36.7506	<.0001
PARENT1	No	1	-0.4341	0.0936	21.4887	<.0001
REVOKED	No	1	-0.8927	0.0906	97.0517	<.0001
URBANICITY	Highly Urban/ Urban	1	2.2644	0.1109	416.9790	<.0001

The maximum likelihood estimates are used to determine the coefficients/estimates, the odd-ratio, probability or fitted values, and the test statistics to assess each parameter and the model.

We can test if individual variables have significant predictive power. The significance of each variable is at the p<.05 level. This is determined by comparing the Wald Chi-Square value for the parameter to the critical Chi-Square value for the relative degrees of freedom (1 for each variable). The Wald Chi-Square value is found by dividing the Estimate by the Standard Error of the parameter and squaring that result. One area of concern with this model is the high number of variables that are not significant. These variables should be considered for removal from the model if we wish to simplify them.

These numeric coefficients can be interpreted as the expected change in the logit for every unit change of the parameter with the other parameters held constant. For example, the expected change in the logit is 0.4145 for every one unit change in the KIDSDRIV variable when all other variables are held fixed.

Odds Ratio Estimates			
Effect	Point Estimate	95% Wald Confidence Limits	
KIDSDRIV	1.514	1.359	1.686
BLUEBOOK	1.000	1.000	1.000
TIF	0.946	0.933	0.960
OLDCLAIM	1.000	1.000	1.000
CLM_FREQ	1.224	1.158	1.294
MVR_PTS	1.123	1.094	1.153
IMP_INCOME	1.000	1.000	1.000
IMP_HOME_VAL	1.000	1.000	1.000
EDUCATION <High School vs z_High School	1.005	0.837	1.208
EDUCATION Bachelors vs z_High School	0.669	0.569	0.787
EDUCATION Masters vs z_High School	0.676	0.535	0.854
EDUCATION PhD vs z_High School	0.673	0.490	0.925
CAR_TYPE Minivan vs z_SUV	0.492	0.416	0.582
CAR_TYPE Panel Truck vs z_SUV	0.904	0.674	1.213
CAR_TYPE Pickup vs z_SUV	0.836	0.697	1.003
CAR_TYPE Sports Car vs z_SUV	1.284	1.061	1.554
CAR_TYPE Van vs z_SUV	0.911	0.720	1.152
CAR_USE Commercial vs Private	2.166	1.823	2.575
IMP_JOB Clerical vs z_Blue Collar	1.102	0.901	1.348
IMP_JOB Doctor vs z_Blue Collar	0.717	0.470	1.095
IMP_JOB Home Maker vs z_Blue Collar	1.030	0.788	1.345
IMP_JOB Lawyer vs z_Blue Collar	0.875	0.644	1.189
IMP_JOB Manager vs z_Blue Collar	0.447	0.348	0.572
IMP_JOB Professional vs z_Blue Collar	0.864	0.694	1.076
IMP_JOB Student vs z_Blue Collar	0.981	0.774	1.244
MSTATUS Yes vs z_No	0.623	0.534	0.726

Odds Ratio Estimates			
Effect	**Point Estimate**	**95% Wald Confidence Limits**	
PARENT1 No vs Yes	0.648	0.539	0.778
REVOKED No vs Yes	0.410	0.343	0.489
URBANICITY Highly Urban/ Urban vs z_Highly Rural/ Rural	9.625	7.745	11.962

Interestingly, by taking the *e* of each parameter's estimate (or coefficient) we can calculate the odds-ratio, which is generally much easier for readers to understand and interpret. For example, for KIDSDRIV the odds-ratio is exp(0.4145) = 1.514. This is easier to interpret as it means that the probability that Y=1 is 1.514times more likely for every unit change of KIDSDRIV when KIDSDRIV is 1 instead of 0. Furthermore, by observing the confidence limits, if the confidence interval does NOT contain the value 1, the variable has a significant effect on the odds ratio. If the interval is below 1 the variable significantly lowers the odds ratio and vice versa if the interval is above 1.

Association of Predicted Probabilities and Observed Responses			
Percent Concordant	80.8	**Somers' D**	0.617
Percent Discordant	19.0	**Gamma**	0.618
Percent Tied	0.2	**Tau-a**	0.240
Pairs	12935224	**c**	0.809

Below, the Association of Predicted Probabilities and Observed Responses table values are used to evaluate the association between the predicted values versus the observed values. These measures rely on concordant and discordant pairs. Concordant pairs are those pairs where the lower ordered response value (often 0) has a lower predicted mean score than the observation with the higher ordered response value. In other words, it is the percent of correctly classified pairs. This is desirable, while discordant pairs have a higher predicted mean score for lower order response values, which is less desirable.

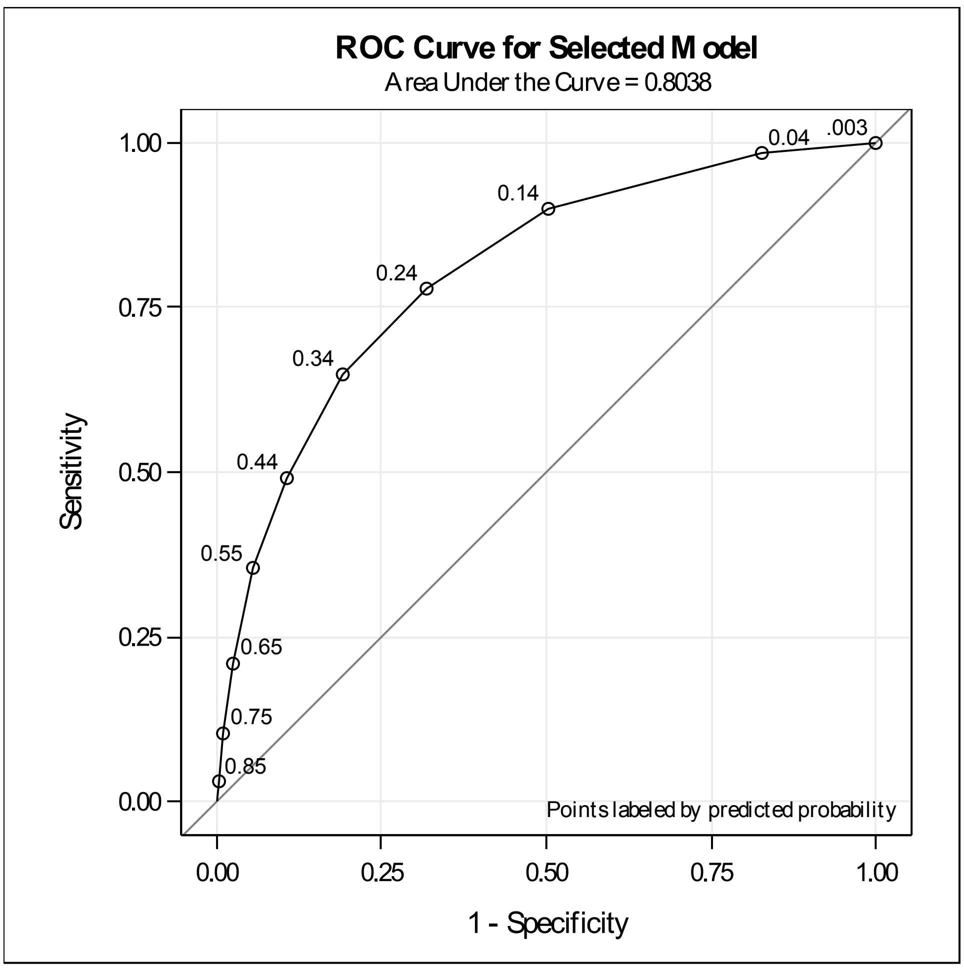
Somers' D is used to determine the strength and direction of relation between pairs of variables. It has a value of -1 to +1 with +1 meaning that all pairs agree or are concordant. A Somers' D value of .617 shows fair concordance with between the predicted and observed responses.

Gamma is similar to Somers' D except that it does not penalize for ties and therefore (using the same scale of -1 to +1) is usually higher value than Somers' D, which is what we see here as well (0.617 vs. 0.618).

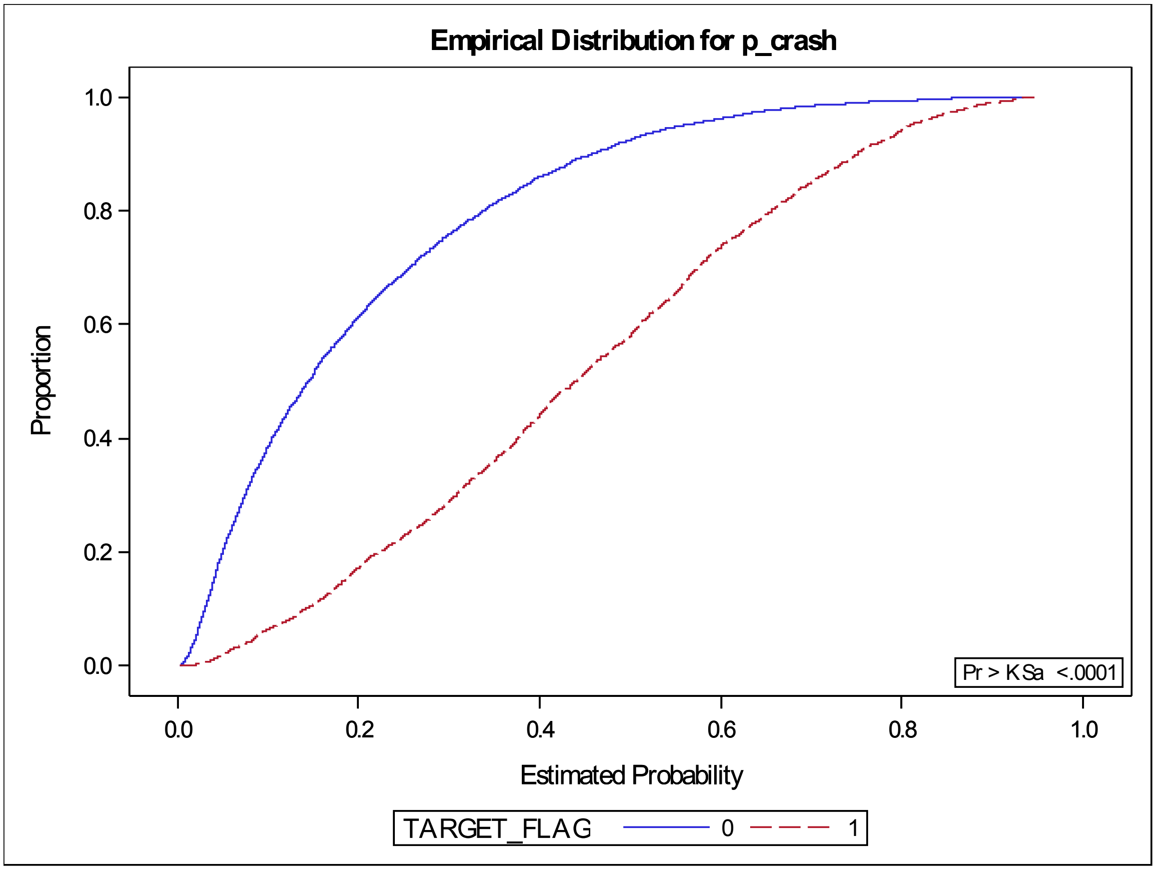
Tau-a is similar to a generalized value of R-square that is derived from the likelihood ratio. It is defined to be the ratio of the difference between the number of concordant pairs minus the discordant pairs divided by the total number of possible pairs.

C is used to determine how well the model can discriminate the response. Its value ranges from 0.5 to 1, where 0.5 is randomly guessing (no predictive power). Thus we want a higher number and our number of .809 shows us that our model does fairly well at discriminating the response value. C is also equivalent to the area under the ROC curve and can be used to compare models.

Thus, we can see that, taken together, based on our concordant/discordant values, Somers' D, Gamma, Tau-a, and C values that we have a strong model for predicting the response variable values correctly.



The area under the ROC curve is .8038 for Model #1. The points on the curve represent probability levels (above graph). This means that at a given probability we will accurately predict the X percent of drivers who crash their car (Sensitivity) and inaccurately predict X drivers who will crash their car (1-Sensitivity).



Kolmogorov-Smirnov Test for Variable p_crash Classified by Variable TARGET_FLAG			
TARGET_FLAG	**N**	**EDF at Maximum**	**Deviation from Mean at Maximum**
0	6008	0.753495	9.709336
1	2153	0.278681	-16.219314
Total	8161	0.628232	
Maximum Deviation Occurred at Observation 5275			
Value of p_crash at Maximum = 0.294123			

Kolmogorov-Smirnov Two-Sample Test (Asymptotic)			
KS	0.209251	**D**	0.474814
KSa	18.903368	**Pr > KSa**	<.0001

The KS value for Model #1 is 20.9251%. We want a higher KS value for our model as it shows a larger difference between our model and a reference value.

Model #2

Model #2 is the same as Model #1 except each variable's histogram has been reviewed to check and remove outliers. Outliers were modified in BLUEBOOK, OLDCLAIM, IMP_INCOME, and IMP_HOME_VAL.

Model Fit Statistics		
Criterion	**Intercept Only**	**Intercept and Covariates**
AIC	9419.962	7434.230
SC	9426.969	7644.444
-2 Log L	9417.962	7374.230

The change in Deviance between Model #1 and Model #2 is 7371.94 vs. 7374.230. Thus, the Deviance is actually higher than before which indicates Model #2 is not as good as Model #1.

Testing Global Null Hypothesis: BETA=0			
Test	**Chi-Square**	**DF**	**Pr > ChiSq**
Likelihood Ratio	2043.7323	29	<.0001
Score	1813.7565	29	<.0001
Wald	1373.1088	29	<.0001

Analysis of Maximum Likelihood Estimates						
Parameter		**DF**	**Estimate**	**Standard Error**	**Wald Chi-Square**	**Pr > ChiSq**
Intercept		1	-0.8414	0.1912	19.3734	<.0001
KIDSDRIV		1	0.4169	0.0550	57.5563	<.0001
BLUEBOOK		1	-0.00002	4.702E-6	24.1695	<.0001
TIF		1	-0.0552	0.00730	57.2554	<.0001
OLDCLAIM		1	-0.00002	6.675E-6	7.9155	0.0049
CLM_FREQ		1	0.2017	0.0306	43.5012	<.0001
MVR_PTS		1	0.1163	0.0136	73.2686	<.0001
IMP_INCOME		1	-3.44E-6	1.158E-6	8.8036	0.0030
IMP_HOME_VAL		1	-1.27E-6	3.283E-7	14.9996	0.0001
EDUCATION	**<High School**	1	0.000830	0.0937	0.0001	0.9929
EDUCATION	**Bachelors**	1	-0.3979	0.0829	23.0353	<.0001
EDUCATION	**Masters**	1	-0.3904	0.1193	10.7106	0.0011
EDUCATION	**PhD**	1	-0.3953	0.1618	5.9708	0.0145
CAR_TYPE	**Minivan**	1	-0.7106	0.0855	69.0986	<.0001
CAR_TYPE	**Panel Truck**	1	-0.0927	0.1497	0.3837	0.5356
CAR_TYPE	**Pickup**	1	-0.1821	0.0927	3.8591	0.0495
CAR_TYPE	**Sports Car**	1	0.2447	0.0974	6.3116	0.0120
CAR_TYPE	**Van**	1	-0.0915	0.1196	0.5855	0.4442
CAR_USE	**Commercial**	1	0.7752	0.0881	77.3610	<.0001
IMP_JOB	**Clerical**	1	0.0900	0.1029	0.7654	0.3816
IMP_JOB	**Doctor**	1	-0.3104	0.2159	2.0668	0.1505
IMP_JOB	**Home Maker**	1	0.0150	0.1369	0.0120	0.9129

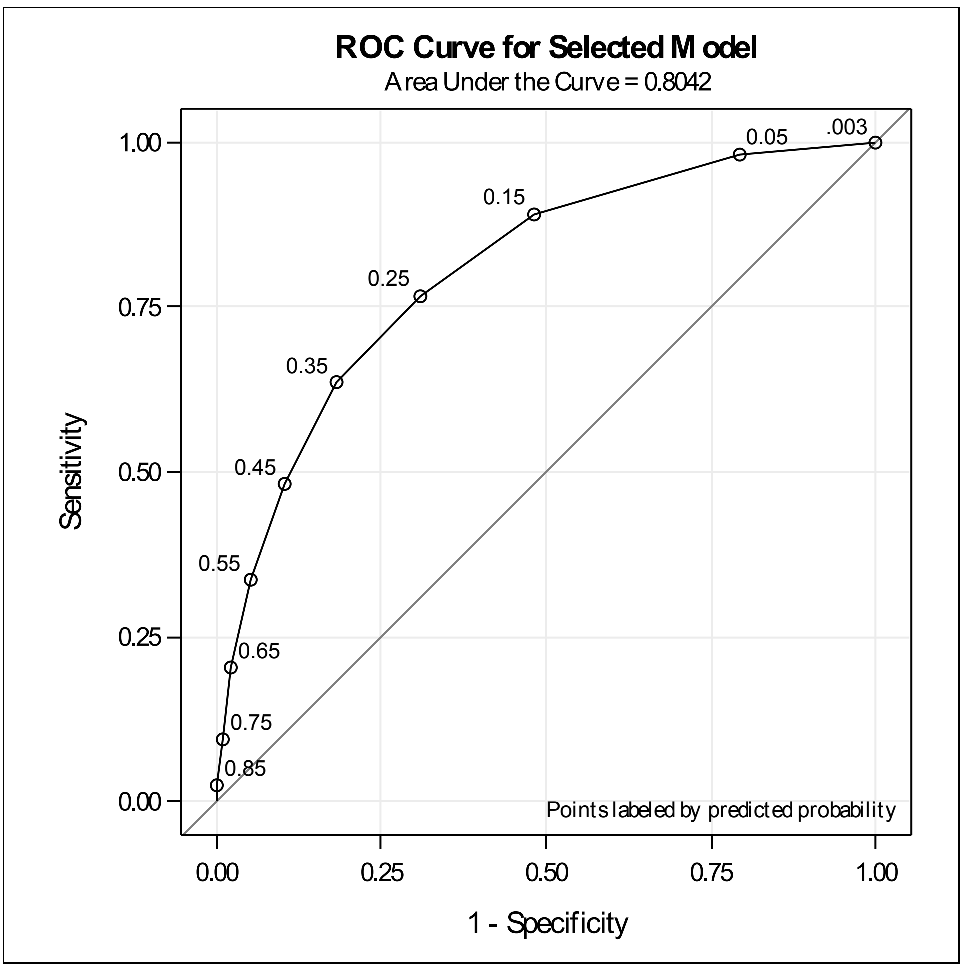
Analysis of Maximum Likelihood Estimates						
Parameter		**DF**	**Estimate**	**Standard Error**	**Wald Chi-Square**	**Pr > ChiSq**
IMP_JOB	**Lawyer**	1	-0.1175	0.1562	0.5659	0.4519
IMP_JOB	**Manager**	1	-0.7987	0.1265	39.8404	<.0001
IMP_JOB	**Professional**	1	-0.1381	0.1118	1.5256	0.2168
IMP_JOB	**Student**	1	-0.0361	0.1211	0.0889	0.7656
MSTATUS	**Yes**	1	-0.4729	0.0781	36.6365	<.0001
PARENT1	**No**	1	-0.4346	0.0936	21.5501	<.0001
REVOKED	**No**	1	-0.8450	0.0897	88.7169	<.0001
URBANICITY	**Highly Urban/ Urban**	1	2.2664	0.1109	417.7842	<.0001

Odds Ratio Estimates			
Effect	**Point Estimate**	**95% Wald Confidence Limits**	
KIDSDRIV	1.517	1.362	1.690
BLUEBOOK	1.000	1.000	1.000
TIF	0.946	0.933	0.960
OLDCLAIM	1.000	1.000	1.000
CLM_FREQ	1.224	1.152	1.299
MVR_PTS	1.123	1.094	1.154
IMP_INCOME	1.000	1.000	1.000
IMP_HOME_VAL	1.000	1.000	1.000
EDUCATION <High School vs z_High School	1.001	0.833	1.203
EDUCATION Bachelors vs z_High School	0.672	0.571	0.790
EDUCATION Masters vs z_High School	0.677	0.536	0.855
EDUCATION PhD vs z_High School	0.673	0.490	0.925
CAR_TYPE Minivan vs z_SUV	0.491	0.416	0.581

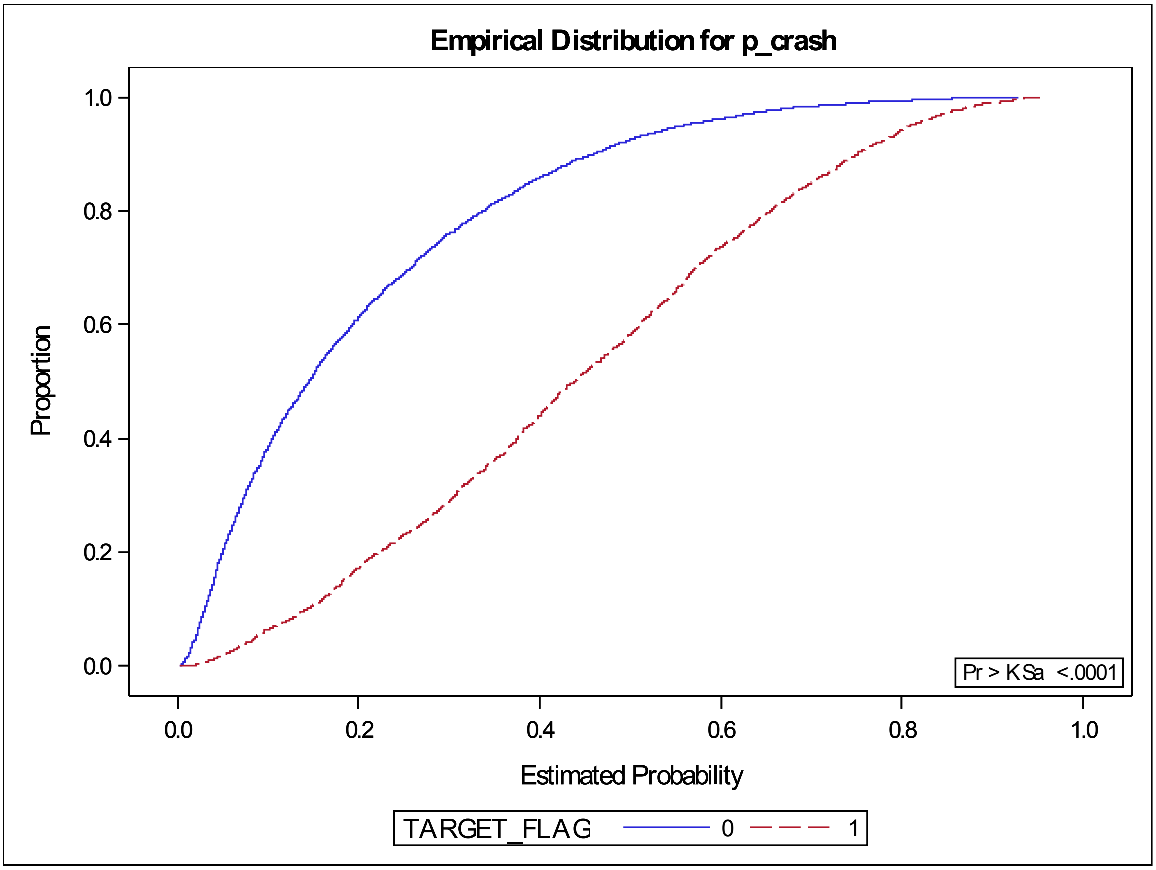
Odds Ratio Estimates			
Effect	Point Estimate	95% Wald Confidence Limits	
CAR_TYPE Panel Truck vs z_SUV	0.911	0.680	1.222
CAR_TYPE Pickup vs z_SUV	0.833	0.695	1.000
CAR_TYPE Sports Car vs z_SUV	1.277	1.055	1.546
CAR_TYPE Van vs z_SUV	0.913	0.722	1.154
CAR_USE Commercial vs Private	2.171	1.827	2.580
IMP_JOB Clerical vs z_Blue Collar	1.094	0.894	1.339
IMP_JOB Doctor vs z_Blue Collar	0.733	0.480	1.119
IMP_JOB Home Maker vs z_Blue Collar	1.015	0.776	1.327
IMP_JOB Lawyer vs z_Blue Collar	0.889	0.655	1.208
IMP_JOB Manager vs z_Blue Collar	0.450	0.351	0.577
IMP_JOB Professional vs z_Blue Collar	0.871	0.700	1.084
IMP_JOB Student vs z_Blue Collar	0.965	0.761	1.223
MSTATUS Yes vs z_No	0.623	0.535	0.726
PARENT1 No vs Yes	0.648	0.539	0.778
REVOKED No vs Yes	0.430	0.360	0.512
URBANICITY Highly Urban/ Urban vs z_Highly Rural/ Rural	9.644	7.760	11.985

Association of Predicted Probabilities and Observed Responses			
Percent Concordant	80.7	**Somers' D**	0.617
Percent Discordant	19.1	**Gamma**	0.618
Percent Tied	0.2	**Tau-a**	0.240
Pairs	12935224	**c**	0.808

Compared to Model #1 the Somers' D, Gamma, Tau-a values are identical and the C value is actually .001 lower than Model #1. These values indicate that the changes to the outliers had a minimal effect, and if anything made the model worse.



Model #2 has a ROC curve where the area under the curve is .8042, which is slightly higher than Model #1which was at .8038.



Kolmogorov-Smirnov Test for Variable p_crash Classified by Variable TARGET_FLAG			
TARGET_FLAG	**N**	**EDF at Maximum**	**Deviation from Mean at Maximum**
0	6008	0.759321	9.657501
1	2153	0.287041	-16.132725
Total	8161	0.634726	
Maximum Deviation Occurred at Observation 1557			
Value of p_crash at Maximum = 0.298548			

Kolmogorov-Smirnov Two-Sample Test (Asymptotic)			
KS	0.208134	**D**	0.472280
KSa	18.802450	**Pr > KSa**	<.0001

The KS value for Model #2 is classified by the TARGET_FLAG variable. Here it is 20.8134%, which is slightly less than Model #1 at 20.8134%.

Model #3-

Model Fit Statistics		
Criterion	**Intercept Only**	**Intercept and Covariates**
AIC	9419.962	7499.959
SC	9426.969	7619.080
-2 Log L	9417.962	7465.959

Model#3 shows a much improved Deviance value compared to both previous models. Deviance for Model #3 is 7465.959 compared to a best of 7371.94 for Model #1. That is not as good but it isn't drastically different.

Testing Global Null Hypothesis: BETA=0			
Test	**Chi-Square**	**DF**	**Pr > ChiSq**
Likelihood Ratio	1952.0031	16	<.0001
Score	1751.1351	16	<.0001
Wald	1336.9367	16	<.0001

Again, like the three values of Liklihood Ratio, Score, and Wald are each significant.

Analysis of Maximum Likelihood Estimates						
Parameter		**DF**	**Estimate**	**Standard Error**	**Wald Chi-Square**	**Pr > ChiSq**
Intercept		1	-1.1307	0.1665	46.0926	<.0001
KIDSDRIV		1	0.3985	0.0543	53.8618	<.0001
BLUEBOOK		1	-0.00001	4.222E-6	11.5075	0.0007
TIF		1	-0.0547	0.00725	56.9603	<.0001
OLDCLAIM		1	-0.00002	6.629E-6	8.1716	0.0043
CLM_FREQ		1	0.2078	0.0303	46.8862	<.0001
MVR_PTS		1	0.1222	0.0135	81.7386	<.0001
IMP_INCOME		1	-4.57E-6	9.213E-7	24.6250	<.0001
HOME_VAL_ZERO		1	0.2718	0.0718	14.3315	0.0002
CAR_TYPE_HRISK	**0**	1	-0.4821	0.0655	54.1930	<.0001
EDUCATION_HS	**0**	1	-0.3784	0.0727	27.1219	<.0001
JOB_LOW	**0**	1	-0.3660	0.0832	19.3632	<.0001
CAR_USE	**Commercial**	1	0.7667	0.0644	141.5309	<.0001
MSTATUS	**Yes**	1	-0.4686	0.0778	36.3243	<.0001
PARENT1	**No**	1	-0.4181	0.0924	20.4732	<.0001
REVOKED	**No**	1	-0.8531	0.0890	91.8673	<.0001
URBANICITY	**Highly Urban/ Urban**	1	2.2280	0.1104	407.2973	<.0001

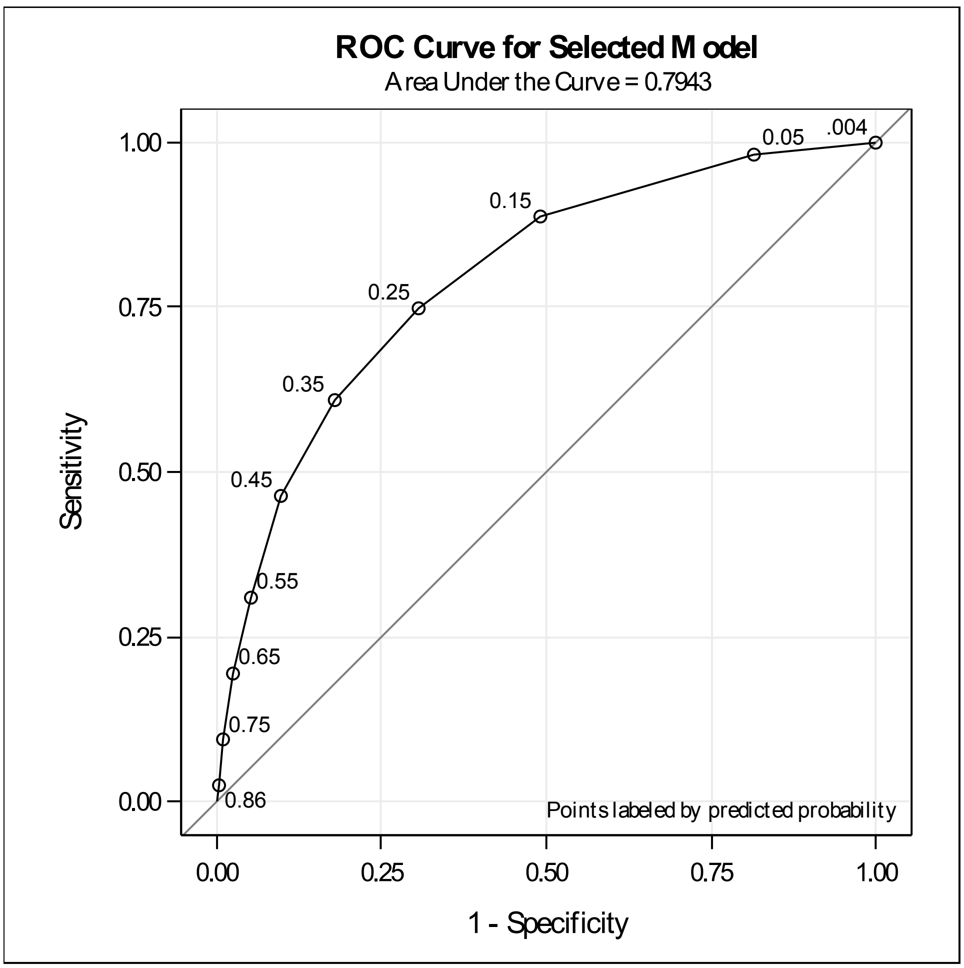
Model#3 is the first model of the three where every variable is significant for the Wald Chi-Square.

Odds Ratio Estimates			
Effect	**Point Estimate**	**95% Wald Confidence Limits**	
KIDSDRIV	1.490	1.339	1.657
BLUEBOOK	1.000	1.000	1.000
TIF	0.947	0.933	0.960
OLDCLAIM	1.000	1.000	1.000
CLM_FREQ	1.231	1.160	1.306
MVR_PTS	1.130	1.100	1.160
IMP_INCOME	1.000	1.000	1.000
HOME_VAL_ZERO	1.312	1.140	1.511
CAR_TYPE_HRISK 0 vs 1	0.618	0.543	0.702
EDUCATION_HS 0 vs 1	0.685	0.594	0.790
JOB_LOW 0 vs 1	0.693	0.589	0.816
CAR_USE Commercial vs Private	2.153	1.897	2.443
MSTATUS Yes vs z_No	0.626	0.537	0.729
PARENT1 No vs Yes	0.658	0.549	0.789
REVOKED No vs Yes	0.426	0.358	0.507
URBANICITY Highly Urban/ Urban vs z_Highly Rural/ Rural	9.282	7.476	11.524

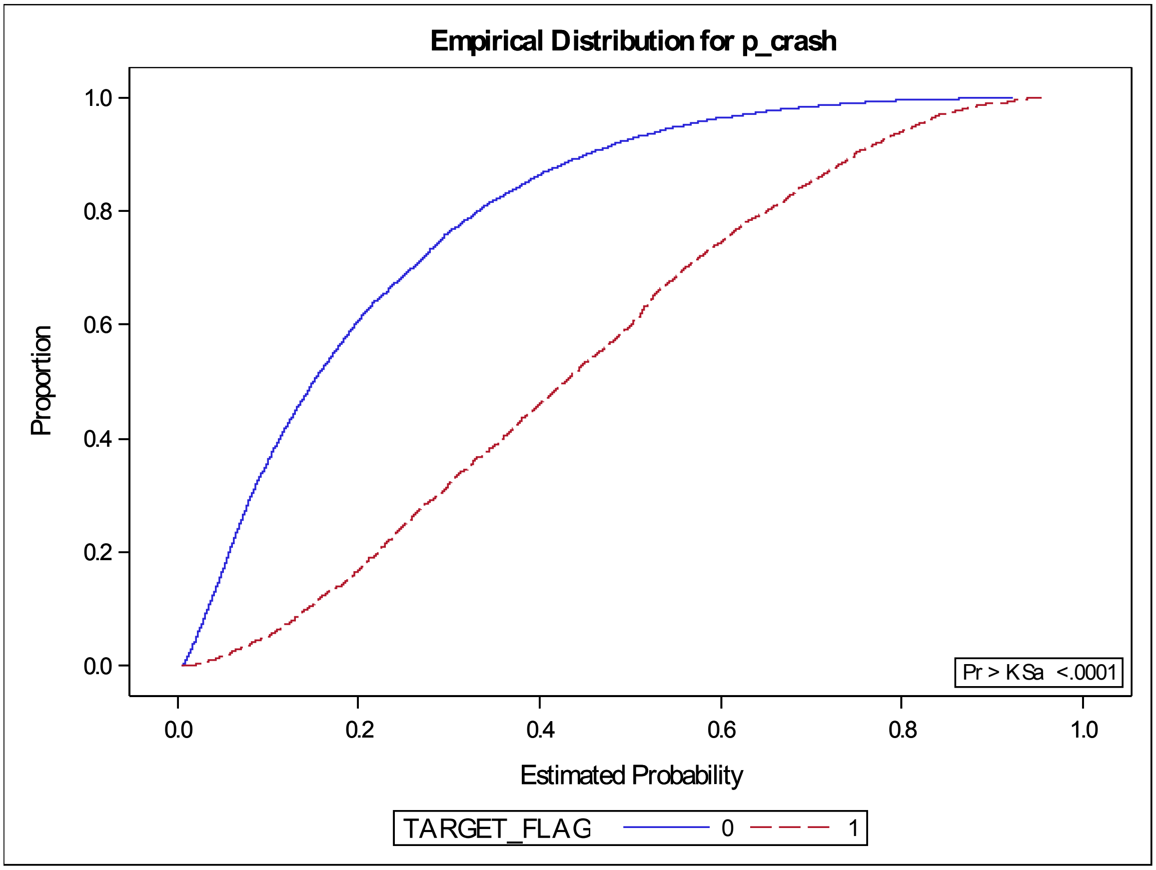
BLUEBOK, IMP_INCOME and IMP_OLDCLAIM have no effect on the odds of crashing a car. These could be removed. The remaining variables each show an effect on the odds of crashing a car.

Association of Predicted Probabilities and Observed Responses			
Percent Concordant	74.3	**Somers' D**	0.589
Percent Discordant	15.5	**Gamma**	0.655
Percent Tied	10.2	**Tau-a**	0.229
Pairs	12935224	**c**	0.794

The one drawback for Model #3 is that the percent concordant is about 6% lower and Somers' D (.589 vs .617), Tau-a (..229 vs. .240), and C (.794 vs. .808) are all lower as well. However, these values are not much lower than the best metrics we found in Model #1.



The area under the ROC curve was slightly lower than Model 1 and Model 2 (.7943 vs. .8038 vs. .8042).



Kolmogorov-Smirnov Test for Variable p_crash Classified by Variable TARGET_FLAG			
TARGET_FLAG	**N**	**EDF at Maximum**	**Deviation from Mean at Maximum**
0	6008	0.641644	9.179219
1	2153	0.192754	-15.333760
Total	8161	0.523220	
Maximum Deviation Occurred at Observation 4954			
Value of p_crash at Maximum = 0.218068			

Kolmogorov-Smirnov Two-Sample Test (Asymptotic)			
KS	0.197826	**D**	0.448890
KSa	17.871269	**Pr > KSa**	<.0001

The KS value for Model #3 is classified by the TARGET_FLAG variable. Here it is 19.78%, which is slightly less than Model #2 at 20.8134% and Model #1 at 20.9251%.

Even though Model #3 didn't have the best metrics in terms of Deviance and area under the curve, I selected Model #3 as the best model due to its much better parsimony (16 variables vs 29 variables), every variable had coefficients that were significant using Chi-Square, and the model evaluation metrics were only slightly less than those in Model #1. In other words, some accuracy was sacrificed slightly for more usability.

Step #10:

Step ten is to deploy the model against our insurance_test dataset. We will show ten observations to demonstrate the model was deployed correctly and that there were not any missing values.

Obs	PROB
1	0.19777
2	0.42628
3	0.14804
4	0.20160
5	0.20562
6	0.21642
7	0.44601
8	0.41157
9	0.08565
10	0.16721

This table shows that each observation has a probability that this driver will crash their car.

Analysis Variable : PROB
N Miss
0

As we can see, no values were missing from our deployed model.

Step 11:

Step eleven is to create a probability/severity model to determine the amount of money it will probably cost the company if the driver crashes their car.

Here we computed the average amount spent to repair a crashed car. We use this average (TARGET_AMT) and multiply it by the probability a driver will have an accident. This gives us the expected loss for each driver and what we can use to determine an insurance rate that balances between affordability and protection for the insurance company.

Here we use a linear regression model based on the scored model we created from the test data. It uses P_TARGET_FLAG and P_TARGET_AMOUNT to identify the probability if a person crashes their car and the probable amount of money it will cost the insurance company to pay for the repairs.

CONCLUSION-

The fickle game played between the auto insurer and those seeking insurance is analogous to the raw nature of an analytic street brawl to achieve a balanced model to satisfy both parties. The insurer doesn't want to lose money insuring the wrong people for the too low a fee, but they don't want to lose customers to lower prices from other companies either.

In the end, our brawl between three models came down to three items: simplicity, significance, and slight differences. Normally the predictive power metrics are the king of the analytic ring, and they were important here too. However, the model we ultimately selected was nearly as accurate in terms of maximizing the predictive metrics but required the company to only gather almost half the amount of information.

When almost thirty pieces of information are required for a model, it is often difficult to gather every piece of information about a driver, which often leads to estimating, guessing, or simply omitting data about a user. By taking a route that is nearly as accurate yet requires gathering only half the information, the selected model should be the easiest to use, especially in terms of having the highest probability of gathering every piece of data for it, and therefore the most practical for the insurance company without sacrificing too much accuracy that could lead to poor estimates. Like the fickle balancing act played between auto insurer, who wants the best accuracy possible, and those seeking insurance, who want practical and affordability, this model balances both accuracy and practicality as well.

**************BINGO BONUS:**

I tried the…
1) GENMOD,
2) used a decision tree approach for one variable imputation,
3) used macros, and
4) tried a different technique to get the KS statistic for each model.

1) GENMOD – AIC, BIC were higher and it included more variables than PROC logistic. It also included a scale variable. GENMOD code is included in SAS code at the end in the BINGO BONUS section.

Criteria For Assessing Goodness Of Fit			
Criterion	**DF**	**Value**	**Value/DF**
Deviance	8136	1211.7012	0.1489
Scaled Deviance	8136	8161.0002	1.0031
Pearson Chi-Square	8136	1211.7012	0.1489
Scaled Pearson X2	8136	8161.0002	1.0031
Log Likelihood		-3797.0509	
Full Log Likelihood		-3797.0509	
AIC (smaller is better)		7646.1017	

Criteria For Assessing Goodness Of Fit			
Criterion	**DF**	**Value**	**Value/DF**
AICC (smaller is better)		7646.2743	
BIC (smaller is better)		7828.2869	

Analysis Of Maximum Likelihood Parameter Estimates								
Parameter		**DF**	**Estimate**	**Standard Error**	**Wald 95% Confidence Limits**		**Wald Chi-Square**	**Pr > ChiSq**
Intercept		1	-1.2460	0.2172	-1.6718	-0.8202	32.90	<.0001
KIDSDRIV		1	0.3677	0.0537	0.2625	0.4730	46.90	<.0001
HOMEKIDS		1	0.0268	0.0297	-0.0313	0.0849	0.82	0.3655
BLUEBOOK		1	-0.0000	0.0000	-0.0000	-0.0000	14.71	0.0001
TIF		1	-0.0530	0.0067	-0.0661	-0.0399	62.88	<.0001
OLDCLAIM		1	-0.0000	0.0000	-0.0000	-0.0000	17.20	<.0001
OLDCLAIM_ZERO		0	0.0000	0.0000	0.0000	0.0000	.	.
CLM_FREQ		1	0.1846	0.0265	0.1326	0.2365	48.44	<.0001
MVR_PTS		1	0.1278	0.0118	0.1047	0.1509	117.79	<.0001
M_YOJ		1	0.1164	0.1123	-0.1037	0.3364	1.07	0.3001
IMP_INCOME		1	-0.0000	0.0000	-0.0000	-0.0000	13.93	0.0002
M_INCOME		1	-0.0191	0.1163	-0.2471	0.2089	0.03	0.8699
HOME_VAL_ZERO		1	0.1293	0.1224	-0.1106	0.3693	1.12	0.2908

Analysis Of Maximum Likelihood Parameter Estimates								
Parameter		**DF**	**Estimate**	**Standard Error**	**Wald 95% Confidence Limits**		**Wald Chi-Square**	**Pr > ChiSq**
IMP_HOME_VAL		1	-0.0000	0.0000	-0.0000	0.0000	2.50	0.1136
M_HOME_VAL		1	-0.1255	0.1079	-0.3370	0.0860	1.35	0.2449
CAR_AGE_NEW		1	0.1075	0.0913	-0.0714	0.2864	1.39	0.2389
IMP_CAR_AGE		1	0.0128	0.0087	-0.0042	0.0299	2.17	0.1409
M_CAR_AGE		1	0.1575	0.1008	-0.0402	0.3551	2.44	0.1183
M_JOB		0	0.0000	0.0000	0.0000	0.0000	.	.
CAR_TYPE_HRISK	0	1	-0.5113	0.0597	-0.6283	-0.3942	73.30	<.0001
EDUCATION_HS	0	1	-0.3884	0.0729	-0.5313	-0.2456	28.41	<.0001
JOB_LOW	0	1	-0.4176	0.0753	-0.5651	-0.2701	30.79	<.0001
JOB_HIGH	0	0	0.0000	0.0000	0.0000	0.0000	.	.
CAR_USE	Commercial	1	0.7997	0.0584	0.6851	0.9142	187.20	<.0001
MSTATUS	Yes	1	-0.4443	0.0750	-0.5913	-0.2974	35.12	<.0001
PARENT1	No	1	-0.4273	0.0949	-0.6132	-0.2413	20.29	<.0001
REVOKED	No	1	-0.8764	0.0763	-1.0259	-0.7269	132.03	<.0001
URBANICITY	Highly Urban/ Urban	1	2.3890	0.1486	2.0977	2.6803	258.42	<.0001
Scale		1	0.3853	0.0030	0.3795	0.3913		

2) Use of small DECISION TREES for a variable imputation

```
if missing(IMP_HOME_VAL) then
        do;

            if YOJ >=4 and INCOME > 50000 and BLUEBOOK > 10000 then
                do;
                    IMP_HOME_VAL=161159.53;
                    *median;
                    M_HOME_VAL=1;
                end;
            else
                do;
                    IMP_HOME_VAL=0;
                    * assuming driver is a home renter;
                    M_HOME_VAL=1;
                end;
        end;
    drop HOME_VAL;

  if missing(IMP_CAR_AGE)then
        do;

            if IMP_INCOME > 86000 then
                do;
                    IMP_CAR_AGE=1;
                    M_CAR_AGE=1;
                end;
            else
                do;
                    IMP_CAR_AGE=9.6;
                    M_CAR_AGE=1;
                end;
        end;
    drop CAR_AGE;
```

3) USE of MACROS

```
%let PATH = /home/derekhughes2014/DATAFILES/;
```

```
%let NAME = mydata;
%let LIB = &NAME..;

libname &NAME. "&PATH." access=readonly;

%let INFILE   = &LIB.logit_insurance;
%let TEST = &LIB.logit_insurance_test;
%let TEMPFILE       = TEMPFILE1;
%let SCRUBFILE1    = SCRUBFILE1;
%let SCRUBFILE2    = SCRUBFILE2;
%let SCRUBFILE3    = SCRUBFILE3;
```

4) Using unique/different approach to find KS statistic

```
* used to find KS statistic for model;
proc npar1way data=bout;
   class target_flag;
   var p_crash;
run;
```

CODE:

```
%let PATH = /home/derekhughes2014/DATAFILES/;
%let NAME = mydata;
%let LIB = &NAME..;

libname &NAME. "&PATH." access=readonly;

%let INFILE   = &LIB.logit_insurance;
%let TEST = &LIB.logit_insurance_test;
%let TEMPFILE       = TEMPFILE1;
%let SCRUBFILE1    = SCRUBFILE1;
%let SCRUBFILE2    = SCRUBFILE2;
%let SCRUBFILE3    = SCRUBFILE3;
```

```
* check to see if can access data;
proc print data=&INFILE.(obs=3);
run;



* Step 1 - EDA showing the contents of the data;

proc contents data=&INFILE. (obs=10);
run;




* Step 2 - Transform data into usable dataset

* convert INFILE to TEMPFILE so can manipulate dataset while preserving original
dataset;
data &TEMPFILE.;
   set &INFILE.;
run;




* Step 3 - Delete variables that will not be used in this model;

data &TEMPFILE.;
   set &TEMPFILE.;
   drop INDEX;
   * INDEX is not needed bc OBS value is the same;
   drop TARGET_AMT;
   * TARGET_AMT is not needed in this model but will be used in the severity model;
run;

* confirming can access TEMPFILE data;
proc print data=&TEMPFILE.(obs=3);
run;



* Step 4 - Explore numeric and character variables to determine adjustments to variables;

* exploring numeric data;
proc means data=&TEMPFILE. n nmiss mean median p1 p25 p75 p99;
```

```
   var _numeric_;
run;

* exploring character data;
proc freq data=&TEMPFILE.;
   table _character_ /missing;
run;

* proc corr run to see correlations between variables to determine how to impute some
variables;
proc corr data=&TEMPFILE.;
run;

* examine histogram of CAR_AGE to determine how to replace missing values;
proc univariate data=&TEMPFILE. plot;
   HISTOGRAM CAR_AGE;
run;



* Step 5 - begin to adjust variables for missing values, impute variables, create flags,
decision trees;

data &SCRUBFILE1.;
   set &TEMPFILE.;

         * Fix numeric missing values;

         * AGE - age imputations;
   IMP_AGE=AGE;
   if missing(IMP_AGE) then
      IMP_AGE=44.7903127; *mean;
   DROP AGE;

   * YOJ - year on job imputations;
   IMP_YOJ=YOJ;
   M_YOJ=0;

   if missing(IMP_YOJ) then
      do;
         IMP_YOJ=10.4992864;
         *mean;
         M_YOJ=1;
      end;
   DROP YOJ;
```

```
* INCOME - income imputations;
IMP_INCOME=INCOME;
M_INCOME=0;

if missing(IMP_INCOME) then
   do;
      IMP_INCOME=61898.10;
      *mean;
      M_INCOME=1;
   end;
drop INCOME;

* HOME_VAL - home value imputations;
IMP_HOME_VAL=HOME_VAL;
M_HOME_VAL=0;
*if missing( IMP_HOME_VAL ) then 161159.53;

if missing(IMP_HOME_VAL) then
   do;

      if YOJ >=4 and INCOME > 50000 and BLUEBOOK > 10000 then
         do;
            IMP_HOME_VAL=161159.53;
            *median;
            M_HOME_VAL=1;
         end;
      else
         do;
            IMP_HOME_VAL=0;
            * assuming driver is a home renter;
            M_HOME_VAL=1;
         end;
   end;
drop HOME_VAL;

* CAR_AGE - car age imputation;
IMP_CAR_AGE=CAR_AGE;
M_CAR_AGE=0;

*if missing( IMP_CAR_AGE )then IMP_CAR_AGE = 8.328;
if missing(IMP_CAR_AGE)then
   do;

      if IMP_INCOME > 86000 then
         do;
            IMP_CAR_AGE=1;
```

```
            M_CAR_AGE=1;
          end;
        else
          do;
            IMP_CAR_AGE=9.6;
            M_CAR_AGE=1;
          end;
      end;
  drop CAR_AGE;

  * Fix character variables;

  * JOB - job type imputation;
  IMP_JOB=JOB;
  M_JOB=0;

  if missing(IMP_JOB) then
    do;
      if IMP_INCOME > 140000 then
        IMP_JOB="Doctor";
      else if IMP_INCOME > 105000 then
        IMP_JOB="Lawyer";
      else if IMP_INCOME > 90000 then
        IMP_JOB="Professional";
      else if IMP_INCOME > 75000 then
        IMP_JOB="Manager";
      else if IMP_INCOME < 30000 AND IMP_INCOME > 10000 then
        IMP_JOB="Clerical";
      else if IMP_INCOME <=10000 AND HOMEKIDS > 0 then
        IMP_JOB="Home Maker";
      else if IMP_INCOME <=10000 then
        IMP_JOB="Student";
      else
        IMP_JOB="z_Blue Collar";
    end;
  drop JOB;

run;

* verify that all numeric and character variables have values for every observation;
proc means data=&SCRUBFILE1. nmiss min mean median;
  var _numeric_;
run;

proc freq data=&SCRUBFILE1.;
  table _character_ /missing;
```

```
run;



* Step 6 - identify variables with predictive power;

* identify category variables with values with much higher percentage crashing than the average for that variable;
proc freq data=&SCRUBFILE1.;
   table (_character_) * TARGET_FLAG /missing;
run;

* identify numeric variables with larger value differences between crashed and no crashed drivers;
proc means data=&SCRUBFILE1. mean median;
   class TARGET_FLAG;
   var _numeric_;
run;


* applying removal of non-predictive character/category and numeric variables from model/dataset;
data &SCRUBFILE1.;
   set &SCRUBFILE1.;
   * category variables dropped;
   drop RED_CAR;
   drop SEX;
   *numeric variables dropped;
   drop TRAVTIME;
   drop IMP_AGE;
         drop IMP_YOJ;
run;



* Step 7 - searching for, investigating, and removing outliers;

proc univariate data=&SCRUBFILE1.;
   class TARGET_FLAG;
   var _numeric_;
   histogram;
run;

* trimming outliers in dataset;
data &SCRUBFILE2.;
   set &SCRUBFILE1.;
```

```
   *outliers modified - numeric variables;
   if BLUEBOOK > 46250 then BLUEBOOK = 46250; * BLUEBOOK max trim;
   if OLDCLAIM > 20000 then OLDCLAIM = 20000; * OLDCLAIM max trim;
   if IMP_INCOME > 210000 then IMP_INCOME = 210000; *IMP_INCOME max
trim;
   if IMP_HOME_VAL > 600000 then IMP_HOME_VAL = 600000;
*IMP_HOME_VAL max trim;

run;


* Step 8 - transforming, combining, creating new variables;

* run proc univariate again to see histograms with outliers removed;
proc univariate data=&SCRUBFILE2.;
   class TARGET_FLAG;
   var _numeric_;
   histogram;
run;

* transforming, combining, creating new variables in dataset;
data &SCRUBFILE3.;
   set &SCRUBFILE2.;

   * new or transformed - numeric variables;
   OLDCLAIM_ZERO = 0; * creating variable for claims of only zero;
         if IMP_OLDCLAIM = 0 then OLDCLAIM_ZERO = 1;


         HOME_VAL_ZERO = 0; * creating variable for home value of only zero (non-
home owners);
         if IMP_HOME_VAL = 0 then HOME_VAL_ZERO = 1;


         CAR_AGE_NEW = 0; * creating variable for new cars only;
         if IMP_CAR_AGE <= 1 then CAR_AGE_NEW = 1;


   * new or transformed - categorical variables;
   CAR_TYPE_HRISK = CAR_TYPE in ("z_SUV","Sports Car","Pickup"); * creating
variable of only high risk car types;
   EDUCATION_HS = EDUCATION in ("<High School","z_High School"); * creating
variable of high school or less education;
   JOB_LOW = IMP_JOB in ("z_Blue Collar","Student","Home Maker","Clerical"); *
creating variable of low paying jobs;
```

```
   JOB_HIGH = IMP_JOB in ("Doctor","Lawyer","Manager","Professional"); * creating
variable of high paying jobs;

run;


/*
* transforming, combining, creating new variables in dataset;
data &SCRUBFILE4.;
   set &SCRUBFILE3.;


* IMP_JOB base is 'Student';
         if (IMP_JOB = 'Student') then IMP_JOB_student=1; else IMP_JOB_student=0;
         if (IMP_JOB = 'Clerical') then IMP_JOB_cleric=1; else IMP_JOB_cleric=0;
         if (IMP_JOB = 'Doctor') then IMP_JOB_doc=1; else IMP_JOB_doc=0;
         if (IMP_JOB = 'Home Maker') then IMP_JOB_home=1; else IMP_JOB_home=0;
         if (IMP_JOB = 'Lawyer') then IMP_JOB_law=1; else IMP_JOB_law=0;
         if (IMP_JOB = 'Manager') then IMP_JOB_mgr=1; else IMP_JOB_mgr=0;
         if (IMP_JOB = 'Professional') then IMP_JOB_pro=1; else IMP_JOB_pro=0;
         if (IMP_JOB = 'z_Blue Collar') then IMP_JOB_bc=1; else IMP_JOB_bc=0;

         drop IMP_JOB;



run;


* convert relevant categorical variables to dummy variables;
/*
* PARENT1 base is 'yes';
if (PARENT1 = 'No') then PARENT1_no=1; else PARENT1_no=0;

* MSTATUS base is 'z_No';
if (MSTATUS = 'Yes') then MSTATUS_yes=1; else MSTATUS_yes=0;

* EDUCATION base is 'PhD';
if (EDUCATION = 'PhD) then EDUCATION_phd=1; else EDUCATION_phd=0;
if (EDUCATION = '<High School') then EDUCATION_lessHS=1; else
EDUCATION_lessHS=0;
if (EDUCATION = 'Bachelors') then EDUCATION_bach=1; else
EDUCATION_bach=0;
if (EDUCATION = 'Masters') then EDUCATION_mast=1; else EDUCATION_mast=0;
if (EDUCATION = 'z_High School') then EDUCATION_zHS=1; else
EDUCATION_zHS=0;
```

```
* CAR_USE base is 'Commercial';
if (CAR_USE = 'Private') then CAR_USE_priv=1; else CAR_USE_priv=0;

* CAR_TYPE base is 'Panel Truck';
if (CAR_TYPE= 'Minivan') then CAR_TYPE_mini=1; else CAR_TYPE_mini=0;
if (CAR_TYPE= 'Pickup') then CAR_TYPE_pick=1; else CAR_TYPE_pick=0;
if (CAR_TYPE= 'Sports Car') then CAR_TYPE_sports=1; else CAR_TYPE_sports=0;
if (CAR_TYPE= 'Van') then CAR_TYPE_van=1; else CAR_TYPE_van=0;
if (CAR_TYPE= 'z_SUV') then CAR_TYPE_suv=1; else CAR_TYPE_suv=0;

* REVOKED base is 'Yes';
if (REVOKED = 'No') then REVOKED_no=1; else REVOKED_no=0;

* URBANICITY base is 'z_Highly Rural/ Rural';
if (URBANICITY = 'Highly Urban/ Urban') then URBANICITY _urban=1; else
URBANICITY _urban=0;

* IMP_JOB base is 'Student';
if (IMP_JOB = 'Clerical') then IMP_JOB_cleric=1; else IMP_JOB_cleric=0;
if (IMP_JOB = 'Doctor') then IMP_JOB_doc=1; else IMP_JOB_doc=0;
if (IMP_JOB = 'Home Maker') then IMP_JOB_home=1; else IMP_JOB_home=0;
if (IMP_JOB = 'Lawyer') then IMP_JOB_law=1; else IMP_JOB_law=0;
if (IMP_JOB = 'Manager') then IMP_JOB_mgr=1; else IMP_JOB_mgr=0;
if (IMP_JOB = 'Professional') then IMP_JOB_pro=1; else IMP_JOB_pro=0;
if (IMP_JOB = 'z_Blue Collar') then IMP_JOB_bc=1; else IMP_JOB_bc=0;

* JOB_HIGH base is 'Doctor';
if (JOB_HIGH = 'Lawyer') then JOB_HIGH_law=1; else JOB_HIGH_law=0;
if (JOB_HIGH = 'Manager') then JOB_HIGH_mgr=1; else JOB_HIGH_mgr=0;
if (JOB_HIGH = 'Professional') then JOB_HIGH_pro=1; else JOB_HIGH_pro=0;

* JOB_LOW base is 'Student';
if (JOB_LOW= 'Clerical') then JOB_LOW_cleric=1; else JOB_LOW_cleric=0;
if (JOB_LOW= 'Home Maker') then JOB_LOW_home=1; else JOB_LOW_home=0;
if (JOB_LOW= 'z_Blue Collar') then JOB_LOW_bc=1; else JOB_LOW_bc=0;

* EDUCATION_HS base is '<High School';
if (EDUCATION_HS = 'z_High School') then EDUCATION_HS_zHS=1; else
EDUCATION_HS_zHS=0;

* CAR_TYPE_HRISK base is 'Sports Car';
if (CAR_TYPE_HRISK= 'z_SUV') then CAR_TYPE_HRISK_suv=1; else
CAR_TYPE_HRISK_suv=0;
if (CAR_TYPE_HRISK= 'Pickup') then CAR_TYPE_HRISK_pick=1; else
CAR_TYPE_HRISK_pick=0;
```

```
*/
*/



* Step 9 - create three logistic regression models

* Model #1 - missing values imputed with averages and
                          the removal of those variables that did not
                          indicate predictive properties;
proc logistic data=&SCRUBFILE1. plot(only)=(roc(ID=prob));
        class EDUCATION CAR_TYPE CAR_USE IMP_JOB MSTATUS PARENT1
REVOKED URBANICITY / param=ref;
   model TARGET_FLAG(ref="0")= KIDSDRIV
                                                            HOMEKIDS
                                                            BLUEBOOK
                                                            TIF
                                                            OLDCLAIM
                                                            CLM_FREQ
                                                            MVR_PTS
                                                            M_YOJ
                                                            IMP_INCOME
                                                            M_INCOME
                                                            IMP_HOME_VAL
                                                            M_HOME_VAL
                                                            IMP_CAR_AGE
                                                            M_CAR_AGE
                                                            M_JOB
                                                            EDUCATION CAR_TYPE
CAR_USE IMP_JOB MSTATUS PARENT1 REVOKED URBANICITY
/selection=backward roceps=0.1;
                                                            output out=bout p=p_crash;

run;


* used to find KS statistic for model;
proc npar1way data=bout;
   class target_flag;
   var p_crash;
run;



* Model #2 - same as Model #1 except includes trimming/removal of outliers;
proc logistic data=&SCRUBFILE2. plot(only)=(roc(ID=prob));
```

```
        class EDUCATION CAR_TYPE CAR_USE IMP_JOB MSTATUS PARENT1
REVOKED URBANICITY / param=ref;
   model TARGET_FLAG(ref="0")= KIDSDRIV
                                HOMEKIDS
                                BLUEBOOK
                                TIF
                                OLDCLAIM
                                CLM_FREQ
                                MVR_PTS
                                M_YOJ
                                IMP_INCOME
                                M_INCOME
                                IMP_HOME_VAL
                                M_HOME_VAL
                                IMP_CAR_AGE
                                M_CAR_AGE
                                M_JOB
                                EDUCATION CAR_TYPE
CAR_USE IMP_JOB MSTATUS PARENT1 REVOKED URBANICITY
/selection=backward roceps=0.1;
                                output out=bout p=p_crash;

run;


* used to find KS statistic for model;
proc npar1way data=bout;
   class target_flag;
   var p_crash;
run;





* Model #3 - same as Model #2 except includes transformed, modified, and new
variables;
proc logistic data=&SCRUBFILE3. plot(only)=(roc(ID=prob));
        class CAR_TYPE_HRISK EDUCATION_HS JOB_LOW JOB_HIGH
CAR_USE MSTATUS PARENT1 REVOKED URBANICITY / param=ref;
   model TARGET_FLAG(ref="0")= KIDSDRIV
                                HOMEKIDS
                                BLUEBOOK
                                TIF
                                OLDCLAIM
```

```
                                                   OLDCLAIM_ZERO

                                                   CLM_FREQ
                                                   MVR_PTS
                                                   M_YOJ
                                                   IMP_INCOME
                                                   M_INCOME
                                                   HOME_VAL_ZERO
                                                   IMP_HOME_VAL
                                                   M_HOME_VAL
                                                   CAR_AGE_NEW
                                                   IMP_CAR_AGE
                                                   M_CAR_AGE
                                                   M_JOB
                                                   CAR_TYPE_HRISK
EDUCATION_HS JOB_LOW JOB_HIGH CAR_USE MSTATUS PARENT1
REVOKED URBANICITY /selection=backward roceps=0.1;
                                                   output out=bout p=p_crash;

run;

* used to find KS statistic for model;
proc npar1way data=bout;
  class target_flag;
  var p_crash;
run;




/*
* Model #4 - same as Model #3 except....;
proc logistic data=&SCRUBFILE4. plot(only)=(roc(ID=prob));
      class CAR_TYPE_HRISK EDUCATION_HS CAR_USE MSTATUS PARENT1
REVOKED URBANICITY / param=ref;
  model TARGET_FLAG(ref="0")= KIDSDRIV
                                                   HOMEKIDS
                                                   BLUEBOOK
                                                   TIF
                                                   OLDCLAIM
                                                   OLDCLAIM_ZERO

                                                   CLM_FREQ
                                                   MVR_PTS
                                                   M_YOJ
                                                   IMP_INCOME
```

```
                                                  M_INCOME
                                                  HOME_VAL_ZERO
                                                  IMP_HOME_VAL
                                                  M_HOME_VAL
                                                  CAR_AGE_NEW
                                                  IMP_CAR_AGE
                                                  M_CAR_AGE
                                                  M_JOB
                                                  IMP_JOB_cleric
                                                  IMP_JOB_doc
                                                  IMP_JOB_law
                                                  IMP_JOB_home
                                                  IMP_JOB_pro
                                                  IMP_JOB_student
                                                  IMP_JOB_bc
                                                  CAR_TYPE_HRISK
EDUCATION_HS CAR_USE MSTATUS PARENT1 REVOKED URBANICITY
/selection=backward roceps=0.1;
run;
*/




* Step 10 - DEPLOY model against the insurance_test dataset;

* finding the mean of TARGET_AMT to find value to multiply against probability of
crash;
proc means data=&INFILE. mean;
var TARGET_AMT;
run;


data SCOREFILE;
   set &TEST.;

         * Fix numeric missing values;

         * AGE - age imputations;
   IMP_AGE=AGE;
   if missing(IMP_AGE) then
      IMP_AGE=44.7903127; *mean;
   DROP AGE;

   * YOJ - year on job imputations;
   IMP_YOJ=YOJ;
```

```
M_YOJ=0;

if missing(IMP_YOJ) then
   do;
      IMP_YOJ=10.4992864;
      *mean;
      M_YOJ=1;
   end;
DROP YOJ;

* INCOME - income imputations;
IMP_INCOME=INCOME;
M_INCOME=0;

if missing(IMP_INCOME) then
   do;
      IMP_INCOME=61898.10;
      *mean;
      M_INCOME=1;
   end;
drop INCOME;

* HOME_VAL - home value imputations;
IMP_HOME_VAL=HOME_VAL;
M_HOME_VAL=0;

*if missing( IMP_HOME_VAL ) then 161159.53;
if missing(IMP_HOME_VAL) then
   do;

      if YOJ >=4 and INCOME > 50000 and BLUEBOOK > 10000 then
         do;
            IMP_HOME_VAL=161159.53;
            *median;
            M_HOME_VAL=1;
         end;
      else
         do;
            IMP_HOME_VAL=0;
            * assuming driver is a home renter;
            M_HOME_VAL=1;
         end;
   end;
drop HOME_VAL;

* CAR_AGE - car age imputation;
```

```
IMP_CAR_AGE=CAR_AGE;
M_CAR_AGE=0;

*if missing( IMP_CAR_AGE )then IMP_CAR_AGE = 8.328;
if missing(IMP_CAR_AGE)then
   do;
      if IMP_INCOME > 86000 then
         do;
            IMP_CAR_AGE=1;
            M_CAR_AGE=1;
         end;
      else
         do;
            IMP_CAR_AGE=9.6;
            M_CAR_AGE=1;
         end;
   end;
drop CAR_AGE;

* Fix character variables;

* JOB - job type imputation;
IMP_JOB=JOB;
M_JOB=0;

if missing(IMP_JOB) then
   do;
      if IMP_INCOME > 140000 then
         IMP_JOB="Doctor";
      else if IMP_INCOME > 105000 then
         IMP_JOB="Lawyer";
      else if IMP_INCOME > 90000 then
         IMP_JOB="Professional";
      else if IMP_INCOME > 75000 then
         IMP_JOB="Manager";
      else if IMP_INCOME < 30000 AND IMP_INCOME > 10000 then
         IMP_JOB="Clerical";
      else if IMP_INCOME <=10000 AND HOMEKIDS > 0 then
         IMP_JOB="Home Maker";
      else if IMP_INCOME <=10000 then
         IMP_JOB="Student";
      else
         IMP_JOB="z_Blue Collar";
   end;
drop JOB;
```

```
   * category variables dropped;
   drop RED_CAR;
   drop SEX;

   *numeric variables dropped;
   drop TRAVTIME;
   drop IMP_AGE;
         drop IMP_YOJ;

   OLDCLAIM_ZERO = 0; * creating variable for claims of only zero;
         if IMP_OLDCLAIM = 0 then OLDCLAIM_ZERO = 1;

         HOME_VAL_ZERO = 0; * creating variable for home value of only zero (non-
home owners);
         if IMP_HOME_VAL = 0 then HOME_VAL_ZERO = 1;

         CAR_AGE_NEW = 0; * creating variable for new cars only;
         if IMP_CAR_AGE <= 1 then CAR_AGE_NEW = 1;

   * new or transformed - categorical variables;
   CAR_TYPE_HRISK = CAR_TYPE in ("z_SUV","Sports Car","Pickup"); * creating
variable of only high risk car types;
   EDUCATION_HS = EDUCATION in ("<High School","z_High School"); * creating
variable of high school or less education;
   JOB_LOW = IMP_JOB in ("z_Blue Collar","Student","Home Maker","Clerical"); *
creating variable of low paying jobs;
   JOB_HIGH = IMP_JOB in ("Doctor","Lawyer","Manager","Professional"); * creating
variable of high paying jobs;


   YHAT=-1.1307                                    +
               0.3985*KIDSDRIV                     +
               -0.00001*BLUEBOOK                       +
               -0.0547*TIF                             +
               -0.00002*OLDCLAIM                       +
               0.2078*CLM_FREQ                         +
               0.1222*MVR_PTS                          +
               -.00000457*IMP_INCOME               +
               0.2718*HOME_VAL_ZERO                +
               -0.4821*(CAR_TYPE_HRISK    in ("0"))        +
               -0.3784*(EDUCATION_HS in ("0"))         +
               -0.3660*(JOB_LOW in     ("0"))          +
               0.7667*(CAR_USE   in ("Commercial"))            +
               -0.4686*(MSTATUS in ("Yes")     )           +
               -0.4181*(PARENT1 in ("No")      )           +
               -0.8531*(REVOKED in ("No"))                 +
```

```
                  2.2280*(URBANICITY in ("Highly Urban/ Urban"));

   YHAT=exp(YHAT);
   PROB=YHAT / (1+YHAT);

   P_TARGET_FLAG = PROB;
         P_TARGET_AMT = P_TARGET_FLAG * 1504.32; * this is the average/mean
amount paid for a crash;
         drop PROB;

         keep INDEX;
         keep P_TARGET_FLAG;
         keep P_TARGET_AMT;

run;


* check that model deploys correctly and there are NO missing values;
proc print data=SCOREFILE(obs=10);
run;

proc means data=SCOREFILE nmiss;
   var P_TARGET_FLAG;
run;







* Step 11 - SCORE MODEL against insurance_test;

*****************************************************;
****     CREATE FILE TO STORE SCORED DATA       ****;
*****************************************************;

* print a few observations to ensure can access the dataset (moneyball_test);
         proc print data=&TEST. (obs=5);
         title10 "Testing Access to Insurance_test - dataset";
         run;

         title10 ;

* code to store scored code into my SAS folder Assignments;
         libname scorelib "/home/derekhughes2014/Assignments";
```

```
	data scorelib.DEREK_HUGHES_FILE_insurance_test;
		set SCOREFILE;
	run;

* view scored data on Moneyball_test - click "download" button
	* in Folders to get this file on local CPU;
	proc print data=scorelib.DEREK_HUGHES_FILE_insurance_test (obs=10);
		title10 "Model#3 vs Dataset in SCOREfILE that's saved to CPU -
(SCOREFILE currently set to insurance_test dataset";
	run;

	title10 ;




*********************************************;
******		BINGO BONUS		**********;
*********************************************;

proc genmod data=&SCRUBFILE3.;
				class CAR_TYPE_HRISK EDUCATION_HS JOB_LOW
JOB_HIGH CAR_USE MSTATUS PARENT1 REVOKED URBANICITY / param=ref;
			model TARGET_FLAG= KIDSDRIV
										HOMEKIDS
										BLUEBOOK
										TIF
										OLDCLAIM

	OLDCLAIM_ZERO
										CLM_FREQ
										MVR_PTS
										M_YOJ

	IMP_INCOME
										M_INCOME

	HOME_VAL_ZERO

	IMP_HOME_VAL

	M_HOME_VAL
```

```
        CAR_AGE_NEW

        IMP_CAR_AGE

        M_CAR_AGE
                                                                M_JOB

        CAR_TYPE_HRISK EDUCATION_HS JOB_LOW JOB_HIGH CAR_USE
MSTATUS PARENT1 REVOKED URBANICITY / link=logit;
            run;
```