HW 03 WINE
Taylor Vender
PREDICT 411 SEC 60

KAGGLE NAME: Taylor V

BINGO BONUS: 55 Total Points

- Hurdle Logistic/Negative Binomial model (20 pts)
- Decision trees to impute missing values for two of the variables (20 pts)
- Use of SAS Macros (10 pts)
- Recreate a model in "R" (5 pts)

INTRODUCTION:

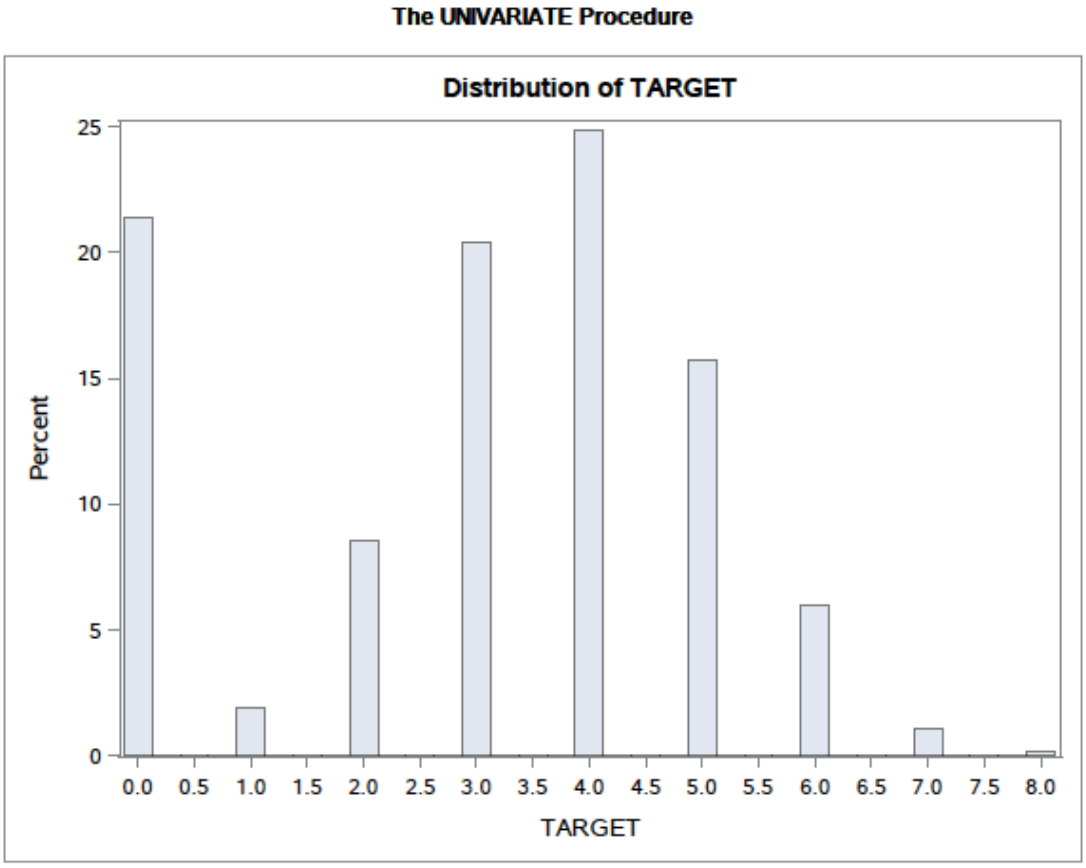
The objective of this assignment is to help a large wine manufacturer predict the number of sample cases of wine that would be purchased by wine distribution companies after sampling a wine. These cases would be used to provide tasting samples to restaurants and wine stores around the United States; so, the more sample cases purchased, the more likely the wine is to be sold at a high-end restaurant. By predicting the number of cases that would be sold given certain properties of the wine, the manufacturer would be able to adjust their wine offering to maximize sales.

First, we will perform an exploratory data analysis and then make the necessary variable transformations and clean up the missing values. Next, we will try various modeling approaches to predict the number of cases of wine sold. Since the number of cases sold is a "count variable" the suitable models we will try are a Logistic Hurdle Model, Linear Regression, Poisson Regression, Negative Binomial Regression, Zero Inflated Negative Binomial Regression, and Zero Inflated Poisson Regression.

RESULTS:

DATA EXPLORATION:

The dataset contains information about 12795 commercially available wines (this is the number of observations in the dataset). The TARGET variable is the number of cases purchased. Before doing a regression, it is a good idea to determine the distribution of the target variable.



Above is the distribution of the wine dataset representing the number of cases sold. Notice the data appears to have a Poisson or a Negative Binomial Distribution, with a spike at the zero values. This implies that we have "ZIP Counting Data" and a zero inflated model will probably work best, although we will try out other models as well.

The predictor variables are mostly related to the chemical properties of the wine being sold and they include

VARIABLE NAME	DEFINITION	THEORETICAL EFFECT
AcidIndex	Proprietary method of testing total acidity of wine by using a weighted average	
Alcohol	Alcohol Content	
Chlorides	Chloride content of wine	
CitricAcid	Citric Acid Content	
Density	Density of Wine	
FixedAcidity	Fixed Acidity of Wine	
FreeSulfurDioxide	Sulfur Dioxide content of wine	
LabelAppeal	Marketing Score indicating the appeal of label design for consumers. High numbers suggest customers like the label design. Negative numbers suggest customes don't like the design.	Many consumers purchase based on the visual appeal of the wine label design. Higher numbers suggest better sales.
ResidualSugar	Residual Sugar of wine	
STARS	Wine rating by a team of experts. 4 Stars = Excellent, 1 Star = Poor	A high number of stars suggests high sales
Sulphates	Sulfate conten of wine	
TotalSulfurDioxide	Total Sulfur Dioxide of Wine	
VolatileAcidity	Volatile Acid content of wine	
pH	pH of wine	

As you can see, we are dealing with only numeric independent variables.

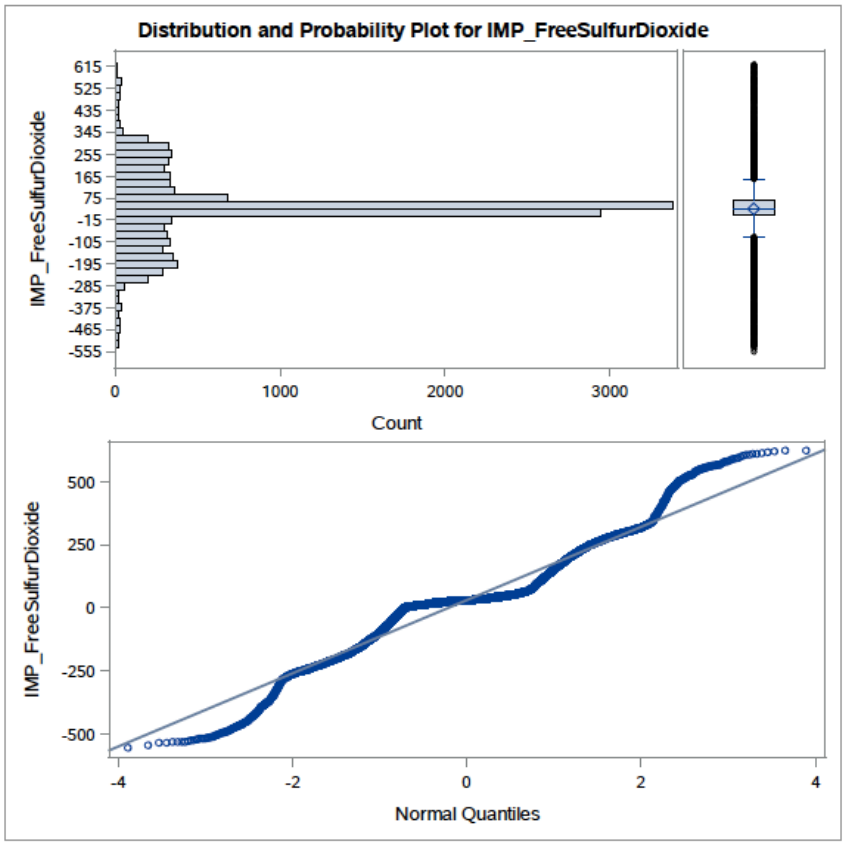
Now, we will examine the predictor variables to see if they have any missing values. Prior to running any regression models, we will replace any missing values with the mean or median of the variable or by using a decision tree.

The MEANS Procedure

Variable	N	N Miss	Mean	Median
INDEX	12795	0	8069.98	8110.00
TARGET	12795	0	3.0290739	3.0000000
FixedAcidity	12795	0	7.0757171	6.9000000
VolatileAcidity	12795	0	0.3241039	0.2800000
CitricAcid	12795	0	0.3084127	0.3100000
ResidualSugar	12179	616	5.4187331	3.9000000
Chlorides	12157	638	0.0548225	0.0460000
FreeSulfurDioxide	12148	647	30.8455713	30.0000000
TotalSulfurDioxide	12113	682	120.7142326	123.0000000
Density	12795	0	0.9942027	0.9944900
pH	12400	395	3.2076282	3.2000000
Sulphates	11585	1210	0.5271118	0.5000000
Alcohol	12142	653	10.4892363	10.4000000
LabelAppeal	12795	0	-0.0090660	0
AcidIndex	12795	0	7.7727237	8.0000000
STARS	9436	3359	2.0417550	2.0000000

While ResidualSugar, Chlorides, FreeSulfurDioxide, TotalSulfurDioxide, pH, Sulphates, Alcohol, and STARS all have missing values, it is important to notice that the number of missing values isn't an overwhelmingly large proportion of the total number of observations for any of the variables. Just based on this observation, I don't find it necessary to remove any of these variables from the model. I am comfortable imputing the variables with their medians or using a decision tree to fix the missing values.

Once the missing values are fixed, I look at plots showing the distributions of each of the predictor variables. I looked at the boxplots, histrograms, and quantiles for each of the variables to check for outliers. For example, below are the plots for IMP_FreeSulfurDioxide, which is the variable FreeSulfurDioxide with the imputated missing values that I replaced with its median.



Quantiles (Definition 5)	
Level	**Quantile**
100% Max	623
99%	464
95%	281
90%	223
75% Q3	64
50% Median	30
25% Q1	5
10%	-165
5%	-220
1%	-382
0% Min	-555

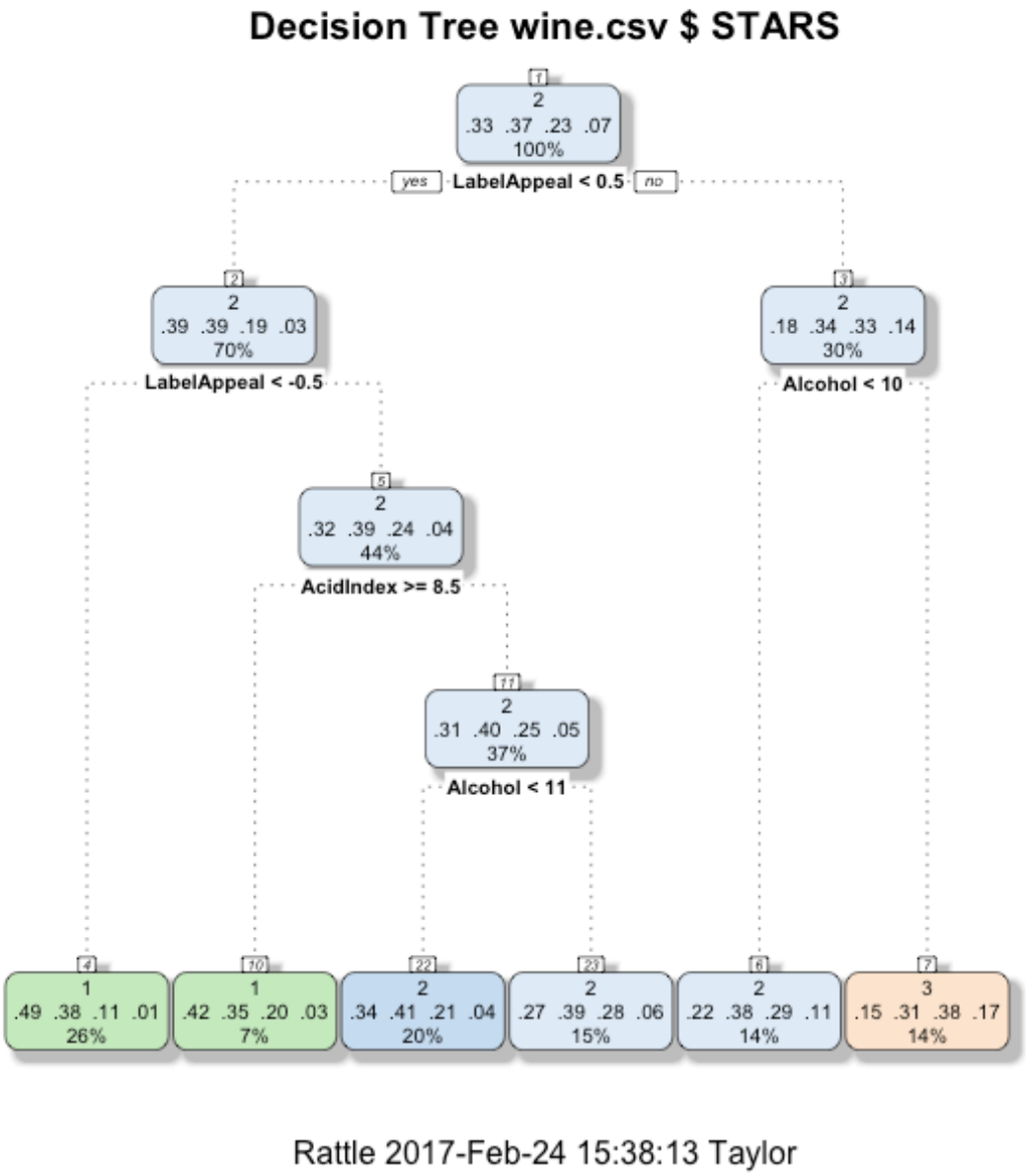
By looking at the plots, we see that the distribution is approaching normal (the 45 degree line), but the range in the histogram is extremely wide and there are a lot of outliers on the boxplot. Since I didn't want to eliminate any potential predictive values, I capped the values at the 1st and 99th percentiles by creating a new variable CAP_IMP_FreeSulfurDioxide. So, if IMP_FreeSulfurDioxide < -382 then CAP_IMP_FreeSulfurDioxide = -382 and if IMP_FreeSulfurDioxide > 464 then CAP_IMP_FreeSulfurDioxide = 464.

Additionally, for the variables FixedAcidity, CitricAcid, IMP_Density, IMP_Sulphates, IMP_pH, IMP_Alcohol, IMP_TotalSulfurDioxide, IMP_Chlorides, and IMP_ResidualSugars (where variables names preceded by IMP are new variables I created with the imputed missing values), I capped the values at the 1st and 99th percentiles to eliminate some of the outliers that I identified during my analysis.

DATA PREPARATION:

My first step in preparing the data is to convert the our target variable (TARGET) into two variables, a binary target variable (TARGET_FLAG) and a count variable (TARGET_AMT). TARGET_FLAG will be "1" if the number of cases sold is greater than 0. Otherwise It will have a value of "0". TARGET_AMT will have a Count value if TARGET_FLAG = 1, otherwise it will be set to missing. We will come back to the importance of this step in the next section when we build our models.

Next, I impute the missing values for the variables discussed in the previous section. Originally, I used a decision tree to impute the missing values for STARS. I chose STARS because it is the variable with the most missing values, and I think that decision trees are slightly more robust in replacing missing values than the means or medians, although those methods are both suitable alternatives. Below is a decision tree with four levels that I created in R in order to predict the missing STARS values based on the other predictor variable values in the record.



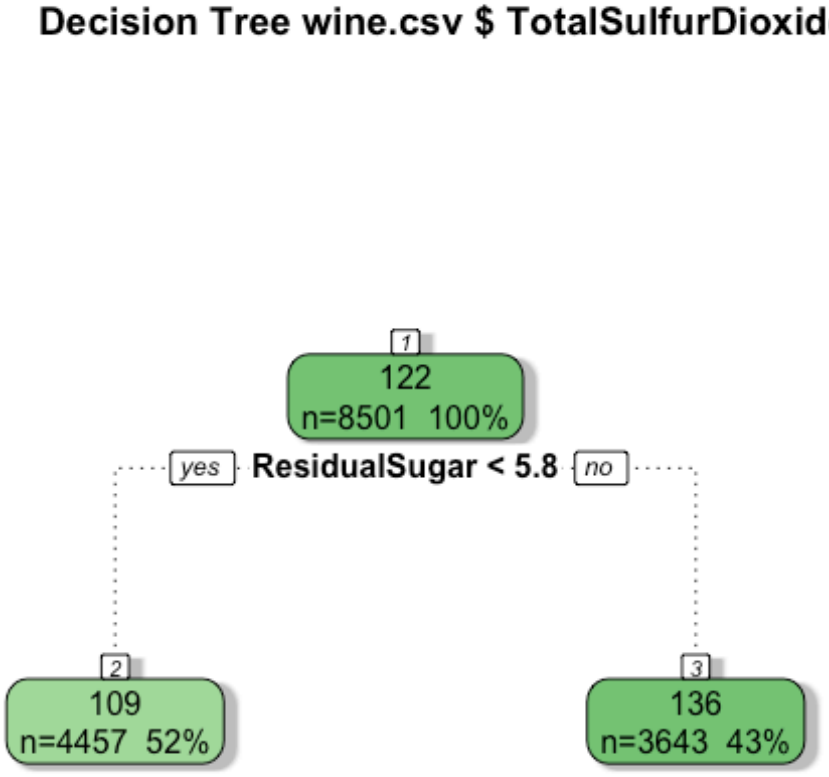
However, after replacing the missing values with the tree, I took a step back and compared the frequency of the values of STARS for both values of our TARGET_FLAG variable with 0 indicating that no cases were sold and 1 indicating that at least one case was sold. The table used for this analysis is shown below.

The FREQ Procedure

Frequency Percent Row Pct Col Pct	Table of STARS by TARGET_FLAG		
	TARGET_FLAG		
STARS	0	1	Total
.	2038 15.93 60.67 74.54	1321 10.32 39.33 13.13	3359 26.25
1	607 4.74 19.95 22.20	2435 19.03 80.05 24.20	3042 23.77
2	89 0.70 2.49 3.26	3481 27.21 97.51 34.60	3570 27.90
3	0 0.00 0.00 0.00	2212 17.29 100.00 21.99	2212 17.29
4	0 0.00 0.00 0.00	612 4.78 100.00 6.08	612 4.78
Total	2734 21.37	10061 78.63	12795 100.00

Looking at the third line of the boxes corresponding to the missing value of stars, we see that for 60.67% of the wines with missing, or no stars, no cases were sold. As compared to the total 21.37% of the wines that weren't sold. This missing value is clearly predictive of whether or not any cases of the wine were sold. Thus, I decided to create a variable IMP_STARS that replaces missing values with a value of 0. Since I created the model with the original variable IMP_STARS as having missing values replaced by the decision tree, I will still show the comparison of the two models although I will not fully go into the building of the model with IMP_STARS using the decision tree.

Even though I didn't use a tree in my final models for the variable STARS, I still wanted to use a tree to impute the missing values for one of my variables I use in my models. So, I decided to impute the missing values of TotalSulfurDioxide.



Interesting enough, although I asked R to create a tree with 4 levels, ResidualSugar was the only variable used to predict the missing values of TotalSulfurDioxide.

Next, I imputed the rest of the missing values for the variables that I discussed in the Data Exploration section using their respective medians.

Once all of the missing values were imputed, I checked the distributions of the predictor variables for outliers, and capped the values where I saw fit. I explained this process in the previous section, Data Exploration.

BUILD MODELS:

First we created a model using Linear regression. As we have discussed, linear regression is not necessarily appropriate for counting data because this violates the assumptions of linear regression. However, linear models are easy to deploy and interpret and, although the assumptions are violated, it is very possible that we could still get good results. So, we will start with a linear regression model, using the automated stepwise variable selection and see how these results compare to "proper" modeling methods.

Parameter Estimates						
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Variance Inflation
Intercept	1	4.03406	0.46336	8.71	<.0001	0
AcidIndex	1	-0.20848	0.00908	-22.97	<.0001	1.05360
CAP_CitricAcid	1	0.02293	0.01411	1.62	0.1043	1.00618
CAP_IMP_Alcohol	1	0.01123	0.00335	3.35	0.0008	1.00608
CAP_IMP_Chlorides	1	-0.12687	0.03903	-3.25	0.0012	1.00275
CAP_IMP_Density	1	-0.83584	0.45667	-1.83	0.0672	1.00335
CAP_IMP_FreeSulfurDioxide	1	0.00032875	0.00008367	3.93	<.0001	1.00315
CAP_IMP_Sulphates	1	-0.03449	0.01372	-2.51	0.0119	1.00224
CAP_IMP_TotalSulfurDioxide	1	0.00023487	0.00005397	4.35	<.0001	1.00436
CAP_IMP_pH	1	-0.03941	0.01811	-2.18	0.0296	1.00494
IMP_STARS	1	0.97670	0.01045	93.45	<.0001	1.12242
LabelAppeal	1	0.43247	0.01366	31.65	<.0001	1.08211
VolatileAcidity	1	-0.09904	0.01498	-6.61	<.0001	1.00628

Unfortunately, the stepwise selection did not eliminate any of the predictor variables, so we will have to use another method of variable selection for the models that follow. Looking at the parameter estimates, we see that IMP_STARS and LabelAppeal both have positive coefficients. According to the theoretical effect of those two variables from the data dictionary shown in the first section of this write up, higher values of STARS and LabelAppeal imply higher sales, so we would expect both of these variables to have positive coefficients.

Next I built a Logistic Hurdle Model. First, we build a Logistic Regression Model to predict whether the number of cases sold is "0" or "not 0". Next, we build a model using Negative Binomial Regression to predict the count value of the number of cases sold (assuming it is not zero). Lastly we multiply the probably of "not 0" by the

count to get the expected number of cases sold. I prefer to build this model first because running logistic regression in our software allows for stepwise variable selection. Since this selection will choose the "best" predictor variables for us, I will continue to use these same selected variables for the rest of my models.

Analysis of Maximum Likelihood Estimates					
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	3.4196	0.2511	185.4419	<.0001
AcidIndex	1	-0.3909	0.0213	337.5715	<.0001
CAP_IMP_Alcohol	1	-0.0224	0.00823	7.4211	0.0064
CAP_IMP_FreeSulfurDi	1	0.000646	0.000207	9.7063	0.0018
CAP_IMP_Sulphates	1	-0.1120	0.0338	10.9703	0.0009
CAP_IMP_TotalSulfurD	1	0.000893	0.000132	45.4803	<.0001
CAP_IMP_pH	1	-0.1984	0.0444	20.0089	<.0001
IMP_STARS	1	2.0534	0.0431	2269.6920	<.0001
LabelAppeal	1	-0.4591	0.0334	188.6516	<.0001
VolatileAcidity	1	-0.1852	0.0368	25.3877	<.0001

The coefficients above are to predict if a case of wine is sold (TARGET > 0). Since we are using this model to predict the probility that the count value (number of cases sold) is greater than zero, we would expect LabelAppeal to have a positive coefficient. I am interested to see if we see this happen with the ZIP model and ZINB that will follow, but for now I do not have any explanation for why this coefficient might be negative.

The table below shows the coefficients to predict the number of cases sold (minus 1), assuming that at least one case was sold.

Analysis Of Maximum Likelihood Parameter Estimates							
Parameter	DF	Estimate	Standard Error	Wald 95% Confidence Limits		Wald Chi-Square	Pr > ChiSq
Intercept	1	0.8179	0.0581	0.7039	0.9318	197.92	<.0001
AcidIndex	1	-0.0207	0.0055	-0.0314	-0.0100	14.44	0.0001
CAP_IMP_Alcohol	1	0.0095	0.0017	0.0062	0.0128	31.32	<.0001
CAP_IMP_FreeSulfurDi	1	0.0000	0.0000	-0.0001	0.0001	0.59	0.4413
CAP_IMP_Sulphates	1	0.0005	0.0070	-0.0132	0.0141	0.00	0.9469
CAP_IMP_TotalSulfurD	1	-0.0000	0.0000	-0.0001	0.0000	1.62	0.2033
CAP_IMP_pH	1	0.0093	0.0092	-0.0087	0.0274	1.03	0.3110
IMP_STARS	1	0.1152	0.0058	0.1038	0.1267	390.55	<.0001
LabelAppeal	1	0.2959	0.0072	0.2818	0.3100	1693.25	<.0001
VolatileAcidity	1	-0.0131	0.0076	-0.0280	0.0019	2.94	0.0866
Dispersion	1	0.0000	0.0001	0.0000	7.07E129		

I built the next model using Zero Inflated Poisson Regression. Similar to the hurdle model, a ZIP Model requires us to build two models. However, this can be done with the addition of a single line of code to predict the probability that the count is a "zero" value, so this model is slightly easier to deploy. The first set of coefficients directly below are for predicting the number of cases of wine sold while the second set of coefficients are for predicting the probability that a value will be zero.

Analysis Of Maximum Likelihood Parameter Estimates							
Parameter	DF	Estimate	Standard Error	Wald 95% Confidence Limits		Wald Chi-Square	Pr > ChiSq
Intercept	1	1.1601	0.0514	1.0594	1.2609	509.40	<.0001
AcidIndex	1	-0.0187	0.0048	-0.0282	-0.0092	14.97	0.0001
CAP_IMP_Alcohol	1	0.0073	0.0015	0.0044	0.0102	23.77	<.0001
CAP_IMP_FreeSulfurDi	1	0.0000	0.0000	-0.0000	0.0001	0.47	0.4909
CAP_IMP_Sulphates	1	0.0004	0.0061	-0.0116	0.0125	0.01	0.9434
CAP_IMP_TotalSulfurD	1	-0.0000	0.0000	-0.0001	0.0000	0.73	0.3943
CAP_IMP_pH	1	0.0055	0.0081	-0.0104	0.0213	0.45	0.5006
IMP_STARS	1	0.1010	0.0052	0.0908	0.1112	377.32	<.0001
LabelAppeal	1	0.2328	0.0063	0.2205	0.2452	1365.38	<.0001
VolatileAcidity	1	-0.0122	0.0067	-0.0254	0.0010	3.30	0.0692
Scale	0	1.0000	0.0000	1.0000	1.0000		

The GENMOD Procedure

Analysis Of Maximum Likelihood Zero Inflation Parameter Estimates							
Parameter	DF	Estimate	Standard Error	Wald 95% Confidence Limits		Wald Chi-Square	Pr > ChiSq
Intercept	1	-4.0264	0.3015	-4.6174	-3.4355	178.35	<.0001
AcidIndex	1	0.4347	0.0253	0.3852	0.4842	296.17	<.0001
CAP_IMP_Alcohol	1	0.0296	0.0099	0.0102	0.0490	8.98	0.0027
CAP_IMP_FreeSulfurDi	1	-0.0008	0.0002	-0.0013	-0.0003	10.09	0.0015
CAP_IMP_Sulphates	1	0.1391	0.0404	0.0599	0.2183	11.85	0.0006
CAP_IMP_TotalSulfurD	1	-0.0010	0.0002	-0.0013	-0.0007	42.43	<.0001
CAP_IMP_pH	1	0.2312	0.0530	0.1273	0.3351	19.03	<.0001
IMP_STARS	1	-2.3788	0.0603	-2.4971	-2.2605	1554.43	<.0001
LabelAppeal	1	0.7235	0.0429	0.6393	0.8076	283.91	<.0001
VolatileAcidity	1	0.1935	0.0439	0.1074	0.2795	19.43	<.0001

Unlike for the logistic portion of the hurdle model, the coefficient estimates in the Analysis of Maximum Likelihood Zero Inflation Parameter Estimates are for predicting the probability that no cases were sold (instead of the probability that TARGET>0). So, the fact that IMP_STARS has a negative coefficient and LabelAppeal has a positive coefficient actually corresponds to the reverse in the previous model.

For predicting the number of cases sold, assuming that any cases were sold in the Hurdle and the ZIP models, the coefficients for IMP_STARS and LabelAppeal are both positive, which is what we would expect.

Now, since the Poisson and Negative Binomial models gave the same results, I wanted to change the input variables so that I could get different models. Based on the Chi-Square test statistics and the associated p-values (PR>ChiSq) from the ZIP model, I was choosing between eliminated CAP_IMP_FreeSulfurDioxide and CAP_IMP_Alcohol. Although CAP_IMP_Alcohol has the larger p-value, I would assume based on prior knowledge that Alcohol content has some weight on deciding whether or not to buy a certain wine. Also, since we are leaving CAP_IMP_TotalSulfurDioxide in the model, we are still accounting for the Total Sulfur Dioxide of wine and probably won't miss out on too much if any accuracy by dropping the Sulfur Dioxide Content in our Zero Inflate Negative Binomial Model.

Analysis Of Maximum Likelihood Parameter Estimates							
Parameter	DF	Estimate	Standard Error	Wald 95% Confidence Limits		Wald Chi-Square	Pr > ChiSq
Intercept	1	1.1630	0.0515	1.0621	1.2638	510.46	<.0001
AcidIndex	1	-0.0185	0.0049	-0.0280	-0.0090	14.56	0.0001
CAP_IMP_Alcohol	1	0.0072	0.0015	0.0043	0.0102	23.34	<.0001
CAP_IMP_Sulphates	1	0.0006	0.0062	-0.0114	0.0127	0.01	0.9195
CAP_IMP_TotalSulfurD	1	-0.0000	0.0000	-0.0001	0.0000	0.74	0.3911
CAP_IMP_pH	1	0.0055	0.0081	-0.0104	0.0214	0.46	0.4984
IMP_STARS	1	0.1002	0.0052	0.0899	0.1104	369.58	<.0001
LabelAppeal	1	0.2323	0.0063	0.2199	0.2447	1349.16	<.0001
VolatileAcidity	1	-0.0121	0.0067	-0.0253	0.0011	3.21	0.0734
Dispersion	0	0.0019	0.0000	0.0019	0.0019		

The GENMOD Procedure

Analysis Of Maximum Likelihood Zero Inflation Parameter Estimates							
Parameter	DF	Estimate	Standard Error	Wald 95% Confidence Limits		Wald Chi-Square	Pr > ChiSq
Intercept	1	-3.9034	0.2905	-4.4727	-3.3341	180.57	<.0001
AcidIndex	1	0.4207	0.0243	0.3731	0.4683	300.06	<.0001
CAP_IMP_Alcohol	1	0.0288	0.0095	0.0101	0.0475	9.11	0.0025
CAP_IMP_Sulphates	1	0.1338	0.0390	0.0573	0.2103	11.74	0.0006
CAP_IMP_TotalSulfurD	1	-0.0010	0.0002	-0.0013	-0.0007	41.89	<.0001
CAP_IMP_pH	1	0.2159	0.0512	0.1156	0.3163	17.78	<.0001
IMP_STARS	1	-2.2174	0.0540	-2.3233	-2.1115	1684.13	<.0001
LabelAppeal	1	0.6868	0.0411	0.6062	0.7673	279.40	<.0001
VolatileAcidity	1	0.1862	0.0424	0.1031	0.2692	19.31	<.0001

Even with dropping a variable, the coefficients of the remaining variables in our ZINB model are very similar to those in the ZIP model, with all of the coefficient signs being the same.

Lastly, even though there is a large spike at zero in the distribution of the target variable, there I definitely similarities between the distributions of a the Poisson and Negative Binomial models. Since both Poisson and Negative Binomial regression models are pretty simple, it is worth it to try both because they might work.

Analysis Of Maximum Likelihood Parameter Estimates							
Parameter	DF	Estimate	Standard Error	Wald 95% Confidence Limits		Wald Chi-Square	Pr > ChiSq
Intercept	1	1.2482	0.0491	1.1520	1.3444	647.01	<.0001
AcidIndex	1	-0.0874	0.0045	-0.0962	-0.0786	379.23	<.0001
CAP_IMP_Alcohol	1	0.0024	0.0015	-0.0005	0.0052	2.66	0.1032
CAP_IMP_FreeSulfurDi	1	0.0001	0.0000	0.0001	0.0002	12.98	0.0003
CAP_IMP_Sulphates	1	-0.0133	0.0060	-0.0250	-0.0016	4.98	0.0256
CAP_IMP_TotalSulfurD	1	0.0001	0.0000	0.0000	0.0001	13.30	0.0003
CAP_IMP_pH	1	-0.0170	0.0079	-0.0325	-0.0016	4.66	0.0308
IMP_STARS	1	0.3117	0.0045	0.3028	0.3205	4736.40	<.0001
LabelAppeal	1	0.1330	0.0061	0.1211	0.1449	481.56	<.0001
VolatileAcidity	1	-0.0337	0.0065	-0.0465	-0.0209	26.74	<.0001
Scale	0	1.0000	0.0000	1.0000	1.0000		

Analysis Of Maximum Likelihood Parameter Estimates							
Parameter	DF	Estimate	Standard Error	Wald 95% Confidence Limits		Wald Chi-Square	Pr > ChiSq
Intercept	1	1.2482	0.0491	1.1520	1.3444	647.01	<.0001
AcidIndex	1	-0.0874	0.0045	-0.0962	-0.0786	379.23	<.0001
CAP_IMP_Alcohol	1	0.0024	0.0015	-0.0005	0.0052	2.66	0.1032
CAP_IMP_FreeSulfurDi	1	0.0001	0.0000	0.0001	0.0002	12.98	0.0003
CAP_IMP_Sulphates	1	-0.0133	0.0060	-0.0250	-0.0016	4.98	0.0256
CAP_IMP_TotalSulfurD	1	0.0001	0.0000	0.0000	0.0001	13.30	0.0003
CAP_IMP_pH	1	-0.0170	0.0079	-0.0325	-0.0016	4.66	0.0308
IMP_STARS	1	0.3117	0.0045	0.3028	0.3205	4736.39	<.0001
LabelAppeal	1	0.1330	0.0061	0.1211	0.1449	481.56	<.0001
VolatileAcidity	1	-0.0337	0.0065	-0.0465	-0.0209	26.74	<.0001
Dispersion	0	0.0000	0.0000	0.0000	0.0000		

Above, the first set of parameter estimates are for the Poisson Model and the second set of parameter estimates are for the Negative Binomial Model. While it does not come as much of a surprise based on the nature of the two models at hand, it is

interesting to see that the output is identical. IMP_STARS and LabelAppeal both have positive coefficients, which aligns with out expectations as well as the other count models for number of cases sold.

SELECT MODELS:

Negative Binomial Model

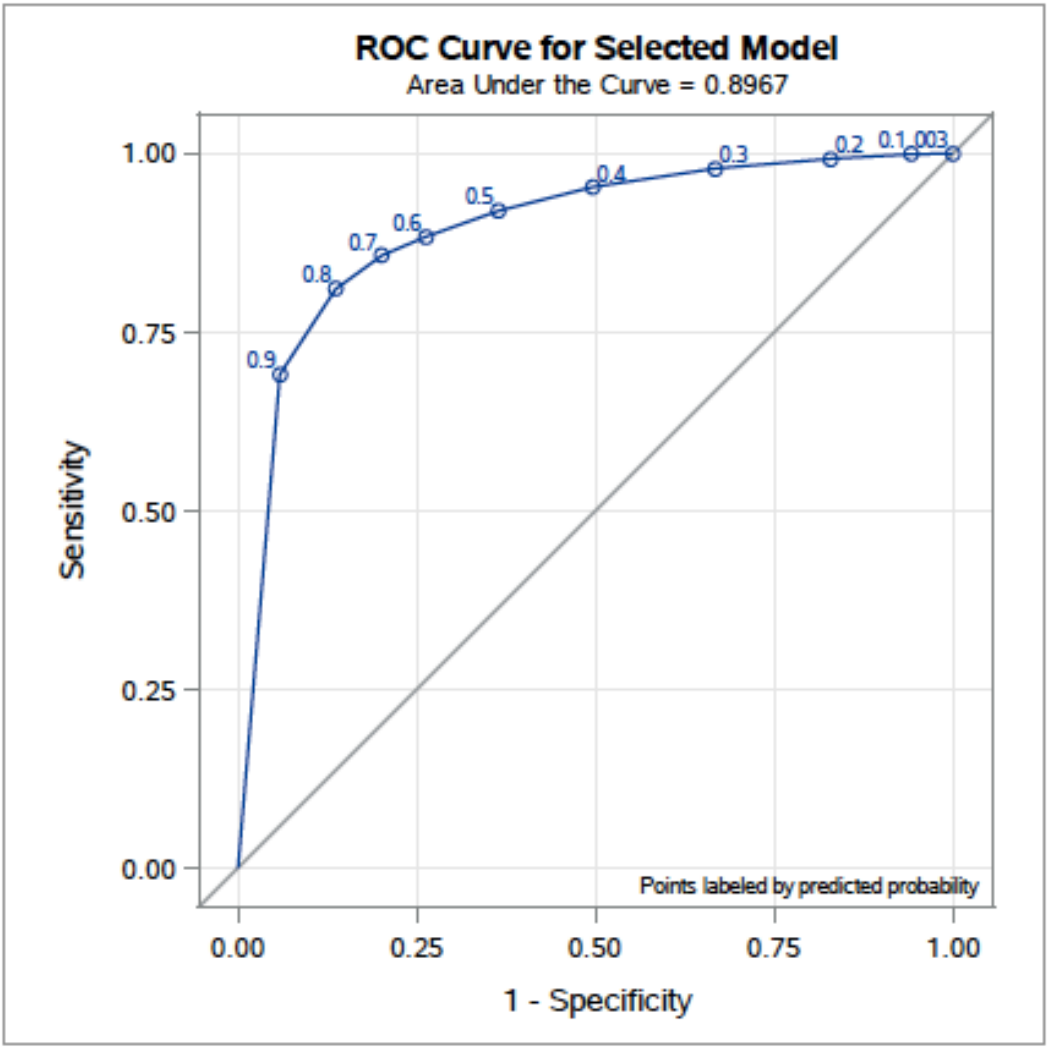
Criteria For Assessing Goodness Of Fit			
Criterion	DF	Value	Value/DF
Deviance	13E3	14738.4244	1.1528
Scaled Deviance	13E3	14738.4244	1.1528
Pearson Chi-Square	13E3	10891.7531	0.8519
Scaled Pearson X2	13E3	10891.7531	0.8519
Log Likelihood		8256.9482	
Full Log Likelihood		-23340.2230	
AIC (smaller is better)		46702.4461	
AICC (smaller is better)		46702.4667	
BIC (smaller is better)		46784.4710	

Poisson Model

Criteria For Assessing Goodness Of Fit			
Criterion	DF	Value	Value/DF
Deviance	13E3	14738.4244	1.1528
Scaled Deviance	13E3	14738.4244	1.1528
Pearson Chi-Square	13E3	10891.7662	0.8519
Scaled Pearson X2	13E3	10891.7662	0.8519
Log Likelihood		8256.9482	
Full Log Likelihood		-23340.2230	
AIC (smaller is better)		46700.4461	
AICC (smaller is better)		46700.4633	
BIC (smaller is better)		46775.0142	

When comparing models I like to use the AIC metric. The AIC for the Negative Binomial Model is 46702.4461 and the AIC for the Poisson Model is 46700.4461. These are extremely close, which implies that the accuracy of the models is very similar. However, due to the "zero inflated" nature of our data, I will not choose either of these models as our "best" model.

Since AIC metrics are only comparable or like models, we will look at the ROC to validate our logistic model portion of the hurdle model. As you can see below, the bow shape of the ROC curve implies that our model is much better than a random model.



Lastly, we will look at the AICs for the ZINB and ZIP model respectively.

Criteria For Assessing Goodness Of Fit			
Criterion	DF	Value	Value/DF
Deviance		40892.8122	
Scaled Deviance		40892.8122	
Pearson Chi-Square	13E3	5761.9322	0.4510
Scaled Pearson X2	13E3	5761.9322	0.4510
Log Likelihood		-20446.4061	
Full Log Likelihood		-20446.4061	
AIC (smaller is better)		40930.8122	
AICC (smaller is better)		40930.8717	
BIC (smaller is better)		41072.4916	

Criteria For Assessing Goodness Of Fit			
Criterion	DF	Value	Value/DF
Deviance		40817.0909	
Scaled Deviance		40817.0909	
Pearson Chi-Square	13E3	5924.4655	0.4638
Scaled Pearson X2	13E3	5924.4655	0.4638
Log Likelihood		11188.6258	
Full Log Likelihood		-20408.5455	
AIC (smaller is better)		40857.0909	
AICC (smaller is better)		40857.1567	
BIC (smaller is better)		41006.2271	

If you recall from the previous section of this write up, the coefficient estimates were very similar for both models but we eliminated one of the predictors for the ZINB model. Since the ZIP model has a smaller AIC, based on that metric it is considered the better model. Further, since including this additional predictor variable seemed to help our model performance, I ultimately chose the Zero Inflated Poisson Regression model as my best model.

CONCLUSION:

In this assignment we used different modeling techniques to build 6 different models to predict the number of cases of wine sold. As discussed in our exploratory data analysis, we were dealing with counting data. Thus, the suggest approach would be to use some sort of Poisson or Binomial model. Since the TARGET variable was "zero inflated" we would assume that the best model would be one of the zero inflated approaches i.e. Zero Inflated Binomial Regression, Zero Inflated Poisson Regression, or Hurdle Logistic Poisson/Negative Binomial Regression.

Overall, the models created using the "zero inflated" approaches performed best. However, the Poisson, Negative Binomial, and even the linear regression model (which is not recommended for counting data) were not too far off. When comparing the predictions for all of the models, we found that all of the predictions were pretty close. This implies that the regression techniques are so robust that, even though the underlying assumptions were violated, we still got good and meaningful results.

BINGO BONUS (ADDITIONAL WORK)

The decision trees for imputing missing values are in the "DATA PREPARATION" section.

LOGISTIC/NEGATIVE BINOMIAL Model created in the "BUILD MODELS" section.

```
> summary(fit)

Call:
glm(formula = TARGET ~ AcidIndex + CAP_IMP_Alcohol + CAP_IMP_FreeSulfurDioxide +
    CAP_IMP_Sulphates + CAP_IMP_TotalSulfurDioxide + CAP_IMP_pH +
    IMP_STARS + LabelAppeal + VolatileAcidity, family = poisson(),
    data = wine_cap)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.9657  -0.7166   0.0708   0.5784   3.2269

Coefficients:
                            Estimate  Std. Error z value    Pr(>|z|)
(Intercept)                1.24822178  0.04907200  25.437     < 2e-16 ***
AcidIndex                 -0.08743238  0.00448973 -19.474     < 2e-16 ***
CAP_IMP_Alcohol            0.00237932  0.00146020   1.629    0.103218
CAP_IMP_FreeSulfurDioxide  0.00013084  0.00003631   3.603    0.000314 ***
CAP_IMP_Sulphates         -0.01329781  0.00595889  -2.232    0.025642 *
CAP_IMP_TotalSulfurDioxide 0.00008599  0.00002358   3.647    0.000265 ***
CAP_IMP_pH                -0.01701280  0.00787809  -2.160    0.030811 *
IMP_STARS                  0.31166486  0.00452859  68.822     < 2e-16 ***
LabelAppeal                0.13303102  0.00606213  21.945     < 2e-16 ***
VolatileAcidity           -0.03369335  0.00651598  -5.171 0.000000233 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 22861  on 12794  degrees of freedom
Residual deviance: 14738  on 12785  degrees of freedom
AIC: 46700

Number of Fisher Scoring iterations: 5
```

Above, is the output for the Poisson model that I recreated in R.

Below is the Model Validation Macro Code I tried to use. I got the errors down from 27 errors to 6 errors, but unfortunately, given time constraints, I was unable to successfully get it to run. I really would’ve liked to compare the model performance based on this output and plan to use this method in the future.

```
%macro FIND_ERROR( DATAFILE, P, MEANVAL);

%let ERRFILE = ERRFILE;
%let MEANFILE = MEANFILE;

data &ERRFILE.;
set &SCOREFILE.;
        ERROR_MEAN              = abs( TARGET - 3.0290739 ) **&P.;
        ERROR_POI       = abs( TARGET - P_GENMOD_POI ) **&P.;
        ERROR_NB        = abs( TARGET - P_GENMOD_NB ) **&P.;
        ERROR_ZIP       = abs( TARGET - P_GENMOD_ZIP ) **&P.;
        ERROR_ZINB      = abs( TARGET - P_GENMOD_ZINB ) **&P.;
        ERROR_HURDLE    = abs( TARGET - P_HURDLE ) **&P.;
```

```
run;


proc means data=&ERRFILE. noprint;
output out=&MEANFILE.;
        mean(ERROR_MEAN)        =       ERROR_MEAN
        mean(ERROR_POI)   =       ERROR_POI
        mean(ERROR_NB)    =       ERROR_NB
        mean(ERROR_ZIP)   =       ERROR_ZIP
        mean(ERROR_ZINB) =       ERROR_ZINB
        mean(ERROR_HURDLE)      =       ERROR_HURDLE
        ;
run;

data &MEANFILE.;
length P 8.;
set &MEANFILE.;
        P               = &P.;
        ERROR_MEAN          = ERROR_MEAN       ** (1.0/&P.);
        ERROR_POI    = ERROR_POI ** (1.0/&P.);
        ERROR_NB            = ERROR_NB  ** (1.0/&P.);
        ERROR_ZIP    = ERROR_ZIP ** (1.0/&P.);
        ERROR_ZINB  = ERROR_ZINB        ** (1.0/&P.);
        ERROR_HURDLE      = ERROR_HURDLE    ** (1.0/&P.);
        drop _TYPE_;
run;

proc print data=&MEANFILE.;
run;

%mend FIND_ERROR;



%FIND_ERROR( &SCOREFILE., 1, 3.0290739 );
%FIND_ERROR( &SCOREFILE., 1.5, 3.0290739 );
%FIND_ERROR( &SCOREFILE., 2, 3.0290739 );
```