Unit 3 Assignment: “Wine Sales Problem”

MSDS 411 Section 56, spring 2019

Prabhat Thakur 05/30/2019

Introduction

The purpose of this project is to build a predictive model to predict the number of wine cases ordered based upon the wine characteristics. The data set contains information on approximately 12,000 commercially available wines. The variables are mostly related to the chemical properties of the wine being sold. The target variable is the number of sample cases of wine that were 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. The more sample cases purchased, the more likely is a wine to be sold at a high end restaurant.

For arriving to a best predictive model, several generalized regression model like Multiple Regression, Poisson, Zero-inflated Poisson, Negative Binomial, Zero-inflated Negative Binomial, Decision Tree and logistic/poisson hurdle models are fit to the data and evaluated. The final model is selected based on Mean Squared Error on test and validation data set.

Bonus Problem

I have attempted following bonus problems:

1. (20 points): Develop a LOGISTIC / POISSON model. This is my champion model as well (Model 7).
2. (20 Points) Use decision tree software for variable selection and missing value imputation. A separate hurdle models is developed using LOGISTIC / POISSON (Model 9) using this method.
3. (20 Points) Build a decision tree model to predict the number of cases sold (Model 8).

Data Exploration

The data for this assignment contains information of chemical properties of wine being sold. There are 12797 observations in the training dataset and 3335 in the test dataset. The Training data set has 16 columns which includes 1 observation identification number INDEX and one response variables

TRAGET (count of number of wine cases sold), and 14 potential predictor variables for building the model. The test dataset contains same columns but TRAGET values are blank. The trained models will be used to predict TRAGET value for the test dataset observations.

Main objective of this assignment is to create generalized regression model to predict count of wine cases sold. The dataset consists continuous and some ordinal variables. Observing the target variable can provide input as to which techniques will fit the data best. Figure 1 shows the distribution of the target in the training data. There is a large mass of observations about 21.3% at zero. One reason for this is that if someone decides to buy a particular wine, they are likely to buy more than one case of it.

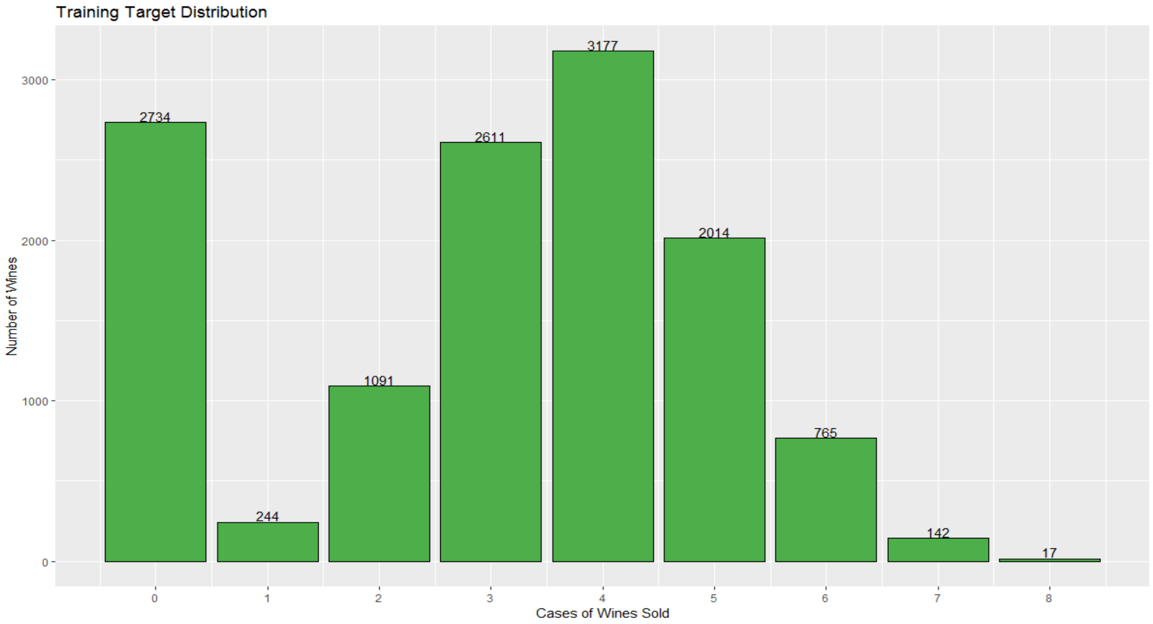

Figure 1 – Distribution of cases of wines sold in the training data

This data structure indicates that zero inflated or hurdle models may improve the fit. For wines that are purchased, the number of cases in excess of one has a mean greater than the variance. This is an important piece of information if a Poisson hurdle model is considered.

To explore predictor variables from training and test data set together, both datasets were merged for data exploration purpose. Below are the boxplot and histogram of continuous predictor variables. Figure 2, shows the distribution of the continuous predictor variables. Based on the box plots, it is likely that the testing data likely comes from the same population as the training data. Figure 3 shows the same data in the form of histograms. From observing the data in this format, there appears to be unrealistic negative values for some of the variables which should be investigated and imputed accordingly. Additionally, several predictors have frequent missing values that suggest that an indicator variable for those values could prove predictive.

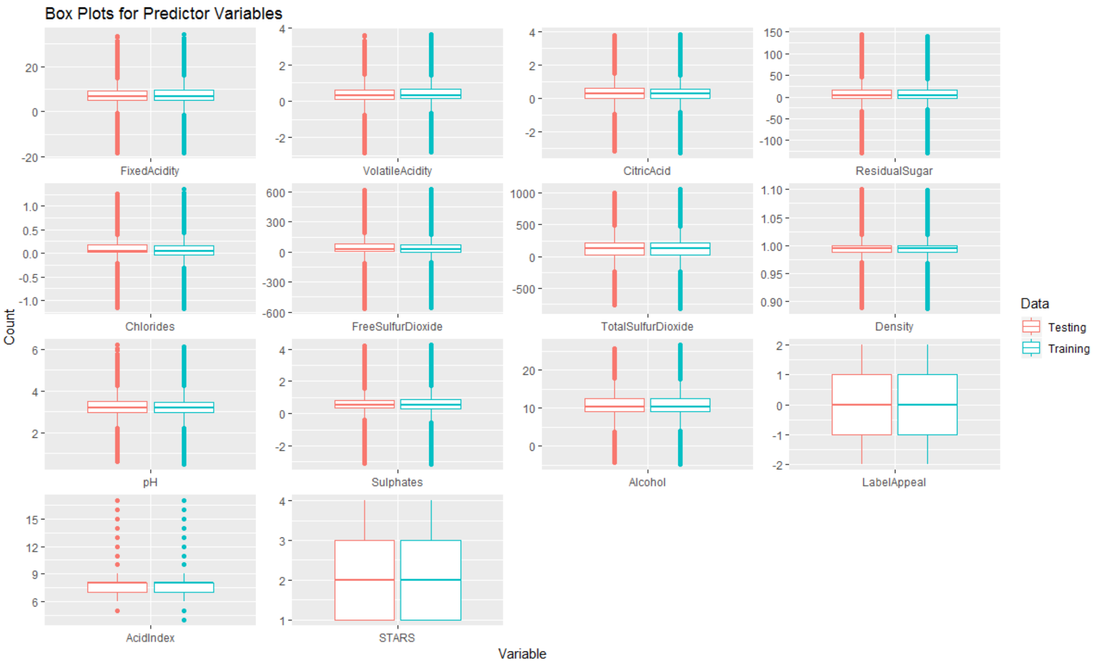

Figure 2 – Continuous Predictor Box Plots

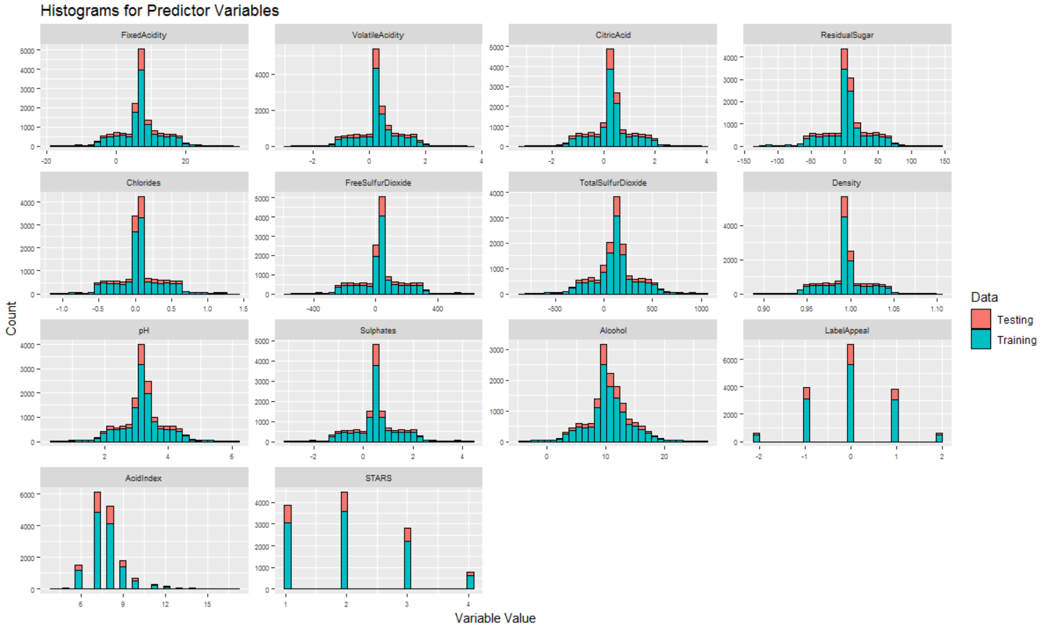

Figure 3 – Continuous Predictor Histograms

The outliers make it difficult to observe the univariate relationships between predictors and the target. An alternative way to do this comparison is to plot the target instead on the x-axis and show the average predictor value for each value of the target on the y-axis. This type of plot is shown in Figure 4. Due to the difference in the plot type, the only interpretation that can be made is that the predictor variables do differ on average by the number of cases sold.

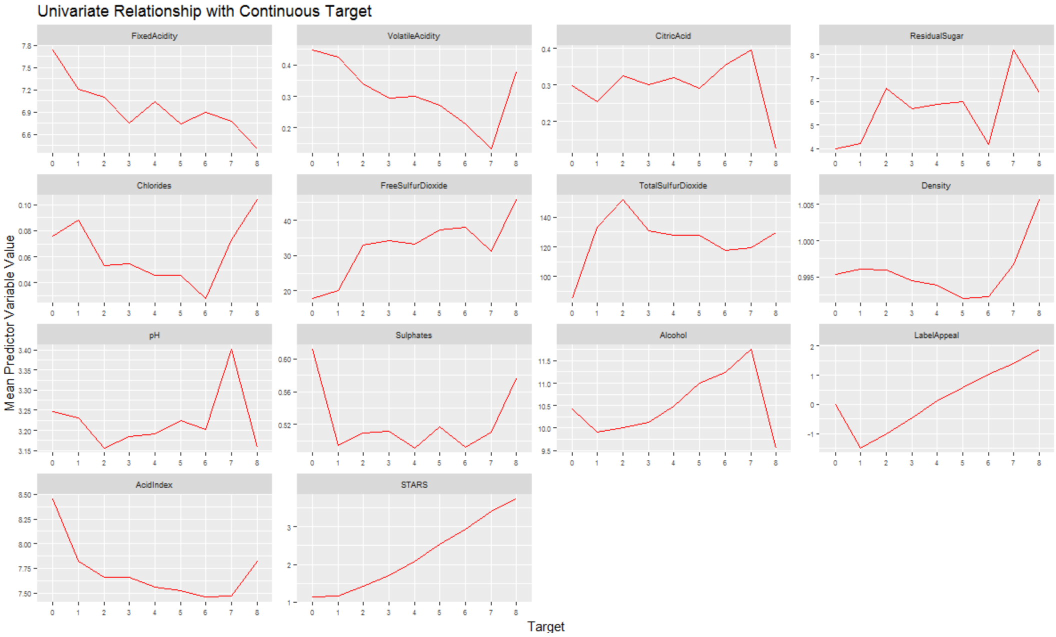

Figure 4 – Univariate Relationship (Axis Switched)

The correlation matrix heat map of training dataset predictor correlation is shown in Figure 5. Multicollinearity appears not to be very likely as measurements lack strong correlations with each other. However, they are also only marginally correlated with the number of cases sold.

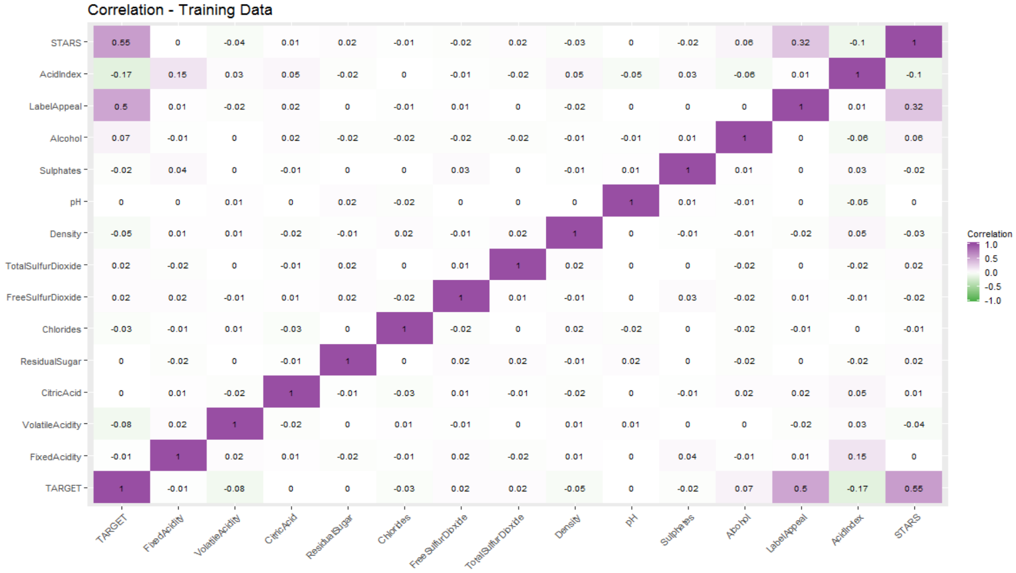

Figure 5 – Training Data Correlation

Next, Figure 6 shows the portion of values missing in the training and test data for variables with at least one missing value. Several of the chemical measurements such as pH and alcohol levels are missing in the data. The presence of missing values does not appear consistent across all measurements as a wine with a missing measurement is likely to have other completed measurements. Approximately a third of the wines are missing a rating as well. This may indicate that not all wines are rated. These missing values are imputed in the Data Preparation section.

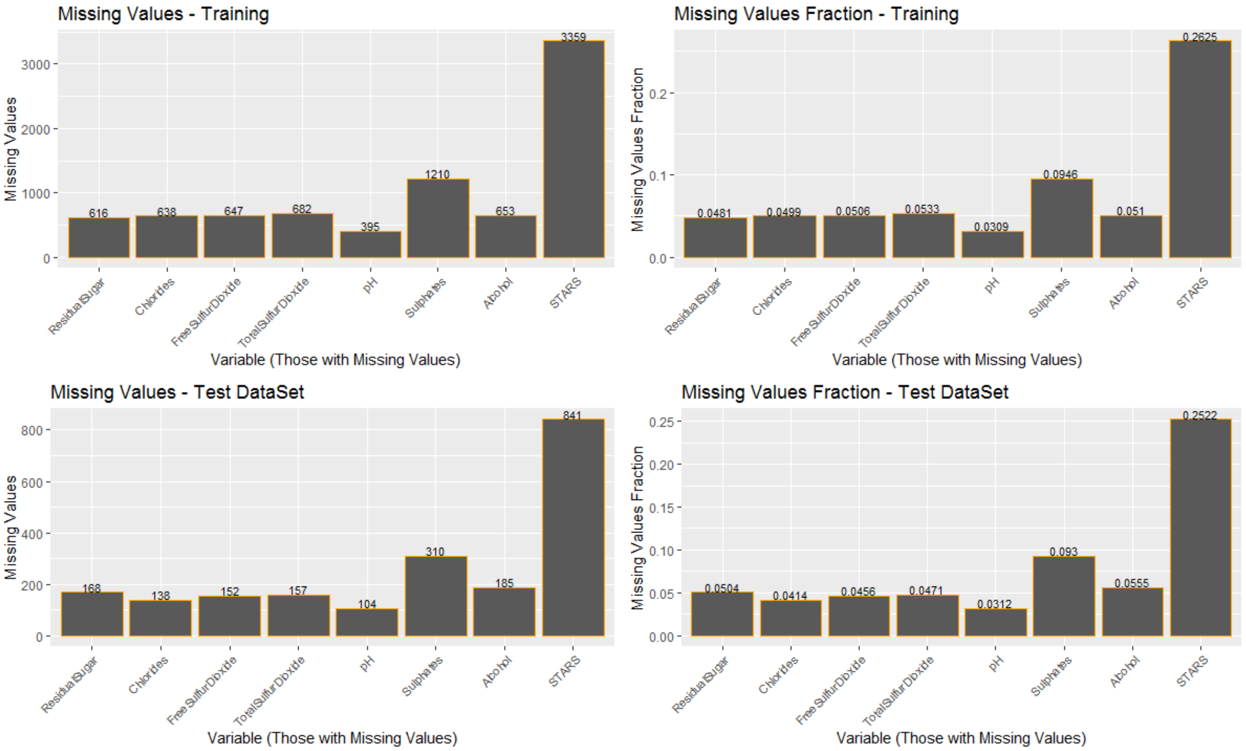

Figure 6 – Missing Value in Training and Test dataset

Data Preparation

To account for missing measurements, missing values are imputed with the mean of the training data. Indicator variables are created based on the presence of a missing value.

The outliers in the data result in the need to cap and floor the predictor variables. The top and bottom 2.5% of the training data define the capped and floored transformed variables. An outlier wine may intuitively be less likely to sell as much, so indicators are created for cases in which a flooring or capping occurs. Figure 7, shows the univariate relationships between predictors and the target after imputing the missing values and outliers capped and floored.

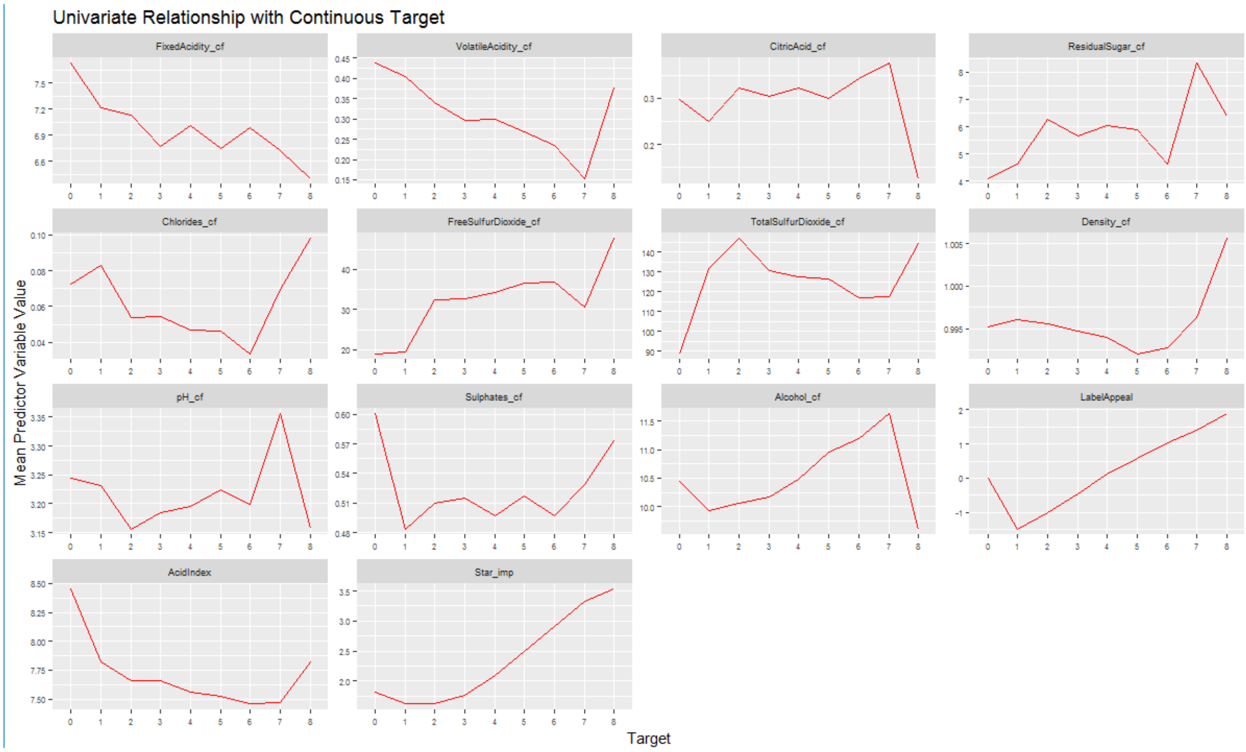

Figure 7 – Univariate Relationship (Axis Switched)

Build Models

The models built on the data are assessed based on their performance on a 30% hold out partition of the training data.

Model 1: The first model is a multiple linear regression baseline including only the variables most correlated with the target: star rating, label appeal, and acid index. All of the variables in the regression are significant, match intuitive relationships, and variance inflation factors (VIF) are all lower than two. The mean squared error (MSE) on the validation set is close to that of the training at 2.6862.

Model 2: This model uses all chemical measurements and missing value indicator variables by applying backwards selection to a multiple linear regression. VIFs continue to stay small which is expected based on the correlation plot. The variables which have intuitive relationships like Star ratings and labels persist. The measurement values and related indicators make it into the model. The training MSE shows very good improvement over model 1 so does the validation MSE's at 1.6801.

Model 3: This model applies backwards selection to all the predictor variables using a Poisson regression instead. Again, the intuitive relationships hold and multicollinearity is not a concern. The validation MSE is little lower than model 2 at 1.6915.

Model 4: This model uses zero-inflated Poisson regression. Due to issues getting the model to converge, manual variable selection is used starting with the variables found in Model 3. The validation MSE is best so far at 1.651. This model could be used as champion model.

Model 5: Tow models are built, first using the same terms as Model 3 and second using stepwise backwards selection to all the predictor variables. Instead of Poisson regression, negative binomial regression is used. The coefficients and fit statistics are identical in this case which is an interesting feature of these kinds of regression. The validation MSE identical for both the models at 1.6915. The coefficients and fit statistics are identical to model 3 which is an interesting feature of these kinds of regression.

Model 6: This model uses Zero-inflated Negative Binomial regression using Model 4 terms due to fit issues. The coefficients and the validation MSE is same as model 4 at 1.6519. This model can also be used as champion model.

Models 7 (Bonus): This model uses hurdle approach where first part is a logistic regression predicting whether or not an individual wine is purchased and second part is a Poisson regression predicting the number of cases in excess of one purchased. Both use backwards stepwise selection.

An advantage of this analysis is that it allows for the interpretation of factors that create sales and factors that lead to larger sales separately. In this case the direction of the variables were similar in almost all cases, but the Poisson regression incorporated fewer variables in the backwards selection. It appears that the information available is better suited to predict a sale than predicting how much is sold. The second part, passion model included the same variables as Model 1 with the addition of capped volatile acidity and the capped TotalSulfurDioxide as variables that negatively impact sales when they increase in value and alcohol level as a variable that increases sales when it increases.

One of the limitations of this approach is that the data in excess of one is actually under-dispersed. This limits the appropriateness of the application of Poisson regression. However, the model may still produce good rankings of wine that will sell.

As expected, the VIFs are low for these models. The validation MSE for the combination is also an improvement over the other models at 1.5485. Out of all the models we have evaluated so far, this is the best model for deployment.

Model 8 (Bonus): As another point of comparison, a decision tree with default parameters from the "rpart" package is fit to the data. The resulting MSE is 1.7068 which is impressive given the simplicity of the algorithm. One element to consider in the future is applying ensemble tree approaches such as Random Forest models.

Model 9 (Bonus): This model is uses the same hurdle approach as model 7 but instead of using mean value to impute missing values, mice package is used which uses decision tree to impute the missing values. Also for both logistic and passion regression models, initial predictors are selected using random forget model from cforest method of party package. Top 12 predictor variables are then passed to stepwise backward variable selection method. The resulting validation set MSE is 1.7473 which can be considered good model. Also calculated the distribution of Target (sale count) using test dataset, following is the result.

```
> summary(scores$P_TARGET)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.

0.05321 2.29188 3.14462 3.06391 3.87254 7.71926
```

Select Model

The different model forms restricted the variety of model metric comparisons to some extent. The focus for model selection was the MSE on the 30% of the data used for validation. Model 7, combination Logistic and Poisson regression model results in a significant improvement in MSE over the other models considered. Additionally, this approach did not have any convergence issues and the hurdle feature adds to the interpretability of the model at large. As a result, this combination is the selected approach.

Below table summarized all the models evaluated and respective MES on training and validation hold out data set.

Models	Description	Model	Training Set MSE	Validation Set MSE	AIC
Model1	Multiple Linear Regression	lm(formula = TARGET ~ LabelAppeal + Star_imp + AcidIndex, data = train_trn)	2.7177	2.6861	34387.9687
Model2	Multiple Linear Regression with backwards selection from the baseline	lm(formula = TARGET ~ LabelAppeal + AcidIndex + VolatileAcidity_cf + CitricAcid_cf + ResidualSugar_mind + Chlorides_cf + FreeSulfurDioxide_cf + TotalSulfurDioxide_mind + TotalSulfurDioxide_cf + Alcohol_cf + Star_imp + Star_mind, data = train_trn)	1.73	1.68	30360.2108
Model3	Poisson regression with backwards selection from the baseline	glm(formula = TARGET ~ LabelAppeal + AcidIndex + VolatileAcidity_cf + Chlorides_cf + FreeSulfurDioxide_cf + TotalSulfurDioxide_cf + Alcohol_cf + Star_imp + Star_mind, family = poisson(link = "log"), data = train_trn)	1.7485	1.6915	32090.1343
Model4	Zero-inflated Poisson regression with some of the Model 3 terms	zeroinfl(formula = TARGET ~ LabelAppeal + AcidIndex + VolatileAcidity_cf + Chlorides_cf + FreeSulfurDioxide_cf + TotalSulfurDioxide_cf + Alcohol_cf + Star_imp + Star_mind | LabelAppeal + Alcohol_cf + Star_imp, data = train_trn, dist = "poisson", link = "log")	1.7312	1.6519	31321.3188
Model5	a. Negative Binomial regression with same terms as Model 3 b. Negative Binomial regression with backwards selection from the baseline	a. glm.nb(formula = model3$formula, data = train_trn, init.theta = 40726.50559, link = log) b. glm.nb(formula = TARGET ~ LabelAppeal + AcidIndex + VolatileAcidity_cf + Chlorides_cf + FreeSulfurDioxide_cf + TotalSulfurDioxide_cf + Alcohol_cf + Star_imp + Star_mind, data = train_trn, init.theta = 40726.43674, link = log)	1.7485 1.7485	1.6915 1.6915	32092.4164 32092.4164
Model6	Zero-inflated Negative Binomial regression using Model 4 terms	zeroinfl(formula = TARGET ~ LabelAppeal + AcidIndex + VolatileAcidity_cf + Chlorides_cf + FreeSulfurDioxide_cf + TotalSulfurDioxide_cf + Alcohol_cf + Star_imp + Star_mind | LabelAppeal + Alcohol_cf + Star_imp, data = train_trn, dist = "negbin", link = "log")	1.7312	1.6519	31323.3194
Model7	Hurdle approach first part logistic regression and second part Poisson regression Logistic regression with backwards selection from the baseline Poisson regression with backwards selection from the baseline	**model7_Logistic:** glm(formula = ifelse(TARGET > 0, 1, 0) ~ LabelAppeal + AcidIndex + VolatileAcidity_cf + Chlorides_cf + FreeSulfurDioxide_cf + TotalSulfurDioxide_mind + TotalSulfurDioxide_cf + pH_cf + Sulphates_cf + Alcohol_cf + Star_imp + Star_mind, family = binomial(link = "logit"), data = train_trn) **model7_Poisson:** glm(formula = (TARGET - 1) ~ LabelAppeal + AcidIndex + VolatileAcidity_cf + TotalSulfurDioxide_cf + Alcohol_cf + Star_imp + Star_mind, family = poisson(link = "log"), data = train_trn[train_trn$TARGET > 0,])	1.6425	1.5485	5386.5 21951
Model8	Decision tree with default options	rpart(formula = TARGET ~ ., data = train_trn)	1.786271	1.706806	

Model9	Hurdle approach first part logistic regression and second part Poisson regression Used decision tree for missing value imputation and variable selection Logistic regression with backwards selection from above selected variables Poisson regression with backwards selection from above selected variables	**model9_Logistic:** glm(formula = ifelse(TARGET > 0, 1, 0) ~ Star_mind + STARS + LabelAppeal + AcidIndex + VolatileAcidity_cf + Chlorides_cf + Alcohol_cf + TotalSulfurDioxide_cf + FreeSulfurDioxide_cf, family = binomial(link = "logit"), data = train_trn) **model9_Poisson:** glm(formula = (TARGET - 1) ~ Star_mind + STARS + LabelAppeal + AcidIndex + Alcohol_cf + Density_cf, family = poisson(link = "log"), data = train_trn[train_trn$TARGET > 0,])	1.729657	1.747398	5933.5 21895

Following is the generalized regression equation of final model which is used to generate scores (predict target flag) on test data set.

```
scores <- test_process1 %>%
 mutate(SCORE_ZERO1 = 2.7677782229 +
       LabelAppeal * -0.4608923867 +
       AcidIndex       *         -0.3828787047    +
       VolatileAcidity_cf          *         -0.1794974473    +
       Chlorides_cf     *         -0.2816060860    +
       FreeSulfurDioxide_cf        *         0.0006699291    +
       TotalSulfurDioxide_mind *         0.2196631550    +
       TotalSulfurDioxide_cf       *          0.0008866712    +
       pH_cf   *         -0.1947236759    +
       Sulphates_cf     *         -0.0987064933    +
       Alcohol_cf       *         -0.0269410701    +
       Star_imp         *         2.5116450460     +
       Star_mind        *         -4.4438135083    ,
     SCORE_ZERO = exp(SCORE_ZERO1) / (1 + exp(SCORE_ZERO1)),
     SCORE_NONZERO = exp(0.83367212809 +
                 LabelAppeal * 0.29246908139 +
                 AcidIndex       *         -0.02100540105  +
                 VolatileAcidity_cf        *         -0.01538854250  +
                 TotalSulfurDioxide_cf  *         -0.00005241572 +
                 Alcohol_cf       *         0.00987772346    +
                 Star_imp         *         0.12190218156    +
                 Star_mind        *         -0.20961823709   ) + 1,
     P_TARGET = SCORE_ZERO * SCORE_NONZERO) %>%
 select(INDEX, P_TARGET)
```

Conclusion

Predicting the number of cases sold for a particular wine involved predicting a zero-inflated target. To account for this, nine different models were fit to the training data that both ignored and explicitly considered the zero-inflated feature. The final selected model was actually a combination of a logistic regression and a Poisson regression. The two models together fit the data well and are extremely interpretable.