REGRESSION MODELS

Wine Demand Prediction

Rafal Decowski

CUNY | DATA MINING

Objective

The objective is to build poisson, negative binomial, and multiple linear regression models, using different variables to predict the number of sample cases of wine that will be purchased by distribution companies.

Contents

DATA EXPLORATION

Dataset

The dataset contains wine attributes such as chemical composition as well as external factors such as label attractiveness to customers and wine-expert ratings. It has 12795 cases across 15 predictor variables and one response variables - TARGET. The all of the predictor variables are of numerical type. Only 50% of all cases are complete which means the dataset may need additional cleansing to either fill out the gaps or drop rows with missing data. The response variable is a count indication of how many cases of wine were requested for sampling.

VARIABLE NAME	*DEFINITION THEORETICAL EFFECT*
TARGET	Number of Cases Purchased None
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
Sulphates	Sulfate content of wine
TotalSulfurDioxide	Total Sulfur Dioxide of Wine
VolatileAcidity	Volatile Acid content of wine
pH	pH of wine

Descriptive Statistics

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

The table below tells me that there are some missing values (NSa column.) The data also highlights a significant issue with the data collection process. This is conclusion is based on the fact that several variables reported negative values in the 'min' column. This applies to several features.

	NAs	*mean*	*sd*	*median*	*trimmed*	*mad*	*min*	*max*	*range*	*skew*	*kurtosis*	*se*
target	0	3.03	1.93	3	3.05	1.48	0	8	8	-0.33	-0.88	0.02
FixedAcidity	0	7.08	6.32	6.9	7.07	3.26	-18.1	34.4	52.5	-0.02	1.67	0.06
VolatileAcidity	0	0.32	0.78	0.28	0.32	0.43	-2.79	3.68	6.47	0.02	1.83	0.01
CitricAcid	0	0.31	0.86	0.31	0.31	0.42	-3.24	3.86	7.1	-0.05	1.84	0.01
ResidualSugar	616	5.42	33.75	3.9	5.58	15.72	-127.8	141.15	268.95	-0.05	1.88	0.31
Chlorides	638	0.05	0.32	0.05	0.05	0.13	-1.17	1.35	2.52	0.03	1.79	0
FreeSulfurDioxide	647	30.85	148.71	30	30.93	56.34	-555	623	1178	0.01	1.84	1.35
TotalSulfurDioxide	682	120.71	231.91	123	120.89	134.92	-823	1057	1880	-0.01	1.67	2.11
Density	0	0.99	0.03	0.99	0.99	0.01	0.89	1.1	0.21	-0.02	1.9	0
pH	395	3.21	0.68	3.2	3.21	0.39	0.48	6.13	5.65	0.04	1.65	0.01
Sulphates	1210	0.53	0.93	0.5	0.53	0.44	-3.13	4.24	7.37	0.01	1.75	0.01
Alcohol	653	10.49	3.73	10.4	10.5	2.37	-4.7	26.5	31.2	-0.03	1.54	0.03
LabelAppeal	0	-0.01	0.89	0	-0.01	1.48	-2	2	4	0.01	-0.26	0.01
AcidIndex	0	7.77	1.32	8	7.64	1.48	4	17	13	1.65	5.19	0.01
STARS	3359	2.04	0.9	2	1.97	1.48	1	4	3	0.45	-0.69	0.01

Missing Values

Taking a closer look at the missing values will help us determine the need and method of handling our dataset. STARS variable is missing over a quarter of the values. If this feature renders a significant correlation and statistical importance to our model we may need to either drop the cases without values or fill the NA's with the mean or another technique.

	Percentage Missing
STARS	0.26
Sulphates	0.09
ResidualSugar	0.05
Chlorides	0.05
FreeSulfurDioxide	0.05
TotalSulfurDioxide	0.05
Alcohol	0.05
pH	0.03
target	0
FixedAcidity	0
VolatileAcidity	0
CitricAcid	0
Density	0
LabelAppeal	0
AcidIndex	0

DATA PREPARATION

Fixing Negative values

In the previous section we noted an important error within our dataset. To fix it, we will replace all negative values with NA's (remove from dataset), with the exception for *LabelAppeal* variable which is a valid scale ranging from -2 to 2 on consumer rating of the label attractiveness.

Outlier Handling

Handling outliers may bring significant changes to the outcome of the regression models. This leads us to 'branching' our dataset into two – one with outliers observed in the original dataset and one with all outliers removed (replaces with NA's). The two branches are *wines_nn* and *wines_no_outliers*.

Missing Values Handling

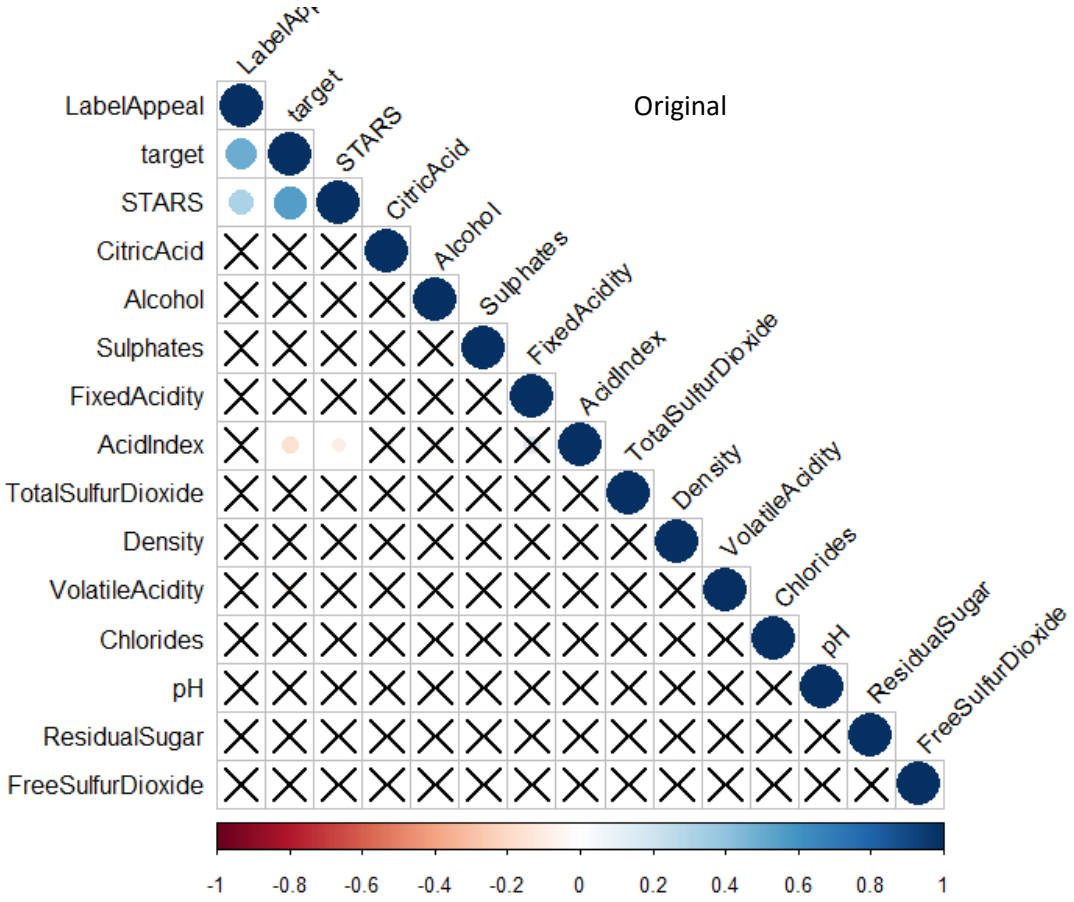
Since we have introduced two transformations that increased the total number of missing values, we must consider how to further increase the quality of the datasets (now two). We will apply a special algorithm called *Multivariate Imputation by Chained Equations*. To do this, we will utilize an R package called '*mice*' that makes it easy to populate the missing values with predictive mean matching for each variable.

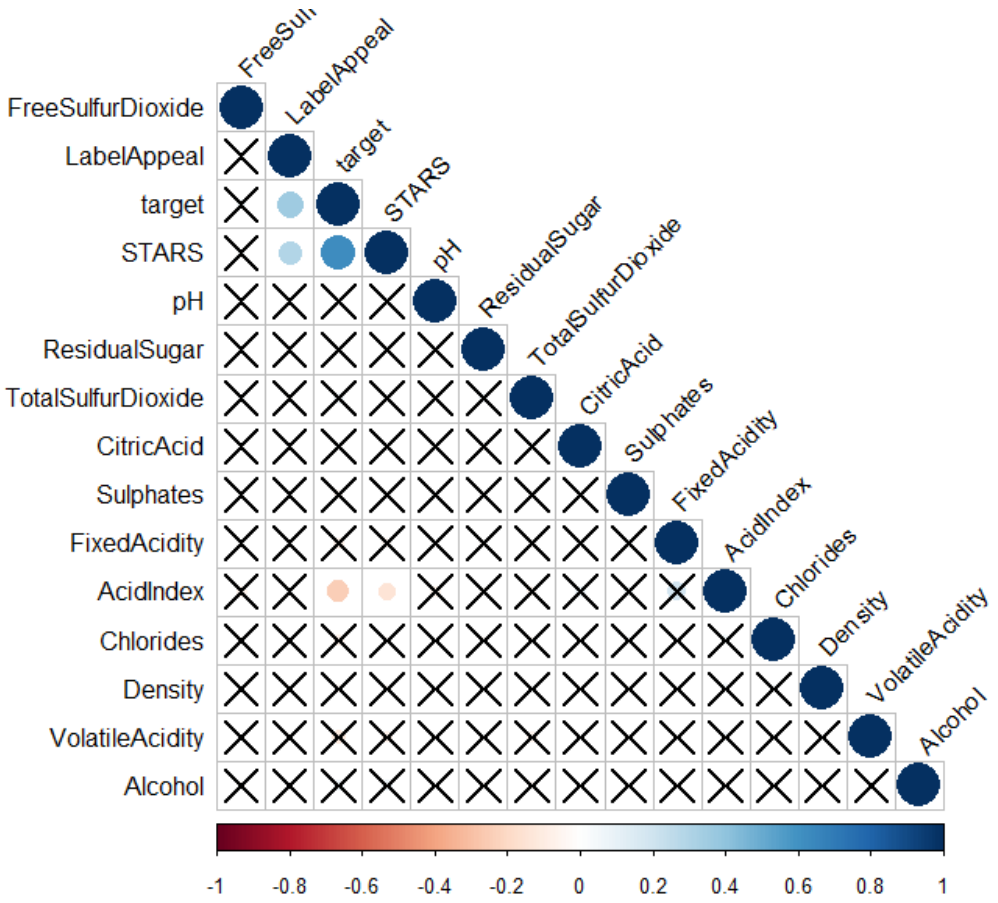
Correlation

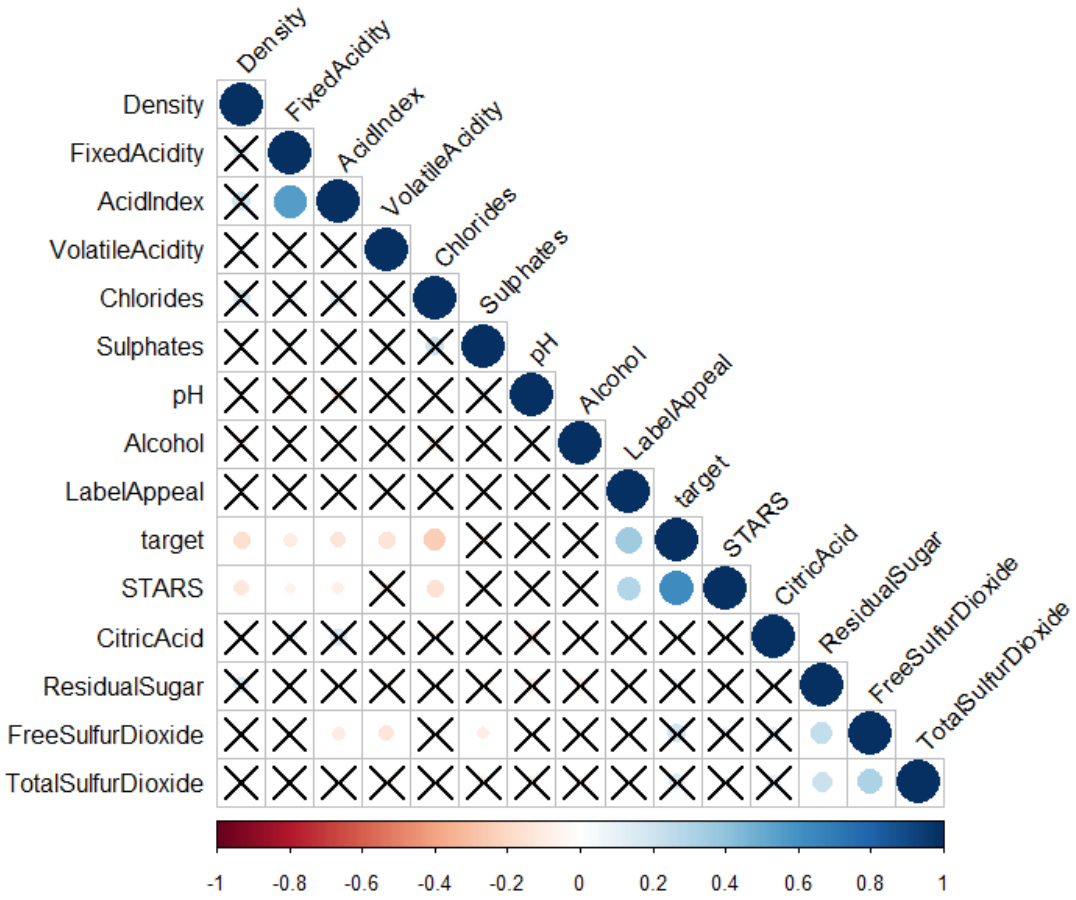
The correlation helps us highlight predictor variables that have a strong relationship with the target variable. It helps us narrow down the important ones and discard the ones that do not significantly affect the prediction results. The image below shows positive (blue) and negative(red) correlation between all variables. The crossed-out fields are rejected by a 95% confidence level. Domain knowledge makes this chart more significant as it helps form more advanced hypotheses and see how variables are related. The confidence level marked multiple fields as statistically insignificant which may help us reduce number of variables included in the models.

There are 3 different correlation graphs:

1. The original dataset
2. Dataset with fixed negative values and populated missing values
3. Dataset with fixed negative values, removed outliers and populated missing values




FreeSulfurDioxide
LabelAppeal
target
STARS
pH
ResidualSugar
TotalSulfurDioxide
CitricAcid
Sulphates
FixedAcidity
AcidIndex
Chlorides
Density
VolatileAcidity
Alcohol
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1


Density
FixedAcidity
AcidIndex
VolatileAcidity
Chlorides
Sulphates
pH
Alcohol
LabelAppeal
target
STARS
CitricAcid
ResidualSugar
FreeSulfurDioxide
TotalSulfurDioxide
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1

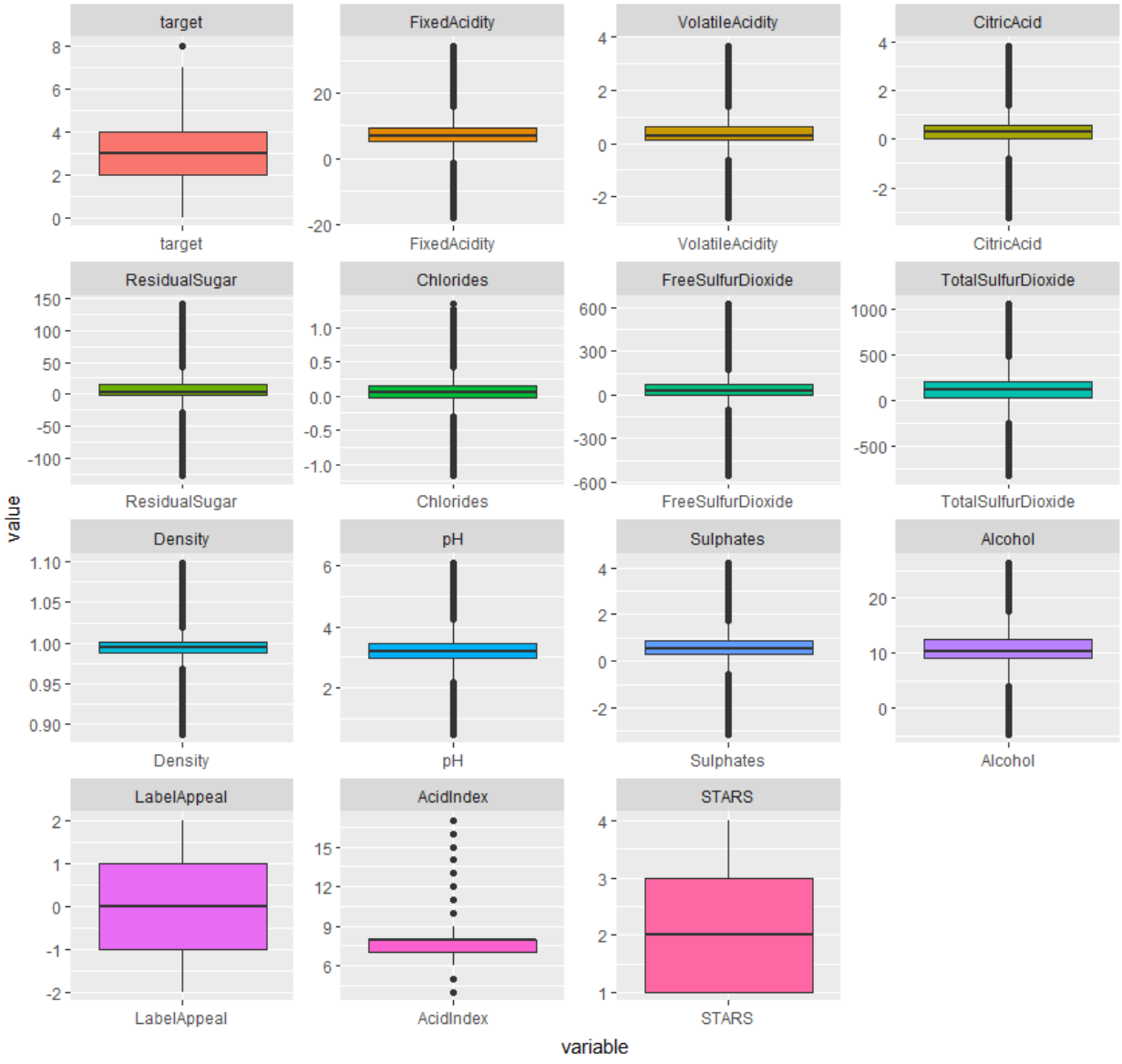
We can observe an interesting trend from the plots and it is even more easily noticeable from the table below. For every applied transformation, two major variable continue to stand out – STARS and LabelAppeal. Interestingly, these variables are **opinion-based** indicators whereas the rest are chemical measurements.

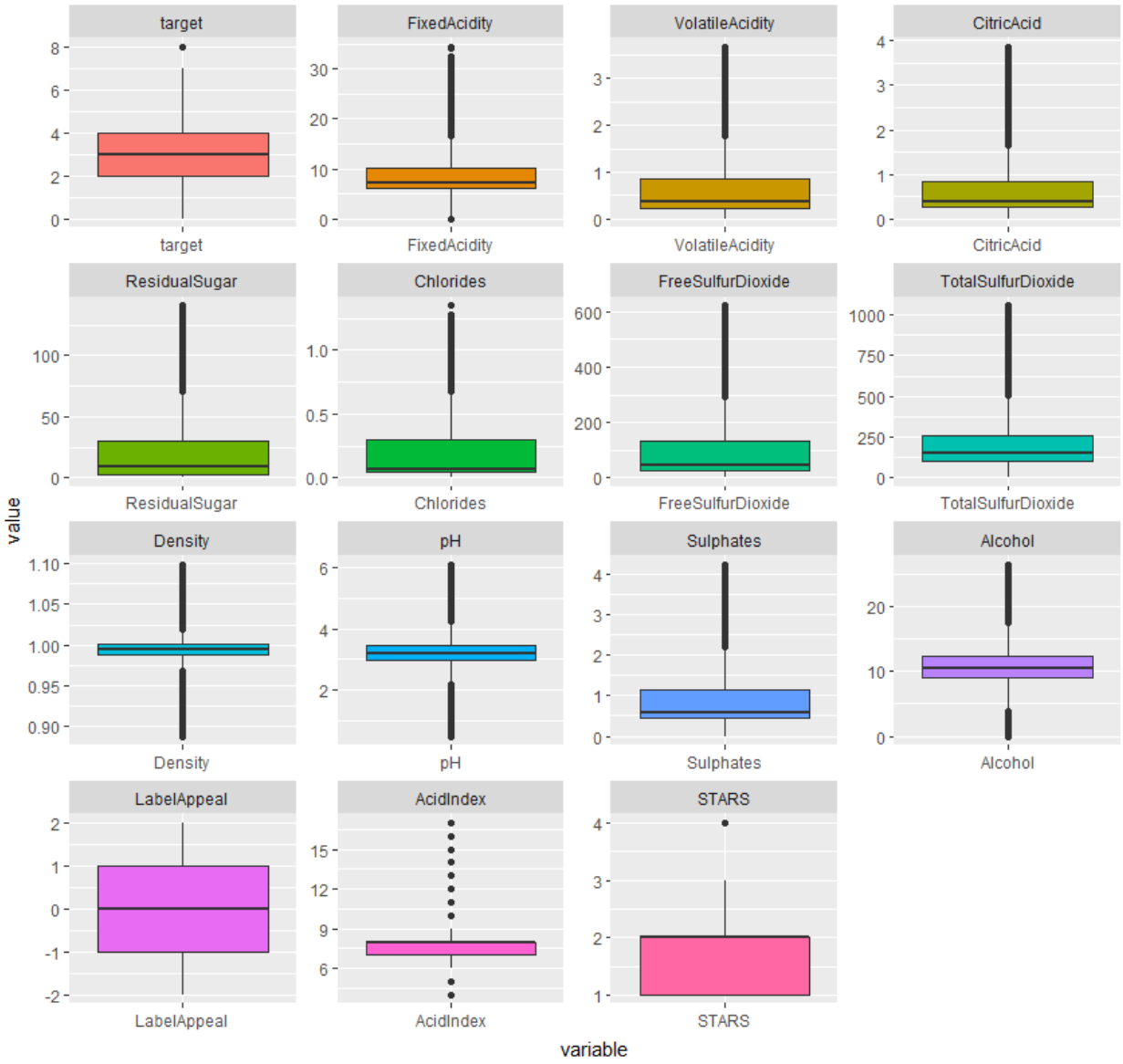
These two categories of variables are a great example for at least two different types of models.

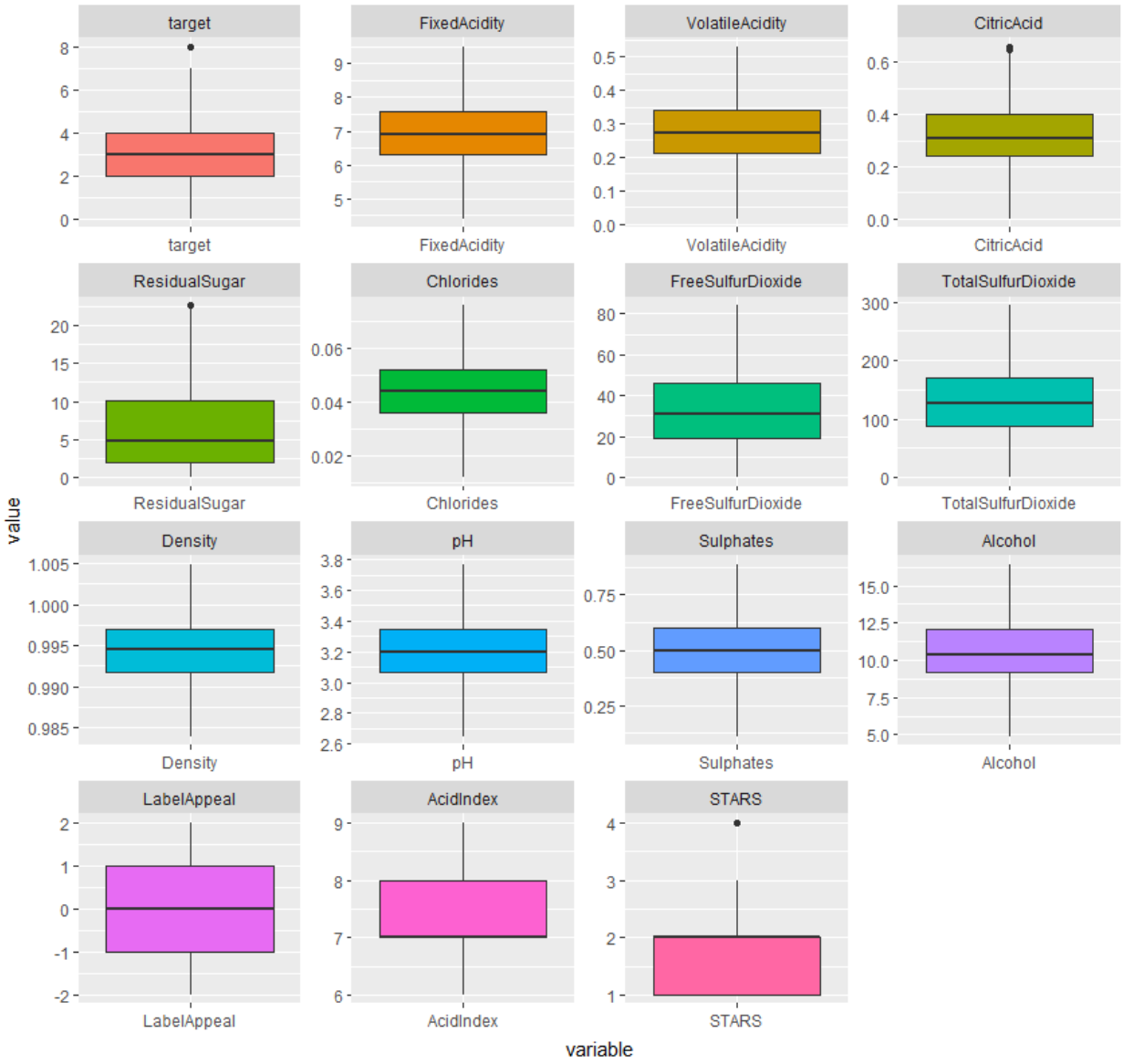
Variable	Original	Fixed Negative & Missing	Fixed Negative & Missing & Outliers
STARS	0.55	0.63	0.62
LabelAppeal	0.50	0.36	0.36
Alcohol	0.07	0.07	0.05
FreeSulfurDioxide	0.02	0.05	0.19
TotalSulfurDioxide	0.02	0.06	0.13
ResidualSugar	0.00	0.01	0.08
CitricAcid	0.00	0.02	0.07
pH	0.00	-0.01	-0.04
FixedAcidity	-0.01	-0.06	-0.11
Sulphates	-0.02	-0.04	-0.08
Chlorides	-0.03	-0.06	-0.24
Density	-0.05	-0.04	-0.16
VolatileAcidity	-0.08	-0.11	-0.15
AcidIndex	-0.17	-0.25	-0.14

Boxplots

The boxplots below help us bring the descriptive statistics from the previous section into neat visuals. We can easily determine ranges, medians and outliers. The transformations for handling negative values as well as outliers are now visualized.




target
FixedAcidity
VolatileAcidity
CitricAcid
ResidualSugar
Chlorides
FreeSulfurDioxide
TotalSulfurDioxide
Density
pH
Sulphates
Alcohol
LabelAppeal
AcidIndex
STARS
value
variable


target
FixedAcidity
VolatileAcidity
CitricAcid
ResidualSugar
Chlorides
FreeSulfurDioxide
TotalSulfurDioxide
Density
pH
Sulphates
Alcohol
LabelAppeal
AcidIndex
STARS
value
variable

Model Building

Poisson Model 1

Poisson model 1 is composed of all variables prior to any transformations. It highlights 5 statistically significant variables within the original dataset – *VolatileAcidity, Alcohol, LabelAppeal, AcidIndex* and *STARS.* 6421 degrees of freedom indicates it discarded almost half of the dataset and captured only the complete cases. This is the first attempt and I do not expect it to be the best performing model.

```
Call:
glm(formula = target ~ ., family = "poisson", data = wines)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2158  -0.2734   0.0616   0.3732   1.6830

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)         1.593e+00  2.506e-01   6.359 2.03e-10 ***
FixedAcidity        3.293e-04  1.053e-03   0.313  0.75447
VolatileAcidity    -2.560e-02  8.353e-03  -3.065  0.00218 **
CitricAcid         -7.259e-04  7.575e-03  -0.096  0.92365
ResidualSugar      -6.141e-05  1.941e-04  -0.316  0.75165
Chlorides          -3.007e-02  2.056e-02  -1.463  0.14346
FreeSulfurDioxide   6.734e-05  4.404e-05   1.529  0.12620
TotalSulfurDioxide  2.081e-05  2.855e-05   0.729  0.46618
Density            -3.725e-01  2.462e-01  -1.513  0.13026
pH                 -4.661e-03  9.598e-03  -0.486  0.62722
Sulphates          -5.164e-03  7.051e-03  -0.732  0.46398
Alcohol             3.948e-03  1.771e-03   2.229  0.02579 *
LabelAppeal         1.771e-01  7.954e-03  22.271  < 2e-16 ***
AcidIndex          -4.870e-02  5.903e-03  -8.251  < 2e-16 ***
STARS               1.871e-01  7.487e-03  24.993  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 5844.1  on 6435  degrees of freedom
Residual deviance: 4009.1  on 6421  degrees of freedom
  (6359 observations deleted due to missingness)
AIC: 23172

Number of Fisher Scoring iterations: 5
```

Poisson Model 2

Poisson model 2 is composed of all variables after fixing the negative values and populating missing values. It highlights many more variables as statistically significant when compared to the previous model. It also includes all 12k+ cases as there are no NA's. However, the residual deviance has dramatically increased. This may indicate decreased overall performance.

```
Call:
glm(formula = target ~ ., family = "poisson", data = wines_nn)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.9205  -0.6818   0.1209   0.6313   2.7156

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)          1.519e+00  1.963e-01   7.741 9.90e-15 ***
FixedAcidity        -6.788e-04  1.078e-03  -0.630  0.52900
VolatileAcidity     -7.272e-02  9.525e-03  -7.635 2.27e-14 ***
CitricAcid           2.277e-02  8.284e-03   2.749  0.00597 **
ResidualSugar        1.354e-04  2.119e-04   0.639  0.52277
Chlorides           -1.142e-01  2.304e-02  -4.955 7.24e-07 ***
FreeSulfurDioxide    1.949e-04  4.740e-05   4.112 3.92e-05 ***
TotalSulfurDioxide   1.401e-04  2.989e-05   4.688 2.76e-06 ***
Density             -3.345e-01  1.923e-01  -1.740  0.08188 .
pH                  -1.665e-02  7.521e-03  -2.213  0.02687 *
Sulphates           -2.383e-02  7.730e-03  -3.082  0.00205 **
Alcohol              4.549e-03  1.437e-03   3.166  0.00155 **
LabelAppeal          1.432e-01  6.096e-03  23.492  < 2e-16 ***
AcidIndex           -9.721e-02  4.528e-03 -21.470  < 2e-16 ***
STARS                3.357e-01  5.609e-03  59.848  < 2e-16 ***
---
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: 15940  on 12780  degrees of freedom
AIC: 47912

Number of Fisher Scoring iterations: 5
```

Poisson Model 3

Poisson model 3 is composed of all variables after fixing the negative values, removing the outliers and populating missing values. It strengthens statistical significance of the variables already brought up to surface with model 2. It also includes all 12k+ cases as there are no NA's. However, the residual deviance is similar model 2. This model's performance may be slightly better than model 2 but not as well as model 1.

```
Call:
glm(formula = target ~ ., family = "poisson", data = wines_no_outliers)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.1734  -0.6801   0.1223   0.6325   2.1750

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)          1.371e+01  1.407e+00   9.744  < 2e-16 ***
FixedAcidity        -1.569e-02  5.974e-03  -2.627  0.00862 **
VolatileAcidity     -4.755e-01  5.005e-02  -9.502  < 2e-16 ***
CitricAcid           1.570e-01  3.799e-02   4.134 3.57e-05 ***
ResidualSugar        4.451e-03  9.704e-04   4.587 4.51e-06 ***
Chlorides           -7.416e+00  4.389e-01 -16.897  < 2e-16 ***
FreeSulfurDioxide    3.455e-03  2.936e-04  11.767  < 2e-16 ***
TotalSulfurDioxide   5.149e-04  8.501e-05   6.057 1.39e-09 ***
Density             -1.270e+01  1.422e+00  -8.933  < 2e-16 ***
pH                  -3.783e-02  2.290e-02  -1.652  0.09846 .
Sulphates           -2.421e-02  3.438e-02  -0.704  0.48138
Alcohol              3.173e-03  2.090e-03   1.518  0.12895
LabelAppeal          1.477e-01  6.107e-03  24.180  < 2e-16 ***
AcidIndex           -3.120e-02  7.575e-03  -4.119 3.80e-05 ***
STARS                3.157e-01  5.720e-03  55.191  < 2e-16 ***
---
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: 15693  on 12780  degrees of freedom
AIC: 47665

Number of Fisher Scoring iterations: 5
```

Poisson Model 4

Poisson model 4 implements the learnings from the previous models as well as hand-picked variable based on the observed correlation. Additionally, it was trained using the original dataset (no transformations) which makes it a good candidate for an easily repeatable training and prediction. This model stresses the importance of wine's appearance, experts' opinion as well as easily accessible and commonly understood alcohol content unit (also included on the label.) Even though the residual deviance is higher than the first model, its increased degrees of freedom and the fact that it only uses 3 variables makes this model one of the best candidates for final selection.

```
Call:
glm(formula = target ~ STARS + LabelAppeal + Alcohol, family = "poisson",
    data = wines)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2417  -0.2625   0.0500   0.3695   1.6198

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) 0.809342   0.020860  38.798  < 2e-16 ***
STARS       0.191450   0.006273  30.520  < 2e-16 ***
LabelAppeal 0.178419   0.006702  26.623  < 2e-16 ***
Alcohol     0.005471   0.001484   3.686 0.000227 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 8176.2  on 8962  degrees of freedom
Residual deviance: 5680.6  on 8959  degrees of freedom
  (3832 observations deleted due to missingness)
AIC: 32357

Number of Fisher Scoring iterations: 4
```

Negative Binomial Model 1

After studying previous models, we can conclude that our dataset does not need extensive transformations. This model was trained using the dataset with fixed negative values and populated NA's using the MICE algorithm. As expected, the pattern of highly important variables, *LabelAppeal* and *STARS* is visible here as well.

```
Call:
glm.nb(formula = target ~ ., data = wines, init.theta = 140198.4536,
    link = log)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2157  -0.2733   0.0616   0.3732   1.6830

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)         1.593e+00  2.506e-01   6.359 2.03e-10 ***
FixedAcidity        3.293e-04  1.053e-03   0.313  0.75446
VolatileAcidity    -2.560e-02  8.353e-03  -3.065  0.00218 **
CitricAcid         -7.259e-04  7.575e-03  -0.096  0.92365
ResidualSugar      -6.141e-05  1.941e-04  -0.316  0.75166
Chlorides          -3.007e-02  2.056e-02  -1.463  0.14347
FreeSulfurDioxide   6.734e-05  4.404e-05   1.529  0.12621
TotalSulfurDioxide  2.081e-05  2.855e-05   0.729  0.46618
Density            -3.725e-01  2.462e-01  -1.513  0.13026
pH                 -4.661e-03  9.598e-03  -0.486  0.62722
Sulphates          -5.164e-03  7.052e-03  -0.732  0.46398
Alcohol             3.948e-03  1.771e-03   2.229  0.02579 *
LabelAppeal         1.771e-01  7.954e-03  22.271  < 2e-16 ***
AcidIndex          -4.870e-02  5.903e-03  -8.251  < 2e-16 ***
STARS               1.871e-01  7.487e-03  24.992  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for Negative Binomial(140198.5) family taken to be 1)

    Null deviance: 5843.9  on 6435  degrees of freedom
Residual deviance: 4009.1  on 6421  degrees of freedom
  (6359 observations deleted due to missingness)
AIC: 23174

Number of Fisher Scoring iterations: 1


              Theta:  140198
          Std. Err.:  234985
Warning while fitting theta: iteration limit reached

 2 x log-likelihood:  -23141.85
```

Negative Binomial Model 2

Negative binomial model 2 is what we can consider as 'cleaned up' version of the previous model. It uses the same dataset (same transformations) but only 3 variables. Based on the residual deviance we can compare it to poisson model 4 and we can expect similar overall performance.

```
Call:
glm.nb(formula = target ~ STARS + LabelAppeal + Alcohol, data = wines,
    init.theta = 142158.6511, link = log)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2417  -0.2625   0.0500   0.3695   1.6198

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) 0.809341   0.020861  38.797  < 2e-16 ***
STARS       0.191451   0.006273  30.519  < 2e-16 ***
LabelAppeal 0.178419   0.006702  26.622  < 2e-16 ***
Alcohol     0.005471   0.001484   3.686 0.000228 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for Negative Binomial(142158.7) family taken to be 1)

    Null deviance: 8176.1  on 8962  degrees of freedom
Residual deviance: 5680.5  on 8959  degrees of freedom
  (3832 observations deleted due to missingness)
AIC: 32359

Number of Fisher Scoring iterations: 1


              Theta:  142159
          Std. Err.:  203705
Warning while fitting theta: iteration limit reached

 2 x log-likelihood:  -32349.38
```

Linear Regression Model 1

Linear regression model 1 utilizes the cleaned up version of the dataset and all variables. It is structured similarly to the negative binomial model 1. Many of the variables yield statistically insignificant which leads us to an improved version in model2. For final comparison we will be using AUC in the next section to compare its performance to other models.

```
Call:
lm(formula = target ~ ., data = wines)

Residuals:
    Min      1Q  Median      3Q     Max
-5.0614 -0.5143  0.1240  0.7170  3.2419

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)         4.563e+00  5.530e-01   8.251  < 2e-16 ***
FixedAcidity        1.685e-03  2.319e-03   0.727   0.4675
VolatileAcidity    -9.466e-02  1.846e-02  -5.129 3.00e-07 ***
CitricAcid         -4.836e-03  1.675e-02  -0.289   0.7728
ResidualSugar      -2.513e-04  4.276e-04  -0.588   0.5567
Chlorides          -1.134e-01  4.546e-02  -2.494   0.0126 *
FreeSulfurDioxide   2.264e-04  9.711e-05   2.332   0.0198 *
TotalSulfurDioxide  7.810e-05  6.288e-05   1.242   0.2142
Density            -1.281e+00  5.435e-01  -2.357   0.0185 *
pH                 -9.441e-03  2.121e-02  -0.445   0.6563
Sulphates          -1.727e-02  1.558e-02  -1.109   0.2676
Alcohol             1.653e-02  3.887e-03   4.252 2.15e-05 ***
LabelAppeal         6.442e-01  1.743e-02  36.947  < 2e-16 ***
AcidIndex          -1.649e-01  1.235e-02 -13.346  < 2e-16 ***
STARS               7.278e-01  1.710e-02  42.571  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.153 on 6421 degrees of freedom
  (6359 observations deleted due to missingness)
Multiple R-squared:  0.445,     Adjusted R-squared:  0.4438
F-statistic: 367.8 on 14 and 6421 DF,  p-value: < 2.2e-16
```

Linear Regression Model 2

This is our 8th model. Structured similarly to the negative binomial model 2 in terms of variables used and the choice of the cleaned up dataset.

```
Call:
lm(formula = target ~ STARS + LabelAppeal + Alcohol, data = wines)

Residuals:
    Min      1Q  Median      3Q     Max
-5.1246 -0.5127  0.1330  0.7444  3.1182

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.904821   0.045789  41.600  < 2e-16 ***
STARS       0.746515   0.014569  51.241  < 2e-16 ***
LabelAppeal 0.651196   0.014931  43.613  < 2e-16 ***
Alcohol     0.021111   0.003304   6.389 1.75e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.168 on 8959 degrees of freedom
  (3832 observations deleted due to missingness)
Multiple R-squared:  0.4341,    Adjusted R-squared:  0.4339
F-statistic:  2291 on 3 and 8959 DF,  p-value: < 2.2e-16
```

Residuals

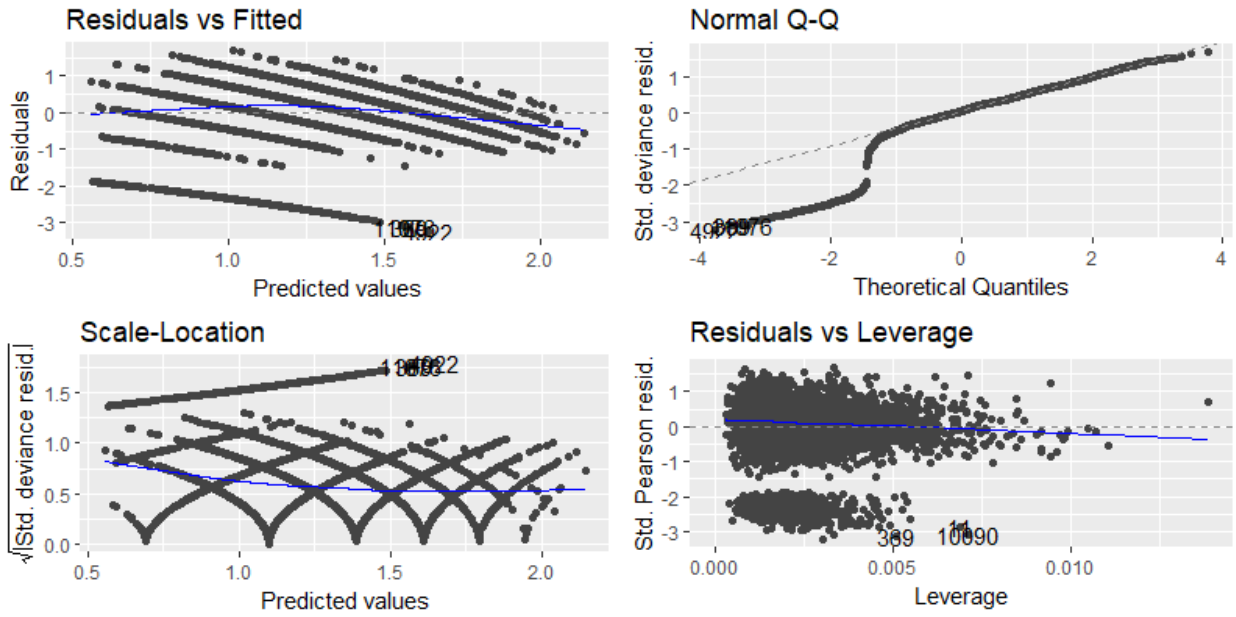
Residual plots listed below in the following order:

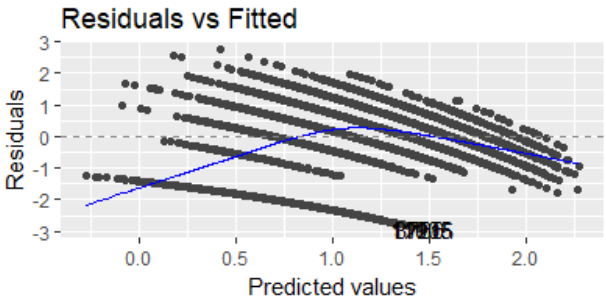
1. Poisson Model 1
2. Poisson Model 2
3. Poisson Model 3
4. Poisson Model 4
5. Negative Binomial Model 1
6. Negative Binomial Model 2
7. Multiple Linear Model 1
8. Multiple Linear Model 2

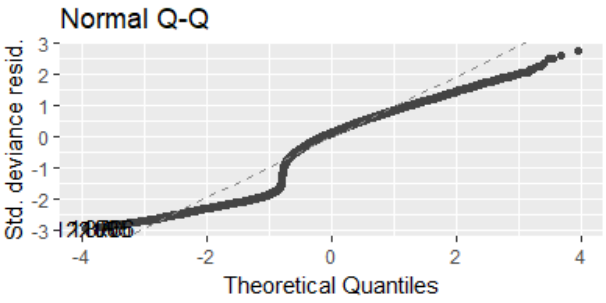
Residual plots highlight the fact that the distribution may be skewed and the it does not represent a perfect model. These fall under the modeling assumptions.

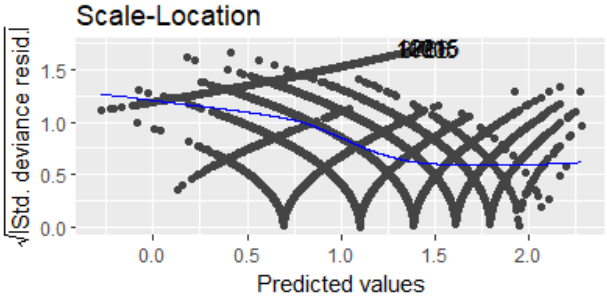
The highest mark for overall fitness of residuals as well as deviance goes to Multiple Linear Model 2

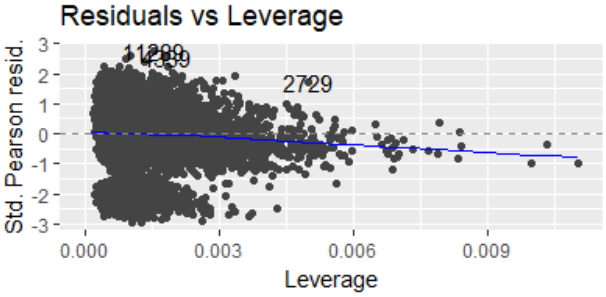
Poisson Residual Plots

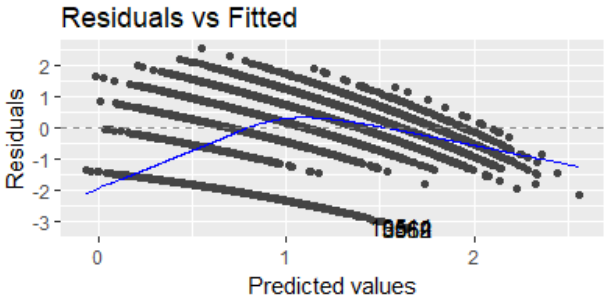


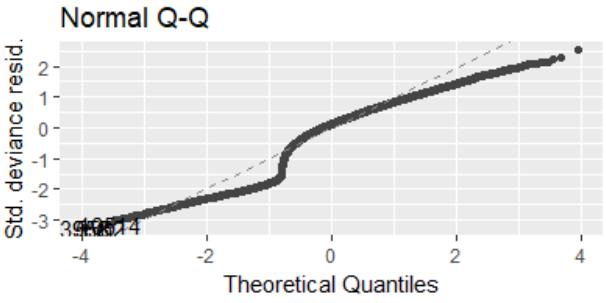

Residuals vs Fitted
Residuals
Predicted values

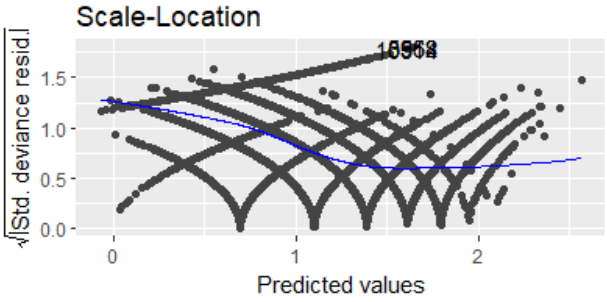

Normal Q-Q
Std. deviance resid.
Theoretical Quantiles

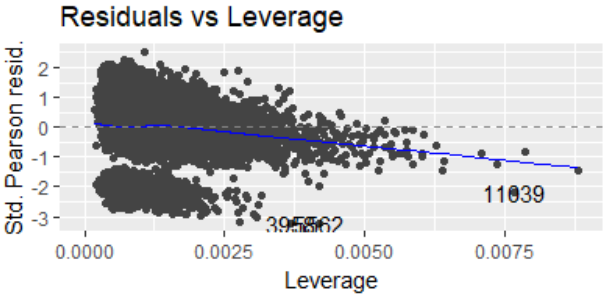

Scale-Location
√|Std. deviance resid.|
Predicted values

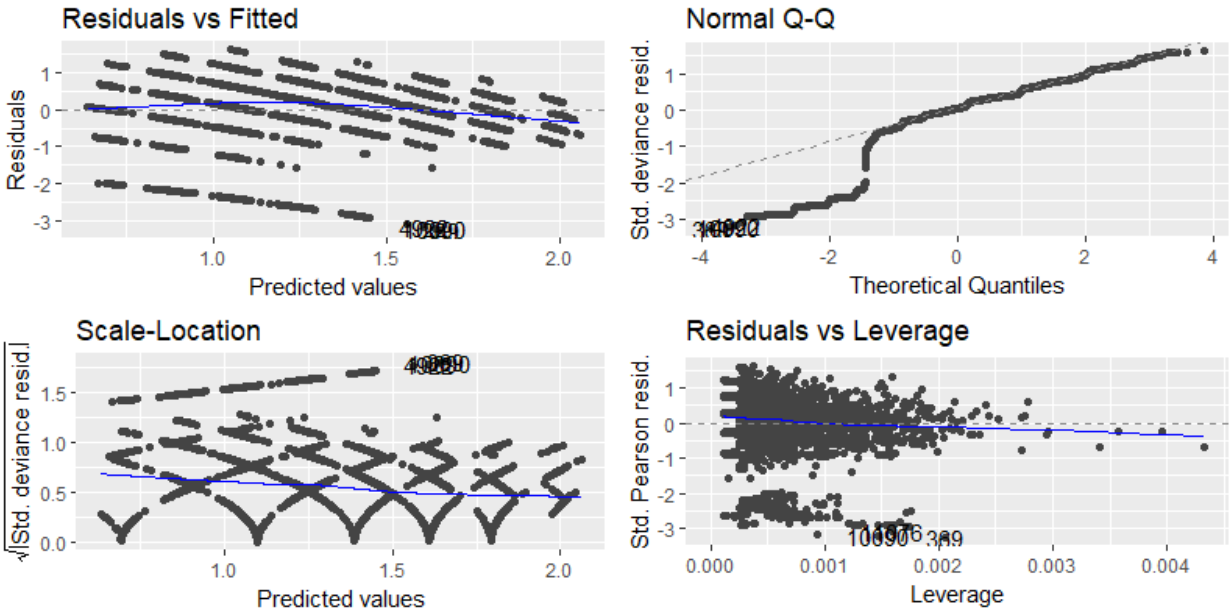

Residuals vs Leverage
Std. Pearson resid.
2729
Leverage


Residuals vs Fitted
Residuals
Predicted values

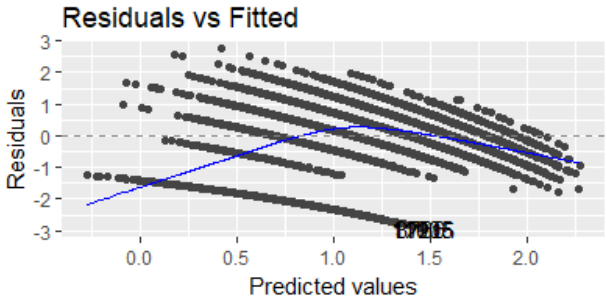

Normal Q-Q
Std. deviance resid.
Theoretical Quantiles

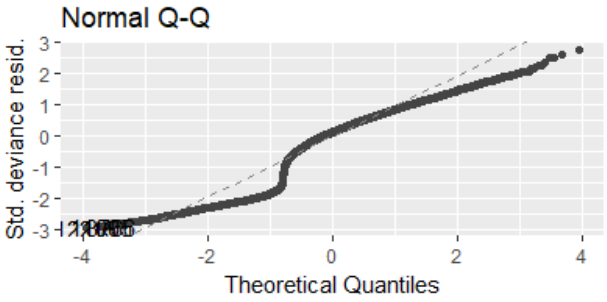

Scale-Location
√|Std. deviance resid.|
Predicted values

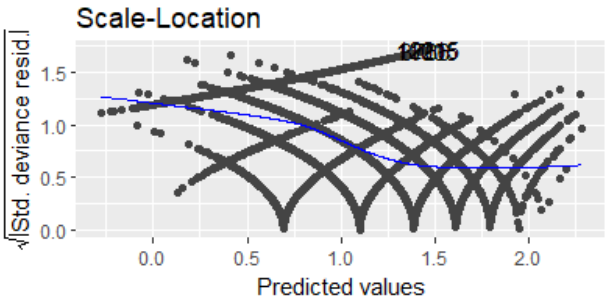

Residuals vs Leverage
Std. Pearson resid.
11039
Leverage

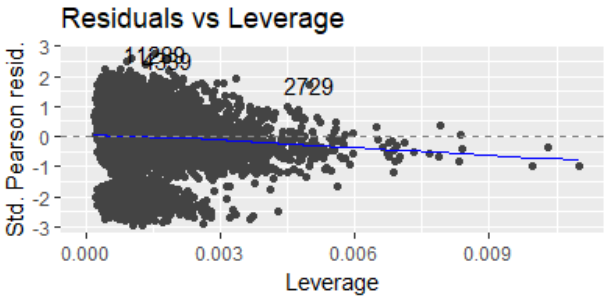

Residuals vs Fitted
Residuals
Predicted values
Normal Q-Q
Std. deviance resid.
Theoretical Quantiles
Scale-Location
√|Std. deviance resid.|
Predicted values
Residuals vs Leverage
Std. Pearson resid.
Leverage

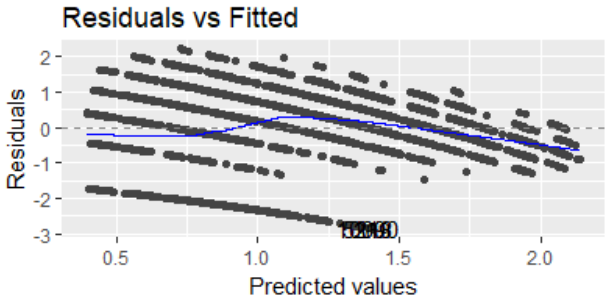
Negative Binomial Residuals Plots

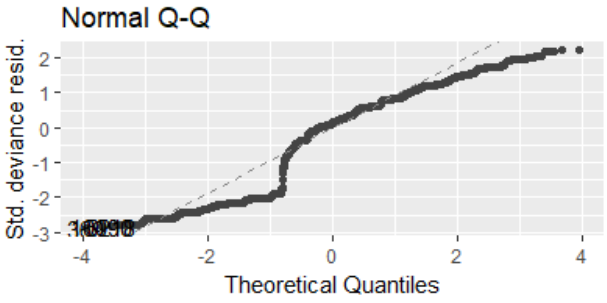


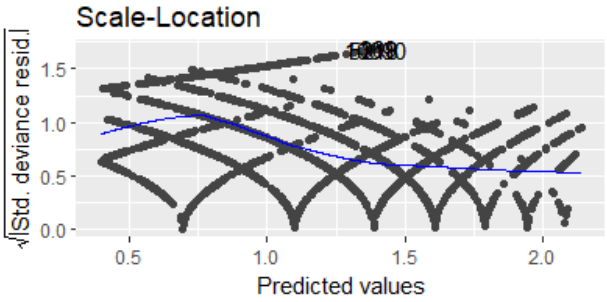


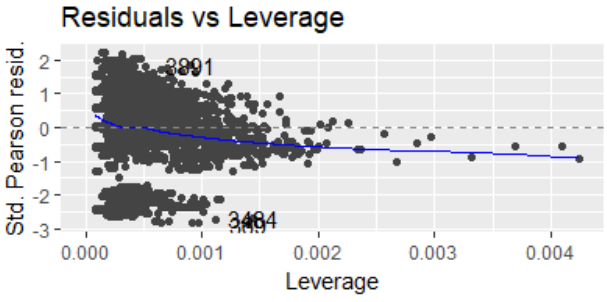




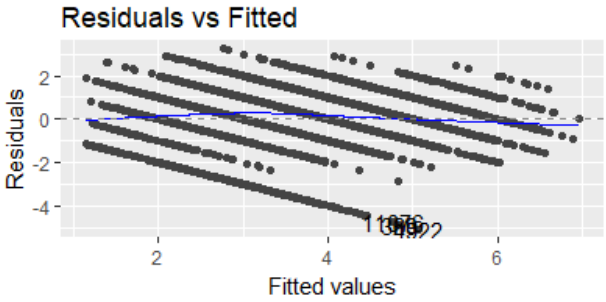


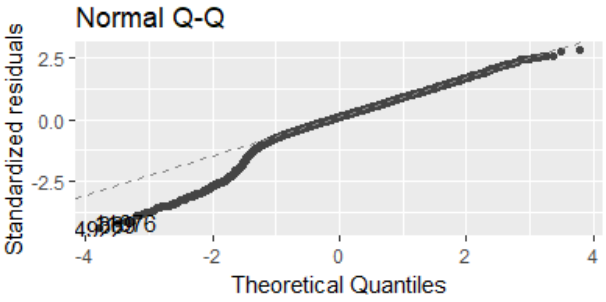


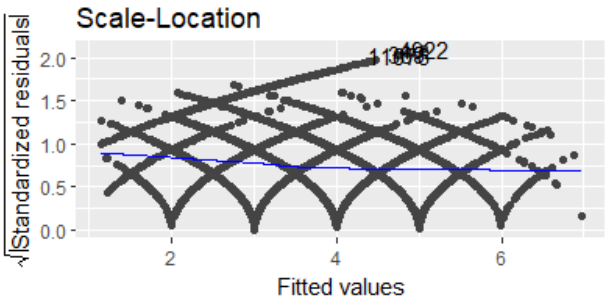


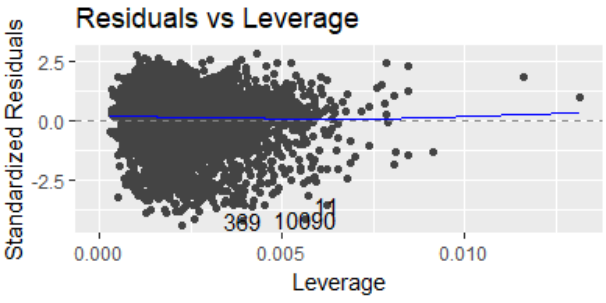


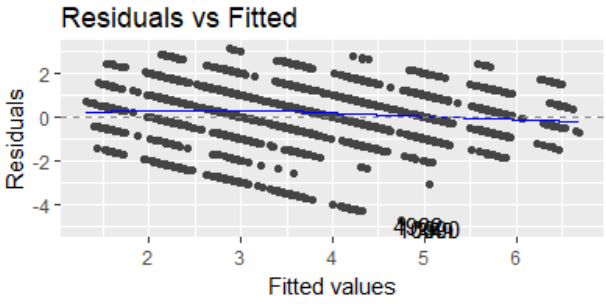
Multiple Linear Regression Residual Plots

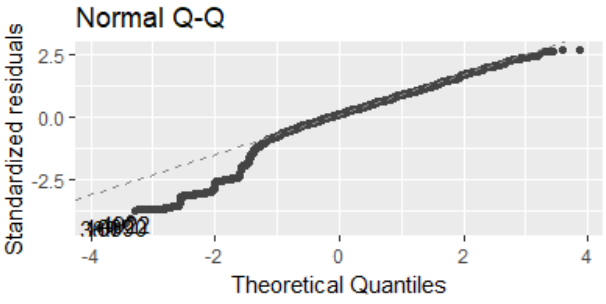


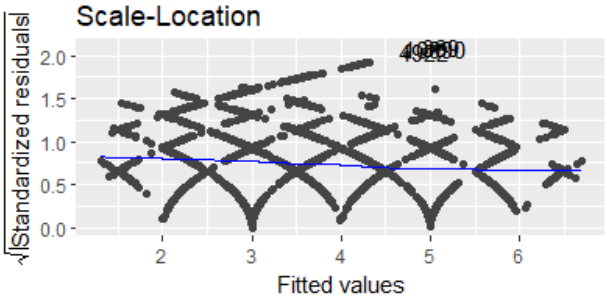


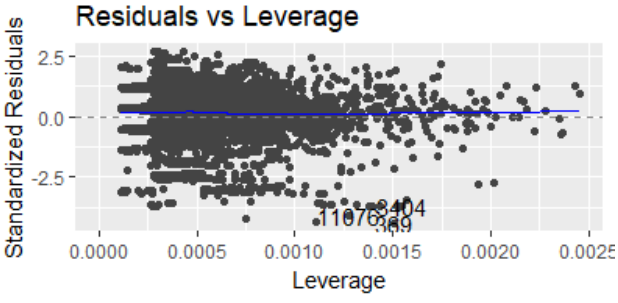








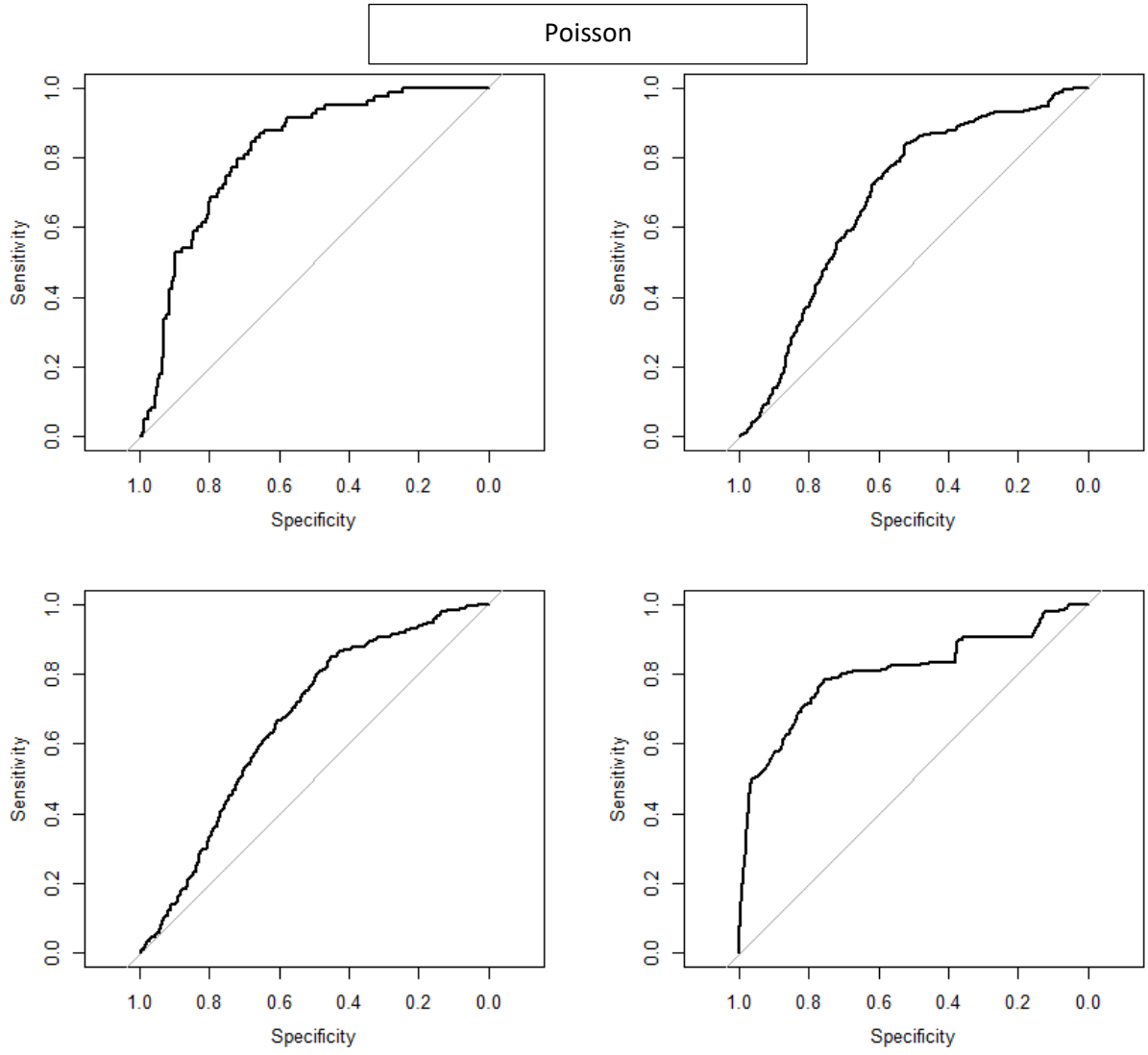




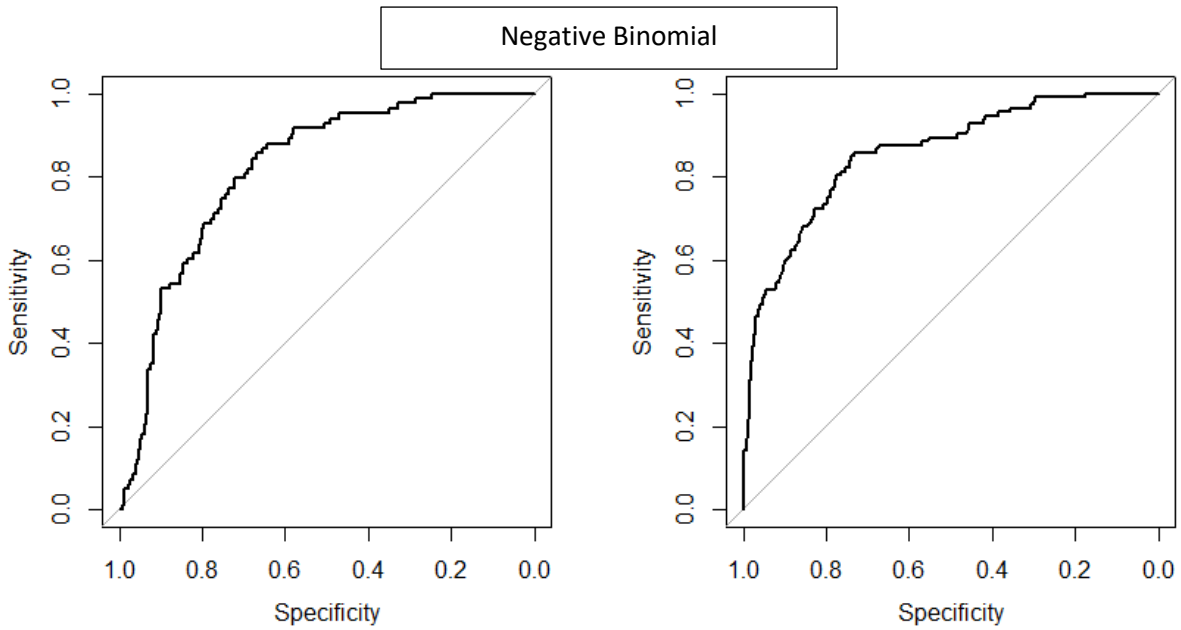
Model Selection

To narrow down to a single model, we will utilize pROC package to study the Area Under the Curve value as well as rendered graphs.

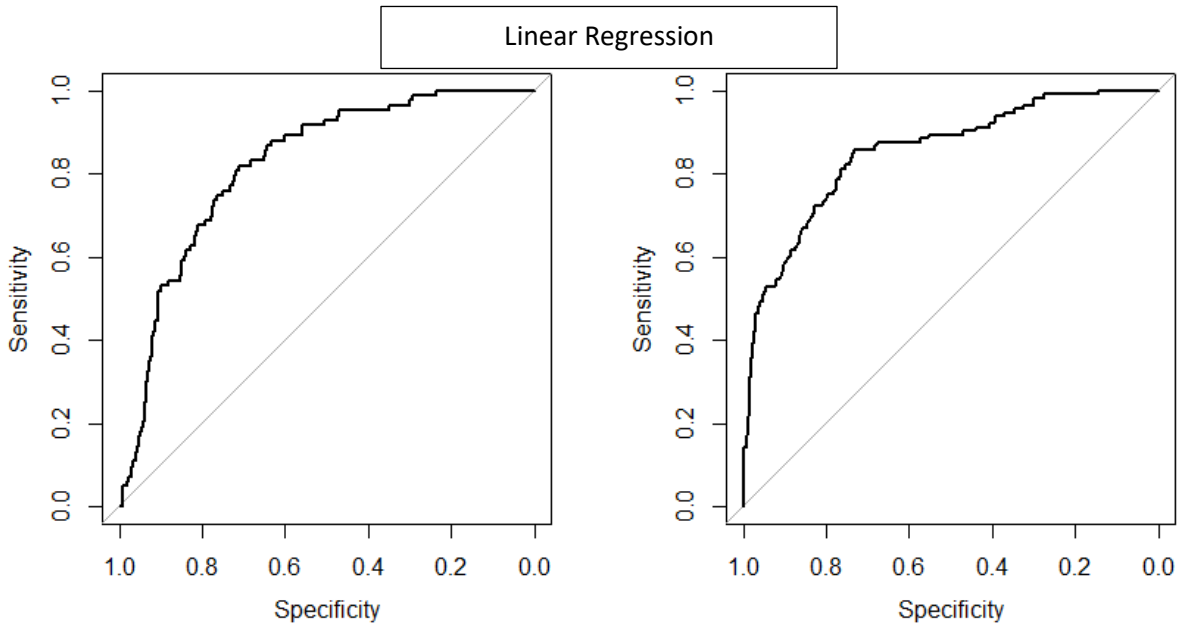
The poisson models 2 & 3 seem to be underperforming when compared to 1 & 4. Furthermore, the fact that model 4 is using only 3 variables makes it a better contender than model 1. Poisson model 4 is the primary choice from this group.



AUC for both Negative Binomial models below is similar. There are no 'sharp' steps as noted in Poisson model 4 and the overall 'increase rate' of sensitivity comes much faster on the specificity x-scale. Model 2 from this group seems to be the best model out of all reviewed so far.



Linear Regression AUC is high, with the first model performing better than the 2nd. LR 1 seems more smooth than previously highest rated NB model 2.



The precise values of the AUC analyzed from the graphs was also captured in the table below. It appears that the 3 'Appearance & Opinion' variables are the most significant when predicting the wine sampling demand. **Multiple Linear Regression Model 2 wins** based on the AUC, variables used, dataset transformation applied, as well as the overall model fitness of residuals and distribution deviance.

variable	*AUC*
NegBinomial2	0.8559
LinRegression2	**0.8518**
LinRegression1	0.8197
Poisson1	0.8168
NegBinomial1	0.8168
Poisson4	0.8046
Poisson2	0.6869
Poisson3	0.6645

Wine Demand Prediction - Regression Models

Rafal Decowski

April 29, 2018

Contents

```
library(dplyr)
library(tidyr)
library(knitr)
library(stringr)
library(reshape2)
library(ggplot2)
library(corrplot)
library(psych)



############################################################
# Loading data and simple transformations
############################################################
wines <- read.csv('wine-training-data.csv', stringsAsFactors=F)

# Drop the index column
wines <- wines[,-1]

# Confirm class of variables
sapply(wines, class)

# Convert class to numeric
wines <- as.data.frame(sapply(wines, as.numeric ))

############################################################
# Descriptive statistics
############################################################
stats <- as.data.frame(describe(wines))


# Compelte cases percentage
cc <- round(sum(complete.cases(wines))/nrow(wines),4)

# Percentage of missing values by variable
missing_values <- as.data.frame(1- round(stats$n/12795,2))
rownames(missing_values) <- rownames(stats)
colnames(missing_values) <- c('Percentage Missing')




############################################################
# Fixing Negative Values
```

```
############################################################

# When negative -> replace with NA's
wines_nn <- data.frame(wines)
wines_nn[wines_nn < 0] <- NA

# Correcting LabelAppeal
wines_nn$LabelAppeal <- wines$LabelAppeal


############################################################
# Outlier Handling
############################################################

remove_outliers <- function(x, na.rm = TRUE) {
  qnt <- quantile(x, probs=c(.25, .75), na.rm = na.rm)
  H <- 1.5 * IQR(x, na.rm = na.rm)
  y <- x
  y[x < (qnt[1] - H)] <- NA
  y[x > (qnt[2] + H)] <- NA
  return(y)
}


# Iterating through every variable and replacing outliers with NA's
# A while loop to eliminate ALL outliers for na_negative df
sumna_na_start <- sum(is.na(wines_nn))
sumna_na_end <- 0

wines_no_outliers <- data.frame(wines_nn)
while (sumna_na_start != sumna_na_end) {
  sumna_na_start <- sum(is.na(wines_no_outliers))
  print(sumna_na_start)
  for(i in c(2:15)){
    wines_no_outliers[,i] <- remove_outliers(wines_no_outliers[,i])
  }
  sumna_na_end <- sum(is.na(wines_no_outliers))
}




############################################################
# Missing values handling
############################################################

library(mice)
wines_nn <- mice(wines_nn, printFlag = FALSE)
wines_nn <- complete(wines_nn,1)

wines_no_outliers <- mice(wines_no_outliers, printFlag = FALSE)
wines_no_outliers <- complete(wines_no_outliers,1)
```

```
##########################################################
# Correlation
##########################################################

# Original Dataset
cormat1 <- cor(wines, use="complete")
res1 <- cor.mtest(cormat1, conf.level = .95)
corrplot(cormat1, type = "lower", order = "hclust", tl.col = "black", tl.srt = 45, p.mat = res1$p, sig.
target_corr1 <- as.data.frame(cormat1['target',])


# First data branch with outliers
cormat2 <- cor(wines_nn, use="complete")
res1 <- cor.mtest(cormat2, conf.level = .95)
corrplot(cormat2, type = "lower", order = "hclust", tl.col = "black", tl.srt = 45, p.mat = res1$p, sig.
target_corr2 <- as.data.frame(cormat2['target',])


# Second data branch without the outliers
cormat3 <- cor(wines_no_outliers, use="complete")
res1 <- cor.mtest(cormat3, conf.level = .95)
corrplot(cormat3, type = "lower", order = "hclust", tl.col = "black", tl.srt = 45, p.mat = res1$p, sig.
target_corr3 <- as.data.frame(cormat3['target',])




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


ggplot(data = melt(as.data.frame(wines)), aes(x=variable, y=value)) +
  geom_boxplot(aes(fill=variable)) +
  theme(legend.position="none") +
  facet_wrap( ~ variable, scales="free")

ggplot(data = melt(as.data.frame(wines_nn)), aes(x=variable, y=value)) +
  geom_boxplot(aes(fill=variable)) +
  theme(legend.position="none") +
  facet_wrap( ~ variable, scales="free")

ggplot(data = melt(as.data.frame(wines_no_outliers)), aes(x=variable, y=value)) +
  geom_boxplot(aes(fill=variable)) +
  theme(legend.position="none") +
  facet_wrap( ~ variable, scales="free")

##########################################################
```

```r
# Histograms
#############################################################


ggplot(data=melt(as.data.frame(wines)), aes(x=value)) +
    geom_histogram(aes(fill=..count..)) +
    theme_minimal() +
    facet_wrap( ~ variable, scales="free")


ggplot(data=melt(as.data.frame(wines_nn)), aes(x=value)) +
    geom_histogram(aes(fill=..count..)) +
    theme_minimal() +
    facet_wrap( ~ variable, scales="free")


ggplot(data=melt(as.data.frame(wines_no_outliers)), aes(x=value)) +
    geom_histogram(aes(fill=..count..)) +
    theme_minimal() +
    facet_wrap( ~ variable, scales="free")







#############################################################
# Poisson regression models
#############################################################
library(MASS)
library(scales)


# All Variables
p_model1 <- glm(target ~ ., wines, family = "poisson")
p_model2 <- glm(target ~ ., wines_nn, family = "poisson")
p_model3 <- glm(target ~ ., wines_no_outliers, family = "poisson")

# Selected Variables
p_model4 <- glm(target ~ STARS + LabelAppeal + Alcohol, wines, family = "poisson")

summary(p_model1)
summary(p_model2)
summary(p_model3)
summary(p_model4)

pred_p_model1 <- predict(p_model1, wines)
pred_p_model2 <- predict(p_model2, wines_nn)
pred_p_model3 <- predict(p_model3, wines_no_outliers)
pred_p_model4 <- predict(p_model4, wines)
```

```
############################################################
# Negative binomial regression
############################################################

nbr1 <- glm.nb(target ~ ., data = wines_nn)
nbr2 <- glm.nb(target ~ STARS + LabelAppeal + Alcohol, data = wines_nn)

summary(nbr1)
summary(nbr2)

pred_nbr1 <- predict(nbr1, wines_nn)
pred_nbr2 <- predict(nbr2, wines_nn)


############################################################
# Multiple linear regression
############################################################

lm1 <- lm(target ~ ., data = wines_nn)
lm2 <- lm(target ~ STARS + LabelAppeal + Alcohol, data = wines_nn)

summary(lm1)
summary(lm2)

pred_lm1 <- predict(lm1, wines_nn)
pred_lm2 <- predict(lm2, wines_nn)




############################################################
# Residual Analysis
############################################################
library(ggfortify)

autoplot(p_model1, title="asdihasd")
autoplot(p_model2)
autoplot(p_model3)
autoplot(p_model4)


autoplot(nbr1)
autoplot(nbr2)


autoplot(lm1)
autoplot(lm2)



############################################################
# ROC
############################################################
```

```
library(pROC)

par(mfrow=c(2, 2))
roc(wines$target ~ pred_p_model1, wines, plot=TRUE)
roc(wines_nn$target ~ pred_p_model2, wines_nn, plot=TRUE)
roc(wines_no_outliers$target ~ pred_p_model3, wines_no_outliers, plot=TRUE)
roc(wines$target ~ pred_p_model4, wines, plot=TRUE)


par(mfrow=c(1, 2))
roc(wines_nn$target ~ pred_nbr1, wines_nn, plot=TRUE)
roc(wines_nn$target ~ pred_nbr2, wines_nn, plot=TRUE)


par(mfrow=c(1, 2))
roc(wines_nn$target ~ pred_lm1, wines_nn, plot=TRUE)
roc(wines_nn$target ~ pred_lm2, wines_nn, plot=TRUE)

auc_df <- data.frame(cbind(Poisson1=0.8168,
                           Poisson2=0.6869,
                           Poisson3=0.6645,
                           Poisson4=0.8046,
                           NegBinomial1=0.8168,
                           NegBinomial2=0.8559,
                           LinRegression1=0.8197,
                           LinRegression2=0.8518))

melt(auc_df)




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


wine_eval <- read.csv2('wine-evaluation-data.csv', sep=',', row.names=NULL)

# Drop the index column
wine_eval <- wine_eval[,-1]
wine_eval <- wine_eval[,-1]


# Convert class to numeric
wine_eval <- as.data.frame(sapply(wine_eval, as.numeric ))

# Populate missing
wine_eval <- mice(wine_eval, printFlag = FALSE)
wine_eval <- complete(wine_eval,1)

predicted_eval <- predict(lm2, wine_eval, type="response")
```

```
wine_eval$Predicted_Cases <- round(predicted_eval,2)

write.csv(wine_eval, 'D:\\Rafal\\CUNY\\621\\hw\\hw5\\wines_predicted.csv')
```