Wine Case Prediction

Eric Lehmphul

5/14/2022

Introduction

This assignment explores a data set containing information on approximately 12,000 commercially available wines. The objective of this assignment is to build a count regression model to predict the number of cases of wine that will be sold given certain properties of the wine.

DATA EXPLORATION

Summary Statistics

The dataset contains multiple variables that are missing values. 8 out of the 15 variables contain missing values. Data imputation will be necessary to proceed to data modeling.

There also appears to be negative values that should not be possible. Such variables where this exists are: `FixedAcidity`, `VolatileAcidity`, `CitricAcid`, `ResidualSugar`, `Chlorides`, `FreeSulfurDioxide`, `TotalSulfurDioxide`, `Sulphates`, and `LabelAppeal`. In the data preprocessing section on unusual negative values, I will adjust them using both absolute value and the absolute value of the minimum value to guarantee that all values make sense. See data preprocessing section for more information.

```
##      TARGET       FixedAcidity     VolatileAcidity     CitricAcid
##  Min.   :0.000   Min.   :-18.100   Min.   :-2.7900   Min.   :-3.2400
##  1st Qu.:2.000   1st Qu.:  5.200   1st Qu.: 0.1300   1st Qu.: 0.0300
##  Median :3.000   Median :  6.900   Median : 0.2800   Median : 0.3100
##  Mean   :3.029   Mean   :  7.076   Mean   : 0.3241   Mean   : 0.3084
##  3rd Qu.:4.000   3rd Qu.:  9.500   3rd Qu.: 0.6400   3rd Qu.: 0.5800
##  Max.   :8.000   Max.   : 34.400   Max.   : 3.6800   Max.   : 3.8600
##
##  ResidualSugar        Chlorides       FreeSulfurDioxide TotalSulfurDioxide
##  Min.   :-127.800   Min.   :-1.1710   Min.   :-555.00   Min.   :-823.0
##  1st Qu.:  -2.000   1st Qu.:-0.0310   1st Qu.:   0.00   1st Qu.:  27.0
##  Median :   3.900   Median : 0.0460   Median :  30.00   Median : 123.0
##  Mean   :   5.419   Mean   : 0.0548   Mean   :  30.85   Mean   : 120.7
##  3rd Qu.:  15.900   3rd Qu.: 0.1530   3rd Qu.:  70.00   3rd Qu.: 208.0
##  Max.   : 141.150   Max.   : 1.3510   Max.   : 623.00   Max.   :1057.0
##  NA's   :616        NA's   :638       NA's   :647       NA's   :682
##     Density             pH          Sulphates          Alcohol
##  Min.   :0.8881   Min.   :0.480   Min.   :-3.1300   Min.   :-4.70
##  1st Qu.:0.9877   1st Qu.:2.960   1st Qu.: 0.2800   1st Qu.: 9.00
##  Median :0.9945   Median :3.200   Median : 0.5000   Median :10.40
```

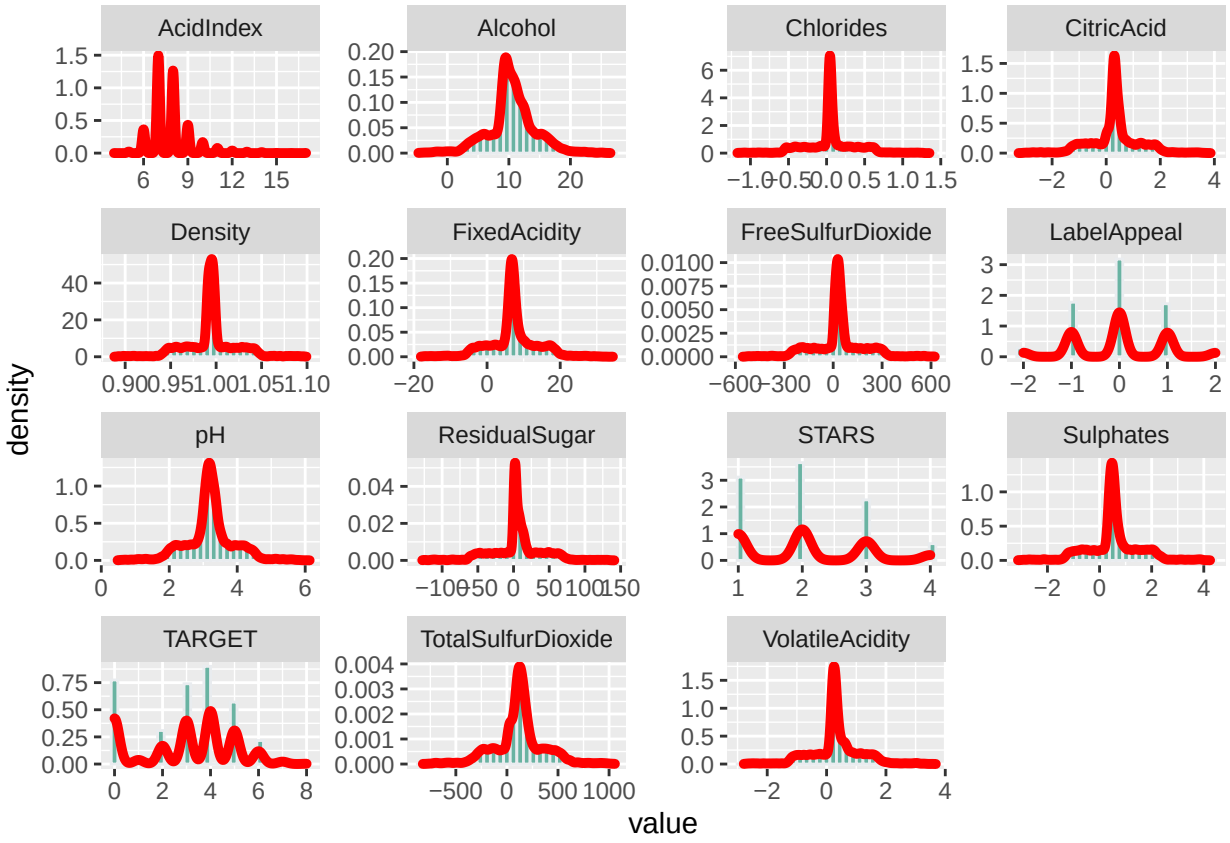
```
##  Mean   :0.9942   Mean   :3.208   Mean   : 0.5271   Mean   :10.49
##  3rd Qu.:1.0005   3rd Qu.:3.470   3rd Qu.: 0.8600   3rd Qu.:12.40
##  Max.   :1.0992   Max.   :6.130   Max.   : 4.2400   Max.   :26.50
##                   NA's   :395     NA's   :1210      NA's   :653
##   LabelAppeal          AcidIndex          STARS
##  Min.   :-2.000000   Min.   : 4.000   Min.   :1.000
##  1st Qu.:-1.000000   1st Qu.: 7.000   1st Qu.:1.000
##  Median : 0.000000   Median : 8.000   Median :2.000
##  Mean   :-0.009066   Mean   : 7.773   Mean   :2.042
##  3rd Qu.: 1.000000   3rd Qu.: 8.000   3rd Qu.:3.000
##  Max.   : 2.000000   Max.   :17.000   Max.   :4.000
##                                       NA's   :3359
```

Variable Distributions

Notable takaways from looking at the data distributions: * `STARS` and `LabelAppear` are categorical variables. * `AcidIndex` looks to follow a poisson distribution shape. * The following variables has a near normal distribution with high kurtosis: `Alcohol`, `Clorides`, `CitricAcid`, `Densitiy`, `FixedAcidity`, `FreeSulfurDioxide`, `pH`, `ResidualSugar`, `Sulphates`, `TotoalSulfurDioxide`, and `VolatileAcidiy`.

```
## Warning: Removed 8200 rows containing non-finite values (stat_bin).
```

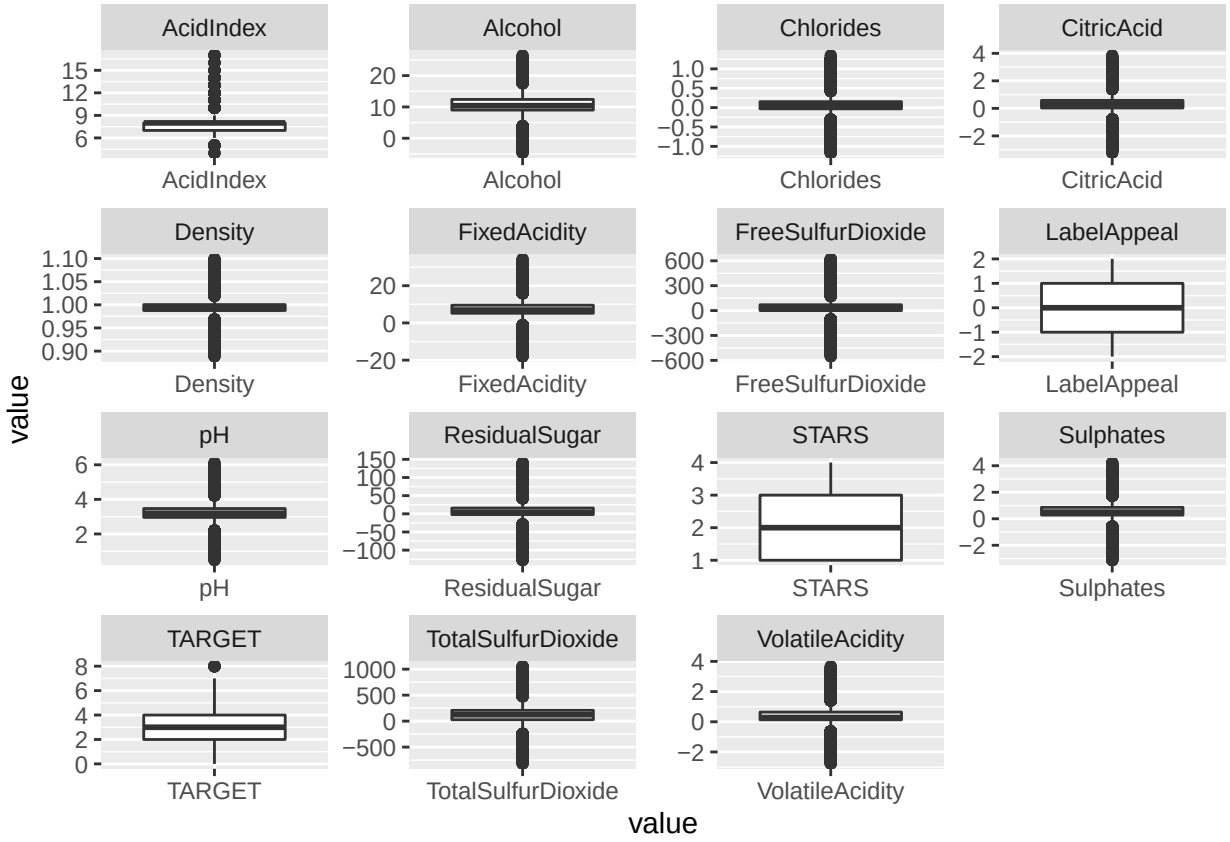
```
## Warning: Removed 8200 rows containing non-finite values (stat_density).
```



Boxplots

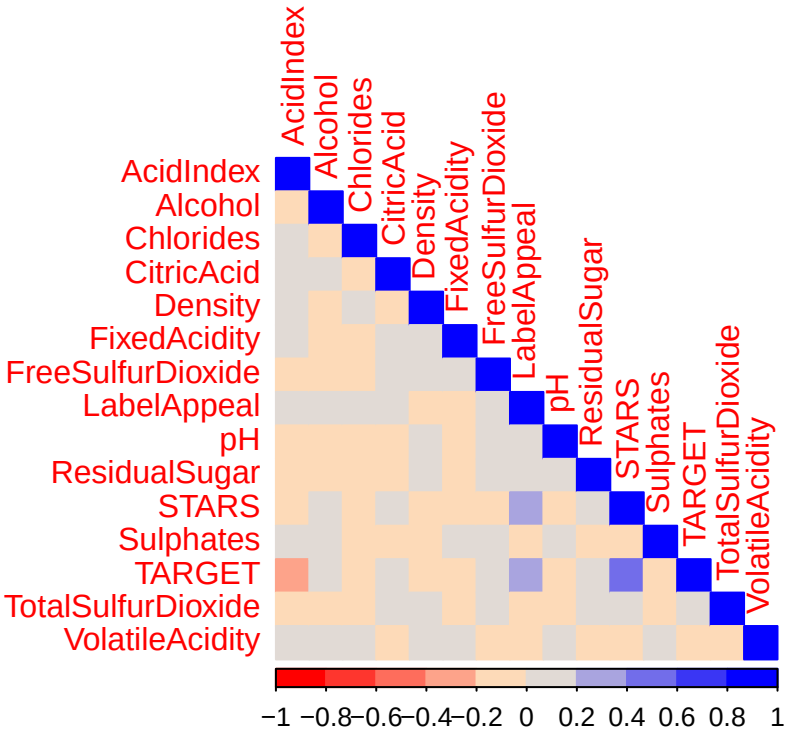
Most of the high kurtosis variables have an abundant amount of outliers, but none appear extreme outliers. Data transformations may not be necessary and outlier removal may not be necessary either.

```
## Warning: Removed 8200 rows containing non-finite values (stat_boxplot).
```



Correlation Matrix to assess Mulicollinearity

There does not seem to be any strongly correlated variables, meaning that multicolinearity is not very likely to occur in the data.



DATA PREPARATION

Removed Unnecessary Variables

The index variable was removed from the dataset as it provides no meaningful relation to the data.

Adjust Data Types

I changed the data type of `STARS` to an ordered factor variable as it only have 4 values: '1', '2', '3', '4'. Least is 1 and most is 4.

`LabelAppeal` was also changed into an ordered factor variable with the values: '-2', '-1', '0', '1', '2'.

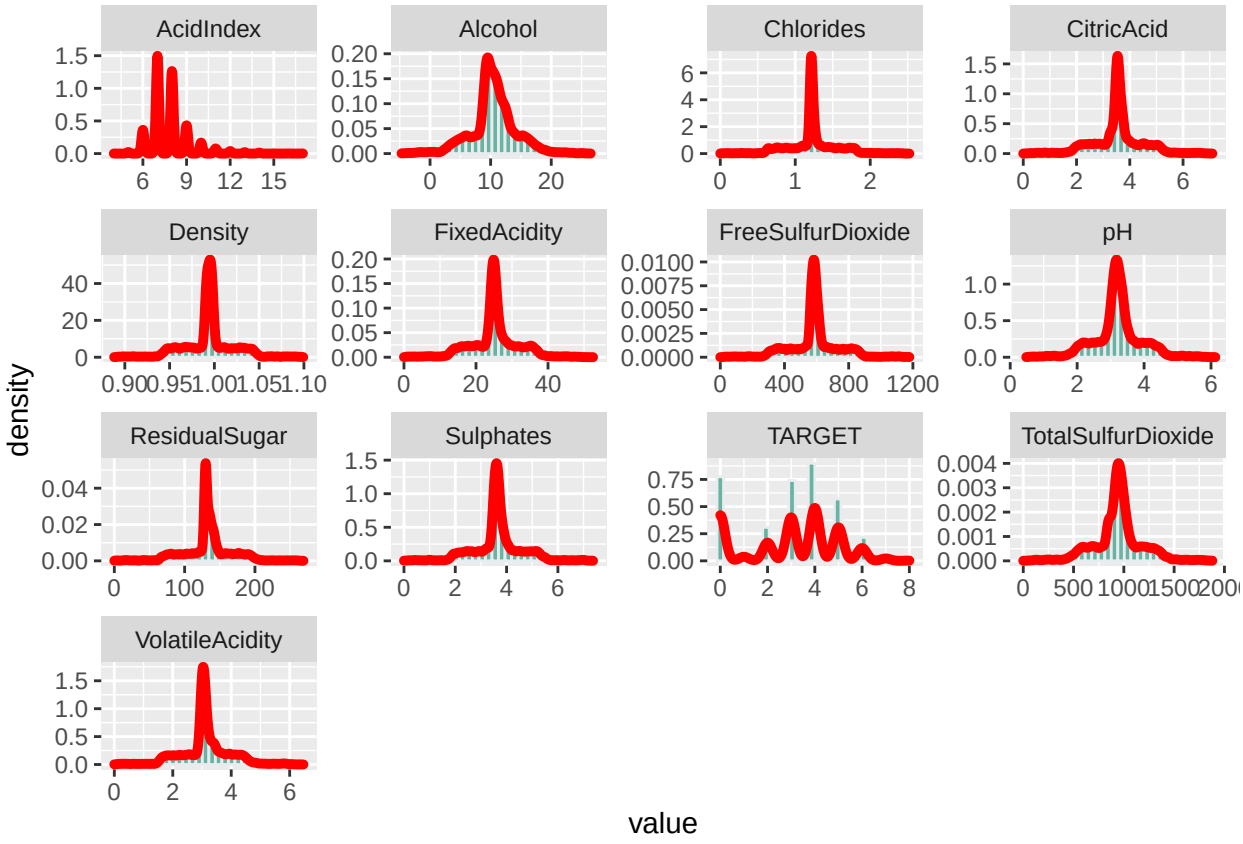
Handle Missing Values

To imputed the missing data I chose to implement KNN imputation for numeric data and imputed the categorical variable `STARS` using the logic that if there was n review, they did not enjoy the product. The numeric data was normalized after KNN imputation. I transformed the data back to its original form for modeling.

Adjusting Invalid Negative Values

I adjusted the strange negative values by by shifting all variables to have a minimum value of 0. I accomplished this by adding the absolute value of the min to each row.

New Data Distributions




```
## [1] "STARS:"
```

```
##    1    2    3    4
## 6401 3570 2212  612
```

```
## [1] "LabelAppeal"
```

```
##   -2   -1    0    1    2
##  504 3136 5617 3048  490
```

BUILD MODELS

Create Train Test Split

I will use a 70% training set and a 30% testing set for creating and evaluating models.

Model 1 - Poisson with all Variables

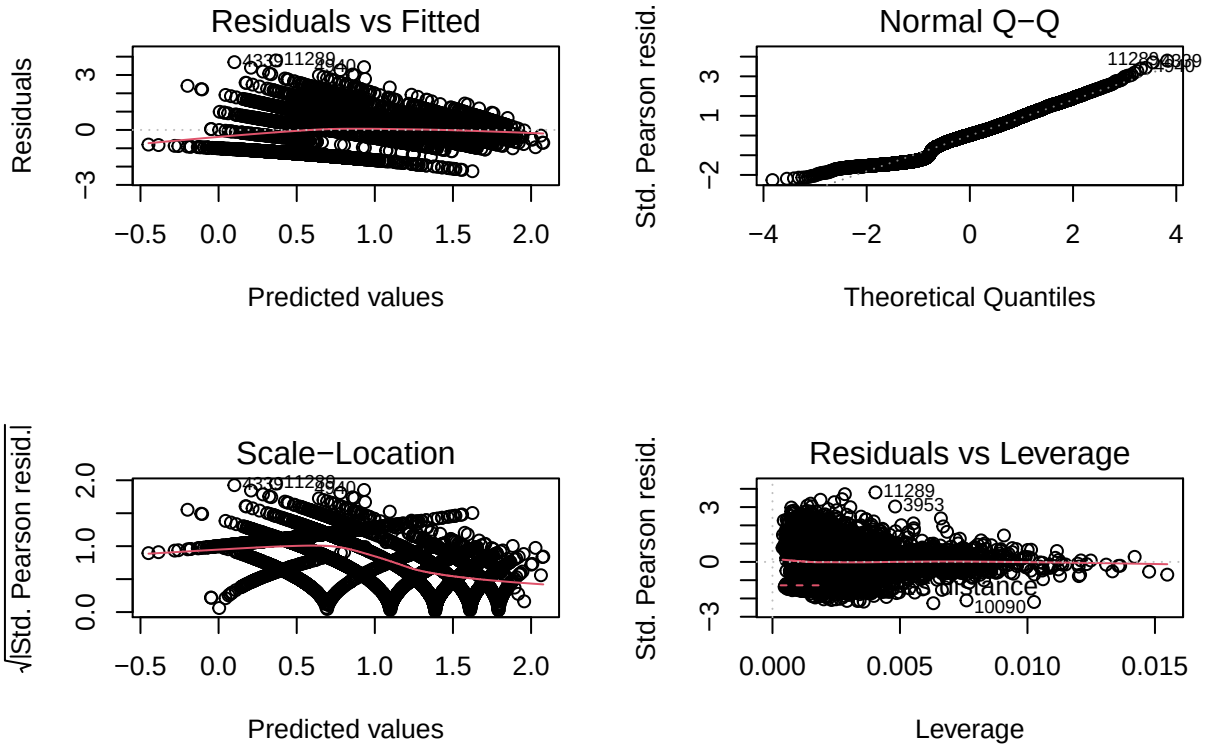
```
##
## Call:
## glm(formula = TARGET ~ ., family = "poisson", data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.1860  -0.6491   0.0369   0.5752   2.8264
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)          2.153e+00  2.600e-01   8.279  < 2e-16 ***
## FixedAcidity        -2.831e-05  1.068e-03  -0.026  0.97886
## VolatileAcidity     -3.723e-02  8.469e-03  -4.396  1.1e-05 ***
## CitricAcid           1.364e-02  7.521e-03   1.814  0.06972 .
## ResidualSugar        7.200e-05  1.968e-04   0.366  0.71443
## Chlorides           -5.194e-02  2.118e-02  -2.453  0.01418 *
## FreeSulfurDioxide    7.930e-05  4.563e-05   1.738  0.08220 .
## TotalSulfurDioxide   7.721e-05  2.908e-05   2.655  0.00792 **
## Density             -3.760e-02  2.470e-01  -0.152  0.87900
## pH                  -1.934e-02  9.914e-03  -1.951  0.05109 .
## Sulphates           -2.084e-02  7.359e-03  -2.832  0.00463 **
## Alcohol              2.247e-03  1.818e-03   1.236  0.21656
## LabelAppeal.L        5.228e-01  3.531e-02  14.808  < 2e-16 ***
## LabelAppeal.Q       -7.813e-02  2.966e-02  -2.634  0.00843 **
## LabelAppeal.C        2.131e-02  2.101e-02   1.014  0.31051
## LabelAppeal^4        1.135e-02  1.337e-02   0.849  0.39591
## AcidIndex           -1.026e-01  5.876e-03 -17.461  < 2e-16 ***
## STARS.L              6.221e-01  1.850e-02  33.623  < 2e-16 ***
## STARS.Q             -2.627e-01  1.545e-02 -17.005  < 2e-16 ***
## STARS.C              1.140e-01  1.275e-02   8.944  < 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: 13824.1  on 7676  degrees of freedom
## Residual deviance:  9187.1  on 7657  degrees of freedom
## AIC: 28377
##
## Number of Fisher Scoring iterations: 5
```

Multicollinearity Check

```
##                        GVIF Df GVIF^(1/(2*Df))
## FixedAcidity       1.026663  1        1.013244
## VolatileAcidity    1.006021  1        1.003006
## CitricAcid         1.005878  1        1.002935
## ResidualSugar      1.004179  1        1.002088
## Chlorides          1.004807  1        1.002401
## FreeSulfurDioxide  1.005294  1        1.002644
## TotalSulfurDioxide 1.003325  1        1.001661
```

```
## Density            1.005003  1        1.002498
## pH                 1.008179  1        1.004081
## Sulphates          1.003344  1        1.001671
## Alcohol            1.013433  1        1.006694
## LabelAppeal        1.124968  4        1.014828
## AcidIndex          1.059705  1        1.029420
## STARS              1.154802  3        1.024278
```

Diagnostic Plots



Model 2 - Backward Selection Poisson

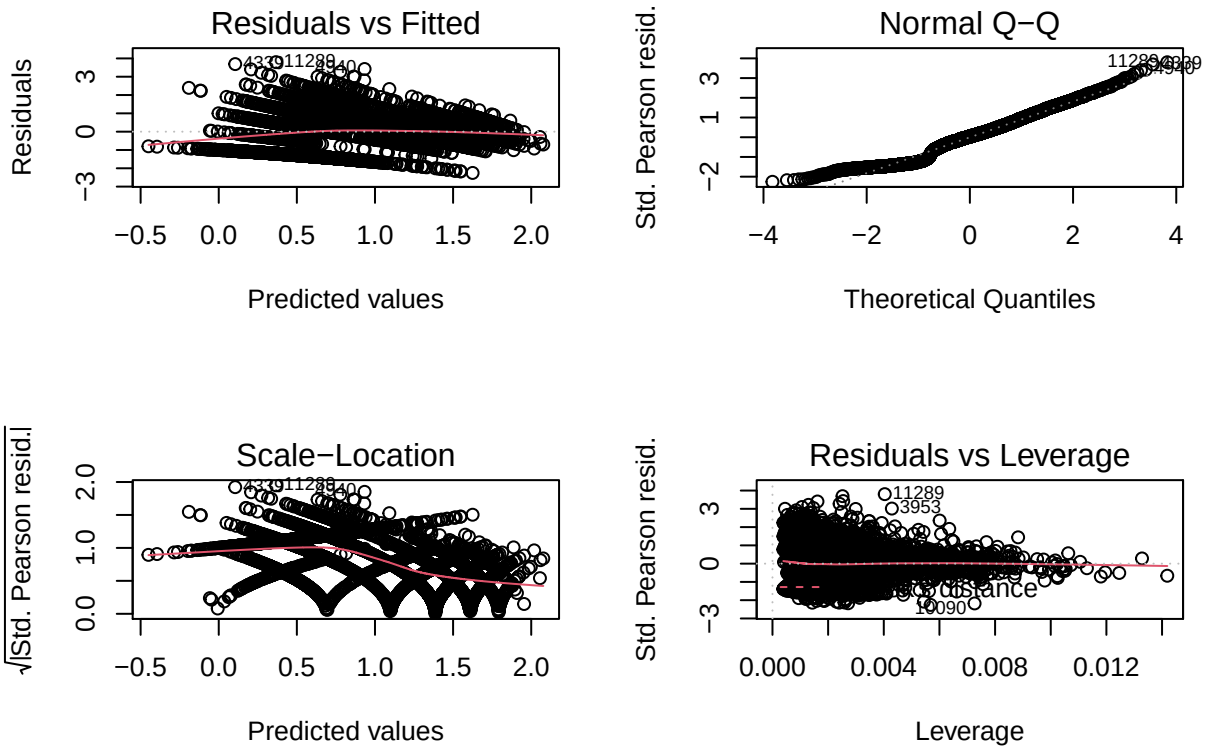
```
##
## Call:
## glm(formula = TARGET ~ VolatileAcidity + CitricAcid + Chlorides +
##     FreeSulfurDioxide + TotalSulfurDioxide + pH + Sulphates +
##     LabelAppeal + AcidIndex + STARS, family = "poisson", data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.1868  -0.6558   0.0366   0.5760   2.8268
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
```

```
## (Intercept)         2.153e+00  8.610e-02  25.009  < 2e-16 ***
## VolatileAcidity    -3.712e-02  8.470e-03  -4.383 1.17e-05 ***
## CitricAcid          1.379e-02  7.518e-03   1.834  0.06672 .
## Chlorides          -5.279e-02  2.116e-02  -2.495  0.01258 *
## FreeSulfurDioxide   7.839e-05  4.560e-05   1.719  0.08560 .
## TotalSulfurDioxide  7.637e-05  2.904e-05   2.630  0.00855 **
## pH                 -1.962e-02  9.904e-03  -1.981  0.04762 *
## Sulphates          -2.072e-02  7.352e-03  -2.819  0.00482 **
## LabelAppeal.L       5.231e-01  3.530e-02  14.817  < 2e-16 ***
## LabelAppeal.Q      -7.742e-02  2.965e-02  -2.611  0.00903 **
## LabelAppeal.C       2.150e-02  2.101e-02   1.023  0.30622
## LabelAppeal^4       1.141e-02  1.337e-02   0.854  0.39319
## AcidIndex          -1.029e-01  5.802e-03 -17.745  < 2e-16 ***
## STARS.L             6.238e-01  1.844e-02  33.825  < 2e-16 ***
## STARS.Q            -2.624e-01  1.545e-02 -16.989  < 2e-16 ***
## STARS.C             1.141e-01  1.274e-02   8.954  < 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: 13824.1  on 7676  degrees of freedom
## Residual deviance:  9188.7  on 7661  degrees of freedom
## AIC: 28371
##
## Number of Fisher Scoring iterations: 5
```

Multicollinearity Check

```
##                        GVIF Df GVIF^(1/(2*Df))
## VolatileAcidity    1.005631  1        1.002811
## CitricAcid         1.005151  1        1.002572
## Chlorides          1.003160  1        1.001579
## FreeSulfurDioxide  1.004155  1        1.002075
## TotalSulfurDioxide 1.001301  1        1.000650
## pH                 1.006421  1        1.003205
## Sulphates          1.001920  1        1.000960
## LabelAppeal        1.122081  4        1.014502
## AcidIndex          1.032499  1        1.016120
## STARS              1.145441  3        1.022890
```

Diagnostic Plots



Model 3 - Negative Binomial

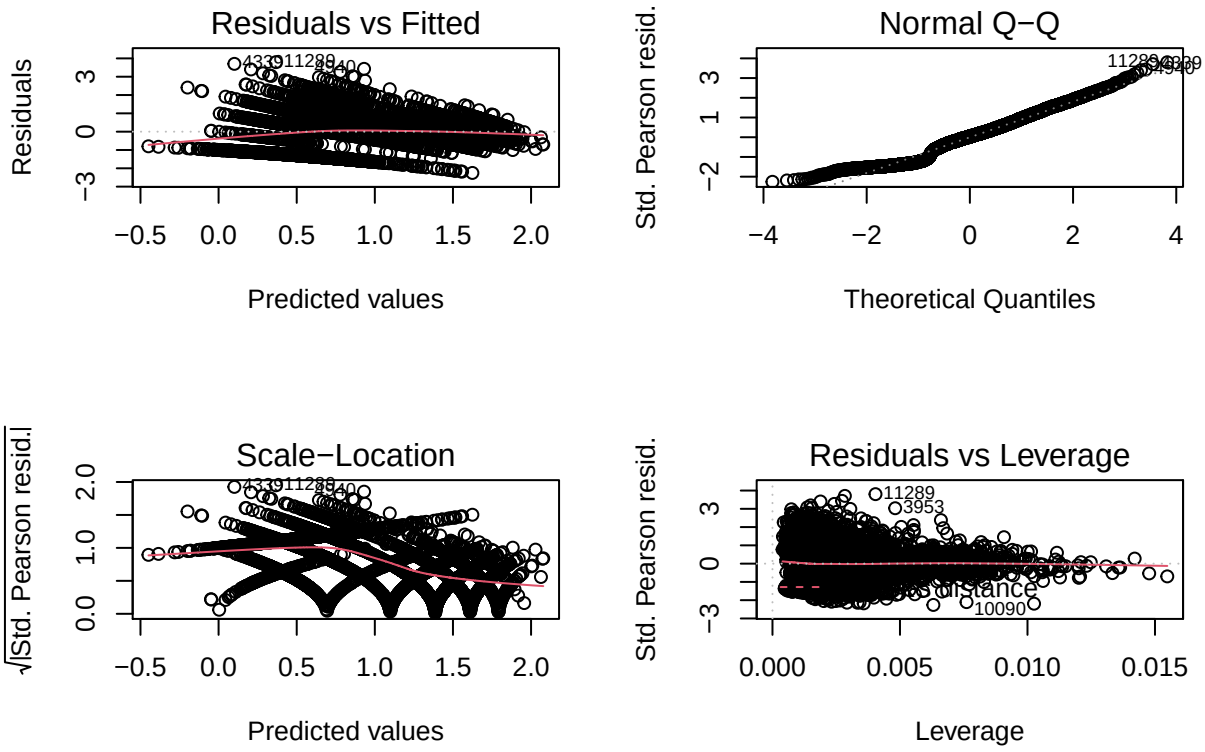
```
##
## Call:
## glm.nb(formula = TARGET ~ ., data = train, init.theta = 47002.99746,
##     link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.1859  -0.6490   0.0369   0.5752   2.8263
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)          2.153e+00  2.600e-01   8.278  < 2e-16 ***
## FixedAcidity        -2.831e-05  1.068e-03  -0.027  0.97885
## VolatileAcidity     -3.723e-02  8.470e-03  -4.396  1.1e-05 ***
## CitricAcid           1.364e-02  7.522e-03   1.814  0.06972 .
## ResidualSugar        7.201e-05  1.968e-04   0.366  0.71442
## Chlorides           -5.194e-02  2.118e-02  -2.453  0.01418 *
## FreeSulfurDioxide    7.931e-05  4.563e-05   1.738  0.08221 .
## TotalSulfurDioxide   7.722e-05  2.908e-05   2.655  0.00792 **
## Density             -3.760e-02  2.470e-01  -0.152  0.87901
## pH                  -1.934e-02  9.915e-03  -1.951  0.05109 .
```

```
## Sulphates          -2.084e-02  7.359e-03  -2.832  0.00463 **
## Alcohol             2.247e-03  1.818e-03   1.236  0.21660
## LabelAppeal.L       5.228e-01  3.531e-02  14.808  < 2e-16 ***
## LabelAppeal.Q      -7.813e-02  2.966e-02  -2.634  0.00843 **
## LabelAppeal.C       2.131e-02  2.101e-02   1.014  0.31051
## LabelAppeal^4       1.135e-02  1.337e-02   0.849  0.39592
## AcidIndex          -1.026e-01  5.876e-03 -17.460  < 2e-16 ***
## STARS.L             6.221e-01  1.850e-02  33.621  < 2e-16 ***
## STARS.Q            -2.627e-01  1.545e-02 -17.004  < 2e-16 ***
## STARS.C             1.140e-01  1.275e-02   8.944  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(47003) family taken to be 1)
##
##     Null deviance: 13823.5  on 7676  degrees of freedom
## Residual deviance:  9186.7  on 7657  degrees of freedom
## AIC: 28379
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  47003
##           Std. Err.:  61095
## Warning while fitting theta: iteration limit reached
##
##  2 x log-likelihood:  -28337.42
```

Multicollinearity Check

```
##                        GVIF Df GVIF^(1/(2*Df))
## FixedAcidity       1.026663  1        1.013244
## VolatileAcidity    1.006021  1        1.003006
## CitricAcid         1.005878  1        1.002934
## ResidualSugar      1.004179  1        1.002087
## Chlorides          1.004807  1        1.002401
## FreeSulfurDioxide  1.005294  1        1.002644
## TotalSulfurDioxide 1.003325  1        1.001661
## Density            1.005003  1        1.002498
## pH                 1.008179  1        1.004081
## Sulphates          1.003344  1        1.001671
## Alcohol            1.013432  1        1.006694
## LabelAppeal        1.124966  4        1.014828
## AcidIndex          1.059705  1        1.029420
## STARS              1.154800  3        1.024278
```

Diagnostic Plots



Model 4 - Backward Selecion Negative Binomial

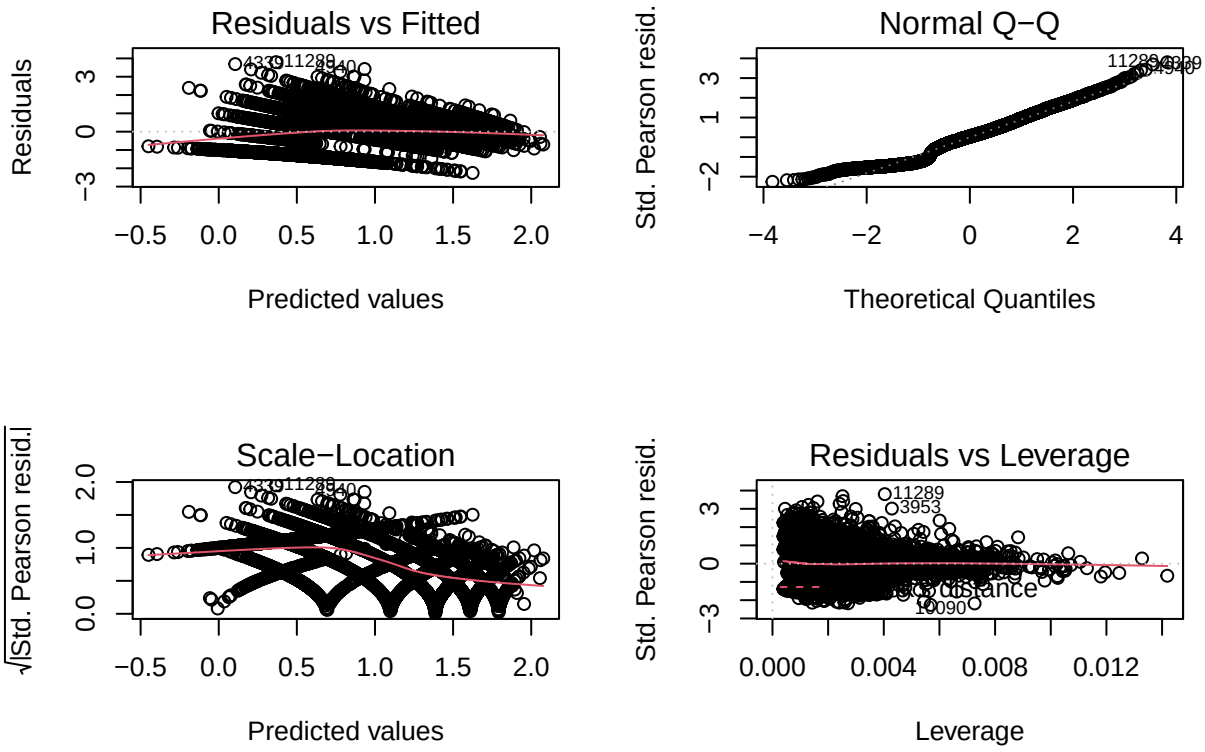
```
##
## Call:
## glm.nb(formula = TARGET ~ VolatileAcidity + CitricAcid + Chlorides +
##     FreeSulfurDioxide + TotalSulfurDioxide + pH + Sulphates +
##     LabelAppeal + AcidIndex + STARS, data = train, init.theta = 46942.99687,
##     link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.1868  -0.6558   0.0366   0.5760   2.8267
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)          2.153e+00  8.610e-02  25.009  < 2e-16 ***
## VolatileAcidity     -3.712e-02  8.470e-03  -4.383 1.17e-05 ***
## CitricAcid           1.379e-02  7.519e-03   1.834  0.06673 .
## Chlorides           -5.279e-02  2.116e-02  -2.495  0.01259 *
## FreeSulfurDioxide    7.840e-05  4.560e-05   1.719  0.08560 .
## TotalSulfurDioxide   7.638e-05  2.904e-05   2.630  0.00855 **
## pH                  -1.962e-02  9.905e-03  -1.981  0.04762 *
## Sulphates           -2.073e-02  7.352e-03  -2.819  0.00482 **
```

```
## LabelAppeal.L       5.231e-01  3.530e-02  14.817  < 2e-16 ***
## LabelAppeal.Q      -7.742e-02  2.965e-02  -2.611  0.00903 **
## LabelAppeal.C       2.150e-02  2.101e-02   1.023  0.30622
## LabelAppeal^4       1.141e-02  1.337e-02   0.854  0.39320
## AcidIndex          -1.030e-01  5.802e-03 -17.744  < 2e-16 ***
## STARS.L             6.238e-01  1.844e-02  33.824  < 2e-16 ***
## STARS.Q            -2.624e-01  1.545e-02 -16.988  < 2e-16 ***
## STARS.C             1.141e-01  1.274e-02   8.954  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(46943) family taken to be 1)
##
##     Null deviance: 13823.5  on 7676  degrees of freedom
## Residual deviance:  9188.4  on 7661  degrees of freedom
## AIC: 28373
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  46943
##           Std. Err.:  61019
## Warning while fitting theta: iteration limit reached
##
##  2 x log-likelihood:  -28339.09
```

Multicollinearity Check

```
##                         GVIF Df GVIF^(1/(2*Df))
## VolatileAcidity    1.005631  1        1.002811
## CitricAcid         1.005151  1        1.002572
## Chlorides          1.003160  1        1.001579
## FreeSulfurDioxide  1.004155  1        1.002075
## TotalSulfurDioxide 1.001301  1        1.000650
## pH                 1.006421  1        1.003205
## Sulphates          1.001920  1        1.000960
## LabelAppeal        1.122079  4        1.014502
## AcidIndex          1.032498  1        1.016119
## STARS              1.145438  3        1.022889
```

Diagnostic Plots



Model 5 - Linear Model

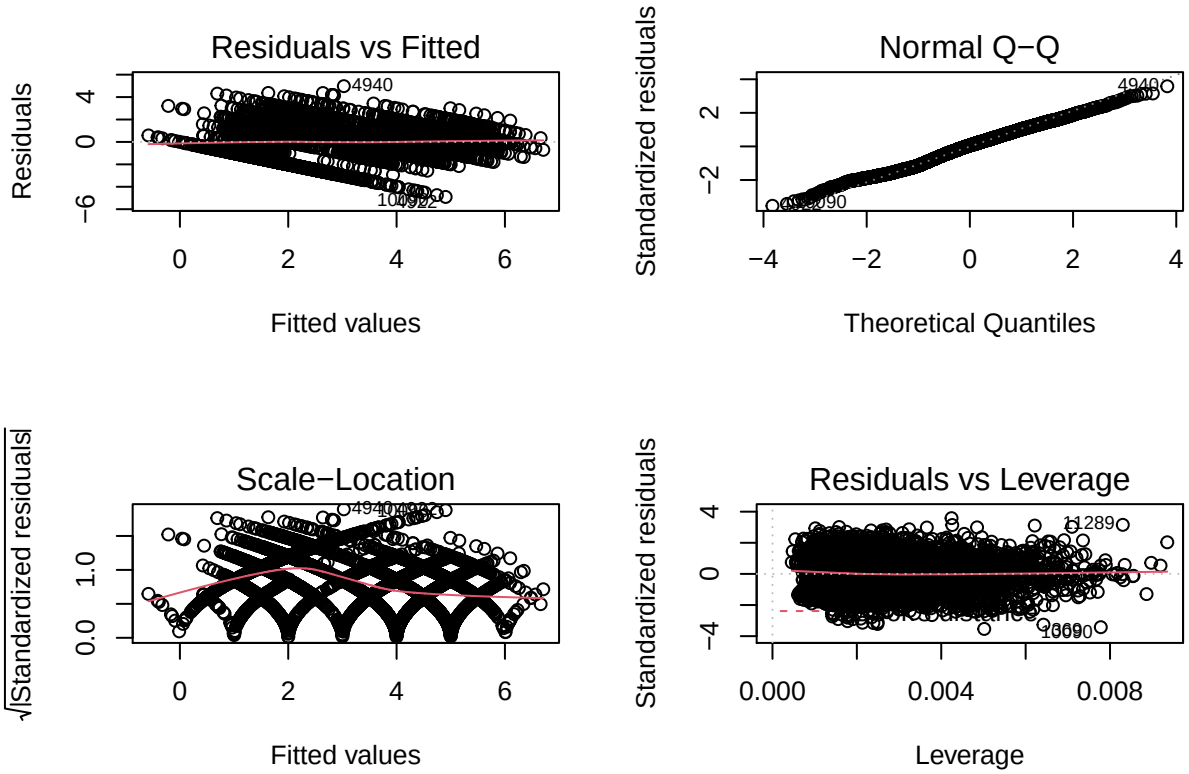
```
##
## Call:
## lm(formula = TARGET ~ ., data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.9060 -1.0057  0.0884  0.9886  4.9735
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          6.112e+00  6.279e-01   9.733  < 2e-16 ***
## FixedAcidity         8.678e-05  2.578e-03   0.034  0.97315
## VolatileAcidity     -1.129e-01  2.049e-02  -5.508 3.75e-08 ***
## CitricAcid           4.155e-02  1.833e-02   2.267  0.02342 *
## ResidualSugar        1.685e-04  4.792e-04   0.352  0.72513
## Chlorides           -1.581e-01  5.099e-02  -3.101  0.00194 **
## FreeSulfurDioxide    2.517e-04  1.103e-04   2.282  0.02255 *
## TotalSulfurDioxide   2.199e-04  7.022e-05   3.132  0.00174 **
## Density             -3.917e-02  5.965e-01  -0.066  0.94765
## pH                  -4.315e-02  2.392e-02  -1.804  0.07123 .
## Sulphates           -5.669e-02  1.775e-02  -3.193  0.00141 **
```

```
## Alcohol             9.225e-03  4.392e-03   2.101  0.03571 *
## LabelAppeal.L       1.422e+00  7.521e-02  18.901  < 2e-16 ***
## LabelAppeal.Q       1.029e-01  6.336e-02   1.625  0.10428
## LabelAppeal.C       1.631e-02  4.616e-02   0.353  0.72383
## LabelAppeal^4       4.176e-02  3.098e-02   1.348  0.17767
## AcidIndex          -2.581e-01  1.242e-02 -20.778  < 2e-16 ***
## STARS.L             2.130e+00  5.447e-02  39.107  < 2e-16 ***
## STARS.Q            -5.500e-01  4.557e-02 -12.071  < 2e-16 ***
## STARS.C             2.692e-01  3.688e-02   7.299 3.20e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.391 on 7657 degrees of freedom
## Multiple R-squared:  0.4837, Adjusted R-squared:  0.4824
## F-statistic: 377.5 on 19 and 7657 DF,  p-value: < 2.2e-16
```

Multicollinearity Check

```
##                        GVIF Df GVIF^(1/(2*Df))
## FixedAcidity       1.035946  1        1.017814
## VolatileAcidity    1.007843  1        1.003914
## CitricAcid         1.006273  1        1.003132
## ResidualSugar      1.004050  1        1.002023
## Chlorides          1.004250  1        1.002123
## FreeSulfurDioxide  1.005333  1        1.002663
## TotalSulfurDioxide 1.005081  1        1.002537
## Density            1.004764  1        1.002379
## pH                 1.006319  1        1.003155
## Sulphates          1.004022  1        1.002009
## Alcohol            1.010037  1        1.005006
## LabelAppeal        1.114739  4        1.013670
## AcidIndex          1.077774  1        1.038159
## STARS              1.149893  3        1.023551
```

Diagnostic Plots



Model 6 - Backward Selection Linear Model

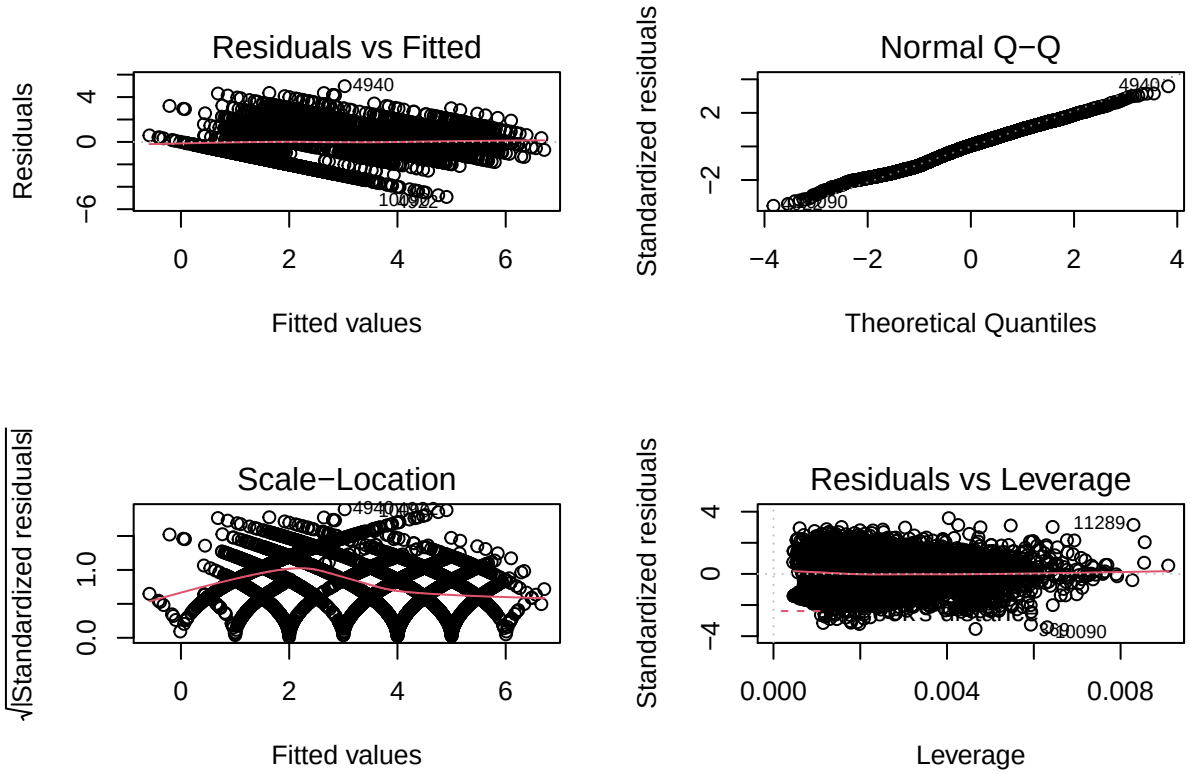
```
##
## Call:
## lm(formula = TARGET ~ VolatileAcidity + CitricAcid + Chlorides +
##     FreeSulfurDioxide + TotalSulfurDioxide + pH + Sulphates +
##     Alcohol + LabelAppeal + AcidIndex + STARS, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.9066 -1.0048  0.0836  0.9886  4.9747
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          6.097e+00  2.091e-01  29.163  < 2e-16 ***
## VolatileAcidity     -1.129e-01  2.048e-02  -5.512 3.66e-08 ***
## CitricAcid           4.149e-02  1.832e-02   2.264  0.02357 *
## Chlorides           -1.581e-01  5.096e-02  -3.103  0.00193 **
## FreeSulfurDioxide    2.525e-04  1.103e-04   2.290  0.02207 *
## TotalSulfurDioxide   2.203e-04  7.016e-05   3.141  0.00169 **
## pH                  -4.302e-02  2.391e-02  -1.799  0.07199 .
## Sulphates           -5.672e-02  1.774e-02  -3.197  0.00139 **
## Alcohol              9.182e-03  4.389e-03   2.092  0.03647 *
```

```
## LabelAppeal.L       1.422e+00  7.519e-02  18.911  < 2e-16 ***
## LabelAppeal.Q       1.032e-01  6.334e-02   1.629  0.10329
## LabelAppeal.C       1.635e-02  4.615e-02   0.354  0.72313
## LabelAppeal^4       4.165e-02  3.097e-02   1.345  0.17868
## AcidIndex          -2.580e-01  1.222e-02 -21.116  < 2e-16 ***
## STARS.L             2.130e+00  5.444e-02  39.128  < 2e-16 ***
## STARS.Q            -5.501e-01  4.556e-02 -12.076  < 2e-16 ***
## STARS.C             2.694e-01  3.686e-02   7.309 2.96e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.391 on 7660 degrees of freedom
## Multiple R-squared:  0.4837, Adjusted R-squared:  0.4826
## F-statistic: 448.5 on 16 and 7660 DF,  p-value: < 2.2e-16
```

Multicollinearity Check

```
##                       GVIF Df GVIF^(1/(2*Df))
## VolatileAcidity    1.007539  1        1.003762
## CitricAcid         1.006039  1        1.003015
## Chlorides          1.003736  1        1.001866
## FreeSulfurDioxide  1.004445  1        1.002220
## TotalSulfurDioxide 1.003658  1        1.001828
## pH                 1.006015  1        1.003003
## Sulphates          1.002642  1        1.001320
## Alcohol            1.009171  1        1.004575
## LabelAppeal        1.113285  4        1.013505
## AcidIndex          1.043535  1        1.021536
## STARS              1.147744  3        1.023232
```

Diagnostic Plots



SELECT MODELS

Identifying Best Model

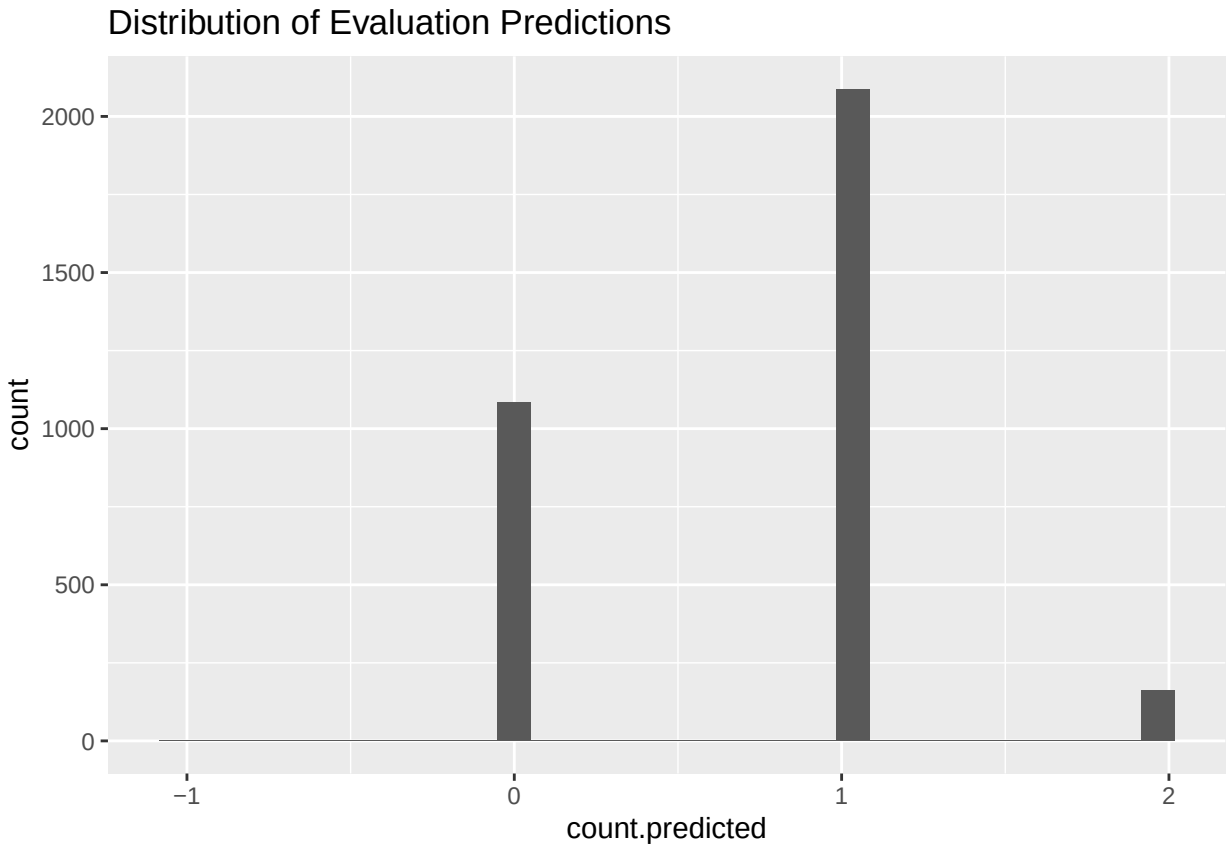
The table below summarizes each of the models performance through the metrics: AIC, BIC, RMSE, R^2, and MAE. The testing data set aside before modeling was used to evaluate model performance to simulate unknown data to the models. The best results are yielded by the linear regression models, but these models are unreliable due to underlying assumptions no being met: the underlying distribution is non-normal, the data is not continuous, and the model residuals appear to foolow a pattern. For this reason I will use the next best set of models for model evaluation. More specifically I will elect to use **Model 2 - Backward Selection Poisson** for predicting evaluation dataset. Model 2 had the highest AIC, BIC, RMSE, and MAE compared to valid / non-bias models. Model 2 was also about equal to other models R^2 values.

	AIC	BIC	RMSE	Rsquared	MAE
Model 1 - Poisson.full	28377.25	28516.17	2.594791	0.4625771	2.268414
Model 2 - Poisson.reduced	28370.91	28482.05	2.594937	0.4618161	2.268399
Model 3 - NegBinom.full	28379.42	28525.29	2.594791	0.4625765	2.268414
Model 4 - NegBinom.reduced	28373.09	28491.17	2.594937	0.4618155	2.268399
Model 5 - Linear.full	26875.42	27021.28	1.392243	0.4721820	1.133501
Model 6 - Linear.reduced	26869.55	26994.58	1.392292	0.4721475	1.133584

Predictions on Evaluation Dataset using Model 2 - Backward Selection Poisson

As explained above, Model 2 is likely to be the best at predicting outside data. The evaluation data underwent the same data preprocessing steps before generating model predictions. Below are the predictions to the evaluation data.

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
## [1] "First 5 rows of predictions:"
```

```
##   prediction count.predicted
## 1  0.4525267               0
## 2  1.0781398               1
## 3  0.3499257               0
## 4  0.2967865               0
## 5  0.2845573               0
## 6  1.3825991               1
```

Appendix

```
library(tidyverse)
library(skimr)
```

```
library(corrplot)
library(caret)
library(RANN)
library(bnstruct)
library(MASS)
library(pscl)
library(car)
library(Metrics)

train.data <- read.csv("wine-training-data.csv")
evaluation.data <-read.csv("wine-evaluation-data.csv")

train.data <- train.data[,-1]



# Introduction



# DATA EXPLORATION

## Summary Statistics


summary(train.data)




## Variable Distributions


train.data %>%
  gather(key = variable, value = value) %>%
  ggplot(., aes(x = value)) +
  geom_histogram(aes(x=value, y = ..density..), bins = 30, fill="#69b3a2", color="#e9ecef") +
  geom_density(aes(x=value), color='red', lwd = 1.75) +
  facet_wrap(~variable, scales ="free", ncol = 4)



## Boxplots


# Create Boxplots
train.data %>%
  gather(key = variable, value = value) %>%
  ggplot(., aes(x = value)) +
  geom_boxplot(aes(x=variable, y = value)) +
  facet_wrap(~variable, scales ="free", ncol = 4)
```

```r
## Correlation Matrix to assess Mulicollinearity


M <- cor(train.data, use = 'pairwise.complete.obs')
corrplot(M, method = 'color', type = 'lower',col= colorRampPalette(c("#FF0000","#FDF6D0", "#0300FF"))(1


# DATA PREPARATION

## Removed Unnecessary Variables



STARS <- c("1","2","3","4")
train.data$STARS <- factor(train.data$STARS, levels = STARS, ordered = T)

LABEL <- c("-2","-1","0","1", "2")
train.data$LabelAppeal <- factor(train.data$LabelAppeal, levels = LABEL, ordered = T)


## Handle Missing Values

preProcValues <- preProcess(train.data[,c("ResidualSugar", "Chlorides", "FreeSulfurDioxide", "TotalSulfu
                            method = c("knnImpute"),
                            k = 10,
                            knnSummary = mean)

imputed.train.data <- predict(preProcValues, train.data,na.action = na.pass)

imputed.train.data$STARS[is.na(imputed.train.data$STARS)] <- "1"



procNames <- data.frame(col = names(preProcValues$mean), mean = preProcValues$mean, sd = preProcValues$s
for(i in procNames$col){
 imputed.train.data[i] <- imputed.train.data[i]*preProcValues$std[i]+preProcValues$mean[i]
}



## Adjusting Invalid Negative Values


imputed.train.data$FixedAcidity <- imputed.train.data$FixedAcidity + abs(min(imputed.train.data$FixedAci

imputed.train.data$VolatileAcidity <- imputed.train.data$VolatileAcidity + abs(min(imputed.train.data$Vo

imputed.train.data$CitricAcid <- imputed.train.data$CitricAcid + abs(min(imputed.train.data$CitricAcid))

imputed.train.data$ResidualSugar <- imputed.train.data$ResidualSugar + abs(min(imputed.train.data$Residu
```

```
imputed.train.data$Chlorides <- imputed.train.data$Chlorides + abs(min(imputed.train.data$Chlorides))

imputed.train.data$FreeSulfurDioxide <- imputed.train.data$FreeSulfurDioxide + abs(min(imputed.train.da

imputed.train.data$TotalSulfurDioxide <- imputed.train.data$TotalSulfurDioxide + abs(min(imputed.train.

imputed.train.data$Sulphates <- imputed.train.data$Sulphates + abs(min(imputed.train.data$Sulphates))



## New Data Distributions


imputed.train.data %>%
  gather(-c(STARS, LabelAppeal), key = variable, value = value) %>%
  ggplot(., aes(x = value)) +
  geom_histogram(aes(x=value, y = ..density..), bins = 30, fill="#69b3a2", color="#e9ecef") +
  geom_density(aes(x=value), color='red', lwd = 1.75) +
  facet_wrap(~variable, scales ="free", ncol = 4)



print("STARS:")
summary(imputed.train.data$STARS)
print("LabelAppeal")
summary(imputed.train.data$LabelAppeal)


# BUILD MODELS

## Create Train Test Split



set.seed(861)
split <- sample(1:nrow(imputed.train.data), .6*nrow(imputed.train.data))

train <- imputed.train.data[split,]
test <- imputed.train.data[-split,]



## Model 1 - Poisson with all Variables


m1 <- glm(TARGET ~ . , data=train, family="poisson")

summary(m1)


### Multicollinearity Check
```

```r
vif(m1)
```

```r
### Diagnostic Plots
```

```r
par(mfrow=c(2,2))
plot(m1)
```

```r
## Model 2 - Backward Selection Poisson
```

```r
all.variables <- glm(TARGET ~ . , data=train, family="poisson")
m2 <- step(all.variables, direction = "backward", trace = 0)
summary(m2)
```

```r
### Multicollinearity Check
```

```r
vif(m2)
```

```r
### Diagnostic Plots
```

```r
par(mfrow=c(2,2))
plot(m2)
```

```r
## Model 3 - Negative Binomial
```

```r
m3 <- glm.nb(TARGET ~ . , data=train)
summary(m3)
```

```r
### Multicollinearity Check
```

```r
vif(m3)
```

```r
### Diagnostic Plots

par(mfrow=c(2,2))
plot(m3)


## Model 4 - Backward Selecion Negative Binomial

all.variables <- glm.nb(TARGET ~ . , data=train)
m4 <- step(all.variables, direction = "backward", trace = 0)
summary(m4)

### Multicollinearity Check

vif(m4)

### Diagnostic Plots

par(mfrow=c(2,2))
plot(m4)


## Model 5 - Linear Model

m5 <- lm(TARGET~., data = train)

summary(m5)

### Multicollinearity Check

vif(m5)

### Diagnostic Plots

par(mfrow=c(2,2))
```

```
plot(m5)


## Model 6 - Backward Selection Linear Model



all.variables <- lm(TARGET ~ . , data=train)

m6 <- step(all.variables, direction = "backward", trace = 0)

summary(m6)


### Multicollinearity Check


vif(m6)


### Diagnostic Plots

par(mfrow=c(2,2))

plot(m6)



# SELECT MODELS

## Identifying Best Model


models <- c("Model 1 - Poisson.full","Model 2 - Poisson.reduced", "Model 3 - NegBinom.full", "Model 4 -

get_evaluation_metrics <- function(model){

  # get predictions
  predictions <- predict(model, newdata=test)
  # store train and test TARGET variable in
  results <- data.frame(obs = test$TARGET, pred=predictions)

  # Get Metrics
  AIC <- AIC(model)
  BIC <- BIC(model)
  RMSE <- rmse(test$TARGET, predictions)
  Rsquared <- cor(test$TARGET, predictions)^2
  MAE <- mae(test$TARGET,predictions)

  evaluation <- cbind(AIC,BIC,RMSE,Rsquared,MAE)

  return(evaluation)
```

```
}




model.evaluations <- rbind(get_evaluation_metrics(m1),
get_evaluation_metrics(m2),
get_evaluation_metrics(m3),
get_evaluation_metrics(m4),
get_evaluation_metrics(m5),
get_evaluation_metrics(m6))

rownames(model.evaluations) <- models

kableExtra::kable(model.evaluations)

# preprocess

STARS <- c("1","2","3","4")
evaluation.data$STARS <- factor(evaluation.data$STARS, levels = STARS, ordered = T)

LABEL <- c("-2","-1","0","1", "2")
evaluation.data$LabelAppeal <- factor(evaluation.data$LabelAppeal, levels = LABEL, ordered = T)


preProcValues1 <- preProcess(evaluation.data[,c("ResidualSugar", "Chlorides", "FreeSulfurDioxide", "Tota
                             method = c("knnImpute"),
                             k = 10,
                             knnSummary = mean)

imputed.evaluation.data <- predict(preProcValues1, evaluation.data,na.action = na.pass)

imputed.evaluation.data$STARS[is.na(imputed.evaluation.data$STARS)] <- "1"



imputed.evaluation.data$FixedAcidity <- imputed.evaluation.data$FixedAcidity + abs(min(imputed.evaluatio

imputed.evaluation.data$VolatileAcidity <- imputed.evaluation.data$VolatileAcidity + abs(min(imputed.eva

imputed.evaluation.data$CitricAcid <- imputed.evaluation.data$CitricAcid + abs(min(imputed.evaluation.da

imputed.evaluation.data$ResidualSugar <- imputed.evaluation.data$ResidualSugar + abs(min(imputed.evalua

imputed.evaluation.data$Chlorides <- imputed.evaluation.data$Chlorides + abs(min(imputed.evaluation.data

imputed.evaluation.data$FreeSulfurDioxide <- imputed.evaluation.data$FreeSulfurDioxide + abs(min(imputed

imputed.evaluation.data$TotalSulfurDioxide <- imputed.evaluation.data$TotalSulfurDioxide + abs(min(imput

imputed.evaluation.data$Sulphates <- imputed.evaluation.data$Sulphates + abs(min(imputed.evaluation.data
```

```r
predictions.evaluation <- predict(m2, newdata=imputed.evaluation.data)

df.eval <- data.frame(prediction = predictions.evaluation,
          count.predicted = round(predictions.evaluation,0))

df.eval %>%
  ggplot(aes(x=count.predicted)) +
  geom_histogram() +
  ggtitle("Distribution of Evaluation Predictions")

print("First 5 rows of predictions:")
head(df.eval)

write.csv(clean_eval_df, 'evaluation_predictions.csv', row.names=F)
```