

Data 621 Homework 5: Wine

*Tommy Jenkins, Violeta Stoyanova, Todd Weigel, Peter Kowalchuk, Eleanor R-Secoquian,
Anthony Pagan*

12/05/2019

OVERVIEW

In this homework assignment, we will explore, analyze and model a data set containing information on approximately 12,000 commercially available wines. The variables are mostly related to the chemical properties of the wine being sold. The response 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. A large wine manufacturer is studying the data in order to predict the number of wine cases ordered based upon the wine characteristics. If the wine manufacturer can predict the number of cases, then that manufacturer will be able to adjust their wine offering to maximize sales.

Objective:

Our objective 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. HINT: Sometimes, the fact that a variable is missing is actually predictive of the target. You can only use the variables given to you (or variables that you derive from the variables provided).

Below is a short description of the variables of interest in the data set:

VARIABLE.NAME	DEFINITION
INDEX	Identification Variable (do not use)
TARGET	Number of Cases Purchased
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 customer satisfaction.
ResidualSugar	Residual Sugar of wine
STARS	Wine rating by a team of experts. 4 Stars = Excellent, 1 Star = Poor
Sulphates	Sulfate content of wine
TotalSulfurDioxide	Total Sulfur Dioxide of Wine
VolatileAcidity	Volatile Acid content of wine
pH	pH of wine

DATA EXPLORATION

Data Summary

With over 12,000 observations in our sample, we must look into the data and explore key summary statistics.

TARGET	FixedAcidity	VolatileAcidity	CitricAcid	ResidualSugar	Chlorides	FreeSulfu
Min. :0.000	Min. :-18.100	Min. :-2.7900	Min. :-3.2400	Min. :-127.800	Min. :-1.1710	Min. :-55
1st Qu.:2.000	1st Qu.: 5.200	1st Qu.: 0.1300	1st Qu.: 0.0300	1st Qu.: -2.000	1st Qu.: -0.0310	1st Qu.: 0
Median :3.000	Median : 6.900	Median : 0.2800	Median : 0.3100	Median : 3.900	Median : 0.0460	Median : 0
Mean :3.029	Mean : 7.076	Mean : 0.3241	Mean : 0.3084	Mean : 5.419	Mean : 0.0548	Mean : 3
3rd Qu.:4.000	3rd Qu.: 9.500	3rd Qu.: 0.6400	3rd Qu.: 0.5800	3rd Qu.: 15.900	3rd Qu.: 0.1530	3rd Qu.: 0
Max. :8.000	Max. : 34.400	Max. : 3.6800	Max. : 3.8600	Max. : 141.150	Max. : 1.3510	Max. : 62
NA	NA	NA	NA	NA's :616	NA's :638	NA's :647

We also calculate the counts for NA's, 0, negative, and unique values.

```
##          vars     n      mean        sd      median
## FixedAcidity    1 12795  7.075717077  6.31764346  6.900000
## VolatileAcidity 2 12795  0.324103947  0.78401424  0.280000
## CitricAcid     3 12795  0.308412661  0.86207979  0.310000
## ResidualSugar   4 12179  5.418733065 33.74937899  3.900000
## Chlorides       5 12157  0.054822489  0.31846729  0.046000
## FreeSulfurDioxide 6 12148 30.845571287 148.71455765 30.000000
## TotalSulfurDioxide 7 12113 120.714232643 231.91321051 123.000000
## Density         8 12795  0.994202718  0.02653765  0.99449
## pH              9 12400  3.207628226  0.67968708  3.200000
## Sulphates      10 11585  0.527111782  0.93212926  0.500000
## Alcohol        11 12142  10.489236260  3.72781904 10.400000
## LabelAppeal    12 12795 -0.009066041  0.89108925  0.000000
## AcidIndex      13 12795  7.772723720  1.32392637  8.000000
## STARS          14  9436  2.041754981  0.90254005  2.000000
##                  trimmed      mad      min      max      range
## FixedAcidity    7.073673928 3.261720e+00 -18.10000 34.40000 52.50000
## VolatileAcidity 0.324388981 4.299540e-01 -2.79000 3.68000 6.47000
## CitricAcid     0.310252027 4.151280e-01 -3.24000 3.86000 7.10000
## ResidualSugar   5.580041047 1.571556e+01 -127.80000 141.15000 268.95000
## Chlorides       0.054015935 1.349166e-01 -1.17100 1.35100 2.52200
## FreeSulfurDioxide 30.933487654 5.633880e+01 -555.00000 623.00000 1178.00000
## TotalSulfurDioxide 120.889536684 1.349166e+02 -823.00000 1057.00000 1880.00000
## Density         0.994213045 9.355206e-03  0.88809 1.09924 0.21115
## pH              3.205570565 3.854760e-01  0.48000 6.13000 5.65000
## Sulphates      0.527145323 4.447800e-01 -3.13000 4.24000 7.37000
## Alcohol        10.501825544 2.372160e+00 -4.70000 26.50000 31.20000
## LabelAppeal    -0.009963857 1.482600e+00 -2.00000 2.00000 4.00000
## AcidIndex      7.643157175 1.482600e+00  4.00000 17.00000 13.00000
## STARS          1.971125828 1.482600e+00  1.00000 4.00000 3.00000
##                  skew      kurtosis      se na_count neg_count
## FixedAcidity   -0.022585961 1.6749987 0.0558515162      0    1621
## VolatileAcidity 0.020379965 1.8322106 0.0069311262      0    2827
## CitricAcid    -0.050307040 1.8379401 0.0076212695      0    2966
## ResidualSugar -0.053122905 1.8846917 0.3058158360    616      NA
## Chlorides      0.030427175 1.7886044 0.0028883621    638      NA
## FreeSulfurDioxide 0.006393010 1.8364966 1.3492769213    647      NA
## TotalSulfurDioxide -0.007179351 1.6746665 2.1071702666    682      NA
## Density        -0.018693764 1.8999592 0.0002346077      0      0
## pH             0.044288014 1.6462681 0.0061037702    395      NA
## Sulphates     0.005911895 1.7525655 0.0086602040   1210      NA
## Alcohol       -0.030715836 1.5394949 0.0338306006    653      NA
## LabelAppeal   0.008429457 -0.2622916 0.0078777294      0    3640
```

```

## AcidIndex      1.648495945 5.1900925 0.0117042526      0      0
## STARS         0.447235292 -0.6925343 0.0092912151    3359     NA
##               zero_count unique_count
## FixedAcidity   548        470
## VolatileAcidity 9982       815
## CitricAcid    9686       602
## ResidualSugar NA          2078
## Chlorides      NA          1664
## FreeSulfurDioxide NA         1000
## TotalSulfurDioxide NA         1371
## Density        9492       5933
## pH              NA          498
## Sulphates      NA          631
## Alcohol         NA          402
## LabelAppeal    5617        5
## AcidIndex      0            14
## STARS          NA          5

```

The dataset consists of two data files: training and evaluation. The training dataset contains 16 columns, and the evaluation dataset also contains 16 columns.

Missing Data

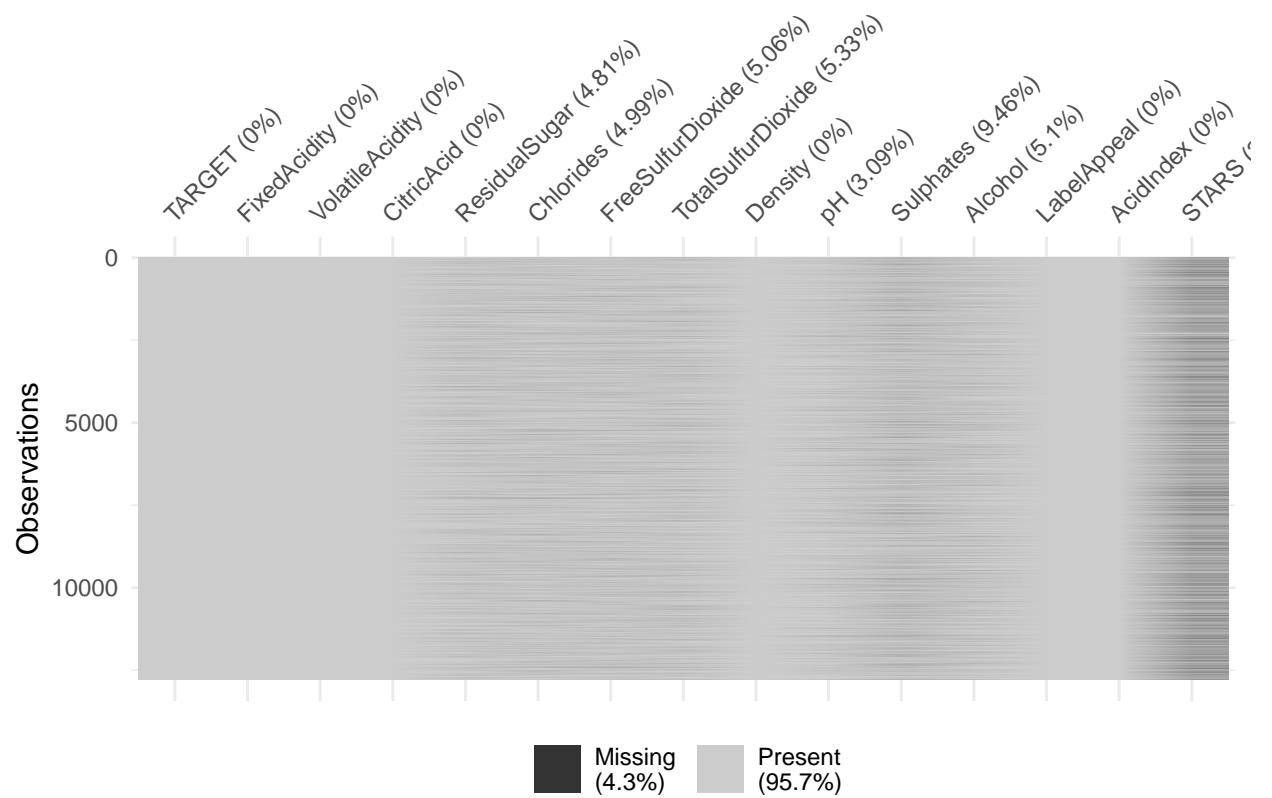
An important aspect of any dataset is to determine how much, if any, data is missing. We look at all the variables to see which if any have missing data. We look at the basic descriptive statistics as well as the missing data and percentages.

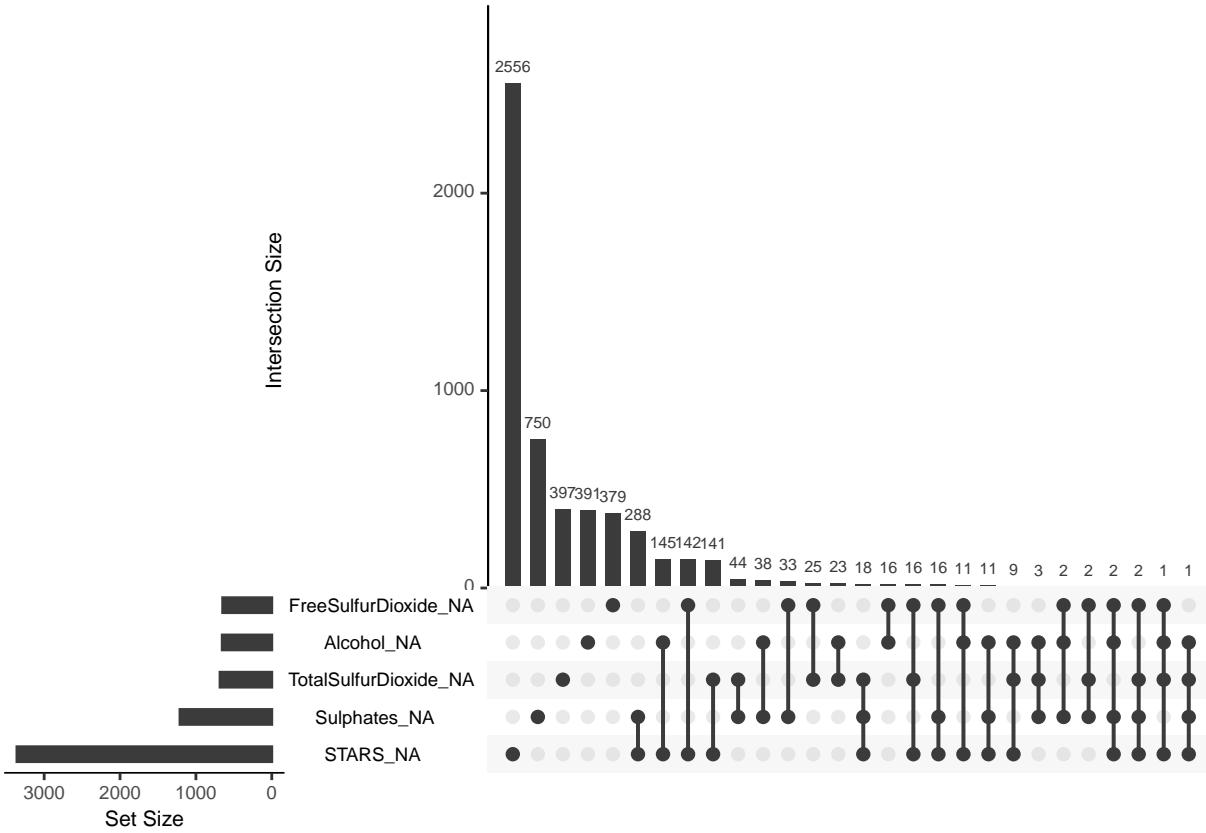
We start by looking at the dataset as a whole and determine how many complete rows, that is rows with data for all predictors we have.

```

##   Mode   FALSE    TRUE
## logical 6359  6436

```





With these results, if we remove all rows with incomplete rows, there will be a total of 6436 rows out of 12795, or 50% of the total dataset. We create a subset of data with complete cases if needed later in our analysis.

```
## Observations: 6,436
## Variables: 15
## $ TARGET <int> 5, 3, 6, 0, 3, 4, 5, 4, 3, 2, 3, 4, 4, 3, 4, 3, ...
## $ FixedAcidity <dbl> 7.1, 5.7, 5.5, -17.2, 6.0, -1.3, 10.0, 6.8, 5.8, ...
## $ VolatileAcidity <dbl> 2.640, 0.385, -0.220, 0.520, 0.330, 0.220, 0.230...
## $ CitricAcid <dbl> -0.88, 0.04, 0.39, 0.15, -1.06, 2.95, 0.27, -0.2...
## $ ResidualSugar <dbl> 14.80, 18.80, 1.80, -33.80, 3.00, -53.00, 14.10, ...
## $ Chlorides <dbl> 0.037, -0.425, -0.277, -0.022, 0.518, 0.541, 0.0...
## $ FreeSulfurDioxide <dbl> 214, 22, 62, 551, 5, -85, -188, -88, 87, 15, 32, ...
## $ TotalSulfurDioxide <dbl> 142, 115, 180, 65, 378, -266, 229, 508, -283, 60...
## $ Density <dbl> 0.99518, 0.99640, 0.94724, 0.99340, 0.96643, 0.9...
## $ pH <dbl> 3.12, 2.24, 3.09, 4.31, 3.55, 3.61, 3.14, 3.23, ...
## $ Sulphates <dbl> 0.48, 1.83, 0.75, 0.56, -0.86, 0.82, 0.88, 0.35, ...
## $ Alcohol <dbl> 22.0, 6.2, 12.6, 13.1, 3.9, 10.0, 11.0, 18.3, 11...
## $ LabelAppeal <int> -1, -1, 0, 1, 1, 0, 1, -1, -1, 0, 0, 1, -1, ...
## $ AcidIndex <int> 8, 6, 8, 5, 7, 8, 11, 8, 6, 7, 8, 7, 7, 8, 6, 9, ...
## $ STARS <int> 3, 1, 4, 1, 2, 3, 2, 2, 1, 1, 2, 2, 1, 3, 1, ...
```

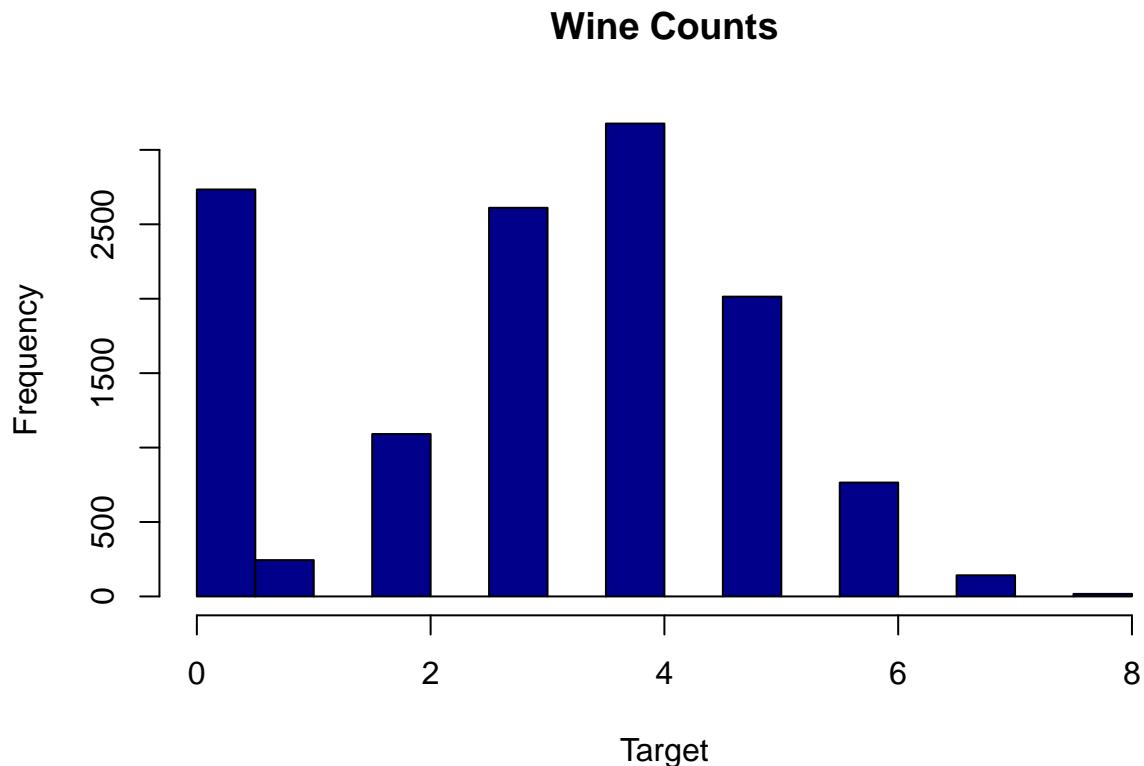
Visualization

We consider each variable

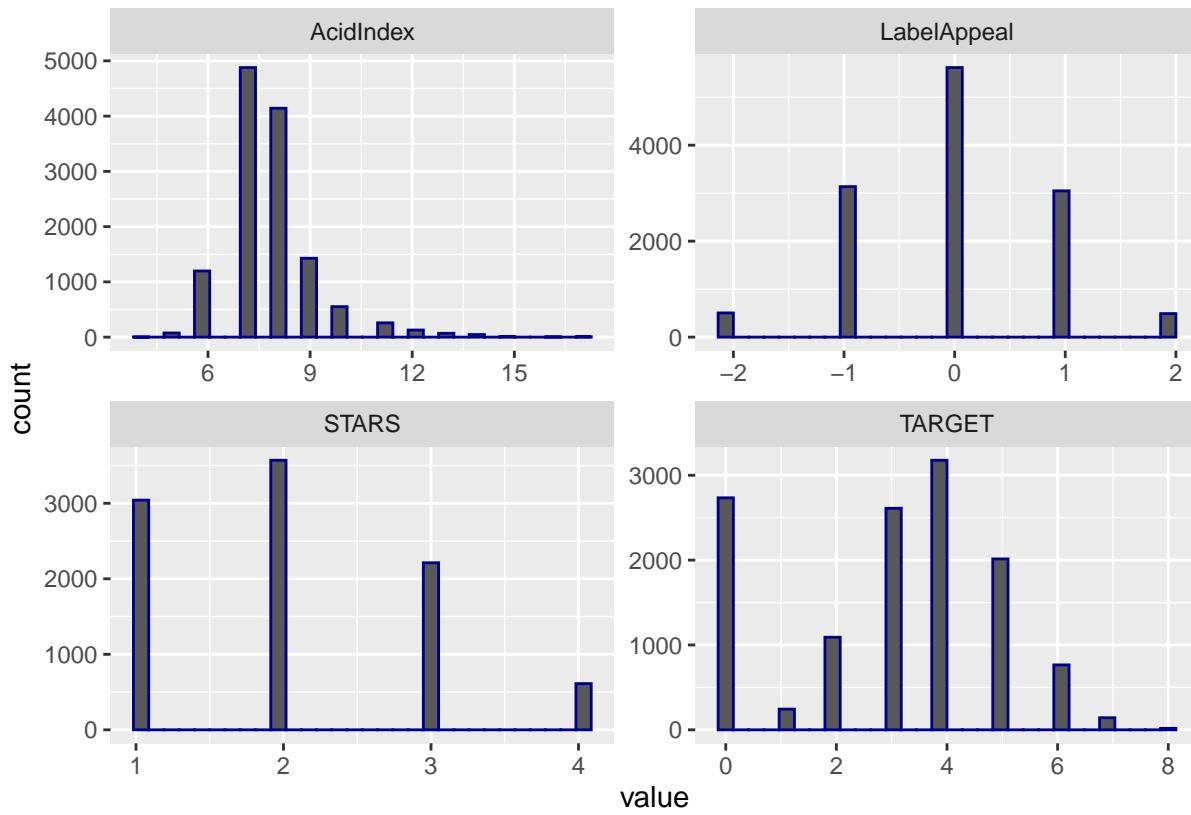
Target Variable

The distribution of our target variable is normal with the exception of many 0 Wine count entries. At such a high percentage, the zero scores likely reflect lack of popularity rather than error, especially if they get low human ratings.

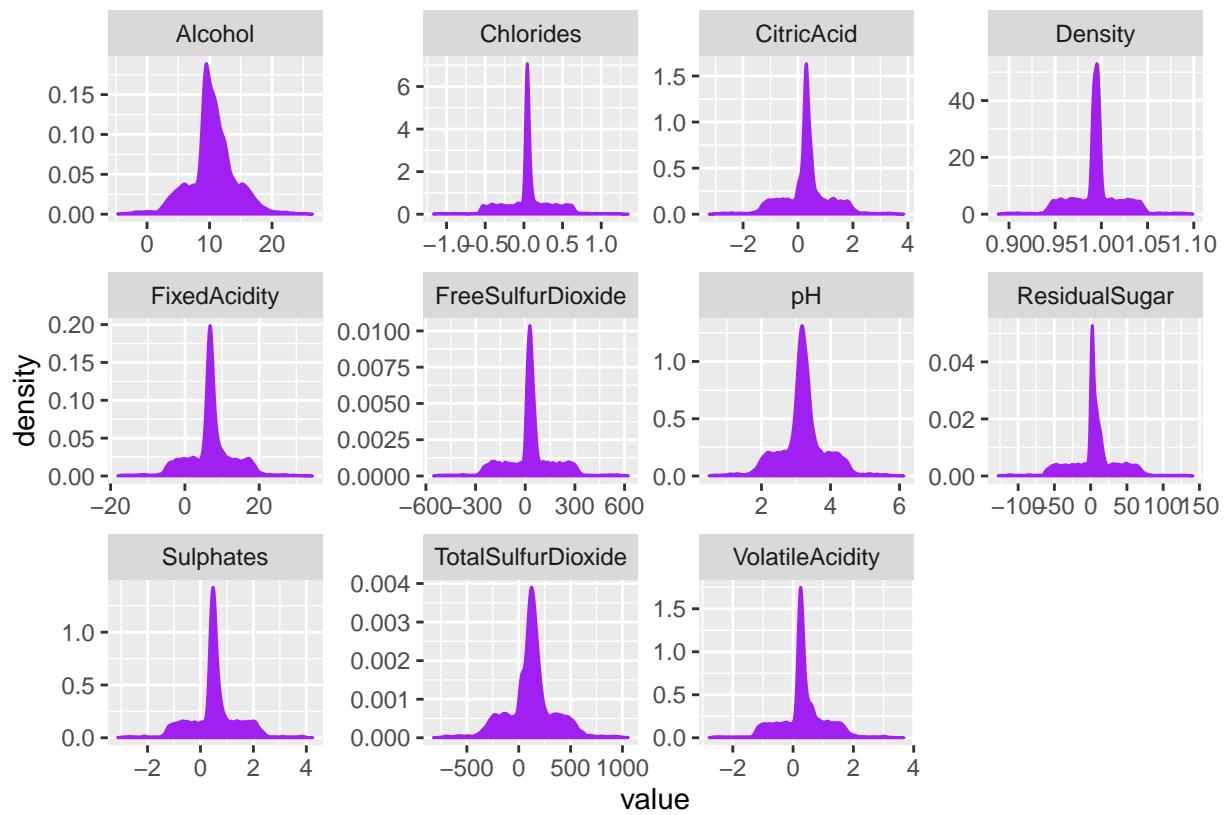
Histogram



Integers The integer variables have a small range and look normal, similar to TARGET. Stars has the least number of values and has many 0 entries. We will treat these as meaningful due to the percentage of NA's. Decision makers who buy wine are similar to the population who creates the integer variables and the range of values is small, so we choose not to impute these.

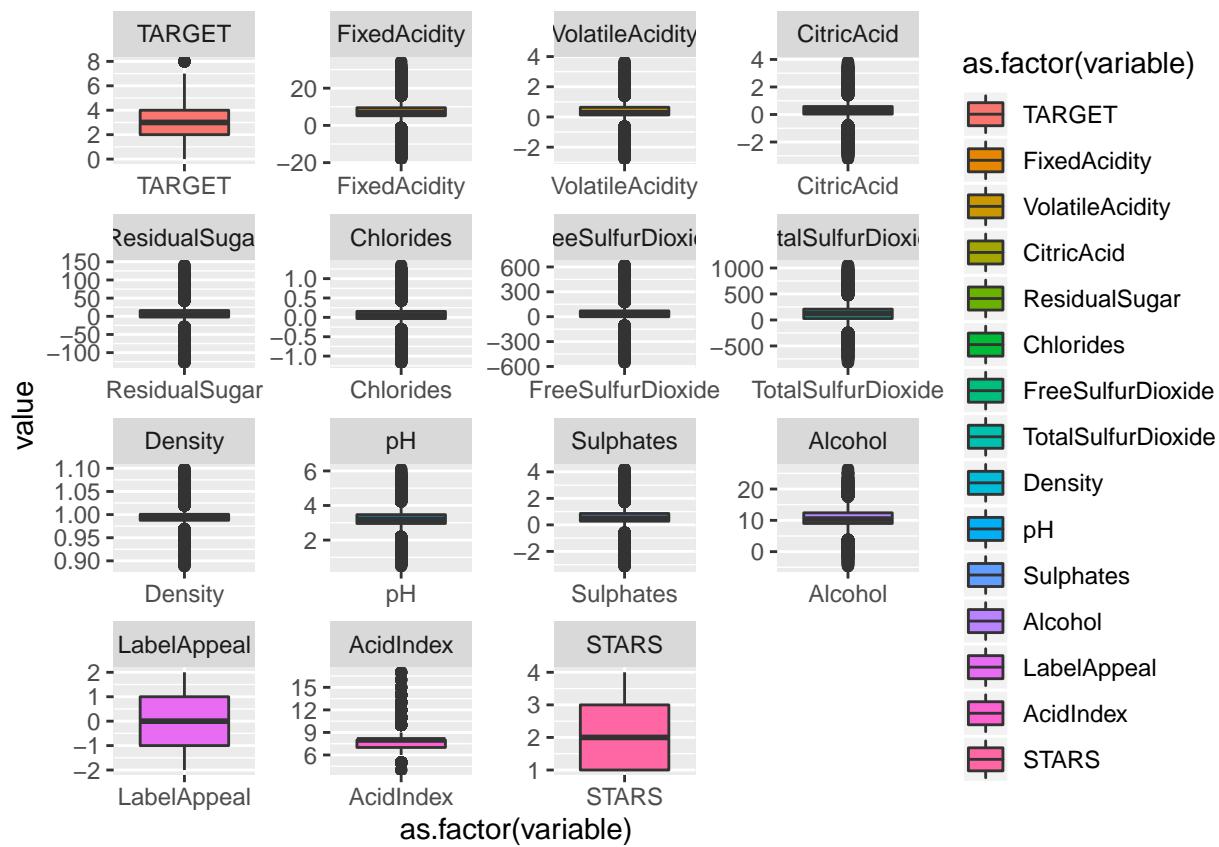


Doubles The Double variable types look very similar to one another, and look somewhat normal. These look okay to impute after we've run our diagnostic plots.

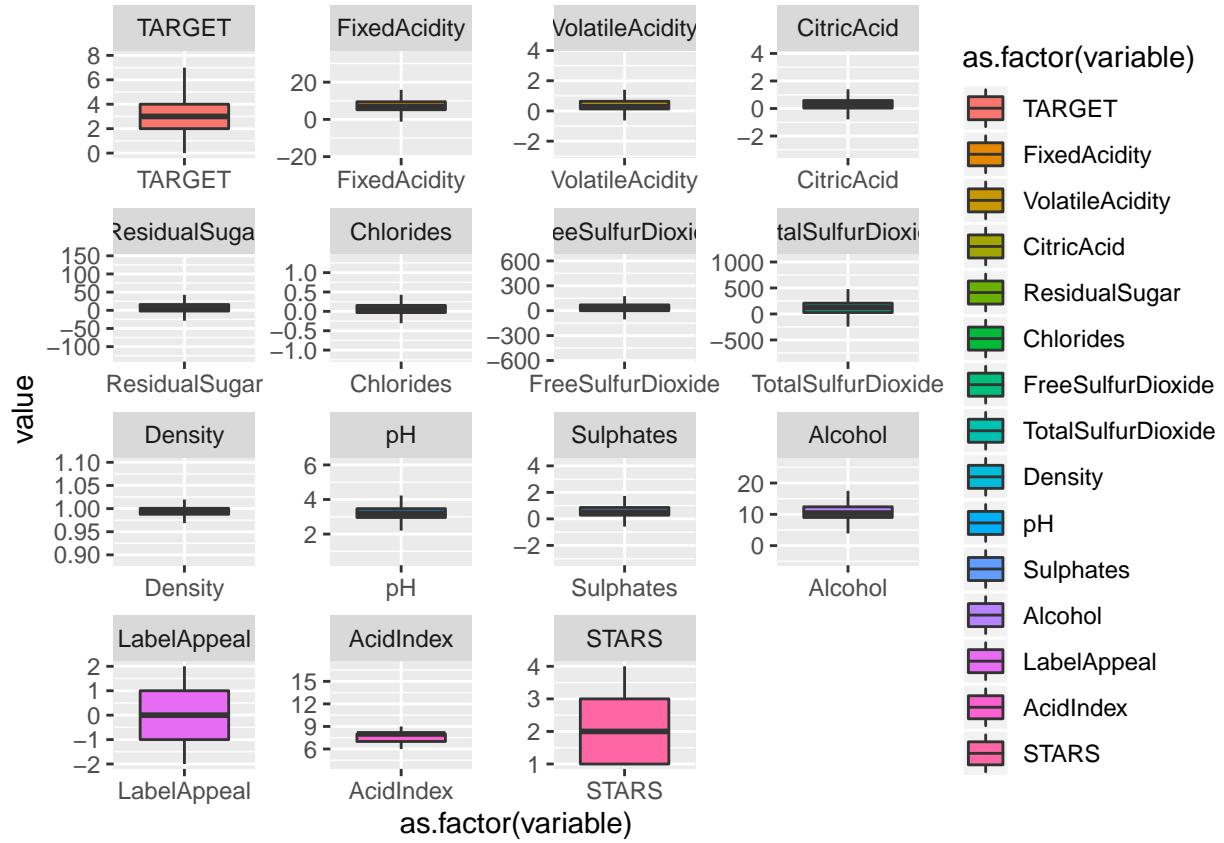


Outliers

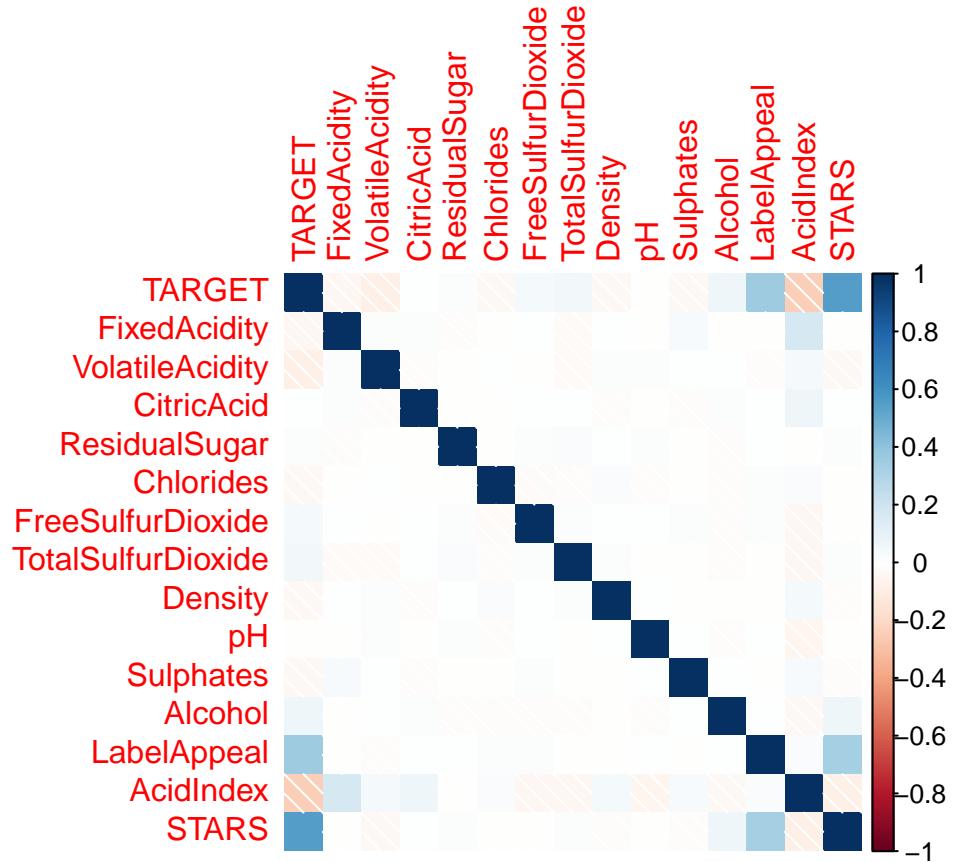
Boxplot



Boxplot Without outliers

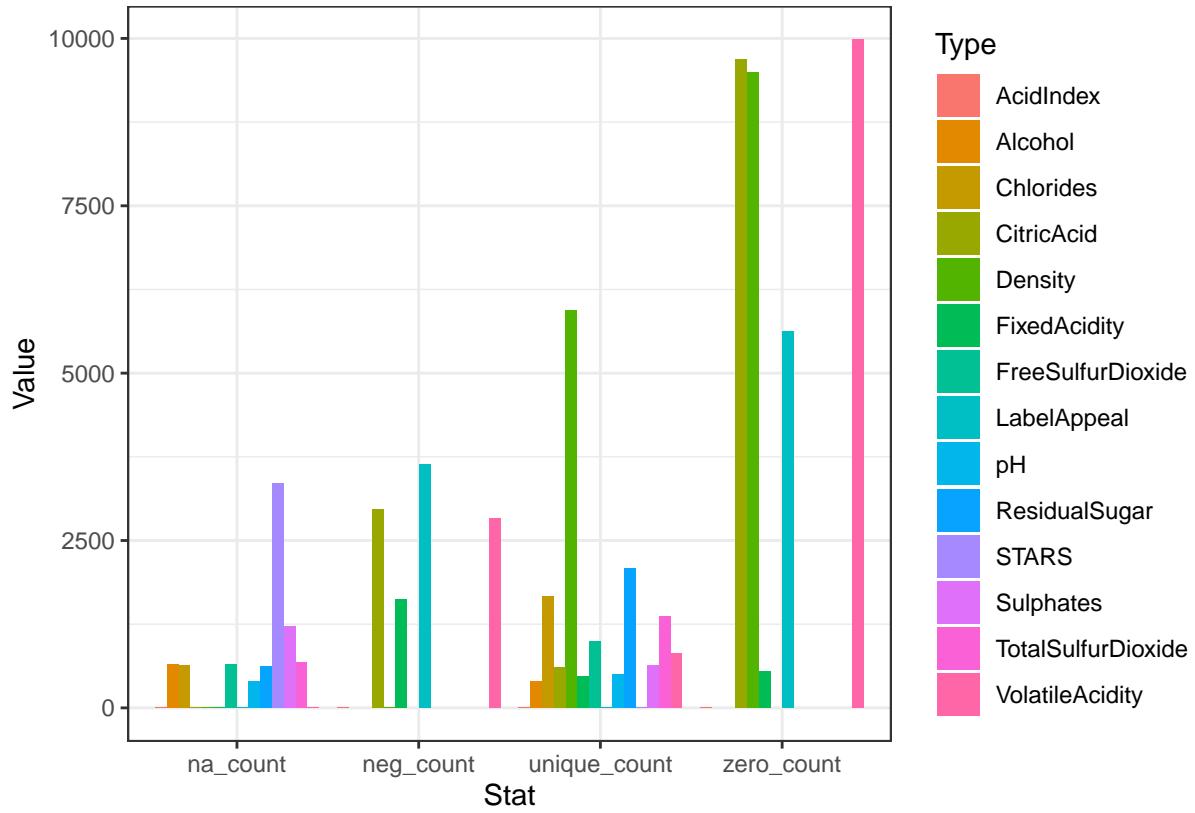


Correlation We note that the human ratings all have high correlations than do our chemical features.



Abnormal Data Finally, we can visualize data abnormalities by visualizing our previously calculated vNA, Negative, Zero, and Unique counts.

```
ab_wine_desc <- wine_desc[,c(-1:-13)]
stat_chart_data <- ab_wine_desc %>% t() %>% as.data.frame() %>% mutate(., Stat=rownames(.))
stat_chart_data %>%
  gather("Type", "Value", -Stat) %>%
  ggplot(aes(Stat, Value, fill = Type)) +
  geom_bar(position = "dodge", stat = "identity", na.rm=TRUE) +
  plot_layout(ncol = 1) +
  theme_bw()
```



DATA PREPERATION

To begin data preparation, we look at some of our abnormal data and consider transformations.

```
##          na_count neg_count zero_count unique_count
## STARS            3359       NA        NA           5
## Sulphates         1210       NA        NA          631
## TotalSulfurDioxide    682       NA        NA         1371
## Alcohol           653       NA        NA          402
## FreeSulfurDioxide    647       NA        NA         1000
## Chlorides          638       NA        NA         1664
## ResidualSugar      616       NA        NA         2078
## pH                 395       NA        NA          498
## FixedAcidity          0      1621       548         470
## VolatileAcidity        0      2827       9982        815
## CitricAcid           0      2966       9686         602
## Density              0          0      9492        5933
## LabelAppeal           0      3640      5617           5
## AcidIndex             0          0          0         14
```

NAs

We recall that STARS has a high correlation with TARGET and we see that it has $r(\text{wine1}["\text{STARS}"], \text{"na_count"}) / \text{nrow}(\text{Wine1})$ NAs and no zero's. We change NA to 0 for STARS.

The remaining NA counts include continuous variables which we can impute via a statistical method.

```

## iter imp variable
## 1 1 ResidualSugar Chlorides FreeSulfurDioxide TotalSulfurDioxide pH Sulphates Alcohol

```

Negatives

While the negative ratings make the data irregular to work with, it is unlikely that so many people ($r(wine1["STARS", "neg_count"] / nrow(WineTrain)) * 100\%$) accidentally used a negative rating. We can consider these for normalization only.

	na_count	neg_count	zero_count	unique_count
## LabelAppeal	0	3640	5617	5
## Chlorides	0	3366	12622	1663
## ResidualSugar	0	3290	147	2077
## FreeSulfurDioxide	0	3209	11	999
## CitricAcid	0	2966	9686	602
## VolatileAcidity	0	2827	9982	815
## TotalSulfurDioxide	0	2651	9	1370
## Sulphates	0	2611	9215	630
## FixedAcidity	0	1621	548	470
## Alcohol	0	122	78	401
## Density	0	0	9492	5933
## pH	0	0	58	497
## AcidIndex	0	0	0	14
## STARS	0	0	3359	5

Zeros

By the same logic we will leave the zero counts alone. We can exclude the TARGET variable unless we will be normalizing it specifically in our later analysis.

	na_count	neg_count	zero_count	unique_count
## Chlorides	0	3366	12622	1663
## VolatileAcidity	0	2827	9982	815
## CitricAcid	0	2966	9686	602
## Density	0	0	9492	5933
## Sulphates	0	2611	9215	630
## LabelAppeal	0	3640	5617	5
## STARS	0	0	3359	5
## FixedAcidity	0	1621	548	470
## ResidualSugar	0	3290	147	2077
## Alcohol	0	122	78	401
## pH	0	0	58	497
## FreeSulfurDioxide	0	3209	11	999
## TotalSulfurDioxide	0	2651	9	1370
## AcidIndex	0	0	0	14

Uniques

We want to take a look at the least unique counts next, and by a large margin LabelAppeal, STARS, and AcidIndex show low unique counts. We see that AcidIndex is a proprietary weighted method for measuring Acid, so we decide not to perform any transformation on AcidIndex.

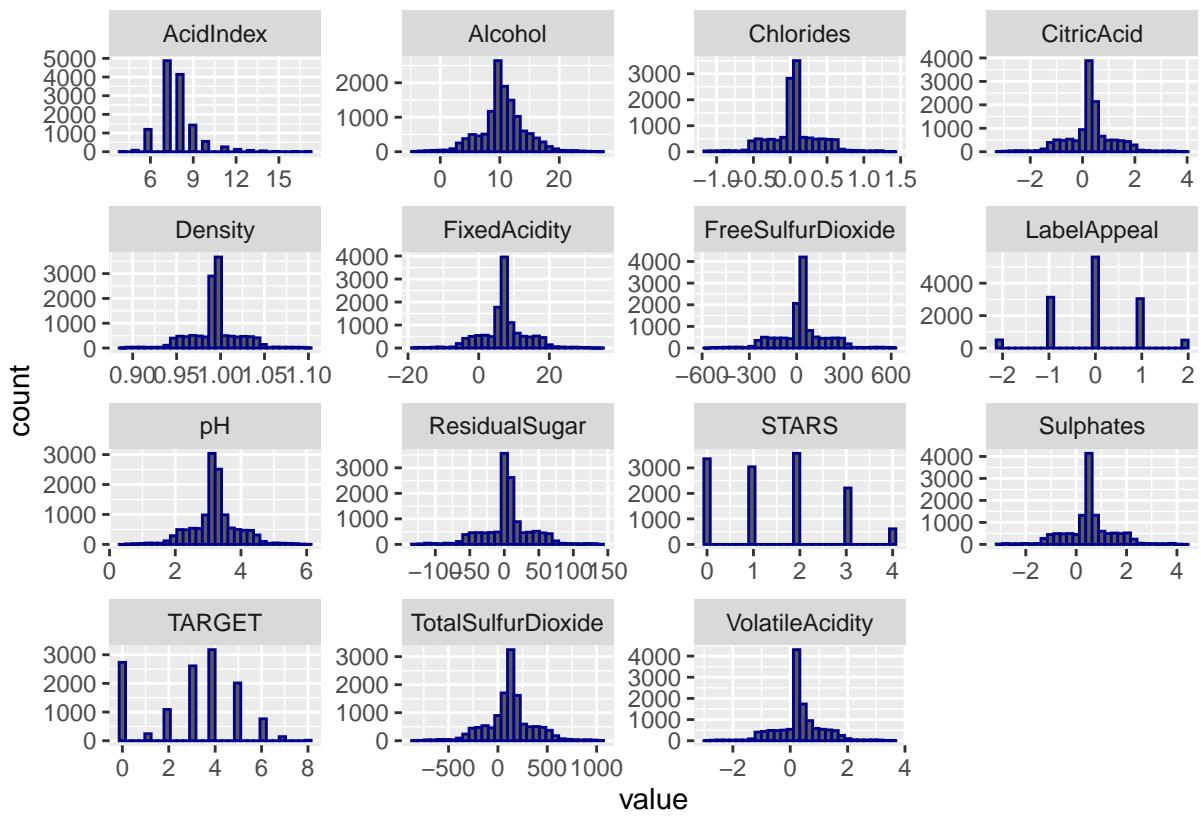
	na_count	neg_count	zero_count	unique_count
## LabelAppeal	0	3640	5617	5
## STARS	0	0	3359	5

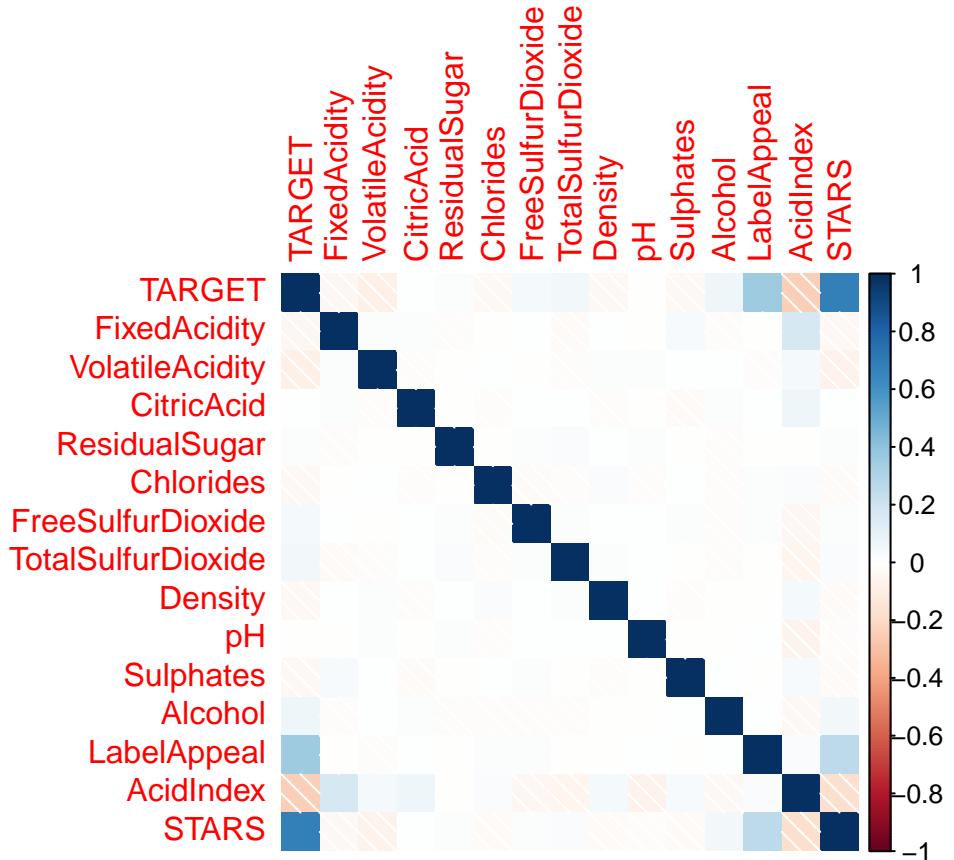
## AcidIndex	0	0	0	14
## Alcohol	0	122	78	401
## FixedAcidity	0	1621	548	470
## pH	0	0	58	497
## CitricAcid	0	2966	9686	602
## Sulphates	0	2611	9215	630
## VolatileAcidity	0	2827	9982	815
## FreeSulfurDioxide	0	3209	11	999
## TotalSulfurDioxide	0	2651	9	1370
## Chlorides	0	3366	12622	1663
## ResidualSugar	0	3290	147	2077
## Density	0	0	9492	5933

Data Finalization

We've finalized our dataset for analysis.

## 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.: -1.950	1st Qu.: -0.0310	1st Qu.: -1.25	1st Qu.: 27.0
## Median : 3.900	Median : 0.0460	Median : 30.00	Median : 123.0
## Mean : 5.396	Mean : 0.0546	Mean : 30.47	Mean : 120.2
## 3rd Qu.: 15.800	3rd Qu.: 0.1540	3rd Qu.: 70.00	3rd Qu.: 207.5
## Max. : 141.150	Max. : 1.3510	Max. : 623.00	Max. : 1057.0
## Density	pH	Sulphates	Alcohol
## Min. :0.8881	Min. :0.480	Min. :-3.1300	Min. :-4.70
## 1st Qu.:0.9877	1st Qu.:2.950	1st Qu.: 0.2800	1st Qu.: 9.00
## Median :0.9945	Median :3.200	Median : 0.5000	Median :10.40
## Mean :0.9942	Mean :3.206	Mean : 0.5271	Mean :10.48
## 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
## LabelAppeal	AcidIndex	STARS	
## Min. :-2.000000	Min. : 4.000	Min. :0.000	
## 1st Qu.:-1.000000	1st Qu.: 7.000	1st Qu.:0.000	
## Median : 0.000000	Median : 8.000	Median :1.000	
## Mean : -0.009066	Mean : 7.773	Mean :1.506	
## 3rd Qu.: 1.000000	3rd Qu.: 8.000	3rd Qu.:2.000	
## Max. : 2.000000	Max. :17.000	Max. :4.000	





BUILD MODEL

Model 1: Poisson Regression (all predictors)

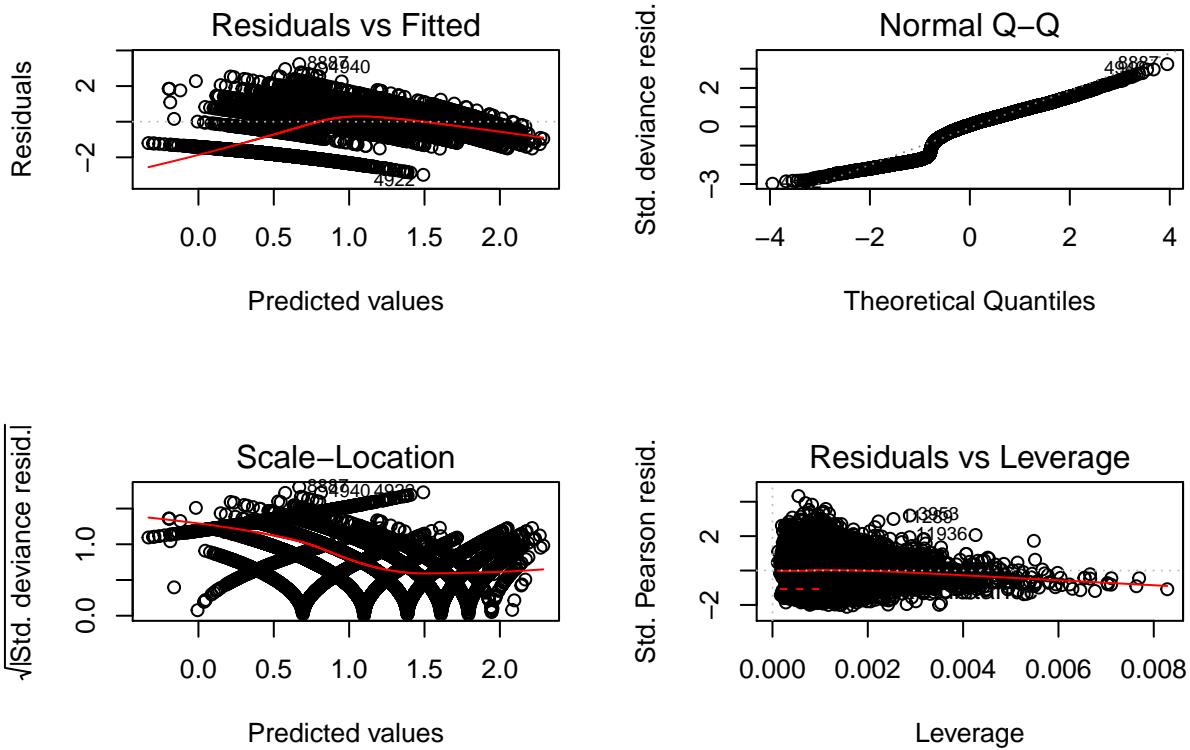
For the first model, we used the Poisson regression and all of the predictors.

```
##
## Call:
## glm(formula = TARGET ~ ., family = poisson, data = WineTrain)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.9819  -0.7234   0.0703   0.5807   3.2360
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)           1.515e+00  1.954e-01  7.755 8.84e-15 ***
## FixedAcidity        -3.217e-04  8.203e-04 -0.392 0.694948
## VolatileAcidity     -3.332e-02  6.517e-03 -5.112 3.18e-07 ***
## CitricAcid          7.587e-03  5.893e-03  1.287 0.197947
## ResidualSugar       5.369e-05  1.507e-04  0.356 0.721693
## Chlorides          -4.517e-02  1.602e-02 -2.820 0.004797 **
## FreeSulfurDioxide   1.242e-04  3.428e-05  3.624 0.000291 ***
## TotalSulfurDioxide  8.689e-05  2.215e-05  3.923 8.76e-05 ***
## Density            -2.829e-01  1.920e-01 -1.474 0.140561
## pH                -1.368e-02  7.503e-03 -1.823 0.068283 .
##
```

```

## Sulphates      -1.021e-02 5.457e-03 -1.871 0.061326 .
## Alcohol        2.349e-03 1.378e-03  1.705 0.088245 .
## LabelAppeal    1.331e-01 6.063e-03 21.955 < 2e-16 ***
## AcidIndex      -8.677e-02 4.550e-03 -19.068 < 2e-16 ***
## STARS          3.112e-01 4.531e-03  68.695 < 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: 14726 on 12780 degrees of freedom
## AIC: 46698
##
## Number of Fisher Scoring iterations: 5

```



Model 2: Poisson Regression (reduced predictors)

For the second model, based on model 1, we reduced the number of predictors.

```

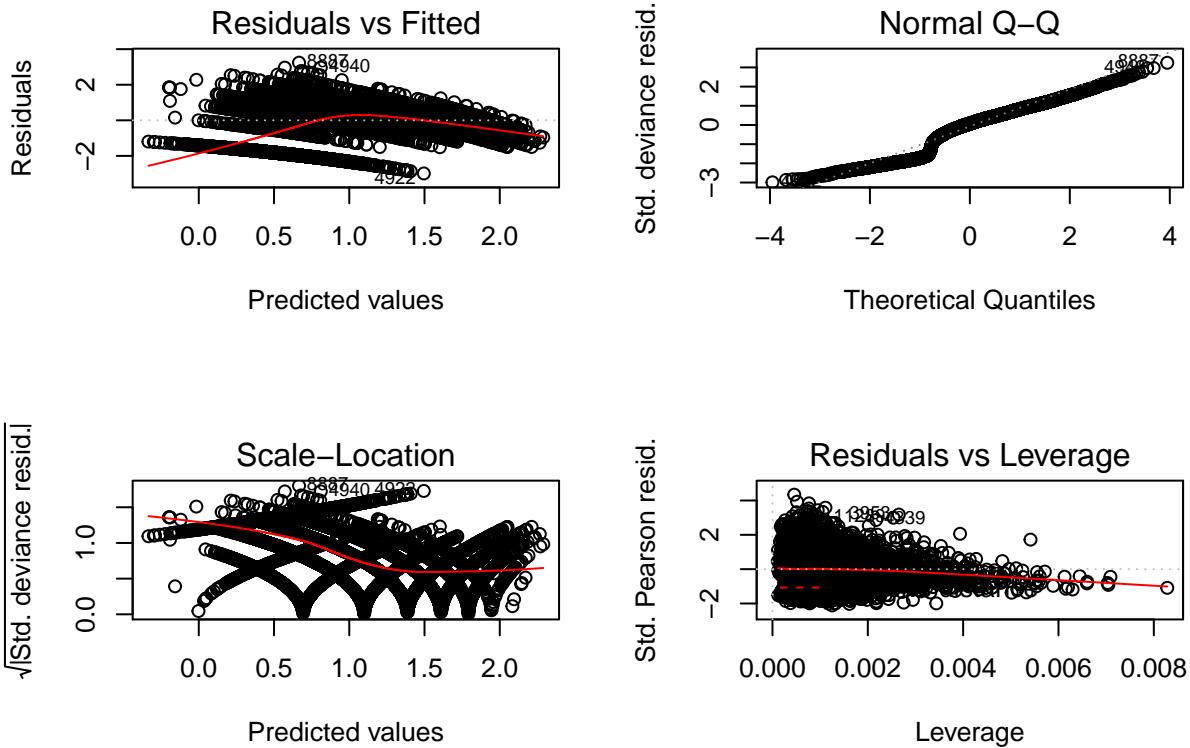
##
## Call:
## glm(formula = TARGET ~ VolatileAcidity + CitricAcid + Chlorides +
##     FreeSulfurDioxide + TotalSulfurDioxide + Density + pH + Sulphates +
##     Alcohol + LabelAppeal + AcidIndex + STARS, family = poisson,
##     data = WineTrain)
##

```

```

## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.9871  -0.7239   0.0698   0.5785   3.2414
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)           1.516e+00  1.954e-01   7.759 8.58e-15 ***
## VolatileAcidity     -3.335e-02  6.517e-03  -5.117 3.10e-07 ***
## CitricAcid          7.540e-03  5.892e-03   1.280 0.200641
## Chlorides          -4.515e-02  1.602e-02  -2.819 0.004814 **
## FreeSulfurDioxide   1.242e-04  3.428e-05   3.622 0.000292 ***
## TotalSulfurDioxide  8.716e-05  2.214e-05   3.936 8.27e-05 ***
## Density            -2.837e-01  1.920e-01  -1.478 0.139457
## pH                 -1.363e-02  7.501e-03  -1.818 0.069127 .
## Sulphates          -1.027e-02  5.456e-03  -1.882 0.059846 .
## Alcohol            2.341e-03  1.377e-03   1.699 0.089269 .
## LabelAppeal        1.331e-01  6.063e-03  21.959 < 2e-16 ***
## AcidIndex          -8.703e-02  4.503e-03 -19.328 < 2e-16 ***
## STARS              3.113e-01  4.530e-03  68.707 < 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: 14726  on 12782  degrees of freedom
## AIC: 46694
##
## Number of Fisher Scoring iterations: 5

```



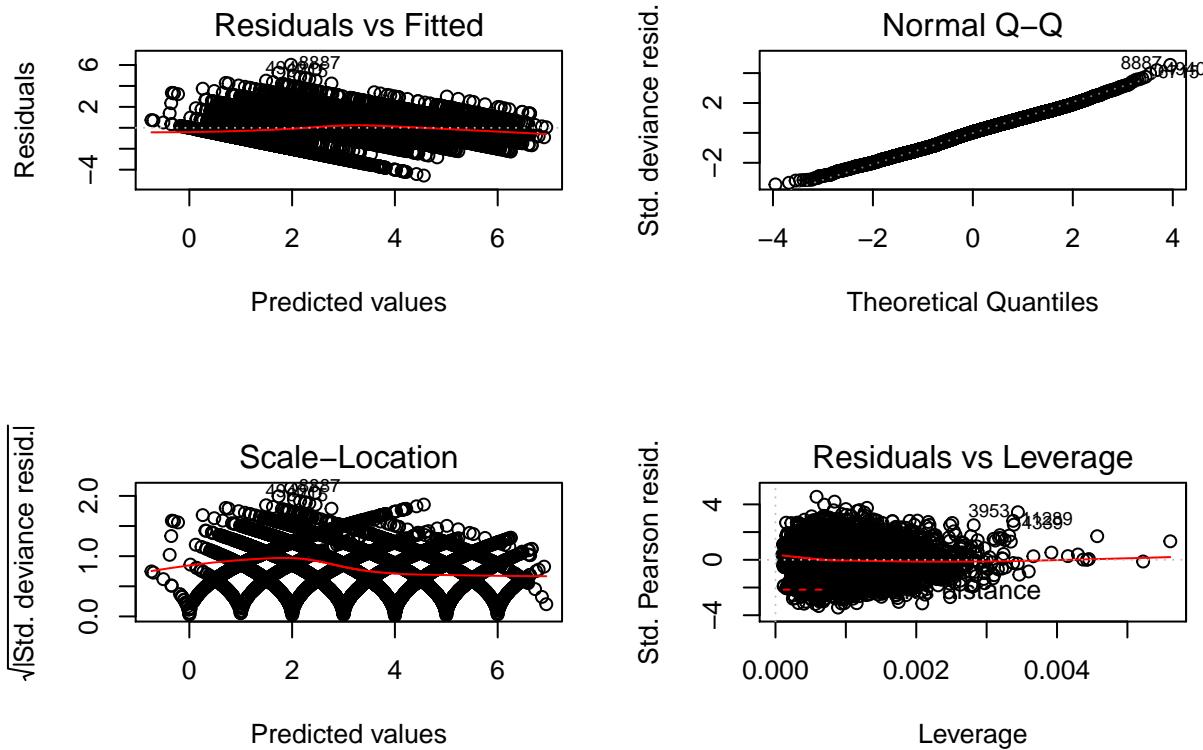
Model 3: Gaussian Regression (significant predictors)

```
##
## Call:
## glm(formula = TARGET ~ VolatileAcidity + FreeSulfurDioxide +
##       TotalSulfurDioxide + Chlorides + Density + pH + Sulphates +
##       LabelAppeal + AcidIndex + STARS, family = gaussian, data = WineTrain)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -4.5629  -0.9516   0.0638   0.9110   6.0223
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)             4.101e+00  4.470e-01   9.174 < 2e-16 ***
## VolatileAcidity        -9.901e-02 1.498e-02  -6.610 4.01e-11 ***
## FreeSulfurDioxide       3.104e-04 7.904e-05   3.927 8.63e-05 ***
## TotalSulfurDioxide     2.321e-04 5.053e-05   4.593 4.40e-06 ***
## Chlorides              -1.374e-01 3.678e-02  -3.735 0.000189 ***
## Density                -8.161e-01 4.420e-01  -1.846 0.064846 .
## pH                     -3.109e-02 1.724e-02  -1.804 0.071282 .
## Sulphates              -2.650e-02 1.255e-02  -2.111 0.034780 *
## LabelAppeal            4.320e-01 1.367e-02  31.601 < 2e-16 ***
## AcidIndex              -2.078e-01 9.055e-03 -22.947 < 2e-16 ***
## STARS                  9.789e-01 1.044e-02  93.772 < 2e-16 ***
## ---
```

```

## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 1.754353)
##
##      Null deviance: 47477  on 12794  degrees of freedom
## Residual deviance: 22428  on 12784  degrees of freedom
## AIC: 43516
##
## Number of Fisher Scoring iterations: 2

```



Model 3 shows a better Q-Q plot than the previous two models.

Model 4: Negative Binomial Regression

```

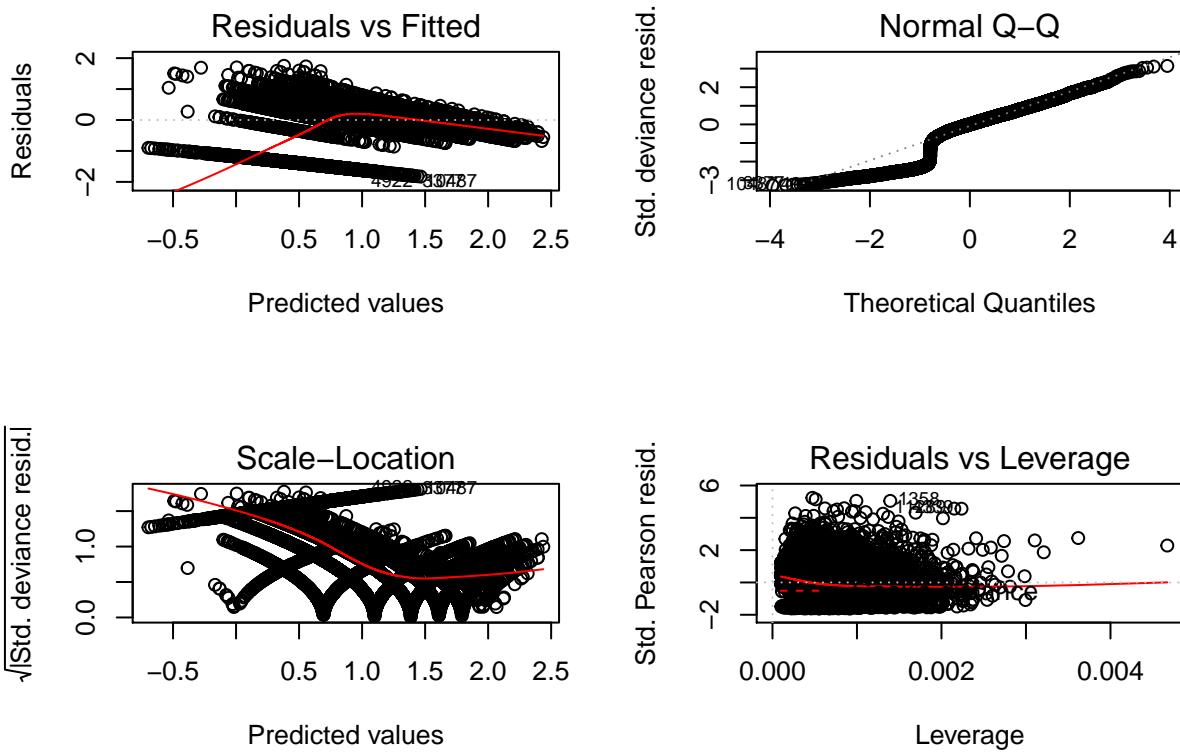
##
## Call:
## glm(formula = TARGET ~ VolatileAcidity + TotalSulfurDioxide +
##       pH + Sulphates + LabelAppeal + AcidIndex + STARS, family = negative.binomial(1),
##       data = WineTrain)
##
## Deviance Residuals:
##      Min        1Q        Median         3Q        Max
## -1.82920   -0.39470    0.00211    0.29978    1.75153
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.435e+00 4.972e-02 28.863 < 2e-16 ***

```

```

## VolatileAcidity -4.523e-02 7.455e-03 -6.067 1.34e-09 ***
## TotalSulfurDioxide 1.319e-04 2.517e-05 5.239 1.64e-07 ***
## pH -2.457e-02 8.573e-03 -2.865 0.00417 **
## Sulphates -1.563e-02 6.243e-03 -2.504 0.01230 *
## LabelAppeal 1.184e-01 6.837e-03 17.316 < 2e-16 ***
## AcidIndex -1.173e-01 4.775e-03 -24.563 < 2e-16 ***
## STARS 3.660e-01 5.170e-03 70.790 < 2e-16 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(1) family taken to be 0.3105383)
##
## Null deviance: 9042.5 on 12794 degrees of freedom
## Residual deviance: 6765.8 on 12787 degrees of freedom
## AIC: 55513
##
## Number of Fisher Scoring iterations: 6

```



SELECT MODEL

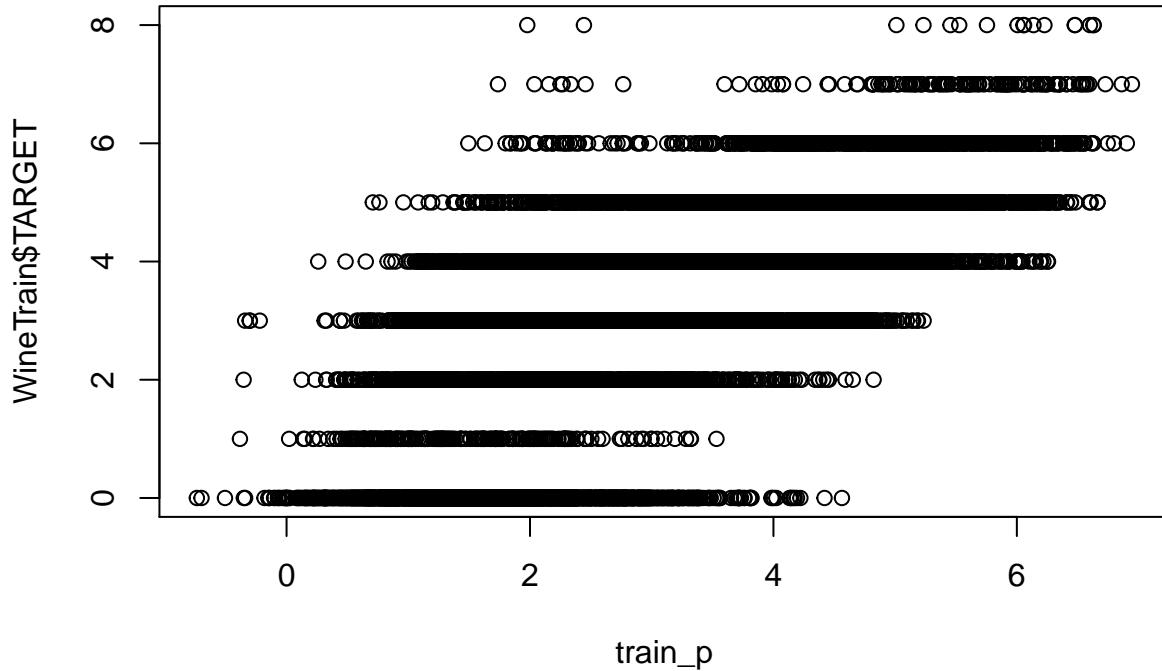
Pick the best regression model

	Model 1	Model 2	Model 3	Model 4
AIC	46697.9566169909	46694.2415354406	43515.7043085365	55513.0040360453
BIC	46809.8087632198	46791.1800621723	43605.1860255196	55572.658514034

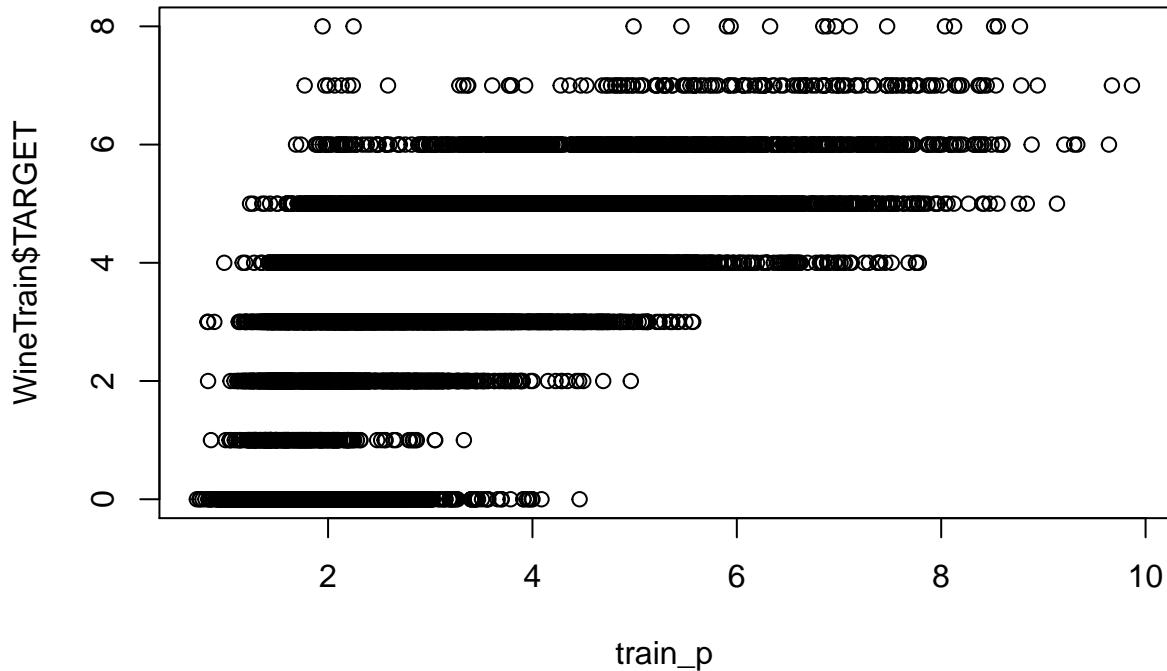
With 4 models computed, we select the model with the lowest combination of AIC and BIC. From the table, we can see the model to pick is model 3

#CONCLUSION

Model 3 showed the best result. We can observe its performance by plotting the datasets TARGET values against the predicted values. One thing we observe is that the model doesn't predict a TARGET of 8.



Other models, although of worse performance according to our selection metric, do show results of TARGET 8, but as can be seen in the graph below, they do not correspond to real TARGET 8 classifications.



APPENDIX

Code used in analysis

```

knitr:::opts_chunk$set(echo = FALSE, warning = FALSE, fig.align='center')
require(knitr)
library(MASS)
library(psych)
library(kableExtra)
library(tidyverse)
library(faraway)
library(gridExtra)
library(reshape2)
library(leaps)
library(caret)
library(naniar)
library(pander)
library(pROC)
library(corrplot)
library(jtools)
library(mice)
#devtools::install_github("thomasp85/patchwork")
library(patchwork)
WineTrain <- read.csv("https://raw.githubusercontent.com/pkowalchuk/CUNY621-HW5/master/wine-training-data.csv")
WineTrain1 <- WineTrain

```

```

WineEval <- read.csv("wine-evaluation-data.csv",na.strings="",header=TRUE)
kable_styling(kable(textbook<-data.frame(VARIABLE.NAME=c("INDEX","TARGET","","","","AcidIndex","Alcohol",""#
#glimpse(WineTrain)
#colnames(WineTrain[-1])<-"INDEX"
WineTrainVars <- WineTrain[-1]
WineTrainFeatures <- WineTrain[-c(1:2)]
kable_styling(kable(summary(WineTrainVars)))

var_stats<- function(WineTrainVars){
  wt <- WineTrainVars
  wine1 <- describe(wt)
  wine1$na_count <- sapply(wt, function(y) sum(is.na(y)))
  wine1$neg_count <- sapply(wt, function(y) sum(y<0))
  wine1$zero_count <- sapply(wt, function(y) sum(as.integer(y)==0))
  wine1$unique_count <- sapply(wt, function(y) sum(n_distinct(y)))

  return(wine1)
}
wine_desc <- var_stats(WineTrainFeatures)

wine_desc %>% as.data.frame()

colsTrain<-ncol(WineTrain)
colsEval<-ncol(WineEval)
missingCol<-colnames(WineTrain) [!(colnames(WineTrain) %in% colnames(WineEval))]
#missingCol
cc<-summary(complete.cases(WineTrainVars))
cWineTrain<-subset(WineTrainVars, complete.cases(WineTrainVars))
cc
vis_miss(WineTrainVars)
gg_miss_upset(WineTrainVars)
glimpse(cWineTrain)
#WineTrain1$INDEX <- NULL
hist(WineTrainVars$TARGET, col='darkblue', xlab = " Target ", main = "Wine Counts")

WineTrainVars %>%
  keep(is.integer) %>%
  gather() %>%
  ggplot(aes(value), main="") +
  facet_wrap(~ key, scales = "free") +
  geom_histogram(color='darkblue') +
  plot_layout(ncol = 1)
WineTrainFeatures %>%
  keep(is.double) %>%
  gather() %>%
  ggplot(aes(value)) +
  facet_wrap(~ key, scales = "free") +
  geom_density(color='purple', fill='purple') +
  plot_layout(ncol = 1)
ggplot(melt(WineTrainVars), aes(x=as.factor(variable), y=value, fill=as.factor(variable))) + facet_wrap
ggplot(melt(WineTrainVars), aes(x=as.factor(variable), y=value, fill=as.factor(variable))) + facet_wrap
corrplot(as.matrix(cor(WineTrainVars, use = "pairwise.complete")),method = "shade")
ab_wine_desc <- wine_desc[,c(-1:-13)]

```

```

stat_chart_data <- ab_wine_desc %>% t() %>% as.data.frame() %>% mutate(., Stat=rownames(.))
stat_chart_data %>%
  gather("Type", "Value", -Stat) %>%
  ggplot(aes(Stat, Value, fill = Type)) +
  geom_bar(position = "dodge", stat = "identity", na.rm=TRUE) +
  plot_layout(ncol = 1) +
  theme_bw()
WineTrainTrans <- WineTrain[-c(1)]
ab_wine_desc <- var_stats(WineTrainTrans)[-c(1),c(-1:-13)]
print(ab_wine_desc[order(-ab_wine_desc$na_count),])
WineTrainTrans$STARS <- sapply(WineTrainTrans$STARS,function(x) ifelse(is.na(x),0,x))
#WineTrain<-as.factor(WineTrain)
WineTrainTrans<-complete(mice(WineTrainTrans, m=1, maxit=1),1)

ab_wine_desc <- var_stats(WineTrainTrans)[-c(1),c(-1:-13)]
print(ab_wine_desc[order(-ab_wine_desc$neg_count),])
ab_wine_desc <- var_stats(WineTrainTrans)[-c(1),c(-1:-13)]
print(ab_wine_desc[order(-ab_wine_desc$zero_count),])
ab_wine_desc <- var_stats(WineTrainTrans)[-c(1),c(-1:-13)]
print(ab_wine_desc[order(ab_wine_desc$unique_count),])
#WineTrainTrans$STARS<-as.factor(WineTrainTrans$LabelAppeal)
#WineTrainTrans$STARS<-as.factor(WineTrainTrans$STARS)
#WineTrainTrans$STARS<-as.factor(WineTrainTrans$AcidIndex)
WineTrain<-WineTrainTrans
summary(WineTrain)

WineTrain %>%
  keep(is.numeric) %>%
  gather() %>%
  ggplot(aes(value), main="") +
  facet_wrap(~ key, scales = "free") +
  geom_histogram(color='darkblue') +
  plot_layout(ncol = 1)

corrplot(as.matrix(cor(WineTrain %>% keep(is.numeric), use = "pairwise.complete")),method = "shade")

m1 <- glm(TARGET ~ ., family = poisson, data = WineTrain)
#m1 <- glm(TARGET ~ ., family = poisson, data = WineTrain)
summary(m1)
par(mfrow = c(2,2))
plot(m1)
m2 <- glm(TARGET ~ VolatileAcidity + CitricAcid + Chlorides + FreeSulfurDioxide
           + TotalSulfurDioxide + Density + pH + Sulphates + Alcohol + LabelAppeal
           + AcidIndex + STARS, family = poisson, data = WineTrain)
summary(m2)
par(mfrow = c(2,2))
plot(m2)
m3 <- glm(TARGET ~ VolatileAcidity + FreeSulfurDioxide + TotalSulfurDioxide + Chlorides + Density + pH +
           + Sulfates + Alcohol + LabelAppeal + AcidIndex + STARS, family = poisson, data = WineTrain)
summary(m3)
par(mfrow = c(2,2))
plot(m3)

```

```

m4 <- glm(TARGET ~ VolatileAcidity + TotalSulfurDioxide +
           pH + Sulphates + LabelAppeal + AcidIndex + STARS, family = negative.binomial(1),
           data = WineTrain)
summary(m4)
par(mfrow = c(2,2))
plot(m4)
m1AIC <- AIC(m1)
m1BIC <- BIC(m1)
m2AIC <- AIC(m2)
m2BIC <- BIC(m2)
m3AIC <- AIC(m3)
m3BIC <- BIC(m3)
m4AIC <- AIC(m4)
m4BIC <- BIC(m4)

AIC <- list(m1AIC, m2AIC, m3AIC, m4AIC)
BIC <- list(m1BIC, m2BIC, m3BIC, m4BIC)
kable(rbind(AIC, BIC), col.names = c("Model 1", "Model 2", "Model 3", "Model 4")) %>%
  kable_styling(full_width = T)

eval_p<-predict(m3,WineEval, type = "response")
write.csv(eval_p,"predicted_eval_values.csv")
train_p<-predict(m3,WineTrain, type = "response")
plot(train_p,WineTrain$TARGET)
train_p<-predict(m2,WineTrain, type = "response")
plot(train_p,WineTrain$TARGET)

```