Assignment 5

Kai Lukowiak

2018-03-25

Abstract

This paper tries to predict the number of cases of wine bought based on certain characteristics of the wine.

Contents

The variables, definitions, and theoretical effects are listed below:

VARIABLE NAME	DEFINITION	THEORETICAL EFFECT
INDEX	Identification Variable (do not use)	None
TARGET	Number of Cases Purchased	None
AcidIndex	Proprietary method of testing total acidity of wine by using a weighted average	No Theoretical Effect
Alcohol	Alcohol Content	No Theoretical Effect
Chlorides	Chloride content of wine	No Theoretical Effect
CitricAcid	Citric Acid Content	No Theoretical Effect
Density	Density of Wine	No Theoretical Effect

VARIABLE NAME	DEFINITION	THEORETICAL EFFECT
FixedAcidity	Fixed Acidity of Wine	No Theoretical Effect
FreeSulfurDioxide	Sulfur Dioxide content of wine	No Theoretical Effect
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	No Theoretical Effect
STARS	Wine rating by a team of experts. 4 Stars = Excellent, 1 Star = Poor	A high number of stars suggests high sales
Sulphates	Sulfate conten of wine	No Theoretical Effect
TotalSulfurDioxide	Total Sulfur Dioxide of Wine	No Theoretical Effect
VolatileAcidity	Volatile Acid content of wine	No Theoretical Effect
pH	pH of wine	No Theoretical Effect

1 Data Exploration

Here is a transposed sample of the data.

TARGET	4.00000	3.0000	5.0000	3.0000	4.0000	6.00000
FixedAcidity	6.50000	-8.6000	7.8000	2.0000	7.2000	5.00000
VolatileAcidity	-2.52000	0.2300	0.9900	0.6750	0.2800	-0.11000
CitricAcid	-0.22000	0.4700	-0.3600	0.3600	0.2600	1.74000
ResidualSugar	121.20000	62.2000	24.3000	2.1000	12.5000	8.30000
Chlorides	0.05900	0.0430	0.2130	0.3480	0.2320	0.41900
FreeSulfurDioxide	33.00000	-70.0000	64.0000	-64.0000	-61.0000	35.00000
TotalSulfurDioxide	101.00000	295.0000	118.0000	43.0000	179.0000	164.00000
Density	1.04196	0.9953	0.9901	0.9976	0.9975	0.99944
pH	3.12000	3.2700	3.0000	3.3100	3.1000	3.53000
Sulphates	-0.74000	-0.0200	-0.4200	0.8700	-0.2600	1.03000
Alcohol	4.60000	11.4000	12.1000	6.8000	9.0000	12.50000
LabelAppeal	0.00000	0.0000	0.0000	1.0000	0.0000	0.00000
AcidIndex	7.00000	7.0000	8.0000	9.0000	8.0000	6.00000
STARS	3.00000	1.0000	3.0000	1.0000	NA	2.00000

The data is all numeric, however, we will need to change `TARGET` to factor for one of the regressions.

1.1 Summary Statistics

The summary statistics for the dataset are:

```
##      TARGET       FixedAcidity     VolatileAcidity      CitricAcid
##  Min.   :0.000   Min.   :-18.10   Min.   :-2.7900   Min.   :-3.1600
##  1st Qu.:2.000   1st Qu.:  5.20   1st Qu.: 0.1300   1st Qu.: 0.0200
##  Median :3.000   Median :  6.90   Median : 0.2800   Median : 0.3100
##  Mean   :3.029   Mean   :  7.11   Mean   : 0.3243   Mean   : 0.3109
##  3rd Qu.:4.000   3rd Qu.:  9.50   3rd Qu.: 0.6400   3rd Qu.: 0.6000
##  Max.   :8.000   Max.   : 34.40   Max.   : 3.6800   Max.   : 3.7700
```

```
##
##  ResidualSugar        Chlorides       FreeSulfurDioxide TotalSulfurDioxide
##  Min.   :-127.800   Min.   :-1.1710   Min.   :-555.00   Min.   :-823.0
##  1st Qu.:  -1.800   1st Qu.:-0.0330   1st Qu.:  -2.00   1st Qu.:  29.0
##  Median :   4.000   Median : 0.0460   Median :  30.00   Median : 125.0
##  Mean   :   5.607   Mean   : 0.0526   Mean   :  29.79   Mean   : 122.5
##  3rd Qu.:  16.050   3rd Qu.: 0.1440   3rd Qu.:  69.75   3rd Qu.: 210.0
##  Max.   : 141.150   Max.   : 1.3510   Max.   : 623.00   Max.   :1054.0
##  NA's   :466        NA's   :464       NA's   :469       NA's   :512
##     Density             pH          Sulphates          Alcohol
##  Min.   :0.8895   Min.   :0.480   Min.   :-3.1200   Min.   :-4.70
##  1st Qu.:0.9882   1st Qu.:2.960   1st Qu.: 0.2700   1st Qu.: 9.00
##  Median :0.9944   Median :3.200   Median : 0.5000   Median :10.40
##  Mean   :0.9942   Mean   :3.207   Mean   : 0.5209   Mean   :10.46
##  3rd Qu.:1.0006   3rd Qu.:3.470   3rd Qu.: 0.8600   3rd Qu.:12.30
##  Max.   :1.0992   Max.   :6.050   Max.   : 4.2100   Max.   :26.50
##                   NA's   :308     NA's   :924       NA's   :500
##   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   :-0.009797   Mean   : 7.784   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   :2512
```

Some of these numbers don't make a ton of sense. For example, how can there be negative alchol content? We will address this in the Data Preperation Section.

1.2 Descpritive Statistics

The descriptive statistics are:

```
##                    vars    n   mean     sd median trimmed    mad     min
## TARGET                1 9595   3.03   1.93   3.00    3.05   1.48    0.00
## FixedAcidity          2 9595   7.11   6.33   6.90    7.10   3.26  -18.10
## VolatileAcidity       3 9595   0.32   0.78   0.28    0.33   0.42   -2.79
## CitricAcid            4 9595   0.31   0.87   0.31    0.31   0.43   -3.16
## ResidualSugar         5 9129   5.61  33.70   4.00    5.69  15.57 -127.80
## Chlorides             6 9131   0.05   0.32   0.05    0.05   0.13   -1.17
## FreeSulfurDioxide     7 9126  29.79 148.72  30.00   30.21  56.34 -555.00
## TotalSulfurDioxide    8 9083 122.45 230.99 125.00  123.39 133.43 -823.00
## Density               9 9595   0.99   0.03   0.99    0.99   0.01    0.89
## pH                   10 9287   3.21   0.68   3.20    3.21   0.39    0.48
## Sulphates            11 8671   0.52   0.93   0.50    0.52   0.44   -3.12
## Alcohol              12 9095  10.46   3.72  10.40   10.48   2.37   -4.70
## LabelAppeal          13 9595  -0.01   0.89   0.00   -0.01   1.48   -2.00
## AcidIndex            14 9595   7.78   1.33   8.00    7.65   1.48    4.00
## STARS                15 7083   2.04   0.90   2.00    1.97   1.48    1.00
##                       max   range  skew kurtosis   se
## TARGET               8.00    8.00 -0.33    -0.88 0.02
## FixedAcidity        34.40   52.50 -0.01     1.65 0.06
## VolatileAcidity      3.68    6.47  0.00     1.76 0.01
## CitricAcid           3.77    6.93 -0.05     1.73 0.01
```

```
## ResidualSugar       141.15  268.95 -0.01     1.82 0.35
## Chlorides             1.35    2.52  0.01     1.86 0.00
## FreeSulfurDioxide   623.00 1178.00 -0.03     1.84 1.56
## TotalSulfurDioxide 1054.00 1877.00 -0.07     1.61 2.42
## Density               1.10    0.21 -0.04     1.91 0.00
## pH                    6.05    5.57  0.05     1.63 0.01
## Sulphates             4.21    7.33 -0.01     1.71 0.01
## Alcohol              26.50   31.20 -0.05     1.59 0.04
## LabelAppeal           2.00    4.00 -0.01    -0.25 0.01
## AcidIndex            17.00   13.00  1.68     5.30 0.01
## STARS                 4.00    3.00  0.44    -0.71 0.01
```

There are quite a few `NA` values:

TARGET	0
FixedAcidity	0
VolatileAcidity	0
CitricAcid	0
ResidualSugar	466
Chlorides	464
FreeSulfurDioxide	469
TotalSulfurDioxide	512
Density	0
pH	308
Sulphates	924
Alcohol	500
LabelAppeal	0
AcidIndex	0
STARS	2512

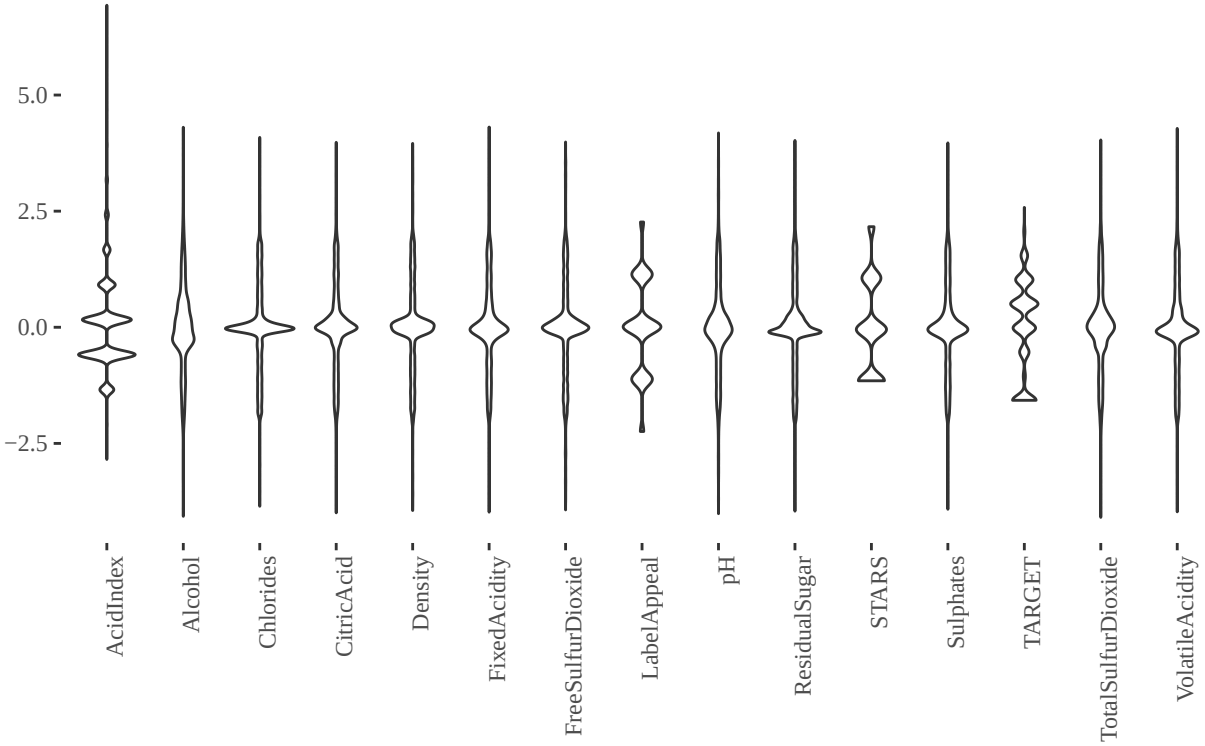
The total percent of rows that have at least on `NA` value is 50%. There is a also significant correlation between NA values and the Target purchase amount. This is problematic for us because dropping the NA rows will result in a biased estimate. Given the high percent of `NA` values and the correlation between missing values and the target, we will have to impute.

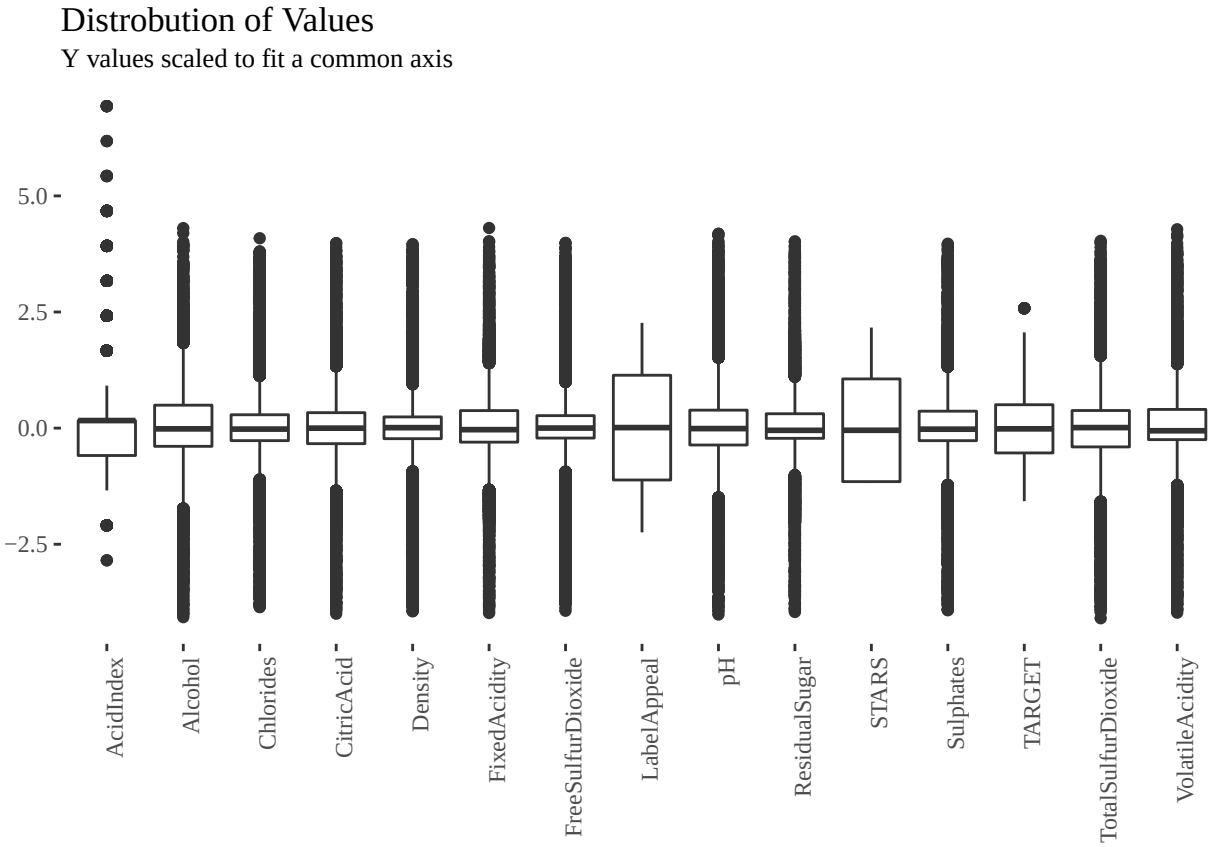
We will have to impute the missing values, but first it's important to make sure the the NA values don't have an explanation. We will explore this more in further sections.

2 Graphical EDA

Distrobution of Values

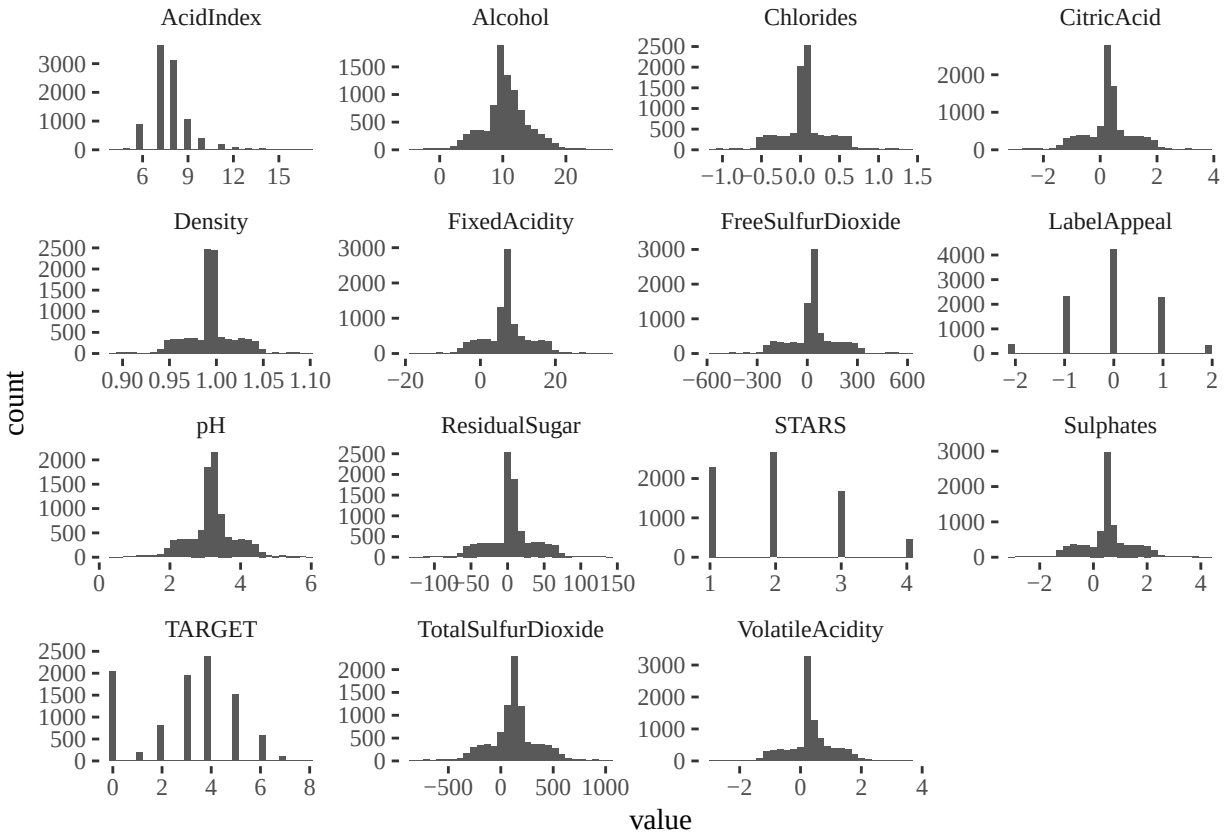
Y values scaled to fit a common axis



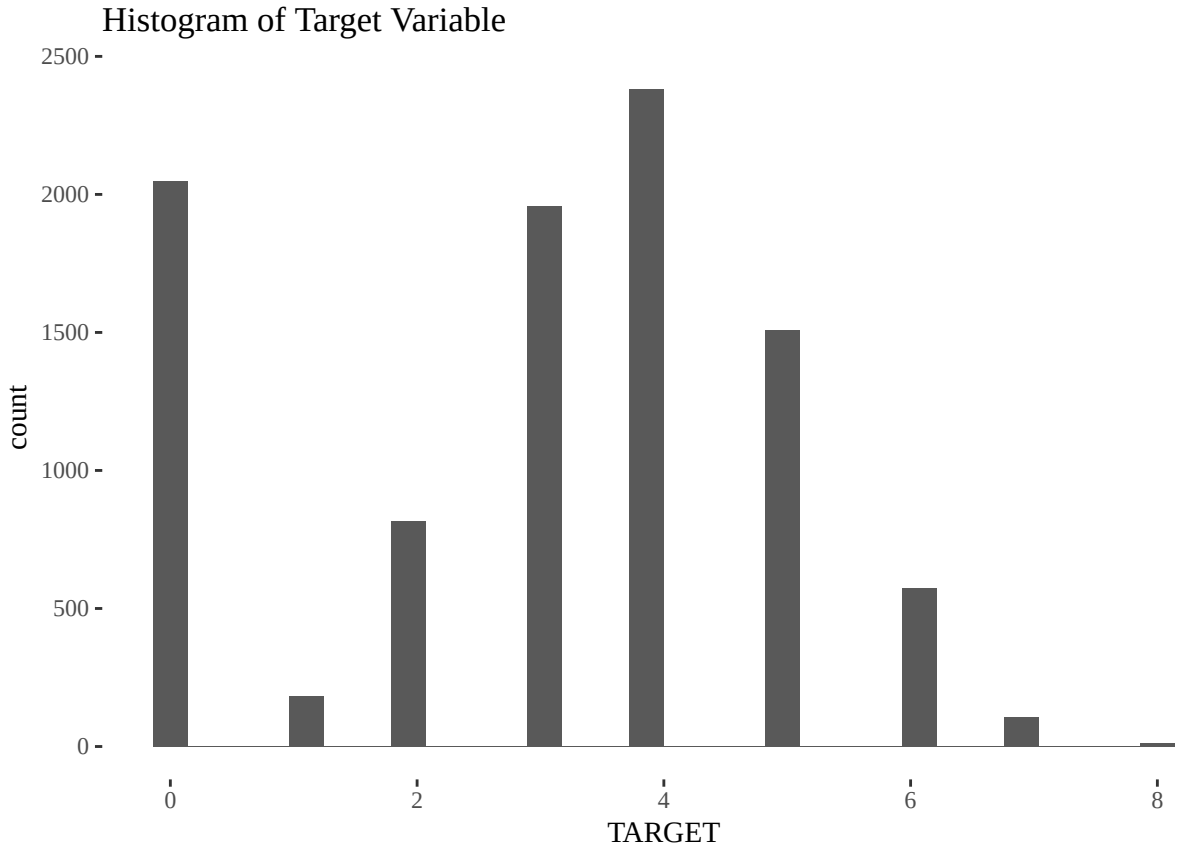


Both the violin and box plots show that most of the variables are normally distributed.

`AcidIndex` seems to be skewed slightly. It also has a skew of 1.68. This is not enought to worry about.

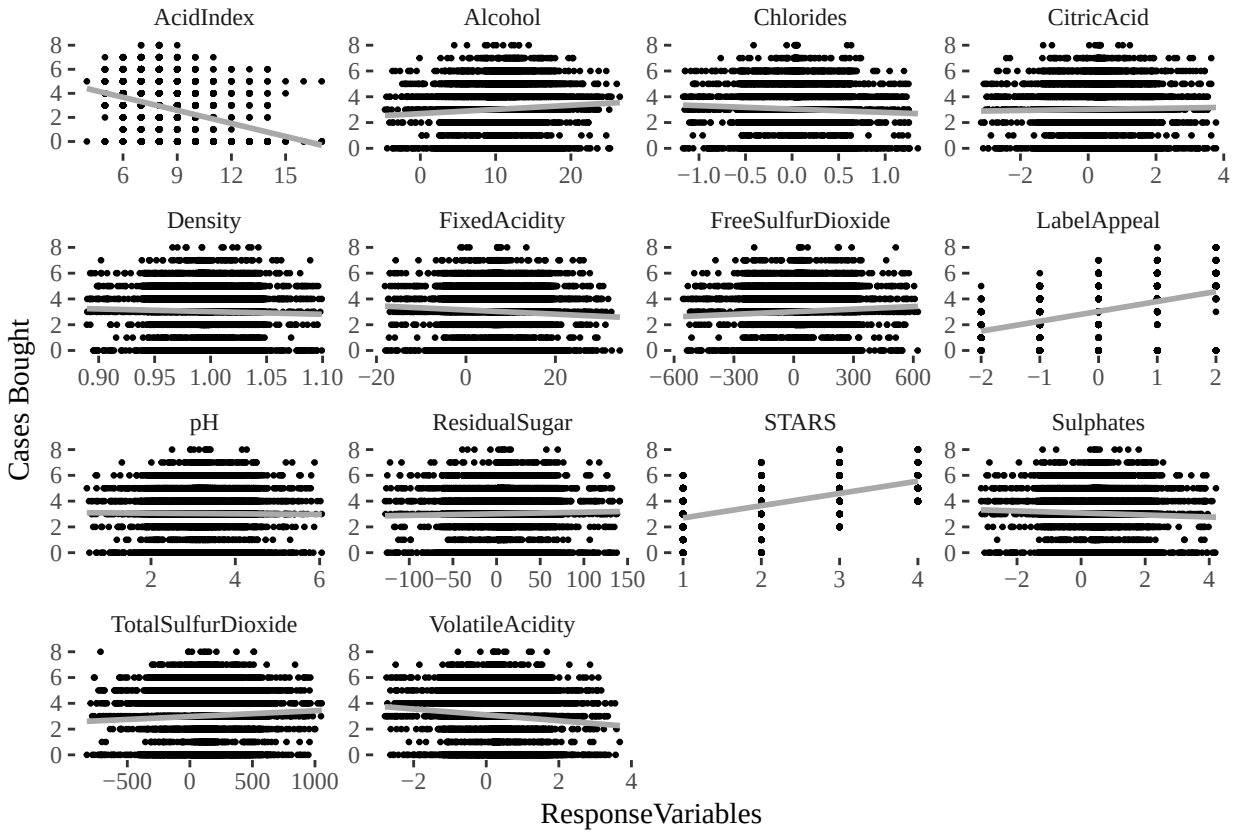


This histogram of the target:



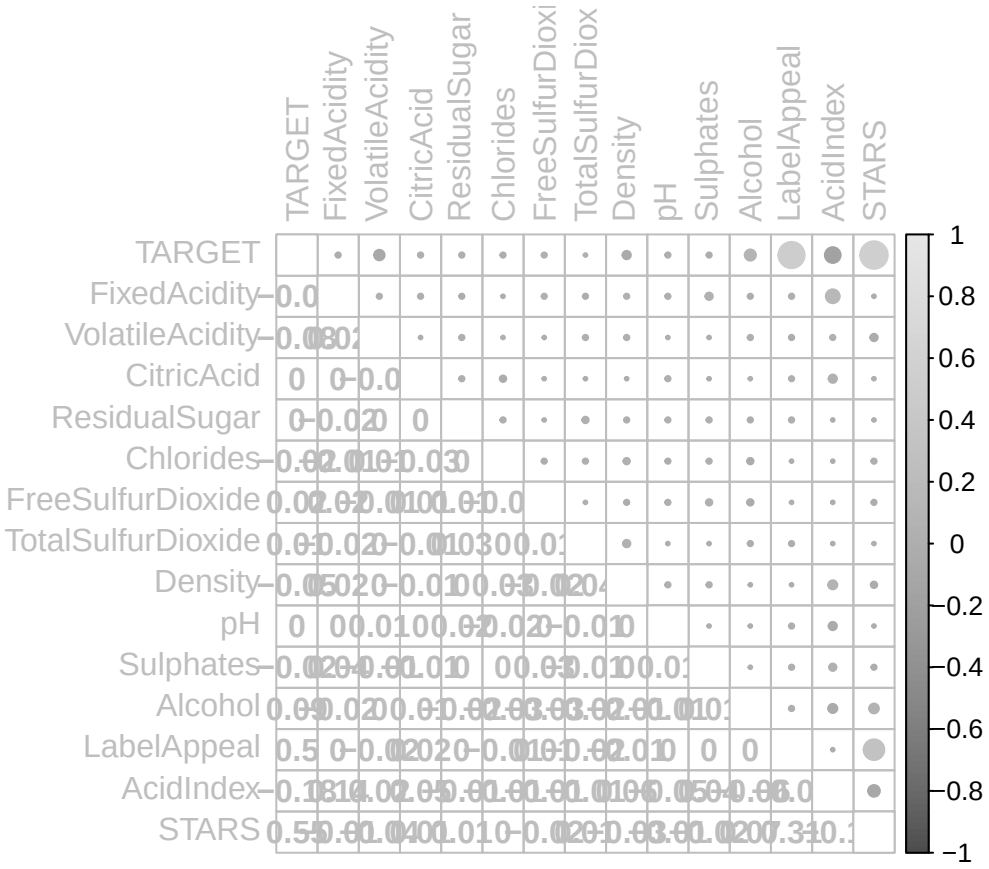
Assides from many wines not being ordered, there rest of the distribution looks normal. We will explor various different models that might deal with this.

Scatter plots against `TARGET`:



There don't seem to be any crazy patterns here. It mostly looks linear which is a good sign for us. `STARS` and `LableAppleal` look like they have the greatest correlation.

Correlation Matrix:



The correlation matrix shows that most values are not that highly correlated.

3 Data Preperation

We will use the `mice` package to impute missing values.

```
##
##  iter imp variable
##   1   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  STA
##   2   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  STA
##   3   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  STA
##   4   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  STA
##   5   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  STA
##
##  iter imp variable
##   1   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  STA
##   2   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  STA
##   3   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  STA
##   4   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  STA
##   5   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  STA
```

3.1 Log Transforms

Given the low correlation between `AcidIndex` and `TARGET` it might not make a huge difference, however, we will log transform it to test.

3.2 Negative Values

There are several variables with negative values that don't necessarily make sense. I am assuming this is due to a normalization procedure.

Thus, transforming them would introduce bias into the model.

4 Building Models

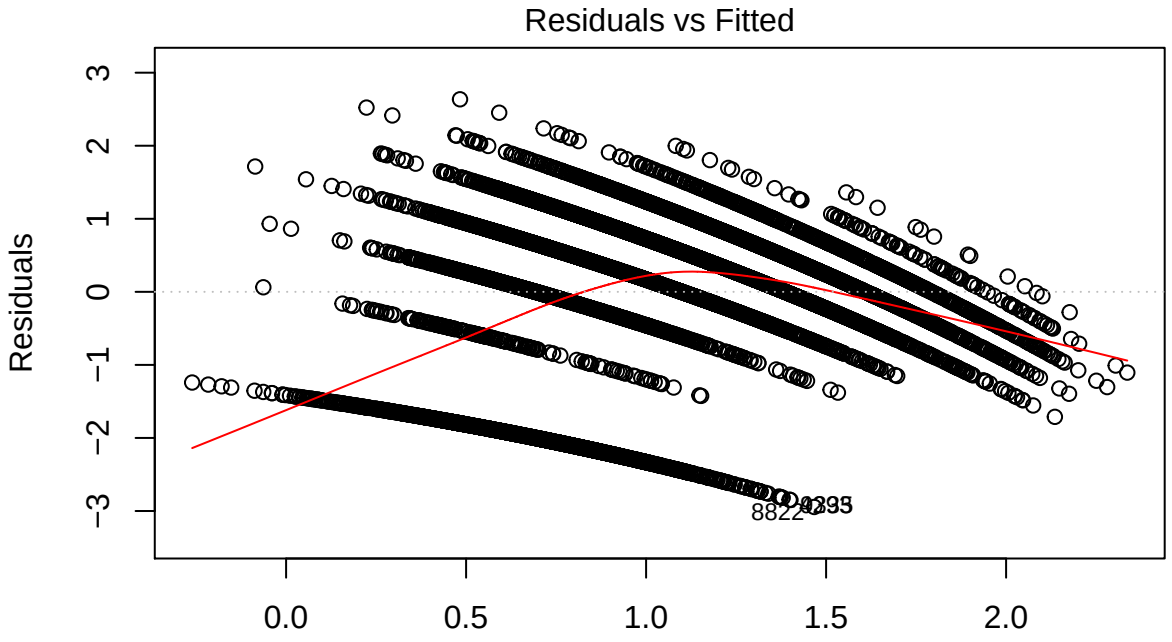
We will build a variety of models using both the imputed and non-imputed data.

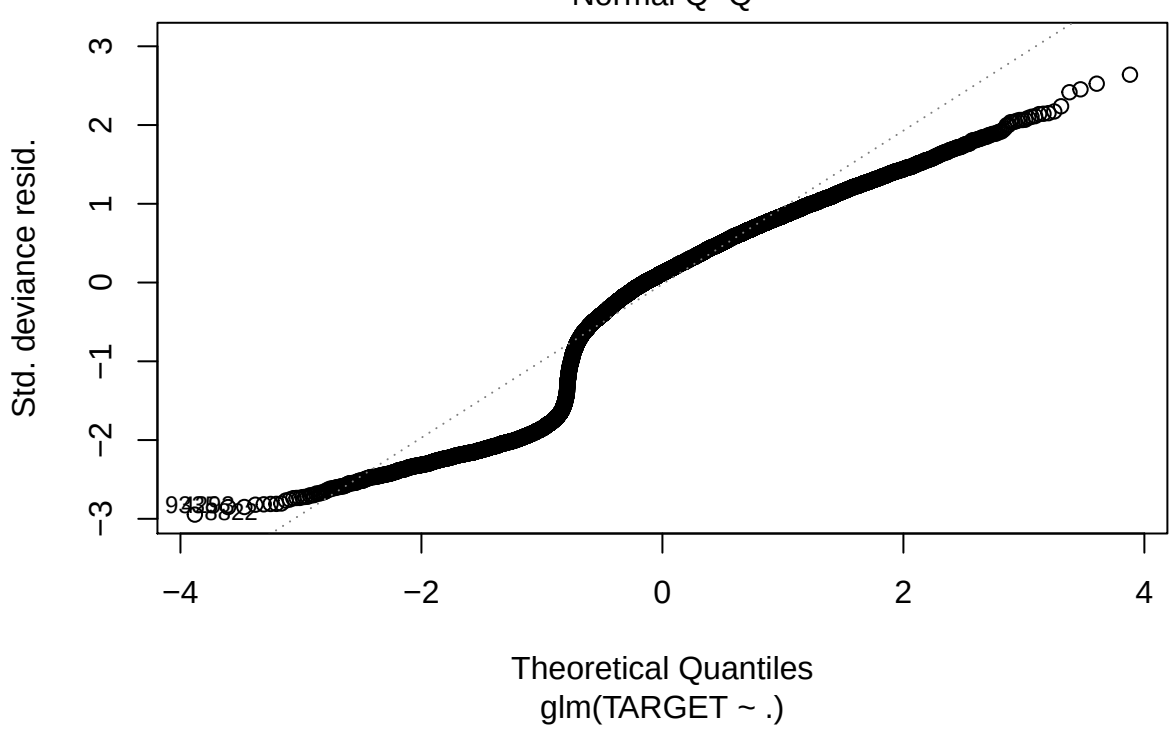
4.1 Poisson 1

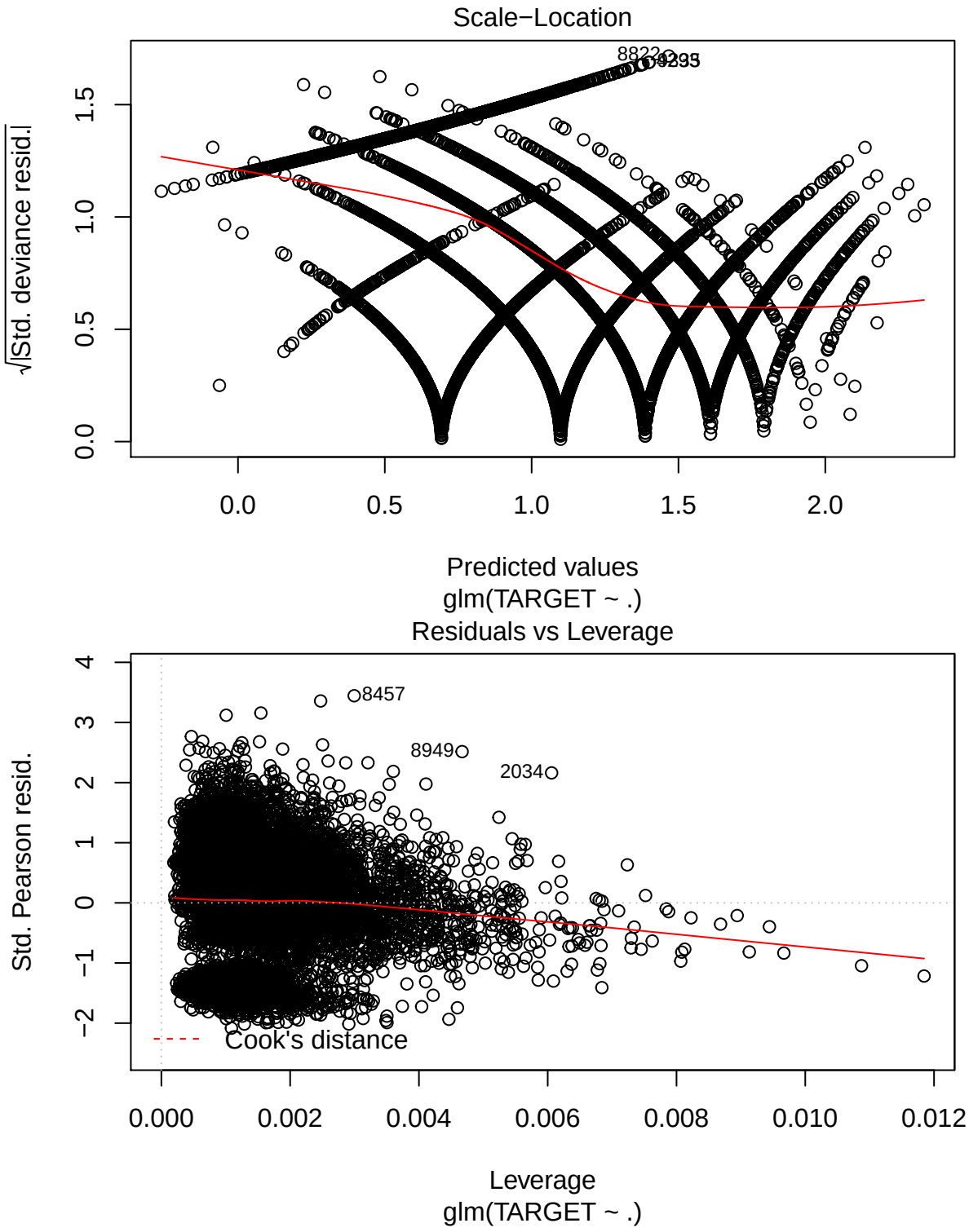
Theoretically, these models should work well given the ranked data.

```
##
## Call:
## glm(formula = TARGET ~ ., family = poisson, data = imputed)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9448  -0.6779   0.1185   0.6363   2.6356
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)          1.437e+00  2.270e-01   6.333 2.40e-10 ***
## FixedAcidity        -3.663e-04  9.460e-04  -0.387  0.69864
## VolatileAcidity     -4.467e-02  7.559e-03  -5.910 3.42e-09 ***
## CitricAcid           1.322e-02  6.766e-03   1.953  0.05077 .
## ResidualSugar        2.089e-04  1.742e-04   1.199  0.23060
## Chlorides           -5.714e-02  1.865e-02  -3.063  0.00219 **
## FreeSulfurDioxide    1.597e-04  3.952e-05   4.040 5.34e-05 ***
## TotalSulfurDioxide   1.113e-04  2.556e-05   4.354 1.34e-05 ***
## Density             -2.479e-01  2.228e-01  -1.113  0.26573
## pH                  -1.488e-02  8.656e-03  -1.719  0.08559 .
## Sulphates           -1.245e-02  6.311e-03  -1.973  0.04850 *
## Alcohol              3.458e-03  1.586e-03   2.180  0.02926 *
## LabelAppeal          1.471e-01  7.029e-03  20.925  < 2e-16 ***
## AcidIndex           -1.001e-01  5.215e-03 -19.184  < 2e-16 ***
## STARS                3.354e-01  6.424e-03  52.202  < 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: 17142  on 9594  degrees of freedom
## Residual deviance: 11995  on 9580  degrees of freedom
```

```
## AIC: 35979
##
## Number of Fisher Scoring iterations: 5
```



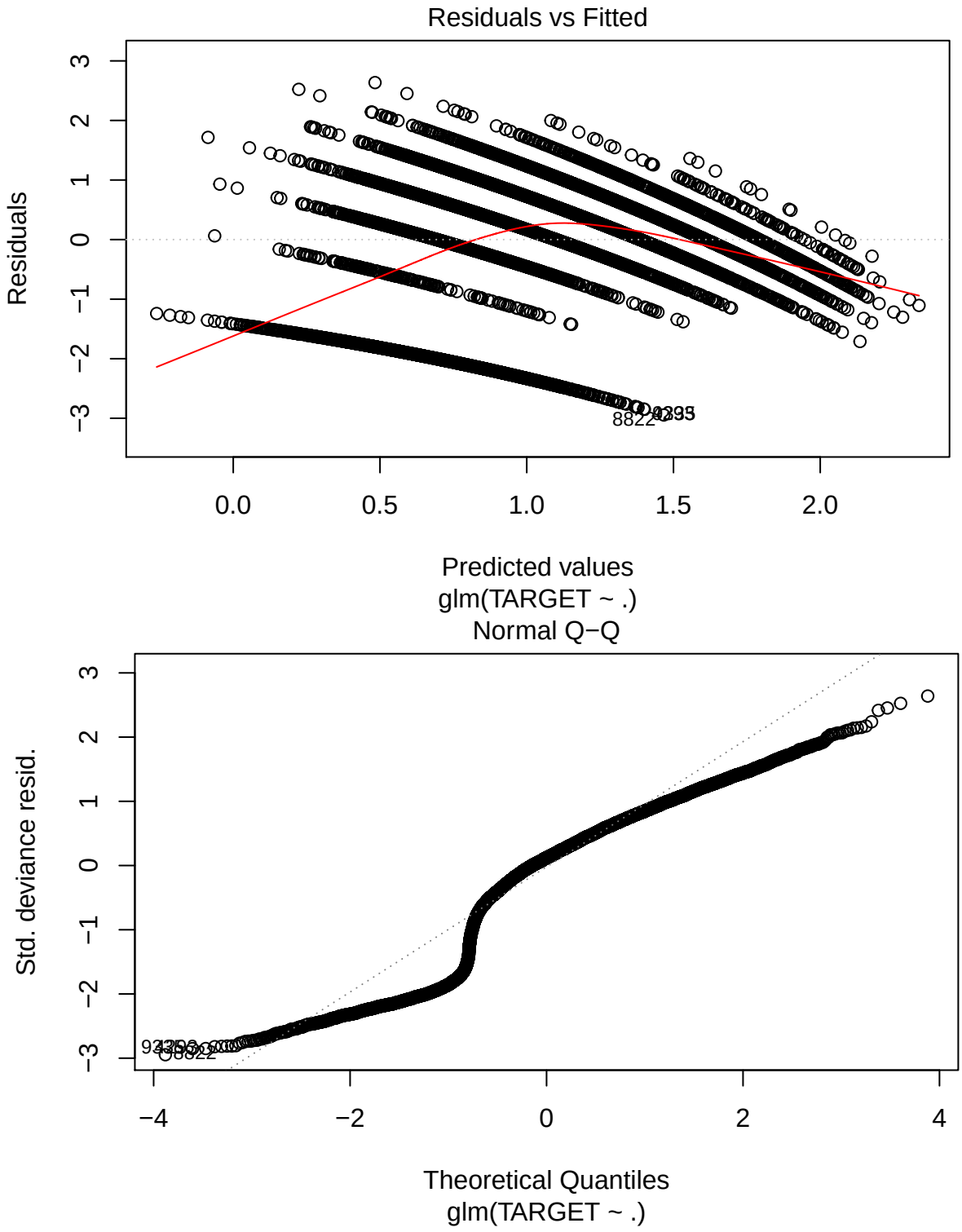


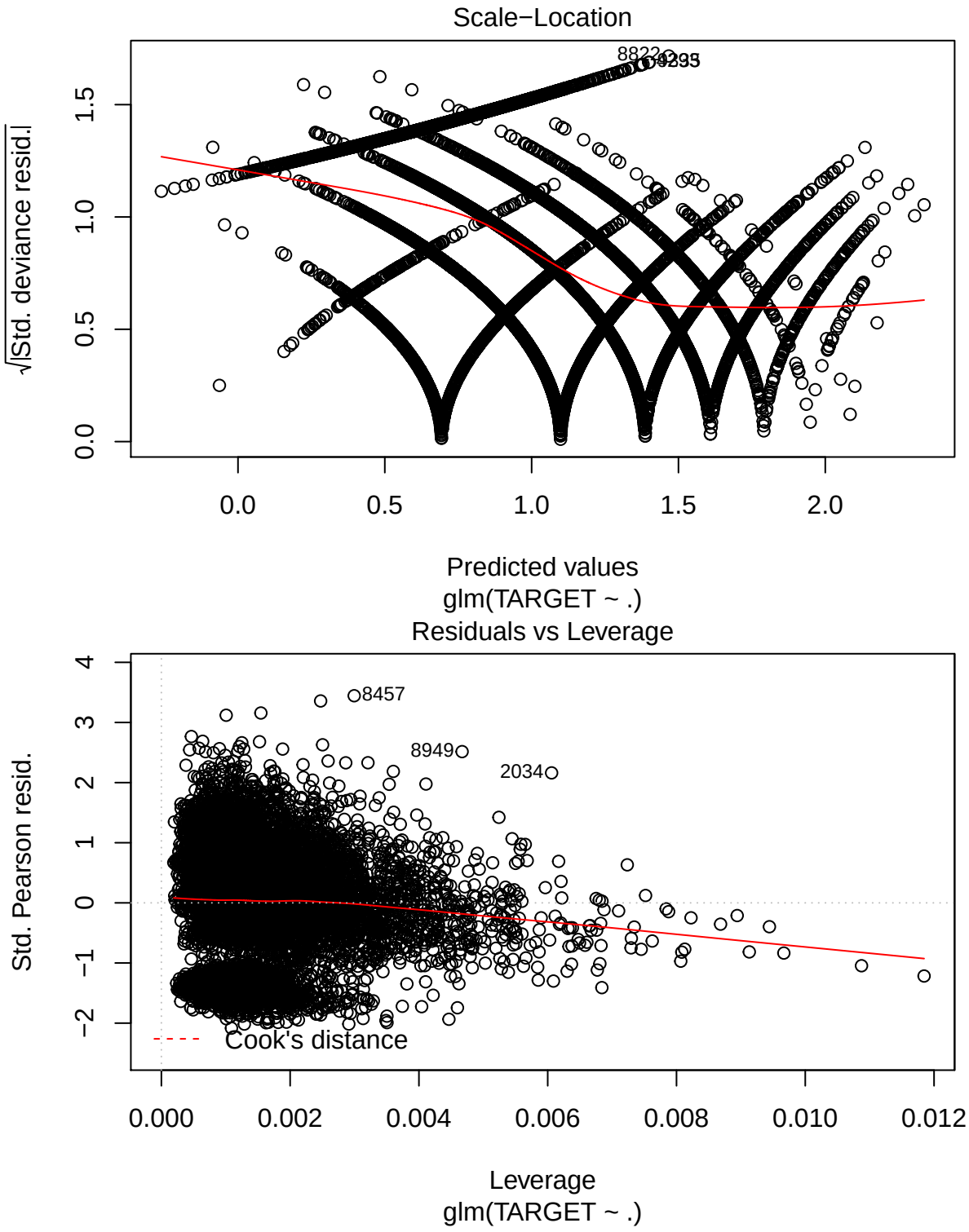


There are some weird diagnostic plots but I don't think there is a ton we can do about that. Also, the lines in the plots are mostly due to the 'categorical' nature of the `TARGET`.

4.2 Poisson with Imputation

```
##
## Call:
## glm(formula = TARGET ~ ., family = poisson, data = imputed)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9448  -0.6779   0.1185   0.6363   2.6356
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)          1.437e+00  2.270e-01   6.333 2.40e-10 ***
## FixedAcidity        -3.663e-04  9.460e-04  -0.387  0.69864
## VolatileAcidity     -4.467e-02  7.559e-03  -5.910 3.42e-09 ***
## CitricAcid           1.322e-02  6.766e-03   1.953  0.05077 .
## ResidualSugar        2.089e-04  1.742e-04   1.199  0.23060
## Chlorides           -5.714e-02  1.865e-02  -3.063  0.00219 **
## FreeSulfurDioxide    1.597e-04  3.952e-05   4.040 5.34e-05 ***
## TotalSulfurDioxide   1.113e-04  2.556e-05   4.354 1.34e-05 ***
## Density             -2.479e-01  2.228e-01  -1.113  0.26573
## pH                  -1.488e-02  8.656e-03  -1.719  0.08559 .
## Sulphates           -1.245e-02  6.311e-03  -1.973  0.04850 *
## Alcohol              3.458e-03  1.586e-03   2.180  0.02926 *
## LabelAppeal          1.471e-01  7.029e-03  20.925  < 2e-16 ***
## AcidIndex           -1.001e-01  5.215e-03 -19.184  < 2e-16 ***
## STARS                3.354e-01  6.424e-03  52.202  < 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: 17142  on 9594  degrees of freedom
## Residual deviance: 11995  on 9580  degrees of freedom
## AIC: 35979
##
## Number of Fisher Scoring iterations: 5
```


Residuals vs Fitted
Residuals
Predicted values
glm(TARGET ~ .)
Normal Q−Q
Std. deviance resid.
Theoretical Quantiles
glm(TARGET ~ .)

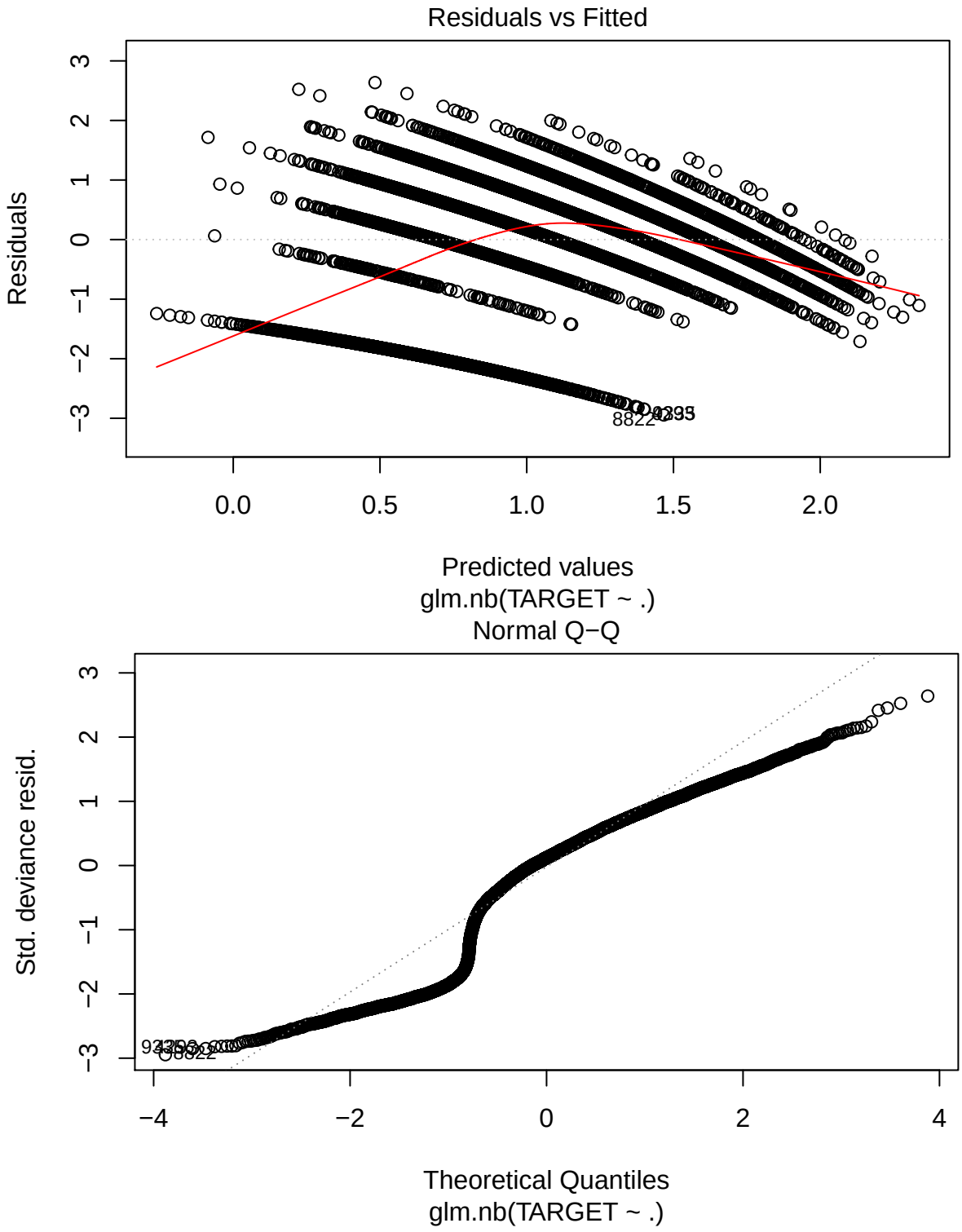


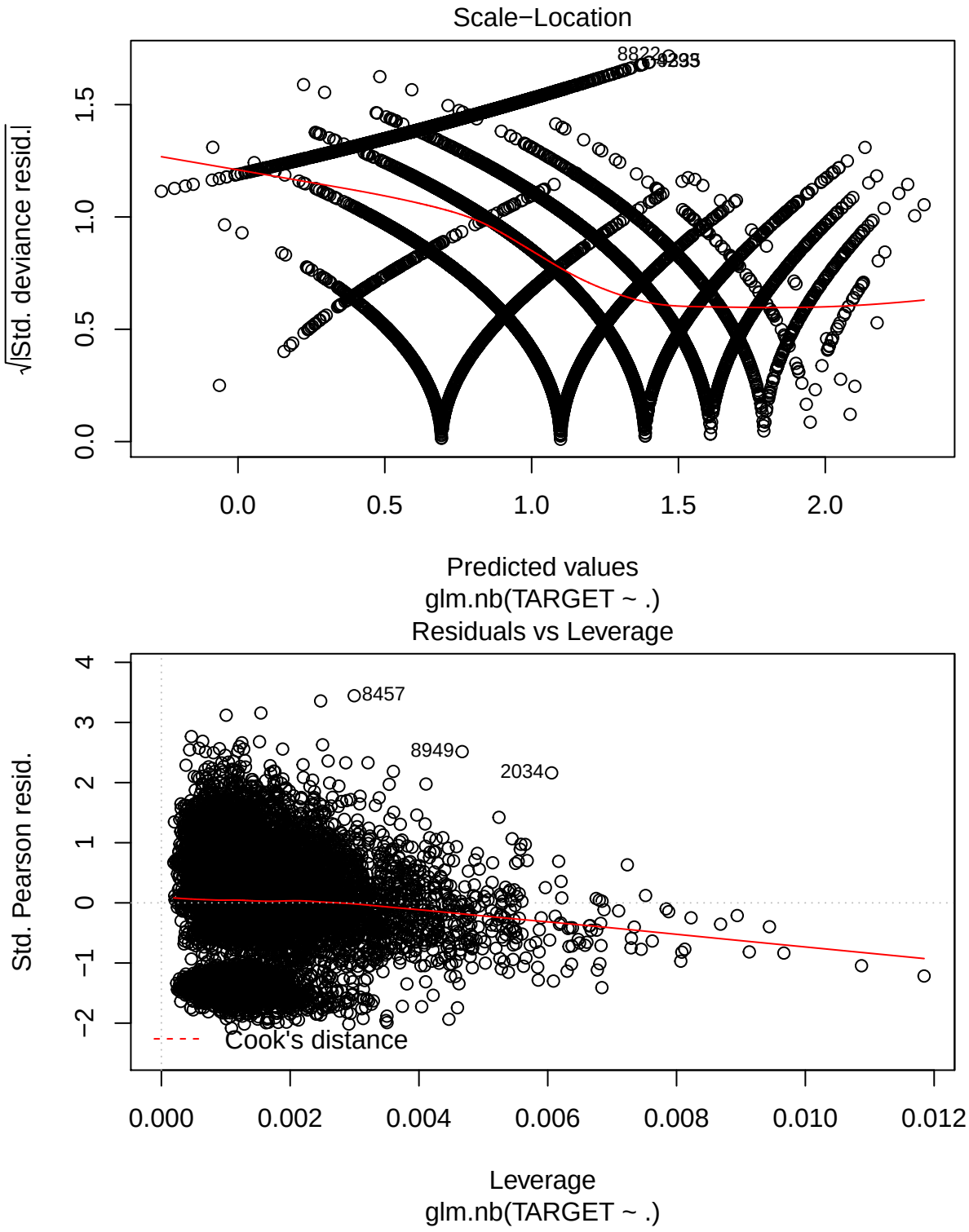
Both these models have similar AICs. I am more comfortable using the imputed data because I think removing a large number of rows will be detrimental.

4.3 Negative Binomial

The Negative Binomial distrobution should also perform well on count variables.

```
##
## Call:
## glm.nb(formula = TARGET ~ ., data = imputed, init.theta = 48536.95165,
##     link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9448  -0.6779   0.1185   0.6363   2.6355
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)          1.437e+00  2.270e-01   6.333 2.41e-10 ***
## FixedAcidity        -3.663e-04  9.461e-04  -0.387  0.69865
## VolatileAcidity     -4.467e-02  7.559e-03  -5.910 3.42e-09 ***
## CitricAcid           1.322e-02  6.766e-03   1.953  0.05078 .
## ResidualSugar        2.089e-04  1.742e-04   1.199  0.23061
## Chlorides           -5.714e-02  1.865e-02  -3.063  0.00219 **
## FreeSulfurDioxide    1.597e-04  3.952e-05   4.040 5.34e-05 ***
## TotalSulfurDioxide   1.113e-04  2.556e-05   4.354 1.34e-05 ***
## Density             -2.479e-01  2.228e-01  -1.113  0.26574
## pH                  -1.488e-02  8.656e-03  -1.719  0.08559 .
## Sulphates           -1.245e-02  6.311e-03  -1.973  0.04850 *
## Alcohol              3.458e-03  1.586e-03   2.180  0.02927 *
## LabelAppeal          1.471e-01  7.029e-03  20.924  < 2e-16 ***
## AcidIndex           -1.001e-01  5.216e-03 -19.184  < 2e-16 ***
## STARS                3.354e-01  6.424e-03  52.201  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(48536.95) family taken to be 1)
##
##     Null deviance: 17141  on 9594  degrees of freedom
## Residual deviance: 11995  on 9580  degrees of freedom
## AIC: 35982
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  48537
##           Std. Err.:  64523
## Warning while fitting theta: iteration limit reached
##
##  2 x log-likelihood:  -35949.57
```


Residuals vs Fitted
Residuals
Predicted values
glm.nb(TARGET ~ .)
Normal Q−Q
Std. deviance resid.
Theoretical Quantiles
glm.nb(TARGET ~ .)

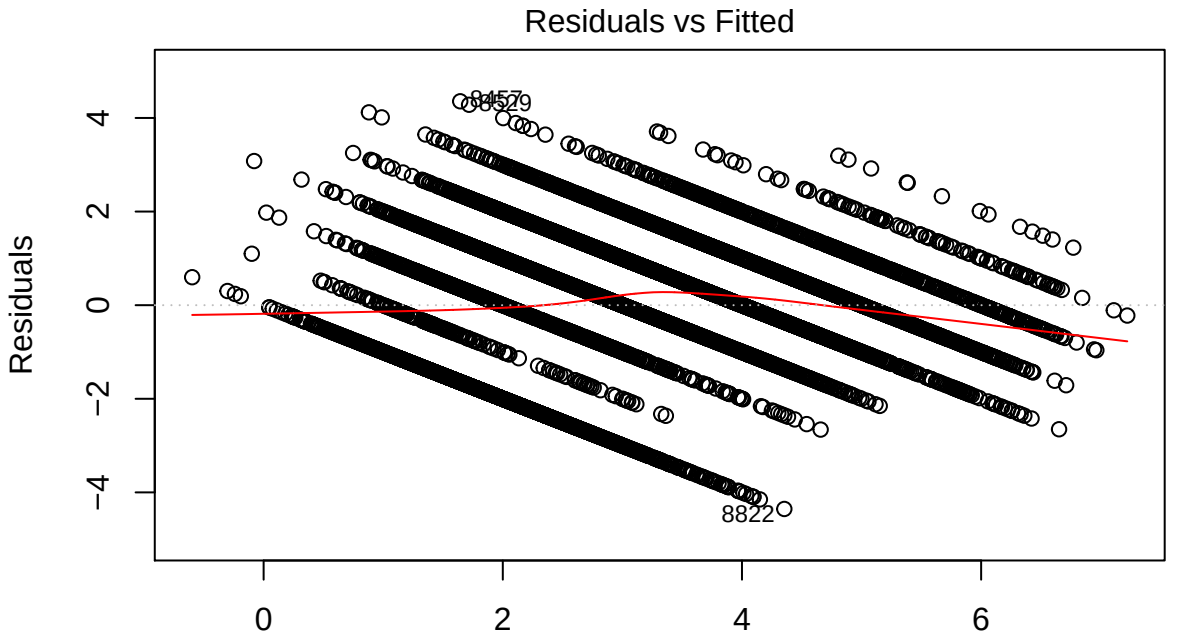


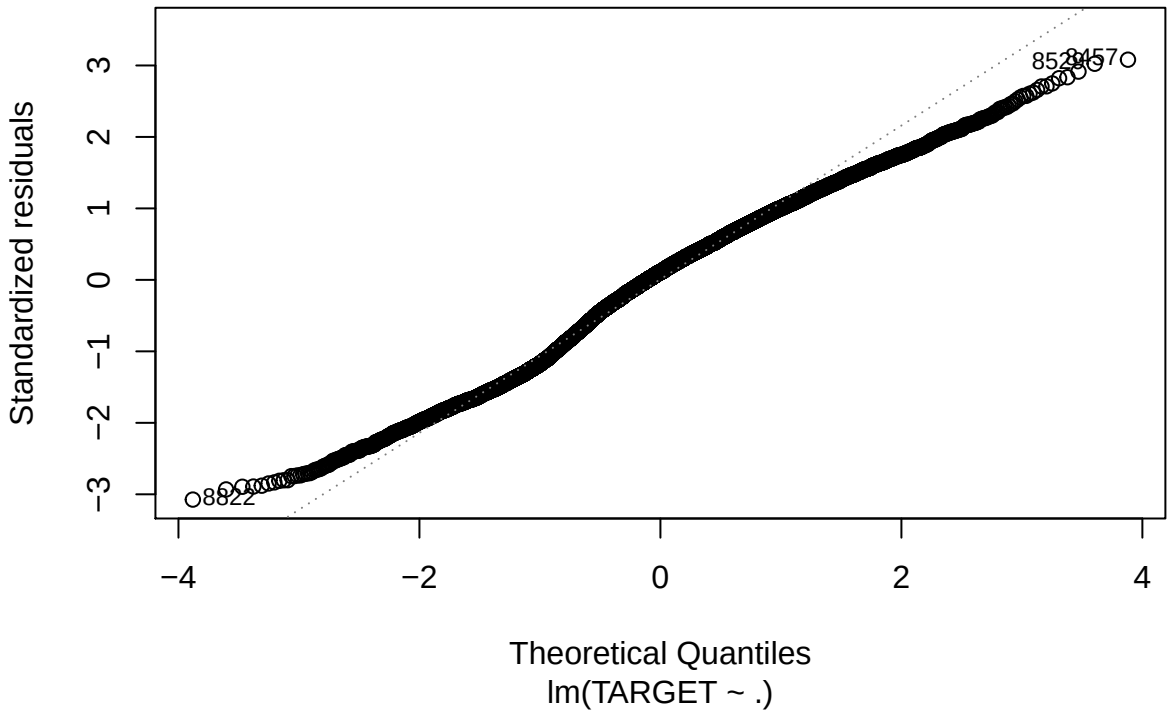
4.4 Linear Model

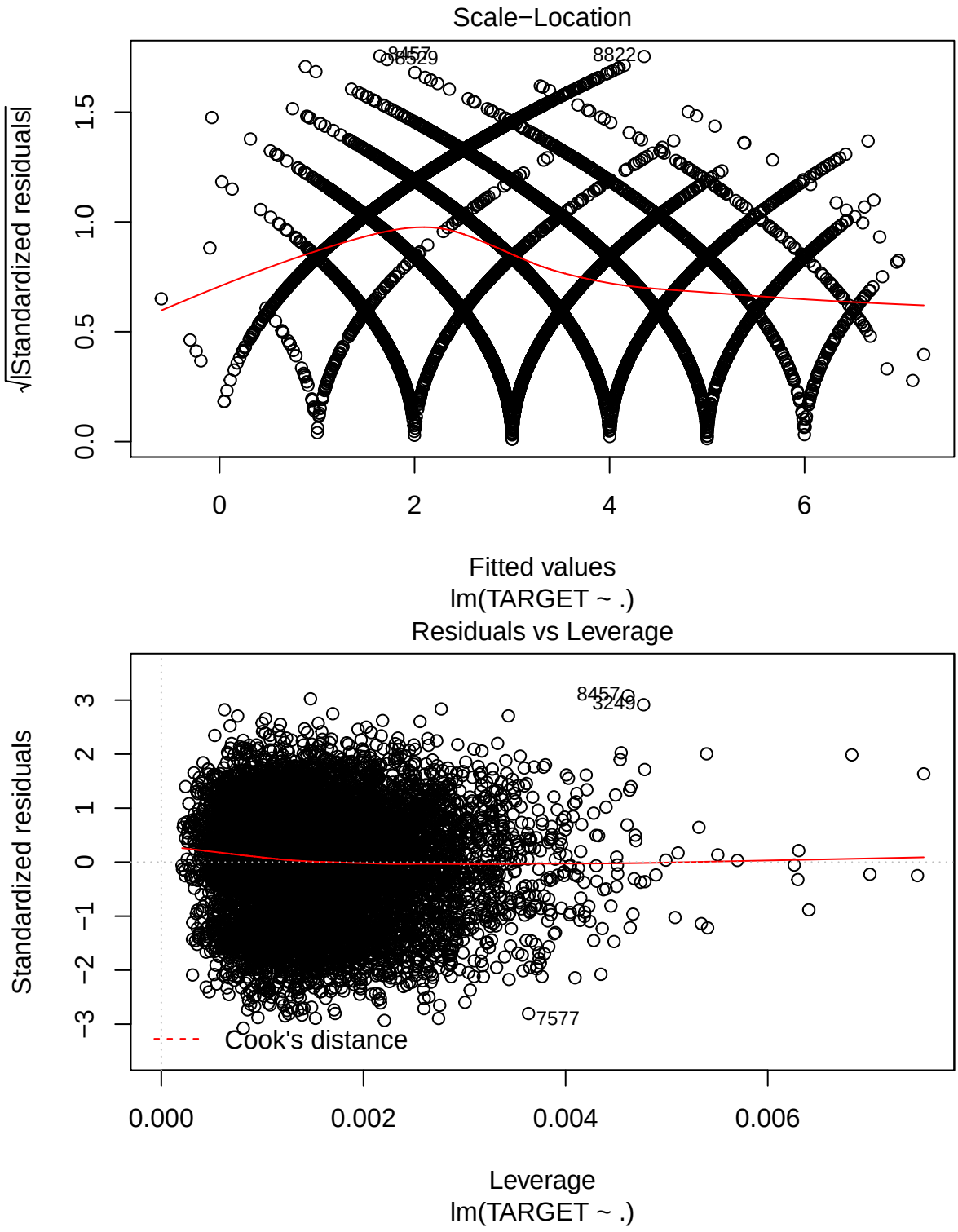
While `TARGET` could be considered a linear response variable, I initially thought it would be a poorer predictor compared to Poisson based models.

However, it actually performed quite well. I think this is because the counts are tighly grouped. If there was more disperse counts, say up to 100 and with few examples of these disperse numbers, I would guess this model would not perform as well.

```
##
## Call:
## lm(formula = TARGET ~ ., data = imputed)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.3541 -1.0133  0.1571  1.0398  4.3560
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.445e+00  5.550e-01   6.208 5.59e-10 ***
## FixedAcidity        -4.059e-04  2.322e-03  -0.175 0.861247
## VolatileAcidity     -1.272e-01  1.851e-02  -6.872 6.71e-12 ***
## CitricAcid           3.666e-02  1.670e-02   2.195 0.028166 *
## ResidualSugar        5.378e-04  4.296e-04   1.252 0.210737
## Chlorides           -1.859e-01  4.568e-02  -4.070 4.74e-05 ***
## FreeSulfurDioxide    4.458e-04  9.728e-05   4.583 4.65e-06 ***
## TotalSulfurDioxide   3.148e-04  6.288e-05   5.006 5.66e-07 ***
## Density             -6.650e-01  5.461e-01  -1.218 0.223375
## pH                  -3.655e-02  2.129e-02  -1.717 0.085982 .
## Sulphates           -3.170e-02  1.554e-02  -2.041 0.041312 *
## Alcohol              1.393e-02  3.880e-03   3.591 0.000331 ***
## LabelAppeal          4.489e-01  1.704e-02  26.345  < 2e-16 ***
## AcidIndex           -2.468e-01  1.127e-02 -21.897  < 2e-16 ***
## STARS                1.149e+00  1.712e-02  67.120  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.417 on 9580 degrees of freedom
## Multiple R-squared:  0.4597, Adjusted R-squared:  0.4589
## F-statistic: 582.1 on 14 and 9580 DF,  p-value: < 2.2e-16
```


Residuals vs Fitted
Residuals
Fitted values
lm(TARGET ~ .)
8457
8822


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(TARGET ~ .)
8457
8822



4.5 Ordinal Logistic Regression

This regression uses ordered factors. I would expect this to be one of the top performers.

```
## Call:
```

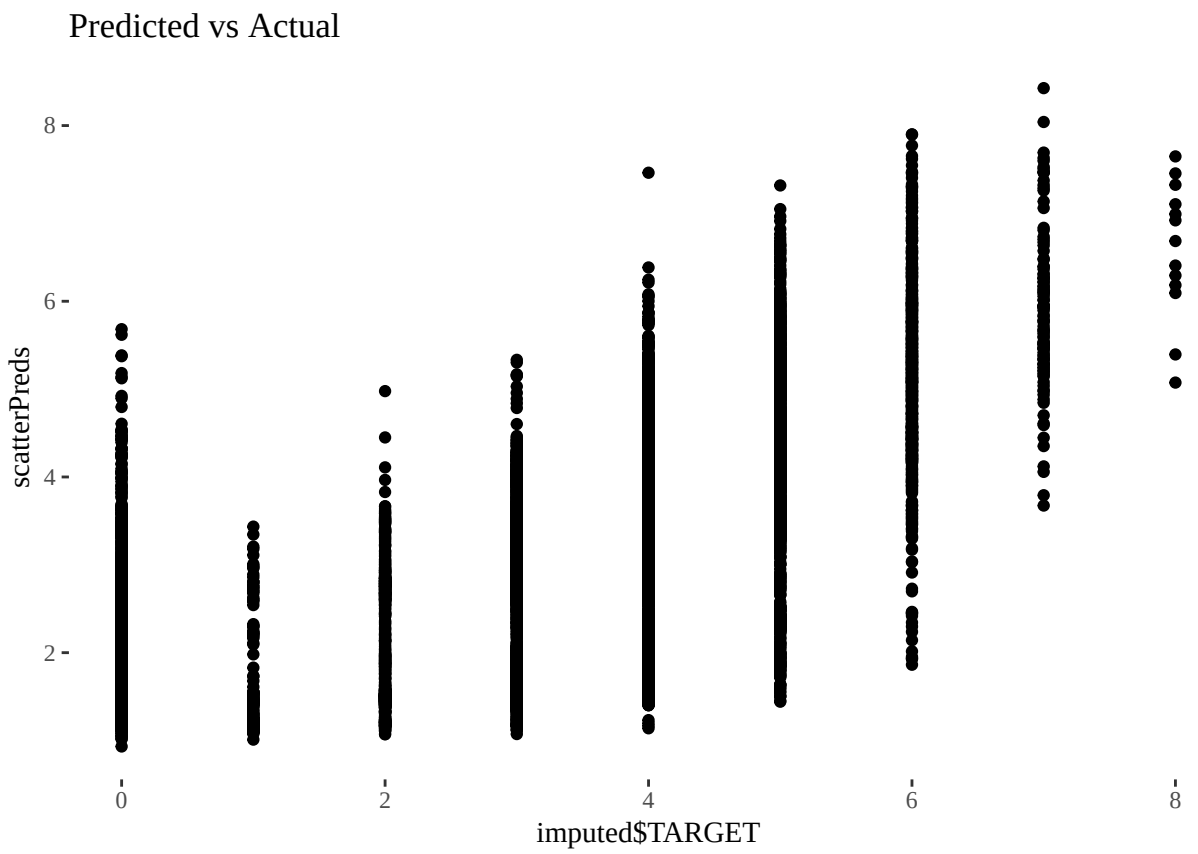
```
## polr(formula = TARGET ~ ., data = polrDF, Hess = TRUE)
##
## Coefficients:
##                         Value Std. Error  t value
## FixedAcidity        0.0012363  2.995e-03   0.4128
## VolatileAcidity    -0.1631205  2.409e-02  -6.7726
## CitricAcid          0.0430326  2.158e-02   1.9940
## ResidualSugar       0.0005946  5.556e-04   1.0701
## Chlorides          -0.2194775  5.860e-02  -3.7456
## FreeSulfurDioxide   0.0005573  1.262e-04   4.4143
## TotalSulfurDioxide  0.0003320  8.173e-05   4.0618
## Density            -0.7878887  8.894e-02  -8.8590
## pH                 -0.0265603  2.751e-02  -0.9656
## Sulphates          -0.0345247  2.026e-02  -1.7045
## Alcohol             0.0297455  5.024e-03   5.9204
## LabelAppeal         0.8423336  2.464e-02  34.1790
## AcidIndex          -0.3280522  1.598e-02 -20.5323
## STARS               1.4242994  2.608e-02  54.6109
##
## Intercepts:
##     Value    Std. Error t value
## 0|1  -2.5419   0.0871   -29.1769
## 1|2  -2.4033   0.0871   -27.5865
## 2|3  -1.7977   0.0873   -20.5959
## 3|4  -0.4448   0.0887    -5.0121
## 4|5   1.3825   0.0941    14.6879
## 5|6   3.3347   0.1052    31.7082
## 6|7   5.5243   0.1385    39.8892
## 7|8   7.8568   0.2969    26.4638
##
## Residual Deviance: 28213.73
## AIC: 28257.73
```

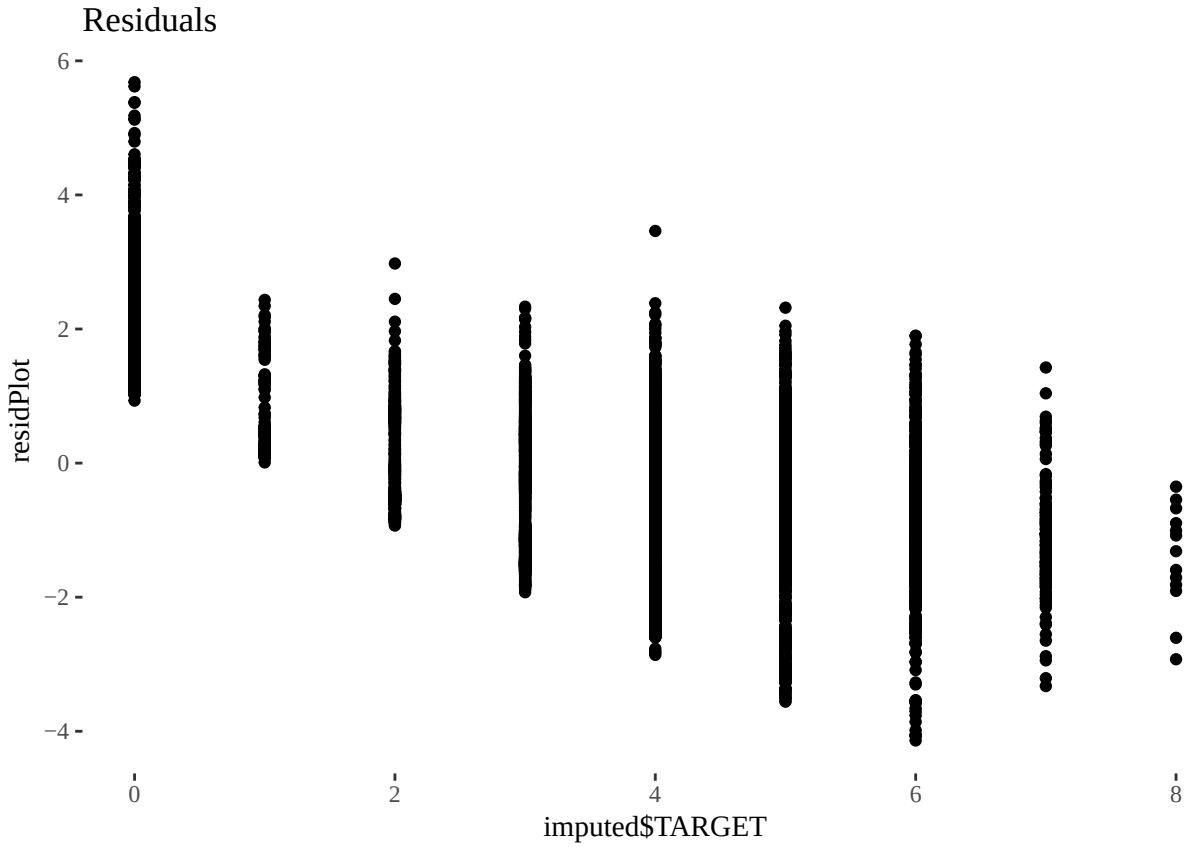
4.6 Zero inflation

Zero inflation understands that some Poisson distrobutions are dominated by many zeros. As such it corrects for this. This is one of the most promissing ones because as we saw in our data exploration, there were more zeros, and then normally distributed data after that.

```
##
## Call:
## zeroinfl(formula = TARGET ~ . | STARS, data = imputed, dist = "negbin")
##
## Count model coefficients (negbin with log link):
##        (Intercept)        FixedAcidity     VolatileAcidity
##          1.551e+00           3.803e-04          -1.865e-02
##         CitricAcid       ResidualSugar           Chlorides
##          2.529e-03           7.168e-06          -3.078e-02
##  FreeSulfurDioxide  TotalSulfurDioxide             Density
##          5.431e-05           1.299e-05          -3.070e-01
##                 pH           Sulphates             Alcohol
##          2.204e-03          -3.623e-03           6.839e-03
##        LabelAppeal           AcidIndex                STARS
##          2.282e-01          -3.625e-02           1.181e-01
```

```
## Theta = 100718789.5354
##
## Zero-inflation model coefficients (binomial with logit link):
## (Intercept)        STARS
##       2.509       -2.848
```





5 Model Selection

We will use the squared difference two select a model (MSE) from predictions on the training sets. (Lower numbers are better.)

5.1 GLM Poisson (Imputed)

A regular Poisson regression does not perform very well.

```
## [1] 6.845792
```

5.2 Negative Binomial

The same can be said for the Negative Binomial.

```
## [1] 6.845788
```

5.3 Linear

The linear model actually performs very well. As I talked about earlier, this is not totally surprising.

```
## [1] 2.004901
```

5.4 Ordenal Logistic Regression

Very surprisinly, this does not work as well as the linear model.

```
## [1] 2.946326
```

5.5 Zero Inflatiion

Zero inflation lives up to it's name. It deals with the zero heavy results very nicely.

```
## [1] 1.982993
```

Because we are not interested in gaining insight into the underlying causes of wine selection, we will use the squared loss. This will tell us how accurate our model is without caring about confidence intervals etc.

Based on this metric, Zero Poission Inflation is the most accurate.

6 Prediction

Predicting using new data:

This model uses the same imputation process as above.

```
##
##  iter imp variable
##    1   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  ST
##    2   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  ST
##    3   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  ST
##    4   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  ST
##    5   1  ResidualSugar  Chlorides  FreeSulfurDioxide  TotalSulfurDioxide  pH  Sulphates  Alcohol  ST
```

Predicted	Rounded
1.645637	2
3.655942	4
1.743480	2
1.507278	2
1.642022	2
5.636340	6

7 Appendix

```
knitr::opts_chunk$set(echo = FALSE)
knitr::opts_chunk$set(warning = FALSE)
knitr::opts_chunk$set(message = FALSE)
library(tidyverse)
library(caTools)
library(corrplot)
library(knitr)
library(psych)
library(kableExtra)
library(ggthemes)
```

```
library(pander)
library(memisc)
library(mice)
dd <- read_csv("/Users/kailukowiak/DATA621/Assignments/Assignment5/Datadict.csv")
dd <- dd %>%
  filter(!is.na(`VARIABLE NAME`)) %>%
  replace_na(list(`THEORETICAL EFFECT` = 'No Theoretical Effect'))
pander::pander(dd, split.cell = 80, split.table = Inf)
LabledDF <- read_csv('wine-training-data.csv')
LabledDF <- LabledDF %>% dplyr::select(-INDEX)
predictDF <- read_csv('wine-evaluation-data.csv')
set.seed(101)
sample = sample.split(LabledDF$TARGET, SplitRatio = .75)
df <- subset(LabledDF, sample == TRUE)
testDF <- subset(LabledDF, sample == FALSE)
temp <- df %>% sample_n(6)
temp %>% t() %>%  kable()
SumTab <- summary(df)
SumTab
dis <- describe(df)
dis
temp <- map(df, ~sum(is.na(.)))
temp <- as.data.frame(temp)
temp %>% t() %>% kable()

df %>%
  scale() %>%
  as_tibble() %>%
  gather() %>%
  ggplot(aes(x = key, y = value)) +
  theme_tufte() +
  geom_violin()+
  #geom_tufteboxplot(outlier.colour="black")+
  theme(axis.title=element_blank()) +
  ylab('Scaled Values')+
  xlab('Variable')+
  theme(axis.text.x = element_text(angle = 90, hjust = 1))+
  ggtitle('Distrobution of Values', subtitle = 'Y values scaled to fit a common axis')
df %>%
  scale() %>%
  as_tibble() %>%
  gather() %>%
  ggplot(aes(x = key, y = value)) +
  # geom_violin()+
  # geom_tufteboxplot(outlier.colour="black", outlier.shape = 22)+
  geom_boxplot()+
  theme_tufte() +
  theme(axis.title=element_blank()) +
  ylab('Scaled Values')+
  xlab('Variable')+
  theme(axis.text.x = element_text(angle = 90, hjust = 1))+
  ggtitle('Distrobution of Values', subtitle = 'Y values scaled to fit a common axis')
df %>%
```

```
  gather() %>%
  ggplot(aes(x=value))+
  geom_histogram()+
  facet_wrap(~key,scales='free')+
  theme(axis.text.x = element_text(angle = 90, hjust = 1))+
  theme_tufte()
ggplot(data = df, aes(x=TARGET)) +
  geom_histogram() +
  ggtitle('Histogram of Target Variable') +
  theme_tufte()
df %>%
  gather(-TARGET, key = "key", value = "ResponseVariables") %>%
  ggplot(aes(x = ResponseVariables, y = TARGET)) +
  geom_point(size = .5) +
  geom_smooth(method='lm',formula=y~x, color = 'dark grey')+
  facet_wrap(~ key, scales = "free")+
  theme_tufte()+
  ylab('Cases Bought')
corr <- round(cor(df, use= 'complete.obs'), 2)
corrplot.mixed(corr, lower.col = 'grey', tl.pos = 'lt' ,upper.col = gray.colors(100), tl.col = 'grey')
imputed <- mice(df, m=1, maxit = 5, seed = 42)
imputed <- complete(imputed)
imputed <- as.data.frame(imputed)

testImput <- test <- mice(testDF, m=1, maxit = 5, seed = 42)
testImput <- complete(testImput)
testImput <- as.data.frame(imputed)
dfFALog <- df %>%
  mutate(AcidIndex = log(AcidIndex))
mod1 = glm(TARGET ~  ., data=imputed, family=poisson)
summary(mod1)
plot(mod1)
mod2 <-  glm(TARGET ~  ., data=imputed, family=poisson)
summary(mod2)
plot(mod2)
mod3 <- glm.nb(TARGET ~ ., data = imputed)
summary(mod3)
plot(mod3)
mod4 <- lm(TARGET ~ ., data = imputed)
summary(mod4)
plot(mod4)
polrDF <- imputed
polrDF$TARGET <- as.factor(polrDF$TARGET)
mod5 <- polr(TARGET ~ ., data = polrDF, Hess=TRUE)
summary(mod5)
library(pscl)
mod6 <- zeroinfl(TARGET ~ . | STARS, data = imputed, dist = 'negbin')
mod6
scatterPreds <- predict(mod6, imputed)
qplot(imputed$TARGET, scatterPreds, main = 'Predicted vs Actual') + theme_tufte()

residPlot <- scatterPreds - imputed$TARGET
qplot(imputed$TARGET, residPlot, main = 'Residuals') + theme_tufte()
```

```r
modelValidation <- function(mod, test){
  preds = predict(mod, test)
  diffMat = as.numeric(preds) - as.numeric(test$TARGET)
  diffMat = diffMat^2
  loss <- mean(diffMat)
  return(loss)
}

modelValidation(mod2, testImput)
modelValidation(mod3, testImput)
modelValidation(mod4, testImput)
polrDFTest <- testImput
polrDFTest$TARGET <- as.factor(polrDFTest$TARGET)
modelValidation(mod5, polrDFTest)
modelValidation(mod6, testImput)

predImputed <- mice(predictDF, m=1, maxit = 5, seed = 42)
predImputed <- complete(predImputed)
predImputed <- as.data.frame(predImputed)

zipPreds <- predict(mod6, predImputed)
zipPreds <- as_data_frame(zipPreds)
colnames(zipPreds) <- 'Predicted'
zipPreds$Rounded <- round(zipPreds$Predicted)
zipPreds %>% head() %>% kable()
```