Finding Fine Wines

Timur Guler, Nick Landi, Spencer Bozsik, Ryan Folks

8/10/2021

Executive Summary

Our team began this project by trying to answer a single question - is it possible to use the physical and chemical properties of red and white wine to predict whether or not the wine is high quality? We took the perspective of a team of data scientists who was hired to advise a boutique wine seller. This seller is not interested in purchasing a high volume of wine, but is very interested in ensuring that they can get the highest ratio of good wines to bad wines in every one of their purchases, so as to maintain their elite reputation. This is a difficult task considering there are far more bad wines in the world than good ones.

The team decided to address this question using linear modeling. We used two data sets, one on red and one on white wines, which featured the wine's chemical and physical properties as well as their 1-10 quality rating. Since our client is interested in categorizing wine into "high" and "low" quality, our team transformed the 1-10 rating, classifying any wine with at least a 7 as "high quality."

Going into the project the team was already skeptical that we would be able to predict both red and white wine quality in one convenient model. Our limited knowledge of wine wasn't enough for us to make this important decision immediately, so we dug deeper. Although red and white wines share the same characteristics, initial data exploration revealed that red and white tend to have drastically different values in several of our potential quality predictors. Some of these characteristics include residual sugar, fixed acidity, and chlorides. Additionally, some characteristics, like fixed acidity, affected quality differently for red and white wines. These factors indicated to our team that red and white wine deserved their own models in order to best meet our client's needs.

Because our client's interest area of quality was binary and categorical, our team decided to use a logistic regression model. This is the most common model type when dealing with these types of questions, as it produces a 0-1 probability score for a wine being "high quality" based on a series of inputs, here, the wine's physical characteristics.

The model building process was an iterative one where we tried to compare several different combinations of the wine characteristics in order to most effectively predict high versus low quality wine. A combination of exploratory data analysis and model selection criteria methods led us to select a 7-predictor model for red wines and a 5-predictor model for whites. These models were validated through diagnostic procedures to ensure the logistic regression assumptions were satisfied, and were compared to other models to ensure maximum predictive and explanatory power. For red wines, alcohol percentage, sulphate levels, fixed acidity, and residual sugar increased the probability of being "high quality", while volatile acidity, density, and chlorides reduced this probability. For white wines, alcohol, sulphates, and pH increased "high quality" likelihood, while volatile acidity and chlorides reduced this likelihood. While these models are somewhat similar, the different magnitudes of predictor effects, as well as the slight differences in the predictors themselves, confirmed our decision to use separate models, and inform our client that there are important differences in how to select quality red and white wines.

After the selection process, our red and white wine quality prediction models were tested against a subset of wine data that had been reserved for this purpose. Our red-wine model correctly identifies "high quality" wines 71% of the time, while our white wine model achieves a 60% rate for top wines.

While our models don't predict high quality wines perfectly, they are an incredible supplement to human judgement alone. We recommend that our client use our models in conjunction with their years of experience with wine. We believe that marrying these two sources of information will lead to a much higher proportion of high-quality wines in stock, which in turn will lead to happy, loyal customers and less money wasted on low-quality wines.

In the future, our team would love to add more characteristics to our wine datasets that aren't limited to the wine's physical and chemical properties. These new predictive variables could include what region the wine comes from, what season it was produced in, and age of the wine. The models that we built aren't only useful to our current client. This information could be informative for a wine producer to see if there are ways they can tweak the wine's characteristics in order to engineer a higher-quality wine.

Exploratory Data Analysis

Our goals: Our team took on the business challenge of advising a boutique wine store who wishes to identify high-quality red and white wines based on their physical and chemical characteristics. As a selective buyer, the agent wishes to only select the upper echelon of wines, and to minimize the number of low quality wines accidentally purchased. In addition to selecting high quality wines, the purchasing agent also wishes to learn which factors determine wine quality, and if and how these differ between white and red wines.

Our team used two data sets to answer these questions. One contained physical and chemical characteristics of red wine (1599 observations), and the other contained physical and chemical characteristics of white wine (4898 observations). This wine type imbalance, while not large enough to prevent meaningful analysis, was critical to keep in mind throughout our process.

Each data set contained 13 variables - an index column, a discrete 1-10 measure of quality, and the following physical/chemical characteristics of the wine:

- fixed acidity
- volatile acidity
- citric acidity
- residual sugar
- chlorides
- free sulfur dioxide
- total sulfur dioxide
- density
- pH
- sulphates
- alcohol

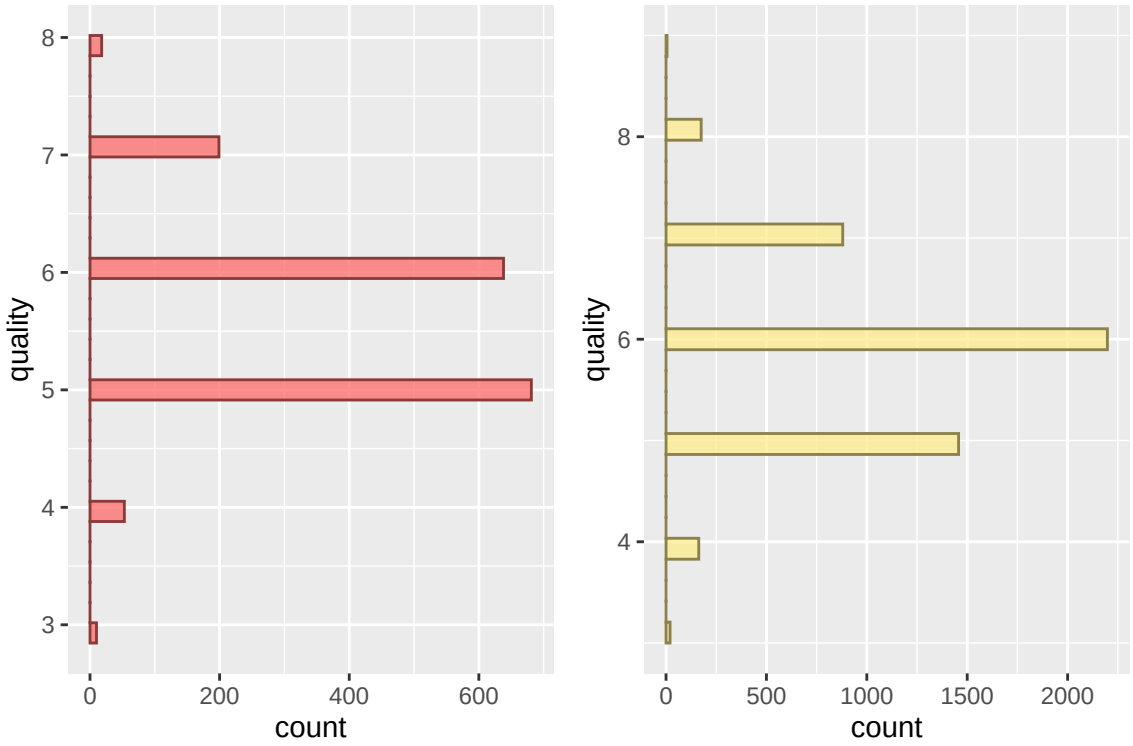
We removed the index column, as well as free sulfur dioxide, due to its high correlation with total sulfur dioxide in both red and white wine. Next, we performed a test-train split for both red and white wines with a fixed seed of 1, using 80% of each group for training data and the remaining 20% as testing data. Further discussions of "the data" during the EDA, model building, and model assumption validation stages will refer to the training data.

One fundamental issue in our analysis was the determination of a cutoff point for "high quality" wine. Based on the distribution for both red and white wines, we picked a cutoff quality rating of 7. This encompassed 13.5% of red wines and 21.4% of white wines. The same cutoff was used for both wine types for several reasons:

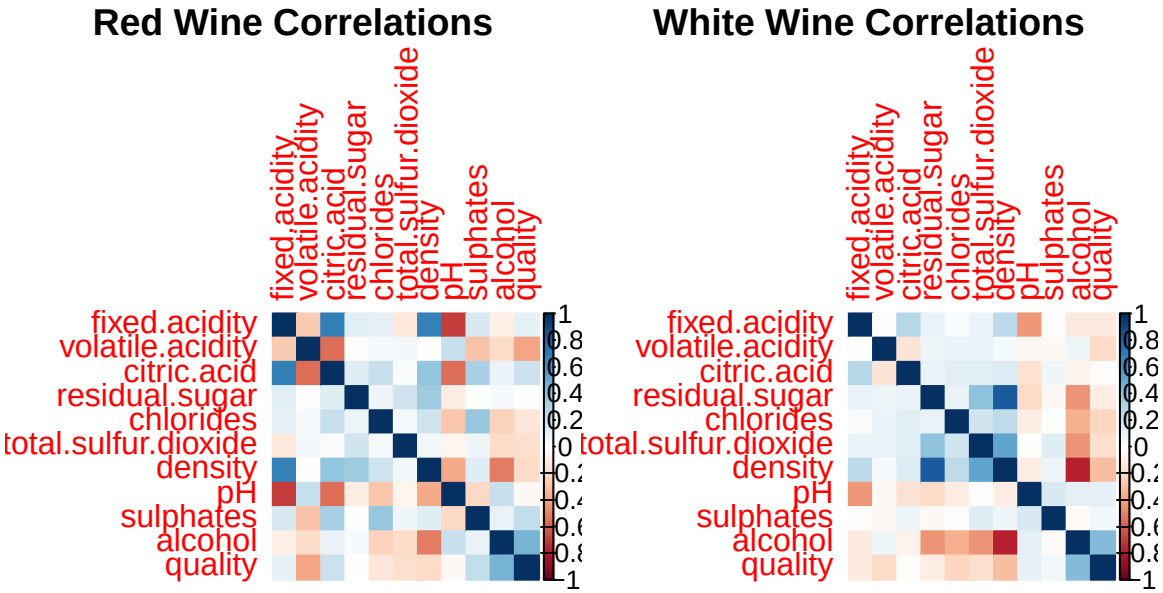
1. These proportions represent a level of distinction without making rarity a statistical concern
2. No determination had been made at this stage as to whether separate models would be used for white and red wine

3. While there was a statistically significant difference in mean quality levels (p=2.2e-16), the actual difference was not practically meaningful (5.877909 for whites vs. 5.636023 for reds).

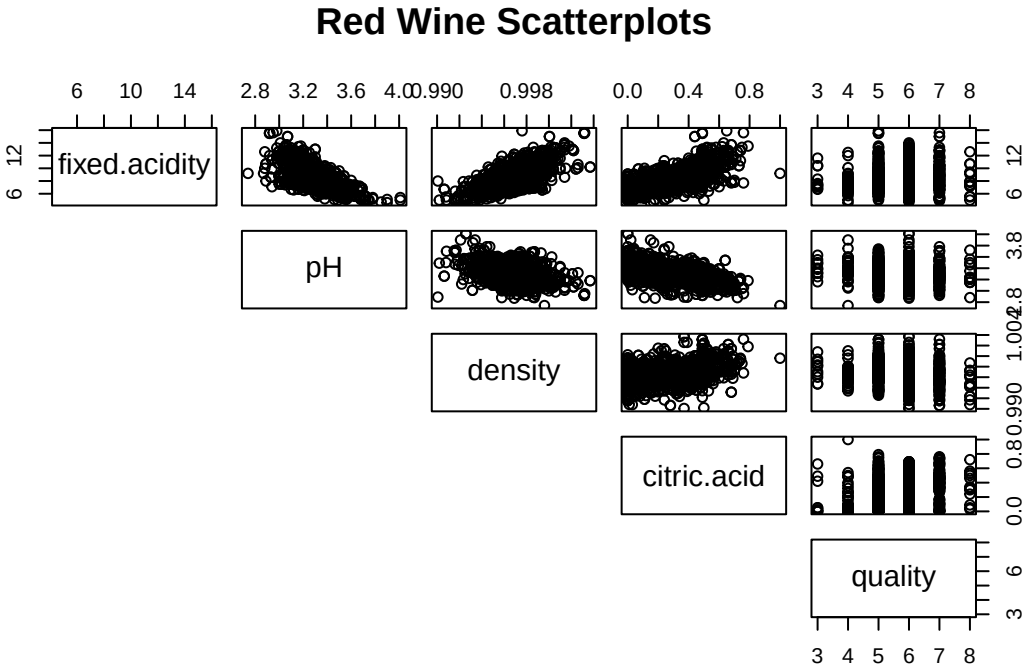
We also added the variable "color" to identify the wines, and created a combined training data set in addition to the separate red and white training sets.

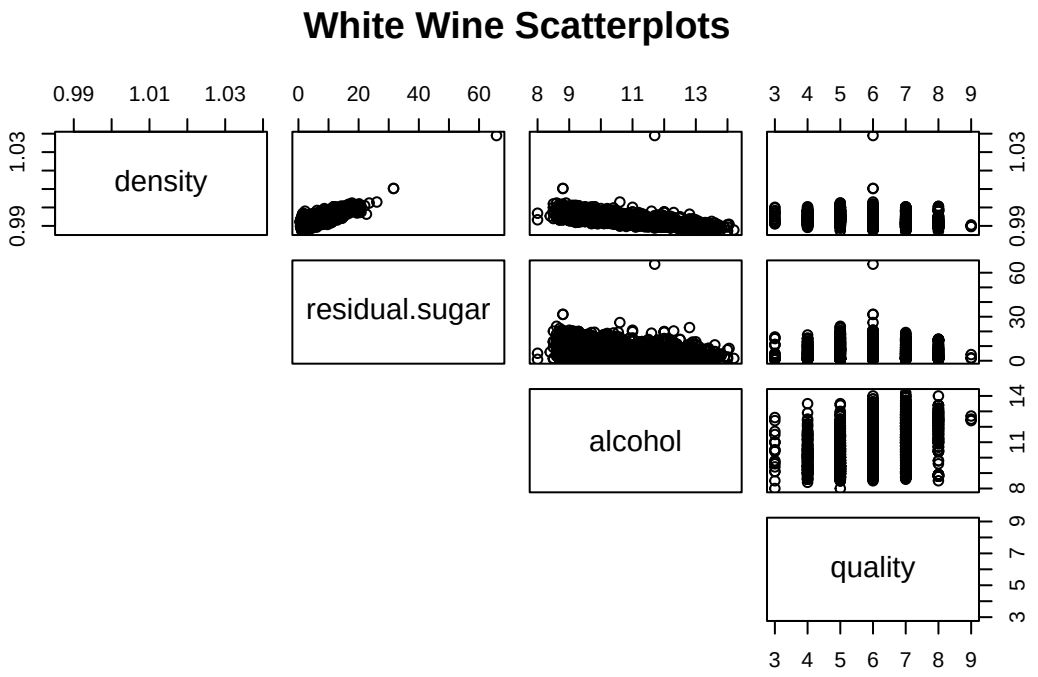


Our team had several goals during the EDA process. Our first goal was to understand the relationships between variables for both red and white wines. We accomplished this by examining correlations and scatterplots.



This shows us that correlations between variables differ somewhat between white and red wines, and that each type of wine has some strong relationships between variables, which we will need to consider when looking for multicollinearity (e.g. fixed acidity with pH, density, and citric acid for red wines, density with residual sugar and alcohol in white wines). Looking at these relationships in more detail with scatterplots, we see that they are fairly tight, and somewhat linear.



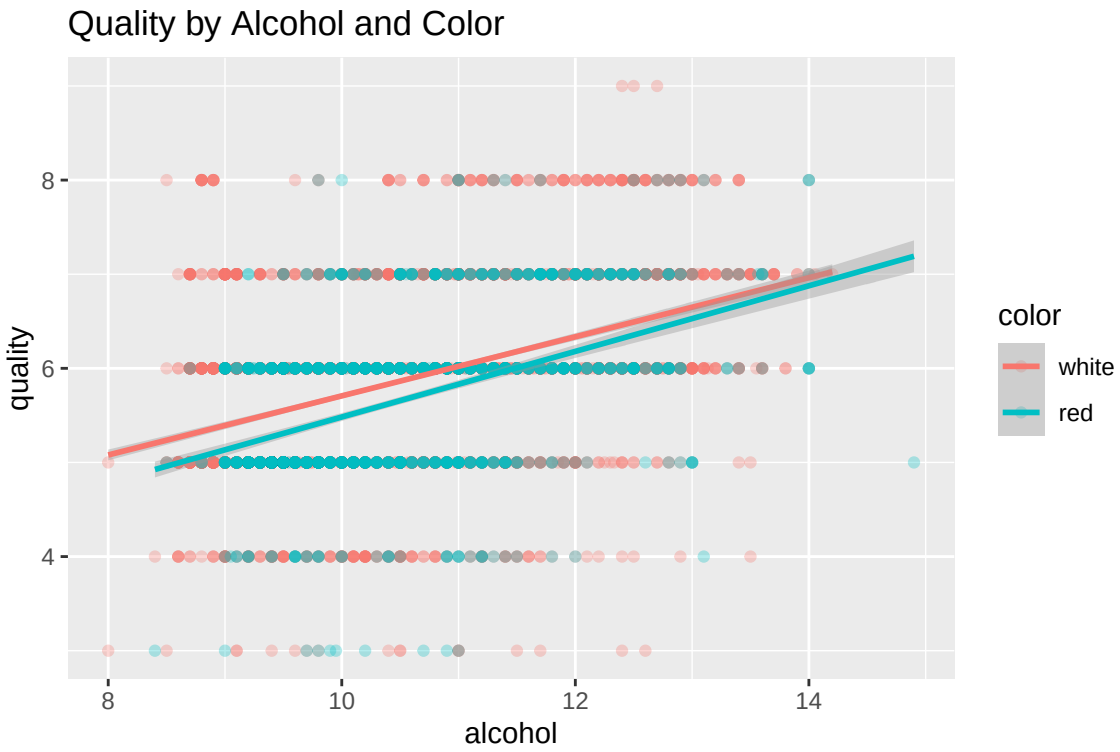


In addition to a general understanding of the variables' distributions and their relationships with each other, we were specifically interested in:

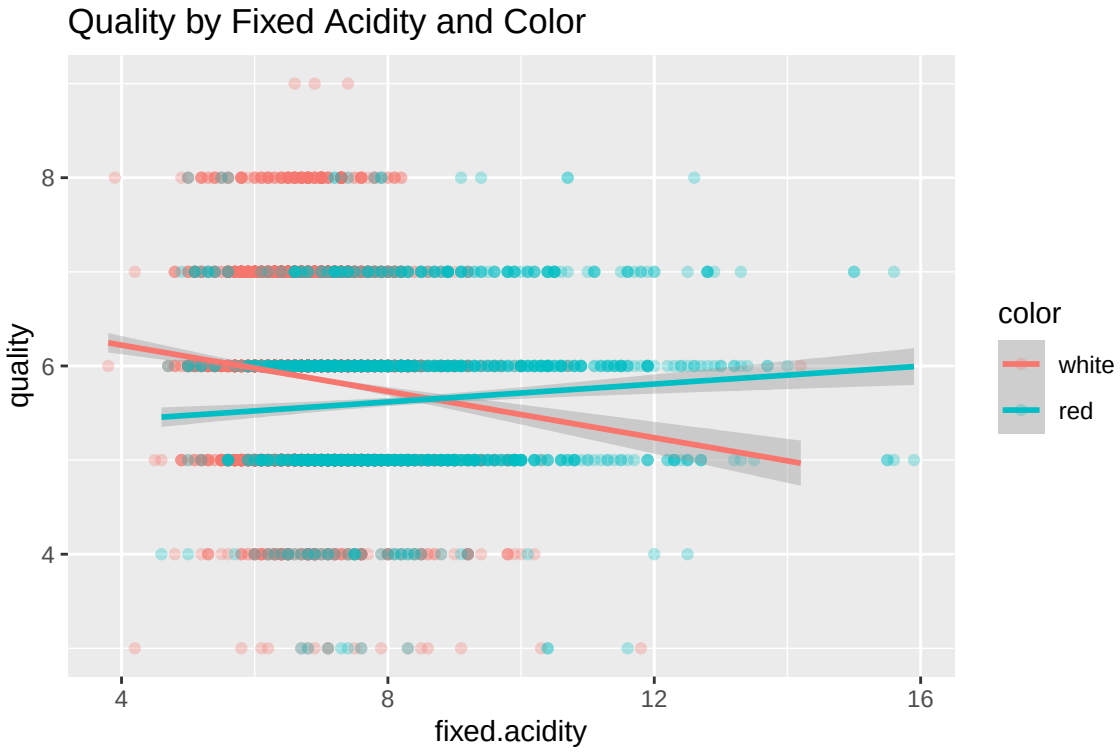
- the relationships between the physical/chemical characteristics of wine and quality
- whether it would be more effective to build a combined model or separate models for red and white

First, we looked at scatterplots of quality vs. physical characteristic by color using best-fit regression lines, as well as boxplots of quality category vs physical characteristic by color. This led to three key takeaways:

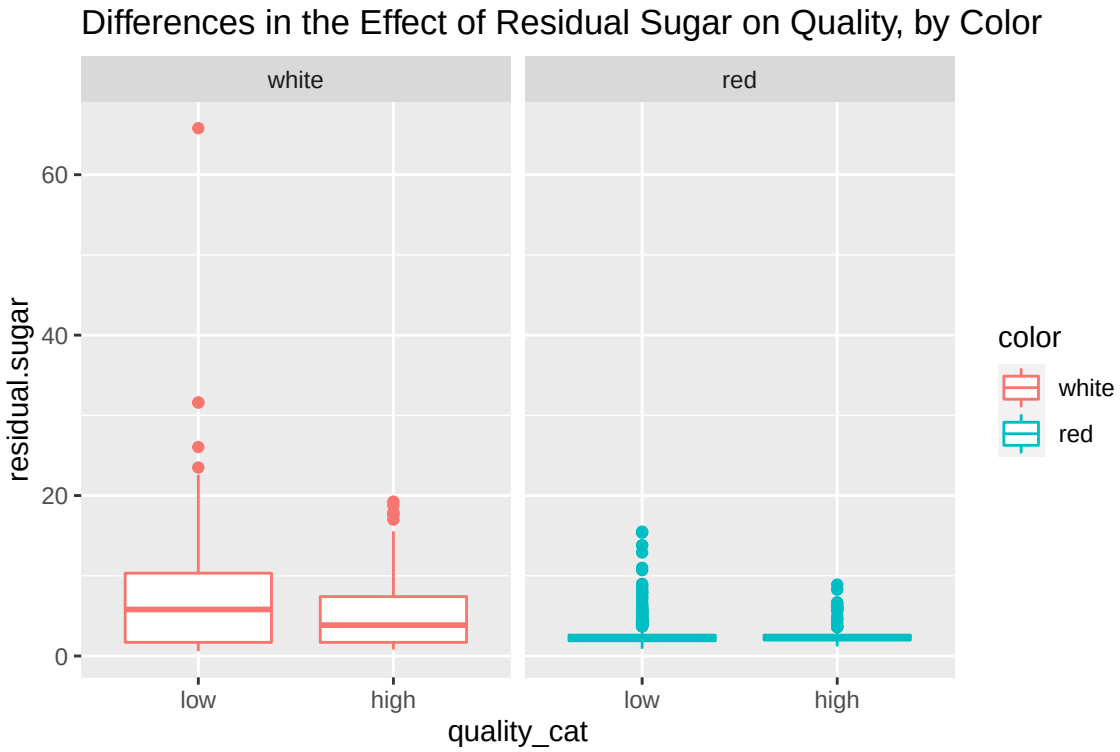
1. While several characteristics seem to have relationships with quality, a visual analysis show low "tightness of fit" for each potential predictor, meaning that it will likely take the combination of several weak predictors to yield a good prediction model



2. Some physical characteristics (notably fixed acidity, citric acidity, chlorides, pH, sulphates) have differing effects on quality in red vs white wines



3. The range for some physical characteristics (notably residual sugar) is markedly different between red and white wines, leading to potential extrapolation issues if a combined model is used.



Based on takeaways (2) and (3), as well as the difference in sizes between the red and white data sets, our team determined that it would be best to use two separate models for predicting high vs. low quality - one for red wines, and one for white wines.

Red Wine Model

Model Building

Our initial Exploratory Data Analysis suggests that we should build two separate models to classify the quality of red and white wines as "low" or "high". We decided to employ the all possible regressions procedure to determine which of the ten predictors may be considered the most important in predicting the quality category of red wine. All-possible-regressions is an automated procedure that tests all possible subsets of the set of potential independent variables, which in the context of our problem is 2^10 = 1024 possible subsets. We set the default value for nbest to 1, which tells the algorithm to return the one best set of predictors (based on R2) for each number of possible predictors. We also set nvmax to 10 so all possible subsets are displayed.

```
## Subset selection object
## Call: regsubsets.formula(quality_cat ~ ., data = train_red, nbest = 1,
##     nvmax = 10)
## 10 Variables  (and intercept)
##                      Forced in Forced out
## fixed.acidity            FALSE      FALSE
## volatile.acidity         FALSE      FALSE
## citric.acid              FALSE      FALSE
```

```
## residual.sugar           FALSE      FALSE
## chlorides                FALSE      FALSE
## total.sulfur.dioxide     FALSE      FALSE
## density                  FALSE      FALSE
## pH                       FALSE      FALSE
## sulphates                FALSE      FALSE
## alcohol                  FALSE      FALSE
## 1 subsets of each size up to 10
## Selection Algorithm: exhaustive
##          fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1  ( 1 )  " "           " "              " "         " "            " "
## 2  ( 1 )  " "           "*"              " "         " "            " "
## 3  ( 1 )  " "           "*"              " "         " "            " "
## 4  ( 1 )  "*"           "*"              " "         " "            " "
## 5  ( 1 )  "*"           "*"              " "         " "            " "
## 6  ( 1 )  "*"           "*"              " "         "*"            " "
## 7  ( 1 )  "*"           "*"              " "         "*"            "*"
## 8  ( 1 )  "*"           "*"              " "         "*"            "*"
## 9  ( 1 )  "*"           "*"              "*"         "*"            "*"
## 10  ( 1 ) "*"           "*"              "*"         "*"            "*"
##          total.sulfur.dioxide density pH  sulphates alcohol
## 1  ( 1 )  " "                  " "     " " " "       "*"
## 2  ( 1 )  " "                  " "     " " " "       "*"
## 3  ( 1 )  " "                  " "     " " "*"       "*"
## 4  ( 1 )  " "                  " "     " " "*"       "*"
## 5  ( 1 )  " "                  "*"     " " "*"       "*"
## 6  ( 1 )  " "                  "*"     " " "*"       "*"
## 7  ( 1 )  " "                  "*"     " " "*"       "*"
## 8  ( 1 )  "*"                  "*"     " " "*"       "*"
## 9  ( 1 )  "*"                  "*"     " " "*"       "*"
## 10  ( 1 ) "*"                  "*"     "*" "*"       "*"

## [1] 9

## [1] 8

## [1] 7

##              Critera Number of predictors selected
## 1                BIC                             7
## 2         Mallows CP                             8
## 3 Adjusted R squared                             9
```

After running the algorithm, we extract the best models based on the following criteria: Adjusted R^2, Mallows C_p, and BIC. For allreg_red, model 9 has the best adjusted R^2, model 8 has the best Mallow's C_p and model 7 has the best BIC. In our case, the model number corresponds to the number of predictors being used. Based on these results, we select the proposed 7-predictor model for our initial model building process. We came to this consensus to reduce model complexity as the three proposed models were deemed valid. The 7-predictor model uses the following predictor variables to classify the quality of red wine as "low" or "high": alcohol, volatile.acidity, sulphates, fixed.acidity, density, residual.sugar and chlorides.

```
##
```

```
## Call:
## glm(formula = quality_cat ~ alcohol + volatile.acidity + sulphates +
##     fixed.acidity + density + residual.sugar + chlorides, family = "binomial",
##     data = train_red)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.1497  -0.4493  -0.2382  -0.1540   2.8257
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)       285.10221  100.45702   2.838  0.00454 **
## alcohol             0.72351    0.12098   5.980 2.23e-09 ***
## volatile.acidity   -2.82651    0.70873  -3.988 6.66e-05 ***
## sulphates           3.23950    0.57682   5.616 1.95e-08 ***
## fixed.acidity       0.36775    0.08657   4.248 2.16e-05 ***
## density          -299.78128  100.88415  -2.972  0.00296 **
## residual.sugar      0.17416    0.07658   2.274  0.02295 *
## chlorides          -7.54886    3.33472  -2.264  0.02359 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1013.65  on 1278  degrees of freedom
## Residual deviance:  722.12  on 1271  degrees of freedom
## AIC: 738.12
##
## Number of Fisher Scoring iterations: 6
```

The fitted 7-predictor logistic regression equation for red wine is displayed as follows:

$\log\left(\frac{\hat{\pi}_{quality\_cat}}{1-\hat{\pi}_{quality\_cat}}\right) = 285.10221 + 0.72351 * x_{alcohol} - 2.82651 * x_{volatile.acidity} + 3.23950 * x_{sulphates} + 0.36775 * x_{fixed.acidity} - 299.78128 * x_{density} + 0.17416 * x_{residual.sugar} - 7.54886 * x_{chlorides}$

The Z test associated with all the coefficients are considered statistically significant at the .05 level, which suggests that all 7 predictor variables should be kept in our model.

Model Diagnostics

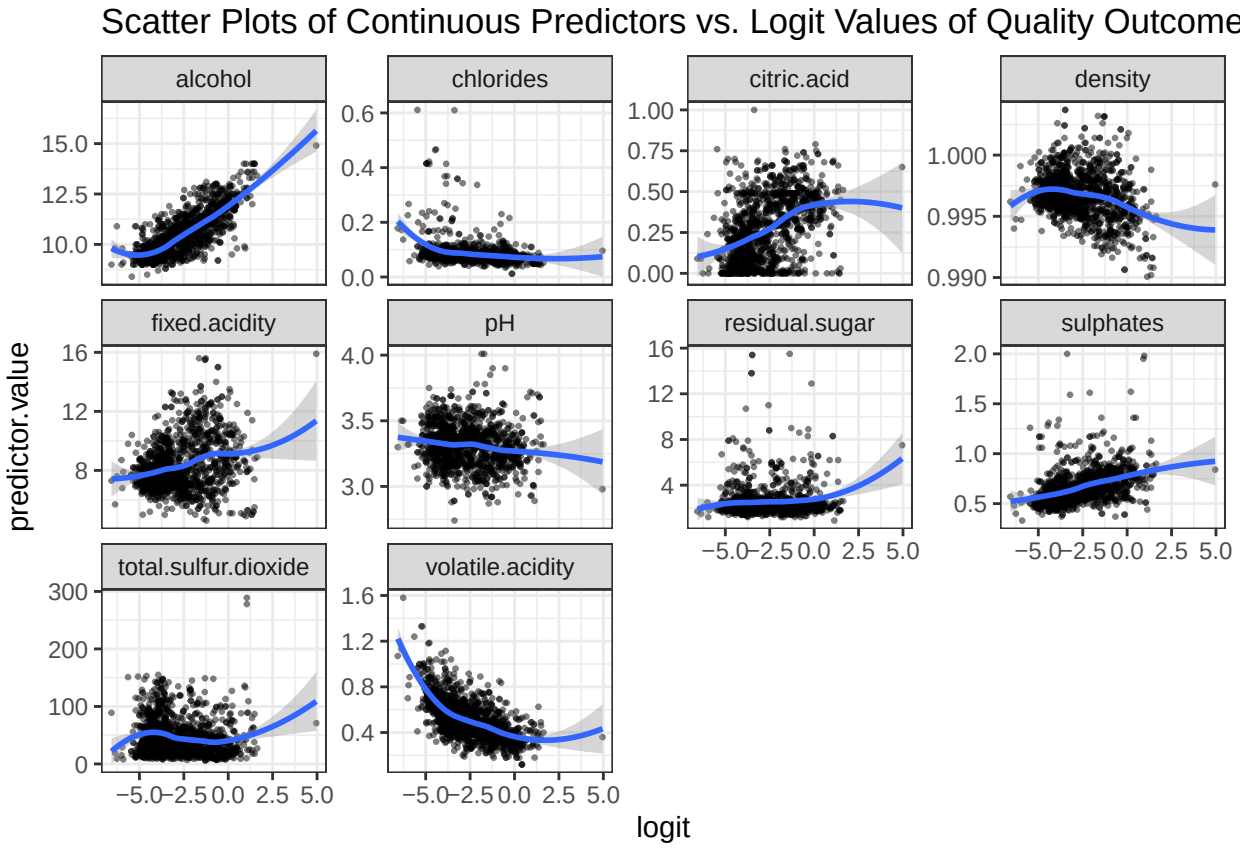
The next step in our procedure is to run model diagnostics to confirm our initial 7-predictor model meets all the assumptions for logistic regressions, which are defined as follows:

1) There is a linear relationship between the logit of the outcome and each predictor variables.
2) There are no influential values (extreme values or outliers) in the continuous predictors
3) There are no high intercorrelations (i.e. multicollinearity) among the predictors.

To determine that the first assumption has been met, we check the linear relationship between continuous predictor variables and the logit of the outcome. This is done by visually inspecting the scatter plot between each predictor variable and the logit values.

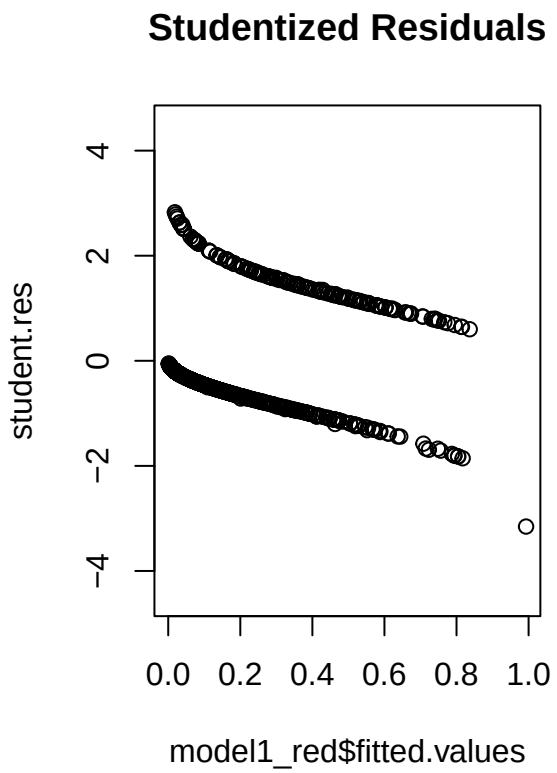
```
##   723  1451   165    21   994    91
## "neg" "neg" "neg" "neg" "neg" "neg"
```

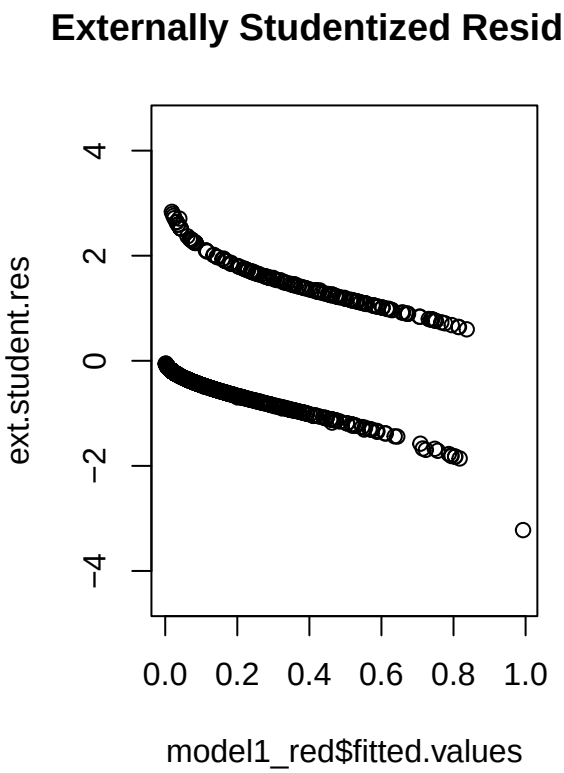
```
## Warning: Predicate functions must be wrapped in `where()`.
##
##   # Bad
##   data %>% select(is.numeric)
##
##   # Good
##   data %>% select(where(is.numeric))
##
## i Please update your code.
## This message is displayed once per session.

## `geom_smooth()` using formula 'y ~ x'
```



The smoothed scatter plots show that alcohol, sulphates and volatile.acidity are quite linearly associated with the quality_cat outcome in logit scale. The smoother scatter plots for the other predictor variables show little to no linear association with the quality_cat outcome in logit scale, which leads us to believe they are weaker predictors. None of the scatterplots lead us to believe that assumption 1 has been violated, and we therefore move on without transforming variables and begin to explore our next assumption.

Next, we generate the residuals, studentized residuals and externally studentized residuals associated with our 7-predictor model to help detect any outliers. Studentized residuals are the quotients resulting from the division of a residual by an estimate of their standard deviation, which as a measure allows us to account for the fact that residuals do not have constant variance. Externally studentized residuals is another method of scaling residuals, and is the deleted residual divided by its estimated standard deviation. We combine these residual values into a dataframe and create scatterplots of the studentized and externally studentized residuals in order to visually inspect for the presence of outliers.





The resulting plots look very similar, which leads us to believe there are no outliers in our model.

Next, we calculate leverages to identify how far each observation is from the center of the predictor space. If the leverage is greater than $\frac{2p}{n}$, the associated observation may be deemed to have high leverage and is considered outlying in the predictor space. In this case p, which corresponds to the number of predictor variables in our model, is 7 and n, the number of records in our training set for red wine, is 1279.

Due to the number of observations that were found to have high leverage, we decided to calculate Cook's Distance in order to determine if these high leverage observations were influential and were consequently negatively affecting our regression model. Cook's distance can be interpreted as the squared Euclidean distance that the vector of fitted values moves when observation i is removed from the regression model. We decided to use Cook's distance measure as we are mainly concerned with the predictive ability of our logistic model. A large value indicates that the observation has a large influence on the results. The Cook's Distance of each observation was compared to the a cut off value generated by the following F distribution: F0.5,p,n-p, where p is 7 and n is 1279.

```
## named numeric(0)
```

None of the generated Cook's distance values exceeded the cutoff value, which leads us to believe that none of our observations in our model are influential. We conclude that assumption 2 has been met, and move on to the final assumption.

Next, we check for multicollinearity in our model. Multicollinearity corresponds to a situation where two or more predictors are linearly dependent on each other. When two or more predictors are linearly dependent, they do not provide independent information to the response, resulting in large standard errors for the coefficients. We can check for multicollinearity by computing the variance inflation factors (VIFs) for each predictor variable. As a rule of thumb, A VIF value that exceeds 5 indicates a problematic amount of collinearity and needs to be explored, whereas a value greater than 10 indicates a serious amount of collinearity and must be immediately addressed.

```
##          alcohol volatile.acidity        sulphates    fixed.acidity
##         2.122107         1.257059         1.412832         2.784768
##          density   residual.sugar        chlorides
##         4.407765         1.415162         1.314060
```

The resulting VIFs are below 5 for all 7 of the predictor variables in our model, which leads us to conclude that the third assumption has been satisfied.

Likelihood Ratio Tests

After checking that all the model assumptions have been met, we move on to comparing our 7-predictor model with other possible logistic regression models using Likelihood Ratio Tests (LRTs) to evaluate if our proposed model is useful in classifying red wine as either "low" or "high" quality (quality_cat).

First, we compare our 7-predictor model to the intercept only model.

The null hypothesis is:

$H_0 : \beta_1 = \beta_2 = \ldots = \beta_7 = 0$

versus the alternative hypothesis:

$H_a : \exists x \in \{1, 2, \ldots, 7\} s.t. \beta_x \neq 0$

We calculate the test statistic, ΔG^2, which measures the difference in loglikelihoods of the models we are comparing, and compare it with a χ^2 distribution, where the degrees of freedom equal the number of coefficients we are testing (7). For this comparison, the test statistic is the difference in the null deviance and residual deviance of the 7-predictor model.

```
## [1] 291.5332
```

```
## [1] 0
```

The generated test statistic is 291.5332 with an associated p-value of 0. Therefore, we reject the null hypothesis and conclude that our 7-predictor model is useful and should be selected over the intercept-only model.

Next, we decide to compare our 7-predictor model to the full (10-predictor) model.

The null hypothesis is:

$H_0 : \beta_{citic.acid} = \beta_{total.sulfur.dioxide} = \beta_{pH} = 0$

versus the alternative hypothesis:

$H_a : \exists x \in \{citric.acid, total.sulfur.dioxide, pH\} s.t. \beta_x \neq 0$

Again we calculate the test statistic, and compare it with a χ^2 distribution with 3 degrees of freedom (as we are testing 3 coefficients). For this comparison, the test statistic is the difference in the residual deviances of both models.

```
## [1] 13.3604
```

```
## [1] 0.003918644
```

The generated test statistic is 13.3604 with an associated p-value of 0.003918644. Therefore, we reject the null hypothesis and conclude that the full (10-predictor) model is more useful than the 7-predictor model.

We found this result to be interesting, and decided to explore the full model in more depth.

```
##
## Call:
## glm(formula = quality_cat ~ ., family = "binomial", data = train_red)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9793  -0.4447  -0.2325  -0.1351   2.7740
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)           3.701e+02  1.206e+02   3.068 0.002153 **
## fixed.acidity         3.678e-01  1.379e-01   2.667 0.007648 **
## volatile.acidity     -2.263e+00  8.422e-01  -2.687 0.007199 **
## citric.acid           8.881e-01  9.443e-01   0.941 0.346946
## residual.sugar        2.687e-01  8.304e-02   3.236 0.001211 **
## chlorides            -8.804e+00  3.527e+00  -2.496 0.012545 *
## total.sulfur.dioxide -1.200e-02  3.643e-03  -3.293 0.000992 ***
## density              -3.870e+02  1.231e+02  -3.143 0.001670 **
## pH                    8.468e-01  1.082e+00   0.782 0.433965
## sulphates             3.853e+00  6.168e-01   6.247 4.18e-10 ***
## alcohol               5.865e-01  1.467e-01   3.998 6.39e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1013.65  on 1278  degrees of freedom
## Residual deviance:  708.76  on 1268  degrees of freedom
## AIC: 730.76
##
## Number of Fisher Scoring iterations: 6
```

We see that the Z test scores associated with the coefficients for predictors pH and citric.acid are insignificant. The next step is to compute VIFs for each of the 10 predictors in the full model to determine if multicollinearity exists between any of the variables.

```
##              alcohol     volatile.acidity            sulphates
##             3.155038             1.744373             1.474041
##        fixed.acidity              density       residual.sugar
##             7.985849             6.512192             1.669128
##            chlorides                   pH          citric.acid
##             1.501571             3.387246             3.186454
## total.sulfur.dioxide
##             1.217643
```

The resulting VIFs suggests that none of the variables are linearly dependent on one another, and we conclude that pH and citric.acid are not significant predictors for classifying red wine as "low" or "high" an drop them from the full model.

Although it is not necessary to test when dropping predictors due to high multicolinearity, we will run the test just to be safe.
To validate dropping both of these predictors from the full model, we compare our 8-predictor model to the full (10-predictor) model.

The null hypothesis is:

$H_0 : \beta_{citric.acid} = \beta_{pH} = 0$

versus the alternative hypothesis:

$H_a : \beta_{citric.acid} \neq 0$ or $\beta_{pH} \neq 0$

Again we calculate the test statistic, and compare it with a χ^2 distribution with 2 degrees of freedom (as we are testing 2 coefficients).

```
##
## Call:
## glm(formula = quality_cat ~ alcohol + volatile.acidity + sulphates +
##     fixed.acidity + density + residual.sugar + chlorides + total.sulfur.dioxide,
##     family = "binomial", data = train_red)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0118  -0.4501  -0.2346  -0.1351   2.7601
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)           3.165e+02  1.030e+02   3.071 0.002131 **
## alcohol               6.657e-01  1.238e-01   5.376 7.61e-08 ***
## volatile.acidity     -2.720e+00  7.113e-01  -3.825 0.000131 ***
## sulphates             3.739e+00  5.985e-01   6.248 4.16e-10 ***
## fixed.acidity         3.373e-01  8.809e-02   3.829 0.000128 ***
## density              -3.304e+02  1.034e+02  -3.194 0.001403 **
## residual.sugar        2.517e-01  7.783e-02   3.234 0.001220 **
## chlorides            -8.472e+00  3.405e+00  -2.488 0.012838 *
## total.sulfur.dioxide -1.188e-02  3.626e-03  -3.275 0.001057 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1013.65  on 1278  degrees of freedom
## Residual deviance:  710.02  on 1270  degrees of freedom
## AIC: 728.02
##
## Number of Fisher Scoring iterations: 6
```

```
## [1] 1.262305
```

```
## [1] 0.5319783
```

The generated test statistic is 1.262305 with an associated p-value of 0.5319783. Therefore, we fail to reject the null hypothesis and select the 8-predictor model over the full model.

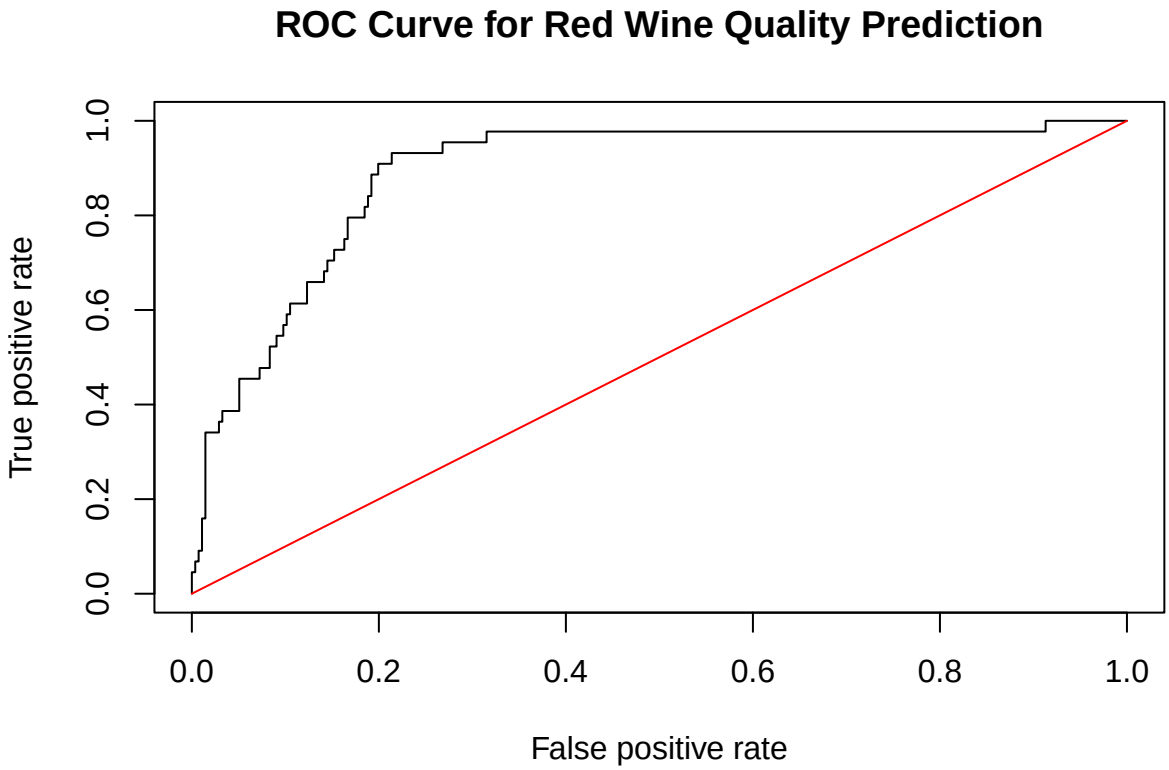
The fitted 8-predictor logistic regression equation for red wine is displayed as follows:

$\log\left(\frac{\hat{\pi}_{quality\_cat}}{1-\hat{\pi}_{quality\_cat}}\right) = 316.5 + 0.6657 * x_{alcohol} - 2.720 * x_{volatile.acidity} + 3.739 * x_{sulphates} + 0.3373 * x_{fixed.acidity} - 330.4 * x_{density} + 0.2517 * x_{residual.sugar} - 8.472 * x_{chlorides} - 0.01188 * x_{total.sulfur.dioxide}$

We now have two candidate models for classifying the quality of red wine: our original 7-predictor model (model1_red) and the 8-predictor model (model2_red). We decide to test both these models using the test_red dataset in order to determine which is more useful in answering our research question.

Model Validation with Test Dataset

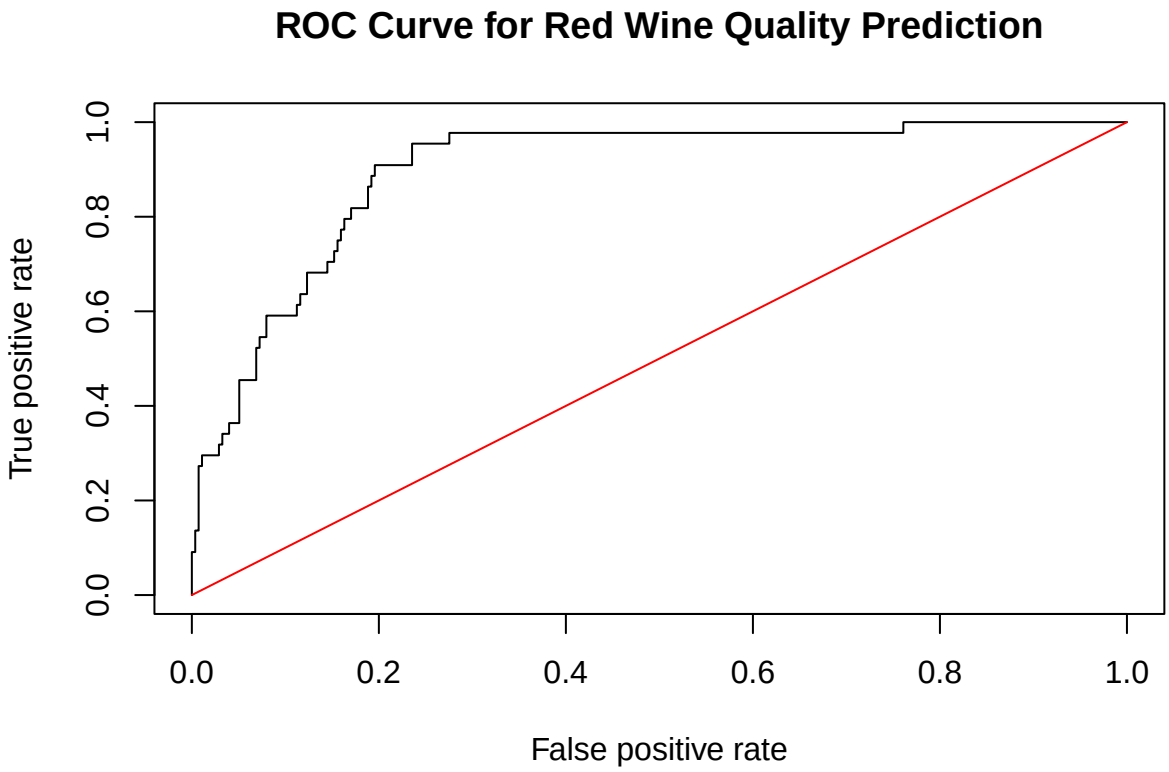
First, we store the predicted probabilities of the test_red data based on our 7-predictor logistic regression model in the variable preds. We then transform this data using the prediction() function and store the values of the true positive rate and false positive rate in the roc_result object. We then use this object to generate a ROC curve for our 7-predictor logistic regression model. The ROC curve plots the true positive rate (TPR) against the false positive rate (FPR) of the model as the threshold value is varied from 0 to 1. The ROC curve generated by our 7-predictor model lies above the diagonal, which indicates the logistic regression does a better job than random guessing when classifying red wines as "low" or "high" quality. We also generate an Area Under the Curve (AUC) associated with this ROC curve. The AUC value for our model is 0.889574, which again provides evidence that our model does better at classifying red wines then random guessing (represented by an AUC score of 0.5). Finally, we produce a confusion matrix, which contains the number of observations in each class ("low" and "high" quality red wine) and the number of predicted observations in each class (for test_red) generated by our 7-predictor model using a threshold value of 0.5.




```
## [[1]]
## [1] 0.8896574

##
##        FALSE TRUE
##   low    270    6
##   high    29   15
```

Next, we follow a similar assessment procedure using the test_red dataset and the 8-predictor model.The ROC curve generated by our 8-predictor model lies above the diagonal, which indicates the logistic regression

does a better job than random guessing when classifying red wines as "low" or "high" quality. The AUC value for this model is 0.8978096, which again provides evidence that our 8-predictor model does slightly better at classifying red wines than our 7-predictor model. Finally, we produce a confusion matrix generated by the 8-predictor model using a threshold value of 0.5.




```
## [[1]]
## [1] 0.8978096

##
##        FALSE TRUE
##   low    267    9
##   high    29   15
```

In terms of our research question, we are most interested in maximizing the ratio of the true positive results to the false positive results. The number of false positive results (low quality wines incorrectly categorized as high quality) is 6 and the number of true positive results (high quality wines correctly categorized as high quality) is 15 for our 7-predictor model, whereas the number of false positive results is 9 and the number of true positive results is 15 for our 8-predictor model. The calculated ratio of true positive results to false positive results is larger for our 7-predictor model, at .714, so we select it over the 8-predictor model, at .625.

White Wine Model

Model Building

Our goal, like with red wines, is to select a model which maximizes the ratio of true positives to false positives.

The model selection process for white wine was by no means straightforward. We began with a 7 predictor model, then moved to the full model, then to an 8 predictor model, and finally settled upon a 5 predictor model which performed nearly identically to any of the previous models, contained fewer predictors, and contained no serious multicolinearity.

60 percent of the wines selected by this final model are truly high quality, which is far higher than the 22 percent that would be average if we were randomly selecting white wines.

Searching all First Order Models

As in the previous section, we begin by seeing what the BIC, Mallows C_p, and adjusted R^2 select as their best model from all the predictors.

```
## Subset selection object
## Call: regsubsets.formula(quality_cat ~ ., data = train_white, nbest = 1,
##     nvmax = 10)
## 10 Variables  (and intercept)
##                       Forced in Forced out
## fixed.acidity             FALSE      FALSE
## volatile.acidity          FALSE      FALSE
## citric.acid               FALSE      FALSE
## residual.sugar            FALSE      FALSE
## chlorides                 FALSE      FALSE
## total.sulfur.dioxide      FALSE      FALSE
## density                   FALSE      FALSE
## pH                        FALSE      FALSE
## sulphates                 FALSE      FALSE
## alcohol                   FALSE      FALSE
## 1 subsets of each size up to 10
## Selection Algorithm: exhaustive
##           fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1  ( 1 )  " "           " "              " "         " "            " "
## 2  ( 1 )  " "           "*"              " "         " "            " "
## 3  ( 1 )  " "           "*"              " "         "*"            " "
## 4  ( 1 )  " "           "*"              " "         "*"            " "
## 5  ( 1 )  " "           "*"              " "         "*"            " "
## 6  ( 1 )  " "           "*"              " "         "*"            " "
## 7  ( 1 )  "*"           "*"              " "         "*"            " "
## 8  ( 1 )  "*"           "*"              " "         "*"            "*"
## 9  ( 1 )  "*"           "*"              " "         "*"            "*"
## 10  ( 1 ) "*"           "*"              "*"         "*"            "*"
##           total.sulfur.dioxide density pH  sulphates alcohol
## 1  ( 1 )  " "                  " "     " " " "       "*"
## 2  ( 1 )  " "                  " "     " " " "       "*"
## 3  ( 1 )  " "                  " "     " " " "       "*"
## 4  ( 1 )  " "                  " "     "*" " "       "*"
## 5  ( 1 )  " "                  "*"     "*" " "       "*"
## 6  ( 1 )  " "                  "*"     "*" "*"       "*"
## 7  ( 1 )  " "                  "*"     "*" "*"       "*"
## 8  ( 1 )  " "                  "*"     "*" "*"       "*"
## 9  ( 1 )  "*"                  "*"     "*" "*"       "*"
## 10  ( 1 ) "*"                  "*"     "*" "*"       "*"

##              Critera Number of predictors selected
```

```
## 1                BIC                              7
## 2         Mallows CP                              7
## 3 Adjusted R squared                              7
```

All the criteria have selected the 7 predictor model as their best, so we will fit that model as our initial model and assess it.

```
##
## Call:
## glm(formula = quality_cat ~ fixed.acidity + volatile.acidity +
##     residual.sugar + density + pH + sulphates + alcohol, family = "binomial",
##     data = train_white)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3135  -0.6685  -0.4075  -0.2159   2.7680
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)       758.80532   97.82192   7.757 8.70e-15 ***
## fixed.acidity       0.61435    0.09825   6.253 4.02e-10 ***
## volatile.acidity   -3.75146    0.51978  -7.217 5.30e-13 ***
## residual.sugar      0.34613    0.03768   9.185  < 2e-16 ***
## density          -784.14780   99.11037  -7.912 2.54e-15 ***
## pH                  3.78385    0.47026   8.046 8.54e-16 ***
## sulphates           2.33484    0.38933   5.997 2.01e-09 ***
## alcohol             0.03626    0.12351   0.294    0.769
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4067.3  on 3917  degrees of freedom
## Residual deviance: 3310.1  on 3910  degrees of freedom
## AIC: 3326.1
##
## Number of Fisher Scoring iterations: 5
```

Alcohol fails the wald test. This is counter intuitive, since alcohol has the highest correlation with quality. Alcohol in white wines is highly correlated with both density and residual sugar as well, so multicolinearity is expected to be present in this model. Therefore, it is likely that alcohol is truly a good predictor, but multicolinearity is causing it to fail the wald test. Before addressing this issue, we should test whether this model is better than the intercept, and better than the model with all predictors.

Is the 7 predictor model better than the intercept only model?

Our hypothesis here are $H_0 : \beta_1 = \beta_2 = \ldots = \beta_7, H_a : \exists x \in \{1, 2, \ldots, 7\} s.t. \beta_x \neq 0$

```
## [1] 757.2668
```

```
## [1] 0
```

The code has produced a χ^2 test statistic of 757.2668, and a corresponding p-value of ≈ 0.
Therefore we reject H_0 and conclude that at least one of our predictors is linearly related to the log odds of our quality category variable.

Is the 10 predictor model better than the 7 predictor model?

Our hypothesis are $H_0 : \beta_8 = \beta_9 = \beta_{10}, H_a : \exists x \in \{8, 9, 10\} s.t. \beta_x \neq 0$

```
##
## Call:
## glm(formula = quality_cat ~ ., family = "binomial", data = train_white)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2947  -0.6730  -0.4100  -0.1802   2.9196
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)           7.148e+02  1.038e+02   6.888 5.64e-12 ***
## fixed.acidity         6.043e-01  1.016e-01   5.947 2.74e-09 ***
## volatile.acidity     -3.949e+00  5.367e-01  -7.357 1.88e-13 ***
## citric.acid          -7.248e-01  4.436e-01  -1.634  0.10228
## residual.sugar        3.276e-01  3.923e-02   8.352  < 2e-16 ***
## chlorides            -1.297e+01  4.251e+00  -3.050  0.00229 **
## total.sulfur.dioxide  1.944e-03  1.327e-03   1.465  0.14297
## density              -7.387e+02  1.052e+02  -7.022 2.19e-12 ***
## pH                    3.604e+00  4.790e-01   7.524 5.31e-14 ***
## sulphates             2.306e+00  3.911e-01   5.896 3.72e-09 ***
## alcohol               5.893e-02  1.260e-01   0.468  0.63997
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4067.3  on 3917  degrees of freedom
## Residual deviance: 3294.9  on 3907  degrees of freedom
## AIC: 3316.9
##
## Number of Fisher Scoring iterations: 6
```

```
## [1] 15.15094
```

```
## [1] 0.001692098
```

We get a χ^2 test statistic of 15.15094 and a corresponding p-value of ≈ 0.001.
Therefore, we reject the null hypothesis and conclude that at least one predictor is significant.
This means that we prefer the full model over the seven predictor model, and should add in the rest of the predictors.

The output shows that the full model has wald tests that fail, and from earlier correlation plots we know that many of the predictors are highly correlated with one another, so we must assess and fix multicolinearity.

Multicollinearity

```
##        fixed.acidity     volatile.acidity          citric.acid
##             2.623734             1.112232             1.171146
##       residual.sugar            chlorides total.sulfur.dioxide
##            11.651111             1.238923             1.438126
##              density                   pH            sulphates
##            25.513449             2.162567             1.137183
##              alcohol
##             7.041047
```

Both residual sugar and density have a VIF higher than 10, meaning that multicolinearity is present, so we should take them out of the model. We ran the hypothesis test for removing the two predictors, and found that the test still prefered the full model. However, we decided to ignore this test for the sake of producing a model that does not contain significant correlation among the predictors.

```
##
## Call:
## glm(formula = quality_cat ~ fixed.acidity + volatile.acidity +
##     citric.acid + chlorides + total.sulfur.dioxide + pH + sulphates +
##     alcohol, family = "binomial", data = train_white)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8837  -0.6874  -0.4266  -0.2088   2.9832
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -11.425022   1.393463  -8.199 2.42e-16 ***
## fixed.acidity          0.018970   0.060068   0.316  0.75215
## volatile.acidity      -3.627411   0.505871  -7.171 7.47e-13 ***
## citric.acid           -0.743569   0.434682  -1.711  0.08715 .
## chlorides            -18.988252   4.270095  -4.447 8.72e-06 ***
## total.sulfur.dioxide   0.001433   0.001224   1.171  0.24172
## pH                     0.854701   0.319972   2.671  0.00756 **
## sulphates              1.112092   0.358293   3.104  0.00191 **
## alcohol                0.787795   0.045869  17.175  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4067.3  on 3917  degrees of freedom
## Residual deviance: 3381.8  on 3909  degrees of freedom
## AIC: 3399.8
##
## Number of Fisher Scoring iterations: 6
```

```
##        fixed.acidity     volatile.acidity          citric.acid
##             1.326713             1.077087             1.162607
##            chlorides total.sulfur.dioxide                   pH
##             1.190123             1.322216             1.293976
##            sulphates              alcohol
##             1.056152             1.433535
```

While we no longer have any multicolinearity, fixed acidity, citric acid, and total sulfur dioxide fail their wald test, so we need to test if we can remove them from the model as well.

```
##
## Call:
## glm(formula = quality_cat ~ volatile.acidity + chlorides + pH +
##     sulphates + alcohol, family = "binomial", data = train_white)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8457  -0.6866  -0.4232  -0.2162   2.9042
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -11.30223    1.03397 -10.931  < 2e-16 ***
## volatile.acidity  -3.38080    0.48649  -6.949 3.67e-12 ***
## chlorides        -18.68852    4.15157  -4.502 6.75e-06 ***
## pH                 0.89333    0.27927   3.199  0.00138 **
## sulphates          1.10611    0.35568   3.110  0.00187 **
## alcohol            0.76496    0.04261  17.953  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4067.3  on 3917  degrees of freedom
## Residual deviance: 3385.7  on 3912  degrees of freedom
## AIC: 3397.7
##
## Number of Fisher Scoring iterations: 5
```

Our hypothesis are $H_0 : \beta_6 = \beta_7 = \beta_8, H_a : \exists x \in \{6, 7, 8\} s.t. \beta_x \neq 0$.

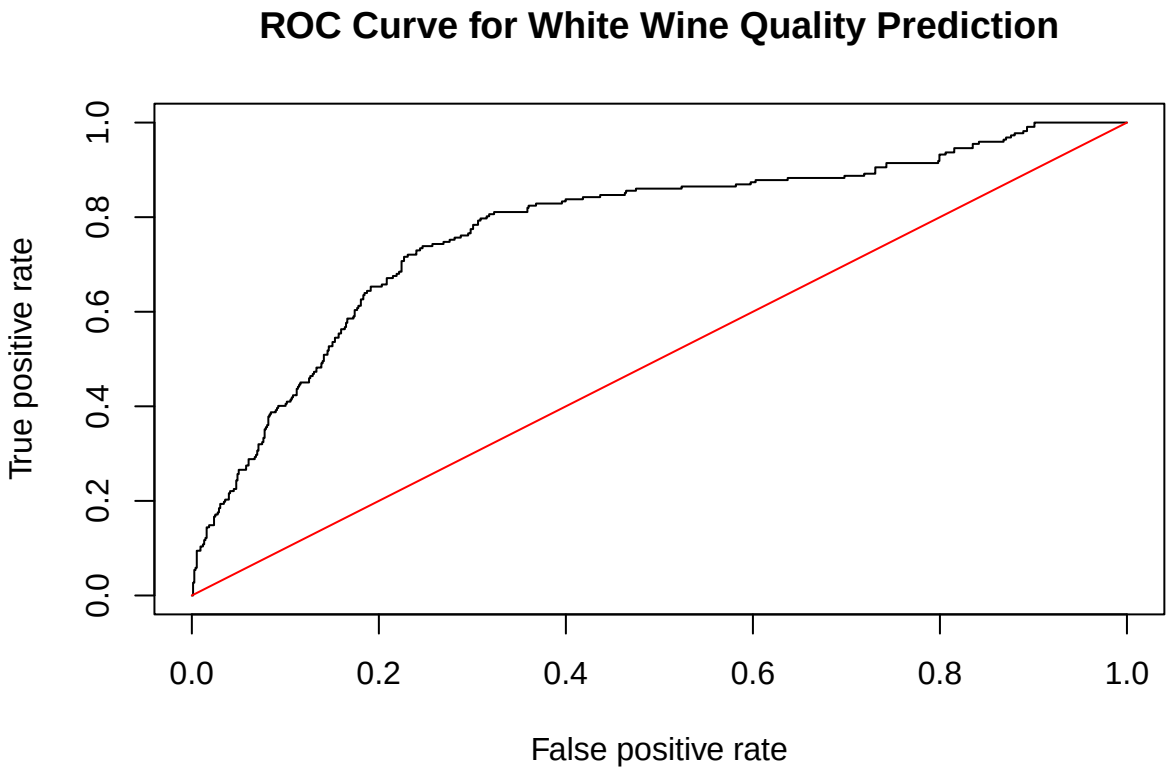
```
## [1] 3.954686
```

```
## [1] 0.266399
```

The χ^2 test statistic is 3.95 and has a corresponding p value of 0.27.

We fail to reject the null, so we may remove these three variables and continue with the 5 predictor model.

Assessing Predictive Accuracy




```
## [[1]]
## [1] 0.7773063
```

Now that we've arrived at this model, we can create an ROC curve for it. This ROC curve has an area of 0.77, which doesn't always translate to high predictive accuracy in our case, it is one metric which suggests that this model predicts well in general.

```
## [1] "Confusion matrix for the 7 predictor model"
```

```
##
##        FALSE TRUE
##   low    715   43
##   high   158   64
```

```
## [1] "Confusion matrix for the full model"
```

```
##
##        FALSE TRUE
##   low    716   42
##   high   159   63
```

```
## [1] "Confusion matrix for the 8 predictor model"
```

```
##
##        FALSE TRUE
##   low    722   36
##   high   167   55
```

```
## [1] "Confusion matrix for the 5 predictor model"
```

```
##
##        FALSE TRUE
##   low    721   37
##   high   166   56
```

The initial model on 7 predictors has a true positive probability of $\frac{64}{64+43} \approx 0.598$
The full model has a true positive probability of $\frac{63}{63+42} = 0.6$
The model on 8 predictors has a true positive probability of $\frac{55}{55+36} \approx 0.604$
Finally, the model on 5 predictors has a true positive probability of $\frac{56}{56+37} \approx 0.602$

Overall, the best model is clearly the five predictor model, which achieves the nearly identical accuracy as the previous models, contains the fewest predictors, and has no multicolinearity.

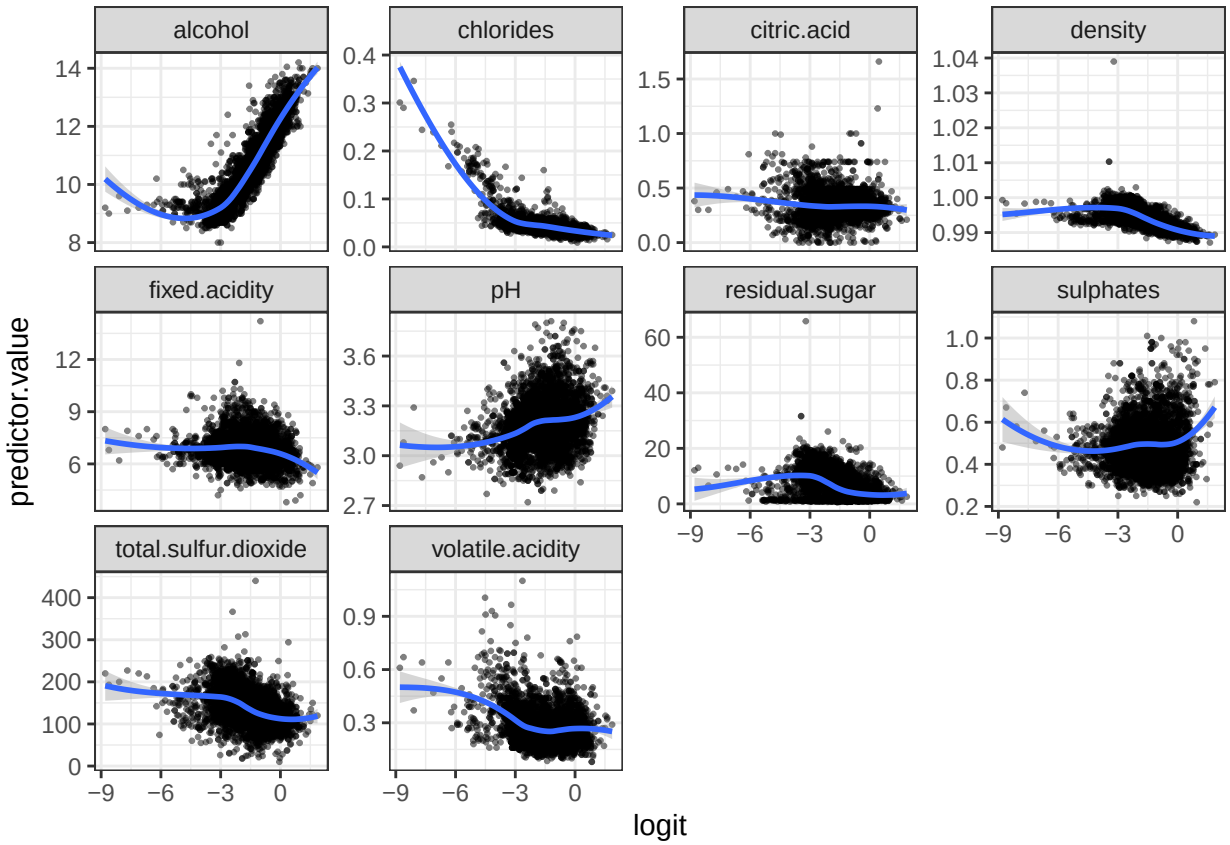
The final model's equation is:
$\log\left(\frac{\hat{\pi}_{quality\_cat}}{1-\hat{\pi}_{quality\_cat}}\right) = -11.30223 - 3.3808 * x_{volatile.acidity} - 18.68852 * x_{chlorides} + 0.89333 * x_{pH} + 1.10611 * x_{sulphates} + 0.76496 * x_{alcohol}$

Outliers or Transformations

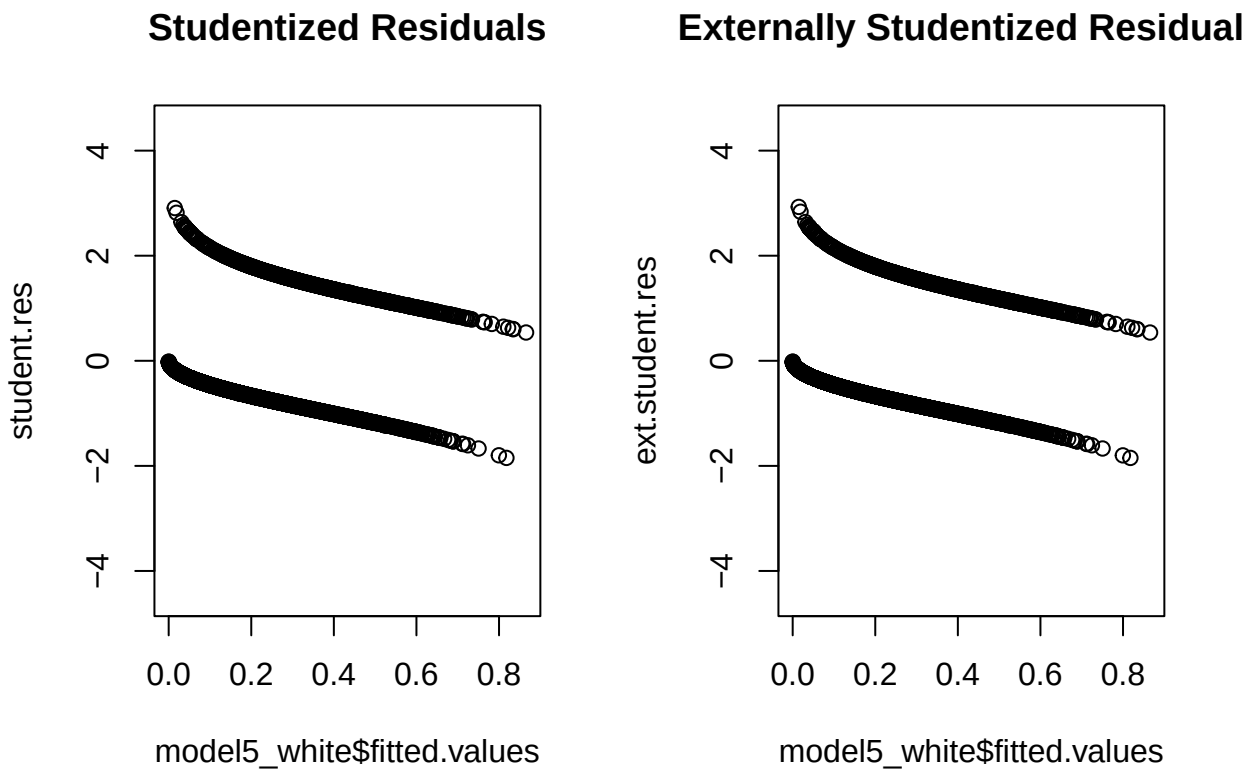
Finally, we need to assess the rest of the regression assumptions.

```
##  1017  4775  2177  1533  4567  2347
## "neg" "neg" "neg" "neg" "neg" "neg"
```

```
## `geom_smooth()` using formula 'y ~ x'
```



None of these scatterplots show any of the predictors are clearly nonlinear, so there is no need to do any transformations. Many transformation were tried in the course of model building, but none were significant, so they were not included in any model.



According to these plots, there doesn't seem to be any outliers in the residuals. Therefore, there shouldn't be any highly influential points in the model.

Final White Wine Model Assessment

The final white wine model does not predict as well as the red wine model. This seems counter intuitive because there are nearly three times more observations in the white wine model, and there were a higher proportion of high quality white wines to low quality ones.

One possible explanation comes from our EDA section. White wine's quality appears slightly less correlated with the predictors, which could explain why it is more difficult to judge them.
Another cause could be that white wines have a small but detectable increase in the variation in quality over red wines (~0.2 of a point). However, this doesn't seem like it would be high enough to justify a predictive accuracy that is this much lower.

Errors in modeling could also cause us not to have found a model which exists and has a very high predictive accuracy. But, this too seems unlikely, since all four of the models fit in this section have nearly identical predictive accuracy.

The true reason why our models for red wine predicted so much better than white could be a combination of all three of these problems and more that we did not list.

Conclusion

We began this project with one goal in mind - to see if we could predict which wines were high-quality based on their physical and chemical characteristics. This knowledge would help our boutique wine shop client with its goal of maximizing the ratio of high-quality wines to low-quality wines in their product mix.

After extensive exploratory data analysis, we were able to determine that the best course of action was to create separate models for both red and white wine due to the fact that many wine characteristics had a substantial difference in their effects on quality depending on the color of the wine, as well as the fact that some of the characteristics had completely different ranges in values across colors.

The process to create both of these models was challenging and took the team several interactions to develop the final models. For the white wine model, we used 5 predictors - alcohol, sulphates, and pH (positive impact on "high" quality probability) as well as volatile acidity and chlorides (negative impact on "high" quality probability). This model was able to perform on par with several more complicated models, while utilizing fewer predictors. Ultimately, for the wines that our model predicts as high-quality, 60 percent of these wines are actually high-quality.

For the red wine model, we used 7 predictors - alcohol percentage, sulphate levels, fixed acidity, and residual sugar (positive impact on "high" quality probability), as well as volatile acidity, density, and chlorides (negative impact on "high" quality probability). The differences between selected predictors and their magnitude shows slight but important differences in the determinants of quality between red and white wines. Our red wine model performed even better than the white wine model, as 71.4% of wines identified as "high quality" were identified correctly.

These are strong results that go a long way towards improving our client's wine selection above random guessing. It makes sense that they are not perfect in their ability to identify high quality wines, as physical characteristics can go only so far in explaining wine quality, and two wines with the same physical characteristics can have differing levels of quality. We see these models as an important supplement to our client's already stellar knowledge of wine. At the very least, they are a powerful sanity check. At their very best, these models can alleviate much of the stress on human judgement and ultimately lead to cost savings on low-quality wine and substantially happier customers.