RED AND WHITE WINE DATA ANALYSIS
PREDICT QUALITY OF WINE

A Thesis
Presented to the
Faculty of
California State Polytechnic University, Pomona

In Partial Fulfillment
Of the Requirements for the Degree
Master of Science
In
Mathematics

By
Gregory D. Nelson
2020
SIGNATURE PAGE

THESIS: RED AND WHITE WINE DATA ANALYSIS
PREDICT QUALITY OF WINE

AUTHOR: Gregory D. Nelson

DATE SUBMITTED: Spring 2020

Department of Mathematics and Statistics

Dr. Jimmy Risk
Thesis Committee Chair
Mathematics & Statistics

Dr. Adam King
Mathematics & Statistics

Dr. Michael L. Green
Mathematics & Statistics
ACKNOWLEDGMENTS

First, I would like to thank my advisor and thesis committee chair Dr. James Risk for his help and advice throughout the process of writing this paper. He always gave me helpful feedback in all of our meetings and helped me understand the big picture, but also the specifics of what I was trying to accomplish. He was always available to answer questions through e-mail in times that were outside our official meetings. I also had the most course work with him and have learned so much that I am applying in this paper. I would especially like to acknowledge his willingness to share his knowledge with me in the form of a paper he’d written and his dissertation as examples for me to emulate.

I would also like to thank Dr. Michael L. Green, who accepted being a member of my thesis committee. I had coursework from him in statistics and appreciated his teaching style and all the material that he has taught me.

I would also like to thank Dr. Adam King, the math department graduate student advisor. While I didn’t have any coursework with him, he was invaluable during this thesis process and as a resource for me to make sure I was on schedule for graduation.

Lastly, I would like to thank my colleagues in the Master’s program, who supported me and gave me advice along this journey. This especially includes Kevin Bailey, William Buehlman and Maria Diaz.
ABSTRACT

Model analysis is wide-spread and has many applications in various fields. The intention of this analysis on red and white wine of the Vinho Verde varietal from Portugal is to predict the quality of the wine based on the predictors. This topic is of interest so results can be shared with patrons that want to buy and consume higher quality wine and how they can find relationships between the variables and the quality. Wine manufacturers would be interested so they can produce wine that more people will enjoy. The analysis is done through comparing several models, including traditional linear models, Least Absolute Shrinkage and Selection Operator (LASSO) models, and the more sophisticated Gaussian process. Model performance will be compared through a training/test set split, as well as $k$-fold cross validation, and, where applicable, the Akaike Information Criterion (AIC).
Contents

Signature Page ii

Acknowledgements iii

Abstract iv

List of Tables vii

List of Figures viii

Chapter 1 Introduction 1

Chapter 2 Regression Problem and Performance Metrics 5

2.1 Akaike Information Criterion (AIC) 7

2.2 Training and Test Set Split 7

2.3 Cross Validation 8

Chapter 3 Regression Modeling Techniques 10

3.1 Linear Regression 10

3.2 Least Absolute Shrinkage and Selection Operator (LASSO) Regression 11

3.3 Gaussian Process 13
Chapter 4  Data Description  

4.1  Description of Variables  

4.2  Relationships Between Variables  

Chapter 5  Model Selection  

5.1  Linear Model  

5.2  Least Absolute Shrinkage and Selection Operator (LASSO) Regression  

5.3  Gaussian Process  

5.4  Training Set Conclusions  

Chapter 6  Wine Quality Prediction  

Chapter 7  Conclusion  

7.1  Predict Quality of Wine  

7.2  Application of Results  

Appendices  

Appendix A  Model Coefficients and Summaries  

A.1  LM$_1$  

A.2  LM$_{LASSO}$  

vi
List of Tables

4.1 Wine Quality data set variable names and units. ........................................ 17
4.2 Five number summary and mean for quantitative variables and Qual-
ity. .................................................................................................................. 18
4.3 Pearson correlation coefficients \( \rho \) in decreasing order for variables
with \( |\rho| \geq 0.5 \). ......................................................................................... 20
5.1 Model comparison on Training set. ............................................................... 26
6.1 Model comparison on Test set. ................................................................. 29
A.1 LM \(_1\) summary. ......................................................................................... 40
A.2 LASSO coefficients. .................................................................................... 41
List of Figures

4.1 Plots describing wine quality based on red v white wine. . . . . . 19
4.2 Select scatter plots for wine data set. . . . . . . . . . . . . . . . 20
5.1 LoWeSS plots showing quadratic patterns. . . . . . . . . . . . . . 22
5.2 Plot of Training set wine quality $y_i$ values on $x$-axis versus their predicted values $\hat{y}_i$ for all four modeling methods. . . . . . . 25
6.1 Plot of Test set wine quality $y_i$ values on $x$-axis versus their predicted values $\hat{y}_i$ for all four modeling methods. . . . . . . 29
Chapter 1

Introduction

This work is endeavoring to predict the quality of wine based on physiochemical data. What is meant by quality of wine can be difficult to articulate, but one such definition is that “ideally, it should be related to intrinsic visual, taste, or aroma characters which are perceived as above average for that type of wine” [11]. The quality of wine this study is trying to predict is pertaining to red and white wine of the Vinho Verde varietal from Portugal [8]. They proposed a data mining approach to predict human wine taste preferences based on easily available analytical tests. They applied three regression techniques under a procedure that performed simultaneous variable and model selection. They used support vector machines, which outperformed multiple regression and neural network methods.

There have been other studies done to analyze different factors that contribute to the quality of wine. It has been discovered that season-warming trends from 1950-1999 caused quality of vintage wine to increase, while year-to-year wine quality decreased. Improved winemaking knowledge and husbandry practices played a role, but it was found that the climate played a significant role in quality variations. Of
note was that the impacts of climate change were not likely to be uniform across all varieties and regions. Further study was suggested [13]. Next, it was found that effects of the environment and management practices that might change grape composition could impact wine quality. The study focused on table wines and explained the contribution climates, sites and viticultural practices make [11]. Other studies examined specific physiochemical properties in the wine that were found to be significant in the endeavor to explain the wine’s quality. There were discovered to be more than one thousand minerals from the grapes and as products of the wine-making process, such as ethanol and glycerol. A major parameter of wine quality is sugar import and the accumulation into a ripe berry. Sugar status was directly related to the wine’s alcoholic content and regulate several genes responsible for development of its aromatic and organoleptic properties. Ripe berries have high sugar levels without losing too much acidity. That particular study focused on key points in grape ripening to improve grape and wine quality and suggested further study [7].

In another study, outliers and anomalies were identified to detect adulteration of wine. Decision Tree and Naive Bayes methods were applied and performances were compared. The Decision Tree outperformed and it was found that Alcohol and Volatile Acidity contributed highly to quality of the wine. White wine was also found to be more sensitive to changes in physiochemistry as opposed to red wine and, therefore, required a higher level of handling care [2]. Other methods that were used to predict human wine taste preferences based on physiochemical properties were Hybrid Fuzzy Logic techniques and Inductive Reasoning. The Fuzzy Inductive Reasoning technique outperformed the other fuzzy approaches, but also other data mining techniques previously applied to the data set [16]. In addi-
tion, Binary and Multi-Class Neural Network Classifiers were applied, in addition to other readily available classification algorithms [1]. Also, Regression Tree algorithms were studied that were able to find a prediction model with over 88% accuracy [10]. Lastly, a Decision Tree was used and found that the results were 60% in agreement with traditional assessment techniques when tested against the Wine Quality dataset from the UC Irvine Machine Learning Repository - the same data set that this paper tests against [15].

Building off of prior research, the analysis will focus on the red and white wine of the Vinho Verde varietal from Portugal that was accessed from the UC Irvine Machine Learning Repository [8]. The goal is to predict the quality of the wine based on the predictors. This topic is of interest so results can be shared with patrons that want to buy and consume higher quality wine. Wine manufacturers would be interested so they can produce wine that more people will enjoy. The analysis is done through comparing several models, including traditional linear models, Least Absolute Shrinkage and Selection Operator (LASSO) models, and the more sophisticated Gaussian process. Model performance will be compared through a training and test set split, as well as k-Fold Cross Validation, and, where applicable, the Akaike Information Criterion (AIC).

In Chapter 2, the regression problem along with its assumptions will be discussed. Performance metrics based on the regression problem will also be introduced. In Chapter 3, specific regression models will be introduced, specifically linear regression, LASSO, and Gaussian process models. In Chapter 4, the data will be described that the regression techniques will be applied to. In Chapter 5, model selection techniques will be applied to the Training set. In Chapter 6, the quality of the wine will be predicted in the Test set based on the model selection
from the previous chapter. Chapter 7 will be the conclusion.
Chapter 2

Regression Problem and Performance Metrics

The goal of this thesis is to find the best model to solve the following regression problem:

\[ y = f(x) + \epsilon, \]  
\[ \text{(2.1)} \]

where \( f \) is some unknown function, \( y \) is observed data, and \( \epsilon \) is a noise term. Toward solving this problem, we assume the observed data is \( D = (x_1, y_1), \ldots, (x_n, y_n), \ x_i \in \mathbb{R}^d \). The process of regression aims to produce a prediction \( \hat{y} \equiv \hat{y}(x') = \hat{f}(x') \), where \( x' \in \mathbb{R}^d \), by fitting an appropriate response function \( \hat{f}(x) \). To determine an appropriate response function to use, some assumptions are typically made on \( f(x) \) in Equation (2.1), and consequently fit according to some minimization problem based on \( D \). Specific modeling techniques are discussed in Chapter 3. The regression setting typically assumes squared-error loss, that is, we want to minimize the Least Squares Residual Sum of Squares (RSS), given by

\[ \text{RSS} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2. \]  
\[ \text{(2.2)} \]
Simply using RSS as an error metric to minimize can lead to overfitting, i.e. a model that performs very well on the data to which it is fit, but predicts poorly out of sample. In fact, if \( y'_i \) is a hypothetical out of sample response associated with \( x_i \), then one can generally show that

\[
E \left[ \sum_{i=1}^{n} (y'_i - \hat{y}_i)^2 \right] = E \left[ \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \right] + \frac{2}{n} \sum_{i=1}^{n} \text{Cov}(\hat{y}_i, y_i),
\]

(2.3)

see section 7.4 from [9]. Since \( \hat{y}_i \) is designed to estimate at location \( x_i \), but itself is built from \( y_i \), they are typically positively correlated, thus we expect \( \text{Cov}(\hat{y}_i, y_i) \geq 0 \), inflating the expected Test error.

One typically defines an averaged version of RSS, the mean square error (MSE), defined as

\[
\text{MSE} = \frac{1}{n} \text{RSS}.
\]

(2.4)

Lastly, one also can compute \( R^2 \), the proportion of the variance in the dependent variable that is predictable from the independent variable(s), which is defined by

\[
R^2 = \frac{TSS - RSS}{TSS}
\]

(2.5)

where RSS is given in Equation (2.2) and

\[
TSS = \sum_{i=1}^{n} (y_i - \bar{y}_i)^2,
\]

(2.6)

i.e. the squared error when no model is fit besides estimating \( \hat{y}_i \) by the sample mean \( \bar{y}_i = \frac{1}{n} \sum_{i=1}^{n} y_i \). \( R^2 \) suffers the same issues as RSS as an error metric, since TSS is fixed regardless of model choice.
2.1 Akaike Information Criterion (AIC)

Akaike Information Criterion (AIC) is a direct estimator of out-of-sample prediction error, attempting to remedy the overfitting problem. Given different models, AIC estimates the quality of each model relative to each other, therefore it is a means for model selection, i.e. choosing between models. The AIC value of a model is given by

\[
\text{AIC} = 2p - 2\ln(\hat{L})
\]

(2.7)

where \( p \) = number of parameters in model, \( \hat{L} \) = maximum value of the likelihood function for the model \([3]\). Since a better fitting model has a higher likelihood, AIC decreases as the fit improves, and penalizes (increases) as the number of parameters \( p \) increases. Thus, the preferred model out of different candidate models is the one with the lowest AIC value. Examining Equation (2.7), AIC rewards goodness of fit, as assessed by the likelihood function, and penalizes based on the number of parameters. When \( \epsilon \) is assumed to be independent and identically distributed (iid) \( N(0, \sigma^2) \), AIC is found to be proportional to \([12]\)

\[
C_p = \frac{1}{n\sigma^2} (\text{RSS} + 2p\sigma^2)
\]

(2.8)

2.2 Training and Test Set Split

Recall that the data is given by \((x_1, y_1), \ldots, (x_n, y_n)\). The goal of prediction is to produce the best out of sample estimate for a new point \( x' \). In this setup, we can artificially introduce out of sample data by splitting our data set.

Without loss of generality, assume that the \( x_i, y_i \) are in a random order. The
training/test set split procedure is done by dividing the data into two sets:

\[
\text{Training set} = (x_1, y_1), \ldots, (x_m, y_m), \quad \text{Test set} = (x_{m+1}, y_{m+1}), \ldots, (x_n, y_n),
\]  

(2.9)

where \(1 < m < n\), so that the data for \(i = m + 1, \ldots, n\) is reserved for out of sample performance assessment and the model is fit to \(x_1, \ldots, x_m\). This has the clear downside of forcing the model to be fit to less data, but offers a valuable out-of-sample performance assessment which is otherwise not possible to obtain without new data. The value \(m\) is typically chosen so that \(2/3\) of the data belongs to the Training set, and \(1/3\) belongs to the Test set, though other values can be chosen. This procedure is useful in reducing sample selection bias [18], as opposed to using different distributions for Training and Test sets, and to prevent over fitting [4].

### 2.3 Cross Validation

The goal of cross validation is to test a model’s ability to predict new data (the Test set) in order to identify problems, such as overfitting or selection bias [5], and to determine how the model will generalize to an independent data set.

The general \(k\)-fold cross validation procedure is done as follows:

1. Denote the data set as \(D = \{(x_1, y_1), \ldots, (x_n, y_n)\}\); without loss of generality assume that it is randomized in order.
2. Divide the data into $K$ folds:

$$
D_1 = \{(x_1, y_1), \ldots, (x_{n_1}, y_{n_1})\},
D_2 = \{(x_{n_1+1}, y_{n_1+1}), \ldots, (x_{n_1+n_2}, y_{n_1+n_2})\},
\vdots
D_K = \{(x_{n_K-1}, y_{n_K-1}), \ldots, (x_{n_K-1+n_K}, y_{n_K-1+n_K})\};
$$

3. For each $k = 1, \ldots, K$,

(a) Fit a model to $D \setminus D_k$, that is, the full data set with the $k$th fold removed.

(b) Predict using the fitted model on the $k$th fold, and compute

$$
MSE_k = \frac{1}{n_k - n_{k-1}} \sum_{i=n_{k-1}+1}^{n_k} (y_i - \hat{y}_i)^2.
$$

4. Average the mean squared errors computed in the last step:

$$
CV_K = \frac{1}{K} \sum_{k=1}^{K} MSE_k,
$$

then the $K$-fold cross validation error is $CV_K$. Typically, the size of each fold $n_k - n_{k-1}$ is chosen to be uniform and equal to $n/K$. In step 3, we are effectively performing a training/test set split $K$ times, and averaging these Test set errors in the final step. $CV_K$ gives an estimate of the model’s predictive performance.
Chapter 3

Regression Modeling Techniques

3.1 Linear Regression

Linear Regression assumes a specific functional form for $f(x)$ in Equation (2.1):

$$f(x) = \sum_{j=1}^{r} \beta_j h_j(x), \quad (3.1)$$

where $\beta = (\beta_0, \ldots, \beta_p)$ and $x = (x_1, \ldots, x_d)^T$ and where $h_j$ are prespecified basis functions. For example, if one suspects a quadratic relationship between $y$ and $x_1$, we could include $h(x) = x_1^2$, or if there is a hypothesized interaction between $x_1$ and $x_2$, one could use $h(x) = x_1 x_2$. This is still called a linear model since it is a linear combination of these basis functions. Without loss of generality, in what follows, we assume that $x_1, \ldots, x_d$ have already been transformed to $p$ predictors $x_1, \ldots, x_p$, in which case Equation (3.1) states

$$f(x) = \beta_0 + \sum_{j=1}^{p} \beta_j x_j = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p = \beta x, \quad (3.2)$$

in which case now $x = (1, x_1, \ldots, x_p)^T$. 

10
Least squares regression is the typical way in which the $\beta_j$ are estimated, which minimizes the sum of squared errors when Equation (3.2) is a surrogate for $y$. If we denote the least squares estimates as $\hat{\beta}_j$, then the minimization problem states

$$\hat{\beta} = \arg\min_\beta \sum_{i=1}^n (y_i - f(x))^2 = \arg\min_\beta \sum_{i=1}^n (y_i - \beta x)^2. \quad (3.3)$$

There is a closed form solution to this problem, in fact

$$\hat{\beta} = (X^T X)^{-1} X^T y, \quad (3.4)$$

where $X$ is the matrix with rows $(1, x_{i1}, \ldots, x_{ip}), i = 1, \ldots, n$ and $y = (y_1, \ldots, y_n)^T$. Thus $\hat{\beta}$ a linear combination of the $y_i$. Predictions are done consequently with

$$\hat{y} = \hat{\beta} x. \quad (3.5)$$

The fact that $\hat{\beta}$ is a linear combination of the $y_i$ yields many nice theoretical properties and calculations, for example $\text{var}(\hat{\beta})$ is relatively easy to calculate, and normality assumptions on $y$ yield $\hat{\beta}$ to be normal as well. On a similar note, if one assumes that the errors $\epsilon_i$ are iid normal random variables, then $\hat{\beta}$ actually is the maximum likelihood estimate based on $y_1, \ldots, y_n$. For more information, please see [12].

### 3.2 Least Absolute Shrinkage and Selection Operator (LASSO) Regression

LASSO and Ridge regression still assumes the parametric linear form in Equation (3.2), but minimizes a slightly different criteria. In particular, LASSO adds a $L^1$
(absolute value) penalty on its coefficients, and Ridge a \( L^2 \) (squared) penalty. The resulting equations to minimize are

\[
\text{RSS} + P_{\text{LASSO}} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \tag{3.6}
\]

\[
\text{RSS} + P_{\text{RIDGE}} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \tag{3.7}
\]

where \( P_{\text{LASSO}} \) and \( P_{\text{RIDGE}} \) are the penalty terms. The idea here is that higher \( \beta_j \) coefficients will also increase the criterion to be minimized, i.e. smaller \( \beta_j \) values are desired. LASSO and Ridge regression are generally used when \( p \), the number of parameters is large, especially relative to \( n \). Specifically, they reduce model complexity and prevent over-fitting that may occur by simply minimizing \( \text{RSS} \) alone. We see that in comparing Equation \( (3.6) \) and Equation \( (3.7) \), the LASSO criterion penalizes by adding absolute coefficients \( (L^1) \), while Ridge adds squared coefficients \( (L^2) \). Remarkably, the \( L^1 \) loss for LASSO will shrink coefficients \( \beta_j \) for unimportant predictors to be identically equal to zero, something that does not occur with Ridge; see [22] for details. This results in a model that is more interpretable. These variables with coefficients of zero, after the shrinkage process, are removed from the model. Variables with non-zero regression coefficients are the variables that are most strongly associated with the response variable [12]. There is no closed form LASSO solution for Equation \( (3.6) \), though for Ridge, the minimizer is

\[
\hat{\beta}_{\text{RIDGE}} = (X^T X + \lambda I)^{-1} X^T y, \tag{3.8}
\]

where in this case \( \beta_0 \) is not included in the minimization and is estimated separately.

Both methods specify a tuning parameter \( \lambda \) to control the strength of the penalty; higher \( \lambda \) means the penalty term is higher and lower \( \lambda \) means to penalty
term is lower, as can be seen in Equations (3.6) and (3.7). For LASSO, as $\lambda$ increases, more coefficients will shrink to zero, which means fewer predictors are selected and there is more shrinkage of the non-zero coefficients. The tuning parameter is typically chosen through cross validation: we fit models for different values of $\lambda$ and choose $\lambda$ to minimize (3.6) or (3.7). This determines the value of the tuning parameter $\lambda$.

### 3.3 Gaussian Process

The objective in modeling with a Gaussian process is to place a prior on the function $f$ in question, and use Bayesian tools to produce a posterior function $\hat{f}$ conditioned on $y_1, \ldots, y_n$. First, some preliminaries:

**Definition 3.3.1.** A Gaussian process is a collection $(f(x))_{x \in \mathbb{R}^d}$ of random variables, any finite number of which have a joint Gaussian distribution.

We define the mean function $\mu(x)$ and covariance function $C(x, x')$ of $f$ as

$$
\mu(x) = \mathbb{E}[f(x)]
$$

$$
C(x, x') = \mathbb{E}[(f(x) - \mu(x))(f(x') - \mu(x'))],
$$

(3.9)

The mean and covariance function are prespecified as priors. One can specify any symmetric positive definite function for $C$, as that is the covariance function of a unique Gaussian process [17]. For simplicity, assume that $\epsilon$ is iid $N(0, \sigma^2)$, thus the Regression Equation (2.1) implies $y|f(x) \sim N(f(x), \epsilon)$. In fact, letting $y = (y_1, \ldots, y_n)$ and $f = (f(x_1), \ldots, f(x_n))$, $y|f$ is multivariate normal. The unconditional distribution of $y$ is easy to obtain (e.g. tower property), yielding the posterior of $f|y$ as a multivariate normal via Bayes’ theorem. Repeating this procedure, we can see that for two test points $x$ and $x'$, the distributions of $f(x)|y$ and
\[ f(x') | y \] are normal with posterior mean and covariance

\[
m(x) = \mu(x) + c(x)^T(C + \Delta)^{-1}(y - \mu);
\]
\[
s^2(x, x') = C(x, x') - c(x)^T(C + \Delta)^{-1}c(x'),
\]

where

- \( c(x)^T = (C(x, x_1), \ldots, C(x, x_n))^T \), the vector of covariances of \( f(x) \) with \( f(x_1), \ldots, f(x_n) \)
- \( C \) is the covariance matrix with entries \( C(x_i, x_j), 1 \leq i, j \leq n \), the prior covariance matrix,
- \( \Delta \) is the diagonal \( n \times n \) (noise variance) matrix with entries \( \sigma^2 \),
- \( \mu = (\mu(x_1), \ldots, \mu(x_n)) \).

The formulas in Equation (3.10) are called the prediction equations.

The covariance function \( C(\cdot, \cdot) \) is an important part of a GP model. The two most common choices for GP modeling are the Matern-\( \nu \) and Gaussian kernels. The DiceKriging package in R [20] uses the Matern-\( \frac{5}{2} \) kernel as its prior by default. It does however, allow fitting of GP models for several kernel families, all by maximum likelihood. The Matern-\( \frac{5}{2} \) covariance kernel, for example, is given in one dimension by the formula

\[
C_j(x_j, x'_j; \theta_j) = 1 + \frac{\sqrt{5}|x_j - x'_j|}{\theta_j} + \frac{5|x_j - x'_j|^2}{3\theta_j^2} \left( \exp \left( -\frac{\sqrt{5}|x_j - x'_j|}{\theta_j} \right) \right)
\]

and the full covariance function for \( x, x' \in \mathbb{R}^p \) is

\[
C(x, x'; \theta) = \eta^2 \prod_{j=1}^{p} C_j(x_j, x'_j; \theta_j)
\]

14
The mean function also influences predictions as seen in Equation (3.10). The Gaussian process reverts to its prior mean as it leaves the design space, since $c(x)^T$ decreases rapidly. For prediction, however, the trend function has little impact in sample and is mostly important for extrapolation [23]. Although that is true, a reasonably accurate mean function is wanted for hyperparameter fitting, since the $\theta$ parameters influence spatial dependence, something affected by detrending [19].
Chapter 4

Data Description

We are analyzing the red and white wine of the Vinho Verde varietal from Portugal, namely the “Wine Quality” data set found at the UCI repository [8]. We first combine the red and white wine data and add a categorical variable indicating if the wine is red or white. The entire data set was split into a Training set, which covered 2/3 of the whole data set and on which all the analysis was done, and a Test set, which covered 1/3 of the whole data set and was only used for final model performance, see Section 2.2.

4.1 Description of Variables

Table 4.1 lists the variables of the data set and their units, and Table 4.2 provides a five-number summary of each quantitative variable and Quality.

Note in Table 4.1 that Quality has no units as it was described to be scored on a scale from 0 to 10 based on sensory data, pH has no units as the quantitative values are on the logarithmic pH scale for acidity and Red has no units because it was assigned a value of 0 or 1 based on if the wine was from red or white wine.
Table 4.1: Wine Quality data set variable names and units.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Variable</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>Quality</td>
<td>No units</td>
</tr>
<tr>
<td>$x_1$</td>
<td>Fixed Acidity</td>
<td>g(tartaric acid)/dm$^3$</td>
</tr>
<tr>
<td>$x_2$</td>
<td>Volatile Acidity</td>
<td>g(acetic acid)/dm$^3$</td>
</tr>
<tr>
<td>$x_3$</td>
<td>Citric Acid</td>
<td>g/dm$^3$</td>
</tr>
<tr>
<td>$x_4$</td>
<td>Residual Sugar</td>
<td>g/dm$^3$</td>
</tr>
<tr>
<td>$x_5$</td>
<td>Chlorides</td>
<td>g(sodium chloride)/dm$^3$</td>
</tr>
<tr>
<td>$x_6$</td>
<td>Free Sulfur Dioxide (FSD)</td>
<td>mg/dm$^3$</td>
</tr>
<tr>
<td>$x_7$</td>
<td>Total Sulfur Dioxide (TSD)</td>
<td>mg/dm$^3$</td>
</tr>
<tr>
<td>$x_8$</td>
<td>Density</td>
<td>g/dm$^3$</td>
</tr>
<tr>
<td>$x_9$</td>
<td>pH</td>
<td>No units</td>
</tr>
<tr>
<td>$x_{10}$</td>
<td>Sulphates</td>
<td>g(potassium sulphate)/dm$^3$</td>
</tr>
<tr>
<td>$x_{11}$</td>
<td>Alcohol</td>
<td>volume %</td>
</tr>
<tr>
<td>$x_{12}$</td>
<td>Red</td>
<td>No units</td>
</tr>
</tbody>
</table>

In Table 4.2, the Quality values span from 3 to 9, evaluated by a minimum of three sensory assessors (using blind tastes), which graded the wine in a scale that ranges from 0 (very bad) to 10 (excellent). The final sensory score is given by the median of these evaluations [8]. What this means is that half of the wine was rated to have lower Quality from 3 to 6 and the half with higher Quality spans from 6 to 9. Quality being skewed left (mean = 5.818 lower than median = 6) shows that the low Quality rated wines have more of a pull to the lower side of the spectrum than do the high Quality rated wines.
Table 4.2: Five number summary and mean for quantitative variables and Quality.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Fixed Acidity</th>
<th>Volatile Acidity</th>
<th>Citric Acid</th>
<th>Residual Sugar</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>3.800</td>
<td>0.0800</td>
<td>0.0000</td>
<td>0.700</td>
</tr>
<tr>
<td>Quartile 1</td>
<td>6.400</td>
<td>0.2300</td>
<td>0.2400</td>
<td>1.800</td>
</tr>
<tr>
<td>Median</td>
<td>7.0000</td>
<td>0.2900</td>
<td>0.3100</td>
<td>3.0000</td>
</tr>
<tr>
<td>Mean</td>
<td>7.2270</td>
<td>0.3411</td>
<td>0.3184</td>
<td>5.4470</td>
</tr>
<tr>
<td>Quartile 3</td>
<td>7.7000</td>
<td>0.4100</td>
<td>0.3900</td>
<td>8.0000</td>
</tr>
<tr>
<td>Maximum</td>
<td>15.9000</td>
<td>1.5800</td>
<td>1.0000</td>
<td>65.8000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variable</th>
<th>Chlorides</th>
<th>FSD</th>
<th>TSD</th>
<th>Density</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>0.0090</td>
<td>1.0000</td>
<td>6.0000</td>
<td>0.9871</td>
</tr>
<tr>
<td>Quartile 1</td>
<td>0.0380</td>
<td>17.0000</td>
<td>76.0000</td>
<td>0.9924</td>
</tr>
<tr>
<td>Median</td>
<td>0.04700</td>
<td>28.0000</td>
<td>118.0000</td>
<td>0.9949</td>
</tr>
<tr>
<td>Mean</td>
<td>0.0564</td>
<td>30.1800</td>
<td>115.0000</td>
<td>0.9947</td>
</tr>
<tr>
<td>Quartile 3</td>
<td>0.06500</td>
<td>41.0000</td>
<td>155.0000</td>
<td>0.9970</td>
</tr>
<tr>
<td>Maximum</td>
<td>0.6110</td>
<td>146.5000</td>
<td>366.5000</td>
<td>1.0390</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variable</th>
<th>pH</th>
<th>Sulphates</th>
<th>Alcohol</th>
<th>Quality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>2.7400</td>
<td>0.2500</td>
<td>8.0000</td>
<td>3.0000</td>
</tr>
<tr>
<td>Quartile 1</td>
<td>3.1100</td>
<td>0.4300</td>
<td>9.5000</td>
<td>5.0000</td>
</tr>
<tr>
<td>Median</td>
<td>3.2100</td>
<td>0.5100</td>
<td>10.3000</td>
<td>6.0000</td>
</tr>
<tr>
<td>Mean</td>
<td>3.2170</td>
<td>0.5327</td>
<td>10.4900</td>
<td>5.8130</td>
</tr>
<tr>
<td>Quartile 3</td>
<td>3.3200</td>
<td>0.6000</td>
<td>11.3000</td>
<td>6.0000</td>
</tr>
<tr>
<td>Maximum</td>
<td>4.0100</td>
<td>2.0000</td>
<td>14.9000</td>
<td>9.0000</td>
</tr>
</tbody>
</table>

### 4.2 Relationships Between Variables

We first aim to see how Red interacts with other predictors of wine quality. Figures 4.1 and 4.2 illustrate this for a few variables.

From Figure 4.1 we see that Fixed Acidity and Volatile Acidity describe Quality differently based on if the data is from red wine or white wine.

Through the process of model selection, it will be determined which predictors, such as Red, are significant in predicting the quality of the wine. Simply studying
Figure 4.1: Plots describing wine quality based on red v white wine.

the summary of a linear model, for example, is not sufficient because there could be multiple predictors that are collinear, meaning they may be individually significant to predict Quality, but not when considered together [14]. That is to say one of them could be significant in predicting the quality of the wine, but not all. Therefore, it is important to keep collinear predictors in mind while model selection is done. Figure 4.2 lists the pairwise relationships between all variables that have a Pearson correlation of $|\rho| \geq 0.5$, and Table 4.3 summarizes these variables and their $\rho$ values.

From Figure 4.2, it appears that Volatile Acidity and Red could be positively correlated, TSD and FSD could be positively correlated, TSD and Red could be negatively correlated and Alcohol and Density also could be negatively correlated.
Table 4.3: Pearson correlation coefficients $\rho$ in decreasing order for variables with $|\rho| \geq 0.5$.

<table>
<thead>
<tr>
<th>Variable 1</th>
<th>Variable 2</th>
<th>$\rho$</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSD</td>
<td>FSD</td>
<td>0.7216</td>
</tr>
<tr>
<td>TSD</td>
<td>Red</td>
<td>-0.6955</td>
</tr>
<tr>
<td>Alcohol</td>
<td>Density</td>
<td>-0.6720</td>
</tr>
<tr>
<td>Volatile Acidity</td>
<td>Red</td>
<td>0.6567</td>
</tr>
</tbody>
</table>

Table 4.3 confirms the appearances from the plots. These possible relationships between variables should be kept in mind as the process of analysis is continued.
Chapter 5

Model Selection

The regression problem now involves predicting $y$, wine quality, using the data $\mathcal{D} = ((x_1, y_1), \ldots, (x_n, y_n))$, where $n = 4331$ and $x \in \mathbb{R}^p$, where $p = 12$. We utilize methods described in Chapter 3 in particular, linear models, LASSO, and Gaussian processes. We analyze each class of model separately, and compare the best performing models in each class on the basis of AIC (when applicable), and otherwise, 5-fold CV error.

5.1 Linear Model

The model selection process was started with a model linear in each $x_j, j = 1, \ldots, d$, using all the predictors, including Red. It showed a preliminary result that even when including all predictors, everything was individually significant when other variables were considered ($\alpha = 0.05$), aside from Citric Acid. We first want to check if linear relationships between predictors is good enough, or if we should use different basis functions.

In an effort to determine if polynomial terms should be included in model selec-
tion, LoWeSS regression [6] was performed. It wasn’t used directly as a prediction model, but it helped identify what polynomial terms to include by smoothing. LoWeSS plots are shown in Figure 5.1.

Figure 5.1: LoWeSS plots showing quadratic patterns.

Additionally, we produced LoWeSS plots for other predictors, but found no clearly defined non-linear relationships in them, other than Chlorides. Thus we consider quadratic terms in FSD, Sulphates, and Chlorides as candidate predictors. Higher order polynomials were also considered but preliminary analysis showed that quadratic was the highest order that should be used.

We also saw from Figures 4.1 and 4.2 varying relationships between predictors and Quality depending on if the wine is red or white, so we also consider interaction terms with Red. It then needs to be determined if other interaction terms should
be included. To this regard, we also consider all other possible interaction terms.

Since there are many possible routes to model fitting at this point, with a clear risk of overfitting by including all terms, we consider two approaches, both utilizing best subset selection [21] on the basis of AIC, which, in short, computes the AIC for all possible subsets of predictors and finds what combination produces the lowest model AIC:

- **LM$_1$**: this model performs best subset selection using *only* linear coefficients on the original $p = 12$ predictors.

- **LM$_{BEST}$**: this model performs best subset selection using all original predictors, the quadratic terms mentioned above, and all possible interaction terms.

LM$_1$ will offer good interpretation of predictor effects on Quality, and has a very low risk of overfitting. LM$_{BEST}$ has an extremely large number of total possible predictors to include; theoretically it should have higher predictive power, at the cost of a higher risk of overfitting. We hope that the choice of AIC remedies the overfitting issue, though to be safe we will also analyze cross validation measures (5-fold CV) and Test set performance as a check.

The result was that LM$_1$ ended up using 11 of the predictors; it left out Citric Acid. The model summary can be found in Appendix [A.1]. For comparison purposes, the AIC values for LM$_1$ and LM$_{BEST}$ were 9510.36 and 9382.96, respectively.
5.2 Least Absolute Shrinkage and Selection Operator (LASSO) Regression

LASSO regression was performed as discussed in Section 3.2 on the full data set, including the aforementioned quadratic variables and interaction terms. As mentioned, LASSO constrains parameter values to zero, reducing the chance of overfitting. The tuning parameter $\lambda$ was optimally determined to be $\lambda = 2.246 \times 10^{-4}$ based on cross validation. We denote the model fitted in this section as LM$_{\text{LASSO}}$. The actual nonzero parameter estimates can be found in Appendix A.2.

The most interpretable part of LM$_{\text{LASSO}}$ is the four original predictors. This model indicates that as Volatile Acidity increases by 1 unit, Quality is predicted to decrease by about 0.039 units, and so forth; see Table A.2.

5.3 Gaussian Process

Generally, Gaussian processes are resilient to overfitting, so we include all predictors, quadratics, and their interaction terms as covariates. Due to the very large amount of observations in the Training set ($n = 4331$), the fitting procedure for the Gaussian process took several hours to run due to inverting the $n \times n$ covariance matrix in Equation (3.12) in finding maximum likelihood parameter estimates. The overall fitting iterative procedure, since the first round, yielded several of the predictors to have produced length scale estimates of $\hat{\theta} = 0$. By looking at the role $\theta$ plays in covariance calculations (see Equation (3.12)), this indicates that these predictors are insignificant. We dropped these from the fitting and repeated the procedure until all variables were deemed significant. We denote the model fitted
in this section as GP.

## 5.4 Training Set Conclusions

All previously mentioned models, LM$_1$, LM$_{BEST}$, LM$_{LASSO}$, and GP, were compared on several bases. First, we show their predictions versus the Training set values; the line $y = x$ indicates what a perfect fit would do. This is seen in Figure 5.2.

![Figure 5.2: Plot of Training set wine quality $y_i$ values on x-axis versus their predicted values $\hat{y}_i$ for all four modeling methods.](image)

Figure 5.2 shows very similar levels of prediction between the methods. It appears that they all do well at predicting Quality for values that have a Quality of 6. All methods clearly have issues when Training set Quality is abnormally large or small.
For Quality less than 6, the predictions $\hat{y}_i$ are mainly above the observed Quality $y_i$. For Quality greater than 6, the predictions $\hat{y}_i$ are mainly below the observed Quality $y_i$.

Next, we compare the models on the basis of MSE$_{Tr}$, $R^2_{Tr}$, which are MSE and $R^2$ evaluated on the Training set, as well as 5—fold CV. This is displayed in Table 5.1 with the best value of each Training set metric in bold. For reference, when all possible predictors, interaction terms and quadratic terms were fit to a linear model, it gave an $R^2_{Tr}$ value of 0.3811, so these “low” $R^2_{Tr}$ values obtained from the linear models should not be dismissed.

Table 5.1: Model comparison on Training set.

<table>
<thead>
<tr>
<th>Model</th>
<th>MSE$_{Tr}$</th>
<th>$R^2_{Tr}$</th>
<th>5—Fold CV</th>
</tr>
</thead>
<tbody>
<tr>
<td>LM$_1$</td>
<td>0.5259</td>
<td>0.2935</td>
<td>0.5291</td>
</tr>
<tr>
<td>LM$_{BEST}$</td>
<td>0.5012</td>
<td>0.3267</td>
<td>0.5149</td>
</tr>
<tr>
<td>LM$_{LASO}$</td>
<td>0.5114</td>
<td>0.3129</td>
<td>0.5055</td>
</tr>
<tr>
<td>GP</td>
<td><strong>0.3527</strong></td>
<td><strong>0.5261</strong></td>
<td><strong>0.4475</strong></td>
</tr>
</tbody>
</table>

It can be seen from Table 5.1 that the the Gaussian process performs the best with the smallest values in MSE$_{Tr}$ and 5—fold CV and largest value in $R^2_{Tr}$. For assessing prediction performance, the 5—fold CV value is what we should consider out of this table because it estimates the Test set error, while MSE$_{Tr}$ and $R^2_{Tr}$ don’t compensate for overfitting. Thus far, the Gaussian process performs the best on the Training set. LM$_{BEST}$ performs next best, followed by LM$_{LASO}$ and LM$_1$.

We can also compare row-wise how the models perform on the basis of MSE$_{Tr}$ versus 5—fold CV; note that the former is computed on the Training set and the latter to estimate MSE$_{Te}$ on the Test set, so they are comparable in magnitude. If
these values differ, it could be indicative of overfitting, since for example if 5-fold CV is larger than the respective $\text{MSE}_{\text{Tr}}$, we estimate that the model does better on the Training set rather than the Test set. All of the linear models have similar values, indicating no real issues with overfitting. GP, on the other hand, has a much inflated 5-fold CV value. For this reason, we should pay close attention to how the GP does on the Test set, as this could suggest overfitting. Despite this, however, GP still does best on the metric of 5-fold CV, so its potential overfitting may not cause predictive issues.
Chapter 6

Wine Quality Prediction

After analyzing the models on the Training set, LM₁, LMₐₜₜₚₜ, LMₐₜₚₜ, and GP were compared to show their predictions versus the Test set values, treating the Test set as a completely new set of model factors in which we are directed to predict wine quality. The plotted results of $y_i$ versus $\hat{y}_i$ are shown in Figure 6.1, which is an analogue to the Training set Figure 5.2.

Figure 6.1 shows very similar levels of prediction, both in regards to comparing to Figure 5.2 and between the models on the Test set. It appears that in the Test set, these models all do well at predicting Quality for values that have a Quality of 6. We see that the models perform quite similarly to how they did on the Training set. For true Quality values in Test set less than 6, the predictions are mainly above the actual Quality. For true Quality values greater than 6, the predictions are mainly below the actual Quality.

Table 6.1 illustrates the comparison of LM₁, LMₐₜₜₚₜ, LMₐₜₚₜ, and GP on the basis of $\text{MSE}_{\text{Tt}}$ and $R^2_{\text{Tt}}$, with the best value of each Test set metric in bold. Table 6.1 can be compared to model performance in the Training set, see Table 5.1.
Table 6.1: Model comparison on Test set.

<table>
<thead>
<tr>
<th>Model</th>
<th>$\text{MSE}_{\text{Te}}$</th>
<th>$R^2_{\text{Te}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>LM$_1$</td>
<td>1.052</td>
<td>0.2186</td>
</tr>
<tr>
<td>LM$_{\text{BEST}}$</td>
<td>0.6239</td>
<td>0.2263</td>
</tr>
<tr>
<td>LM$_{\text{LASSO}}$</td>
<td>0.6177</td>
<td>0.2986</td>
</tr>
<tr>
<td>GP</td>
<td>0.4706</td>
<td>0.4107</td>
</tr>
</tbody>
</table>

Analyzing Table 6.1, it is of note that all four models perform worse on basis of $R^2_{\text{Te}}$ as compared to $R^2_{\text{Tr}}$ (see Table 5.1). On the basis of $\text{MSE}_{\text{Te}}$ and $R^2_{\text{Te}}$, GP performs the best and LM$_1$ performs the worst. It is of note that a linear model
is probably not the best approach here, since none of the linear models seemed to perform very well compared to the GP. In addition, this helps alleviate the possible concern of the GP overfitting - it does extremely well on the Test set.
Chapter 7

Conclusion

The analysis of this paper focused on the red and white wine of the Vinho Verde varietal from Portugal that was accessed from the UC Irvine Machine Learning Repository \[8\]. The goal was to predict the quality of the wine based on the predictors. The analysis was done through more traditional methods, such as fitting a linear models with best subset selection and LASSO models. In addition to these, the more sophisticated Gaussian process was applied to the data set. On the Training data, we compared performance mostly on the basis of $K$-fold cross validation error, and also used a training/test set split to examine genuine out of performance prediction ability of the models.

7.1 Predict Quality of Wine

The goal of the paper was model selection to best predict quality of wine. We split the original data set into a Training set ($n = 4331$) and Test set ($n = 2166$) so that we could assess actual out of sample prediction performance. On the Training set, the models were compared on the basis of their $5$–fold CV values because it
estimates Test set error. The $\text{MSE}_{\text{Tr}}$ and $R^2_{\text{Tr}}$ values of the models are used to supplement, but it is worth noting they don’t compensate for overfitting. Best subset selection process was implemented between different one-predictor models. On the basis of AIC, the best 11-predictor model, referred to as LM$_1$, was originally selected, but it was found that a model with interaction terms and quadratic terms labeled LM$_\text{BEST}$ performed the best. Through LASSO Regression, a model LM$_{\text{LASSO}}$ with four original predictors and 20 interaction terms, 8 including quadratics was obtained. The last model considered was not a linear model but a Gaussian process, named GP. It used the original predictors and the interaction terms and quadratic interaction terms. Its 5-fold CV was calculated on the Training set, and was found to be the lowest of all models, as seen on Table 5.1. GP also had the lowest $\text{MSE}_{\text{Tr}}$ and highest $R^2_{\text{Tr}}$. In the Training set, the best models on the basis of 5-fold CV were, in order: GP, LM$_{\text{LASSO}}$, LM$_\text{BEST}$ and LM$_1$.

These models were further compared by applying them to the Test set. As can be seen on Table 6.1, the best models on the basis of $\text{MSE}_{\text{Te}}$ were, in order: GP, LM$_{\text{LASSO}}$, LM$_\text{BEST}$ and LM$_1$, with the GP significantly outperforming the linear models. These results also agree with the Training set results. Further study is suggested to analyze this data set further by using different methods to compare the quality of the wine. Our results suggest that a linear model is probably not the best approach, since none of the linear models seemed to capture the relationships between variables very well compared to the GP. It is recommended to consider other non-linear or non-parametric methods, such as splines.
7.2 Application of Results

These results can now be shared with patrons that want to buy and consume higher quality wine and they can understand how to predict the wine’s quality. They can look for wines that include more of the physiochemical properties that predict higher quality and that include less of the properties that predict lower quality of wine. Wine manufacturers would also be interested so they can produce wine with the properties that more people will enjoy and that have higher quality.
Bibliography


Appendices
Appendix A

Model Coefficients and Summaries

A.1 LM$_1$

Table A.1 shows the model coefficients for LM$_1$.

A.2 LM$_{LASSO}$

Table A.2 shows the model coefficients for LM$_{LASSO}$.
<table>
<thead>
<tr>
<th>Variable</th>
<th>Coefficient</th>
<th>SE(Coefficient)</th>
<th>P-Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>95.61</td>
<td>16.23</td>
<td>$4.17 \times 10^{-9}$</td>
</tr>
<tr>
<td>$x_1$</td>
<td>0.0761</td>
<td>0.0182</td>
<td>$2.89 \times 10^{-5}$</td>
</tr>
<tr>
<td>$x_2$</td>
<td>-1.439</td>
<td>0.0924</td>
<td>$&lt; 2 \times 10^{-16}$</td>
</tr>
<tr>
<td>$x_4$</td>
<td>0.0579</td>
<td>0.0070</td>
<td>$&lt; 2 \times 10^{-16}$</td>
</tr>
<tr>
<td>$x_5$</td>
<td>-1.142</td>
<td>0.3890</td>
<td>0.0034</td>
</tr>
<tr>
<td>$x_6$</td>
<td>0.0055</td>
<td>0.0010</td>
<td>$9.73 \times 10^{-9}$</td>
</tr>
<tr>
<td>$x_7$</td>
<td>-0.0015</td>
<td>0.0004</td>
<td>$1.14 \times 10^{-4}$</td>
</tr>
<tr>
<td>$x_8$</td>
<td>-94.59</td>
<td>16.51</td>
<td>$1.08 \times 10^{-8}$</td>
</tr>
<tr>
<td>$x_9$</td>
<td>0.3545</td>
<td>0.1091</td>
<td>0.0012</td>
</tr>
<tr>
<td>$x_{10}$</td>
<td>0.6528</td>
<td>0.0913</td>
<td>$1.02 \times 10^{-12}$</td>
</tr>
<tr>
<td>$x_{11}$</td>
<td>0.2281</td>
<td>0.0208</td>
<td>$&lt; 2 \times 10^{-16}$</td>
</tr>
<tr>
<td>$x_{12}$</td>
<td>0.4153</td>
<td>0.0662</td>
<td>$3.78 \times 10^{-10}$</td>
</tr>
</tbody>
</table>
Table A.2: LASSO coefficients.

<table>
<thead>
<tr>
<th>Predictor</th>
<th>Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Intercept)</td>
<td>15.67</td>
</tr>
<tr>
<td>Volatile Acidity</td>
<td>-0.03898</td>
</tr>
<tr>
<td>Residual Sugar</td>
<td>0.0002857</td>
</tr>
<tr>
<td>Density</td>
<td>-12.61</td>
</tr>
<tr>
<td>Alcohol</td>
<td>0.1796</td>
</tr>
<tr>
<td>Fixed Acidity:Residual Sugar</td>
<td>0.003126</td>
</tr>
<tr>
<td>Fixed Acidity:Red</td>
<td>0.008542</td>
</tr>
<tr>
<td>Volatile Acidity:TSD</td>
<td>-0.001581</td>
</tr>
<tr>
<td>Volatile Acidity:Density</td>
<td>-0.06782</td>
</tr>
<tr>
<td>Volatile Acidity:pH</td>
<td>-0.2825</td>
</tr>
<tr>
<td>Residual Sugar:Alcohol</td>
<td>0.00005373</td>
</tr>
<tr>
<td>Chlorides:TSD</td>
<td>-0.009502</td>
</tr>
<tr>
<td>Chlorides:Alcohol</td>
<td>-0.0001134</td>
</tr>
<tr>
<td>FSD:Alcohol</td>
<td>0.001412</td>
</tr>
<tr>
<td>pH:Alcohol</td>
<td>0.01347</td>
</tr>
<tr>
<td>Sulphates:Alcohol</td>
<td>0.06872</td>
</tr>
<tr>
<td>Sulphates:Red</td>
<td>0.1674</td>
</tr>
<tr>
<td>TSD:Sulphates$^2$</td>
<td>-0.002013</td>
</tr>
<tr>
<td>Fixed Acidity:FSD$^2$</td>
<td>-1.531 x 10$^{-7}$</td>
</tr>
<tr>
<td>Volatile Acidity:FSD$^2$</td>
<td>-0.0001208</td>
</tr>
<tr>
<td>Residual Sugar:FSD$^2$</td>
<td>-4.776 x 10$^{-7}$</td>
</tr>
<tr>
<td>FSD:FSD$^2$</td>
<td>-2.712 x 10$^{-7}$</td>
</tr>
<tr>
<td>TSD:FSD$^2$</td>
<td>-2.690 x 10$^{-7}$</td>
</tr>
<tr>
<td>Red:FSD$^2$</td>
<td>-0.00006813</td>
</tr>
<tr>
<td>Sulphates$^2$:Chlorides$^2$</td>
<td>-0.6287</td>
</tr>
</tbody>
</table>