Assessing the Impact of Physicochemical Properties on Red Wine Quality: A Multivariate (OLS and Lasso) Regression and Bayesian Approach

Jukun Zhang[1]

[1] Rutgers University

Abstract

This study mainly uses three models-OLS, Lasso regression and Bayes model. After preliminary exploratory analysis (descriptive statistics, box plots, correlation heat maps, paired scatter plots), OLS can explain 36% of the variance (adjusted $R^2 = 0.356$), thus determining that alcohol, volatile acid, sulfate, chloride, pH, free sulfur dioxide and total sulfur dioxide are important predictors. Lasso regression reduces 3 variables by setting the L1 penalty coefficient. Although the model is further optimized, the results are basically the same as OLS. However, the Bayes model extracts more accurate prediction variables through 95% confidence interval and KDE test, and believes that the positive impact of wine quality comes from free SO (FSO2), sulfates (Sulphates) and alcohol (Alcohol), and the negative impact comes from volatile acid (VolAcid), chloride (Chlor), and total SO (TSO2). Finally, the data_wine_tidy data is binarized, and Logistic regression is used to verify that the prediction accuracy of the final confirmed 6 variables for the quality of red wine is 0.818, indicating that these six variables can provide a certain quantitative basis for the prediction of red wine quality.

Keywords: OLS; Lasso Regression; Bayes Regression; Logistics Regression
Word count: 4084

Correspondence concerning this article should be addressed to Jukun Zhang. E-mail: jz1250@scarletmail.rutgers.edu

2 Introduction & Literature Review

The quality of red wine not only affects consumers' tasting experience and health perception, but is also directly related to the improvement direction of grape planting and brewing technology and the optimization of the industrial value chain. With the continuous advancement of analytical instruments and data acquisition technology, researchers can obtain a series of physical and chemical indicators including fixed acidity, volatile acidity, citric acid content, residual sugar, chloride, free sulfur dioxide, total sulfur dioxide, density, pH value, sulfate and alcohol content. These indicators have their own characteristics in different regions, different years and different brewing process conditions, and have a complex and multidimensional impact on the sensory score (quality) of the final wine.

Traditional research mainly uses ordinary least squares (OLS) regression to evaluate the linear relationship between various physical and chemical properties and scores. Some scholars combine ridge regression or principal component analysis to deal with multicollinearity problems. In recent years, Lasso regression has been widely used in high-dimensional data modeling because of its variable selection and regularization capabilities; while Bayesian regression provides a framework for uncertainty quantification for parameter estimation through prior distribution and posterior inference. However, there are few existing literatures that systematically compare the model performance and variable importance differences of OLS, Lasso and Bayesian methods on the same data set.

Based on the red wine quality data set (n = 1,599) provided by UCI, this study uses OLS, Lasso and Bayesian regression as analysis tools. Through descriptive statistics, visual exploration and cross-validation, this study comprehensively evaluates the relative impact of various physical and chemical properties on wine quality scores, and compares the advantages and disadvantages of the three methods in terms of prediction accuracy and variable explanatory power, providing a quantitative basis for red wine quality control and optimization.

3 Data & Exploratory Analysis

3.1 Data Source & Pre-processing

My data source was find from “UCI Red Wine Quality” link. It has 1599 data points and I choose “Quality” as the response and other 11 variables as predictors which include fixed.acidity, volatile.acidity, citric.acid, residual.sugar, chloride, free.sulfur.dioxide, total.sulfur.dioxide, density, pH, sulfate and alcohol. After preprocessing the data, I first read the raw data as *data_wine_raw*, and then checked whether there were missing values in the raw data. I used “anyNA(data_wine_raw)” and “colSums(is.na(data_wine_raw))” to test whether there were missing values and which column of variables had missing values. The test results showed that the data was very intact and there were no missing values. The next step was to continue to abbreviate the column names to make it easier to represent in the output image and make the results easier to read. Part of the data_wine_tidy are show here:

Table 1
data_wine_tidy (Part vars)

FixAcid	VolAcid	Sugar	Chlor	Alc	Qual
7.4	0.70	1.9	0.076	9.4	5
7.8	0.88	2.6	0.098	9.8	5
7.8	0.76	2.3	0.092	9.8	5
11.2	0.28	1.9	0.075	9.8	6
7.4	0.70	1.9	0.076	9.4	5
7.4	0.66	1.8	0.075	9.4	5

3.2 Descriptive Statistics

Then I exported Five-Number Summaries to further analyze the data.

The Five-Number Summaries (Table 2) highlight the central tendency and variance of each physicochemical variable in the red wine dataset. For most acidity measurements (fixed acidity, volatile acidity, citric acid), the interquartile ranges (IQRs) were moderate, about 2.1 units for fixed acidity and 0.25 units for volatile acidity, indicating a fairly consistent distribution of acidity across samples, although the maximum values (15.9 for fixed acidity and 1.58 for volatile acidity) indicate a right skew and a high acidity outlier. Sugar content was clearly positively skewed. The median sugar content was only 2.2 g/L, but the maximum reached 15.5 g/L, reflecting the small number of very sweet wines. Chloride and pH show narrow IQRs (0.02 g/L and 0.19 pH units, respectively), indicating strict quality control over salinity and acidity balance. Density was fairly constant (only 0.996-0.998 from Q1-Q3), confirming little variation in ethanol to water ratio. Sulfate (IQR = 0.18 g/L) and alcohol (IQR = 1.6 % v/v) both show moderate distributions, with the upper quartile (11.1 %) and maximum value (14.9 %) of alcohol highlighting that wines with higher alcohol content may have higher quality scores. Finally, total SO and free SO showed the largest absolute ranges (TSO up to 289 mg/L, FSO up to 72 mg/L), indicating different SO conservation strategies, while the mass itself ranged from 3 to 8, but was concentrated at 5 to 6 (IQR = 1). Combining these summaries allows us to predict which predictors contribute to quality changes and which nonlinear effects or outliers require further investigation.

Table 2

Table 2. Five-Number Summaries for All Variables

variable	Min	Q1	Median	Q3	Max
FixAcid	4.60	7.10	7.90	9.20	15.90
VolAcid	0.12	0.39	0.52	0.64	1.58
CitAcid	0.00	0.09	0.26	0.42	1.00
Sugar	0.90	1.90	2.20	2.60	15.50
Chlor	0.01	0.07	0.08	0.09	0.61
FSO2	1.00	7.00	14.00	21.00	72.00
TSO2	6.00	22.00	38.00	62.00	289.00
Dens	0.99	1.00	1.00	1.00	1.00
pH	2.74	3.21	3.31	3.40	4.01
Sulph	0.33	0.55	0.62	0.73	2.00
Alc	8.40	9.50	10.20	11.10	14.90
Qual	3.00	5.00	6.00	6.00	8.00

3.3 Visualization & Correlation

After completing the above mean and median analysis, I output boxplots, heat maps, and Pairwise Scatterplot Matrix of some variables and wine quality, hoping to use these visualizations to analyze the correlation between each variable and response (quality) and check whether there is multicollinearity. The output images are as follows:

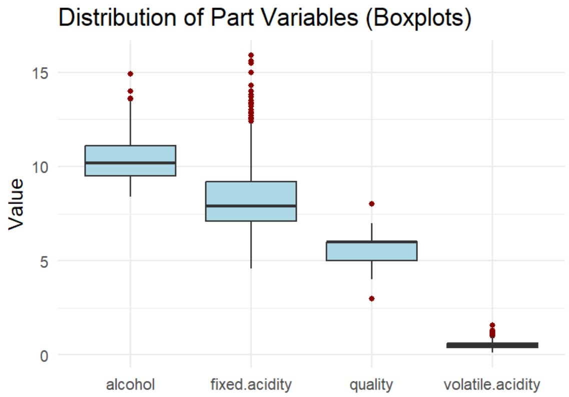

Figure 1. Boxplot of part Variables

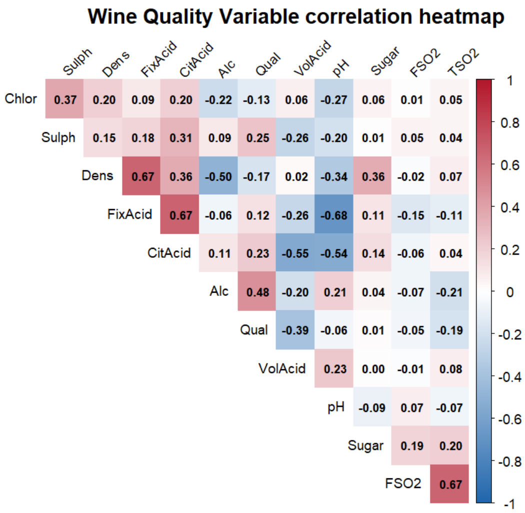

Figure 2. Heatmap of correlations between variables

Figure 2 shows the Pearson correlation coefficient matrix of all eleven variables and the dependent variable wine quality. From the figure, we can see that alcohol and taste show a strong positive correlation (r +0.48), which indicates that wines with higher alcohol content tend to get higher quality scores. On the other hand, volatile acidity is significantly negatively correlated with health (r -0.39), indicating that the higher the volatile acidity, the lower the quality score of the wine. We also found that there may be some multi-collinearity between the predictors - free SO2 and total SO2 (r = +0.67) and fixed acidity and density (r = +0.67). In short, based on this figure, we have a deeper understanding of the data, and the conclusions we get will help guide the subsequent OLS and Lasso regression methods.

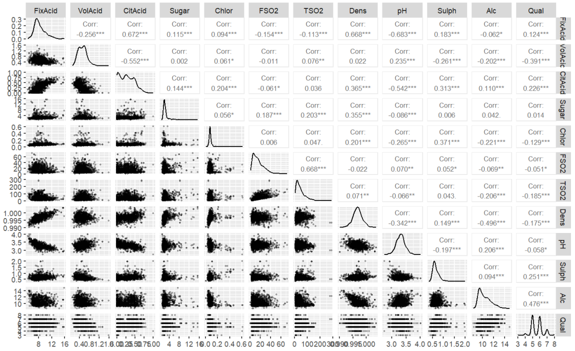

Figure 3. Pairwise Scatterplot Matrix for all variables

Figure 3 uses GGally::ggpairs to output the Pairwise Scatterplot Matrix, where the lower left corner of the diagonal outputs the scatter plots between the variables. From the scatter plots of the variables, we can see that the points of Fixed Acidity and Density are dense and distributed in an upward diagonal, reflecting the conclusion drawn from the heat map that the two variables may be highly collinear. The output in the middle of the diagonal is the kernel density estimates of each variable's marginal distribution, and the output in the upper right corner of the diagonal is the same Pearson correlation coefficient as the heat map. The smooth density curves of each variable reveal the distribution characteristics: for example, Sugar shows an obvious right-skewed long-tail distribution, while Density is highly concentrated in the range of 0.997–0.998. This matrix not only further confirms the known strong associations such as Sulphates and Quality, but also once again shows the multicollinearity problem and the abnormal marginal distribution of some variables, providing a key basis for variable screening and diagnosis of subsequent regression models.

4 Methodology & Results

4.1 Ordinary Least Squares Model (OLS)

I first used the OLS model to linearly model the dependent variable wine quality for all 11 variables. The regression equation is as follows:

$$\mathrm{Qual}_i = \beta_0 + \beta_1\,\mathrm{FixAcid}_i + \beta_2\,\mathrm{VolAcid}_i + \cdots + \beta_{11}\,\mathrm{Alc}_i + \varepsilon_i, \quad \varepsilon_i \sim N(0,\sigma^2). \tag{1}$$

Hypothesis tests are performed on the parameters to test the significance of the parameters. The results can be obtained from the P-value column in Table 3. For each regression coefficient β_j, test:

Null Hypothesis(H_0) : $\beta_j = 0$ vs Alternative Hypothesis(H_a) : at least one $\beta_j \neq 0$

Calculate the t statistic $t_j = \hat{\beta}_j/\mathrm{SE}(\hat{\beta}_j)$ and report the two-tailed p value. The results show:

Table 3
OLS regression coefficient estimation and significance test

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	21.97	21.19	1.04	0.30
FixAcid	0.02	0.03	0.96	0.34
VolAcid	-1.08	0.12	-8.95	0.00
CitAcid	-0.18	0.15	-1.24	0.21
Sugar	0.02	0.02	1.09	0.28
Chlor	-1.87	0.42	-4.47	0.00
FSO2	0.00	0.00	2.01	0.04
TSO2	0.00	0.00	-4.48	0.00
Dens	-17.88	21.63	-0.83	0.41
pH	-0.41	0.19	-2.16	0.03
Sulph	0.92	0.11	8.01	0.00
Alc	0.28	0.03	10.43	0.00

Among them, we can find that the parameters of volatile acidity, chloride, total SO_2, free SO_2, pH, sulphates and alcohol are significant in the model, while other variables are not significant at the 5% level. We can preliminarily conclude that these variables shown as significant above have a certain effect on the quality score of red wine. The fitting results of the OLS model show that the coefficient of determination $R^2 = 0.3606$ indicates that the model can explain about 36.1% of the total variance of the quality score. However, after adjustment, $R^2 = 0.3561$, and the F statistic of the model = 81.35 (df = 11, 1587), $p < 2.2 \times 10^{-16}$, rejecting the null hypothesis, indicating that the model is significant.

After that, the OLS model residuals were further tested using the diagnosis() function in the library (ds4ling). The following is the output of the OLS residual model diagnostic graph:

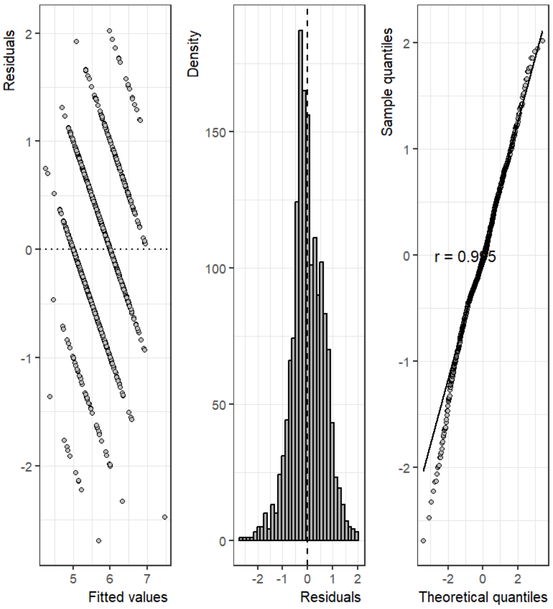

Figure 4. Residual diagnosis for OLS model

The residuals vs. fitted values plot on the left of the test plot shows randomly distributed scatter with no particularly obvious pattern, so the assumption of linearity can be considered supported. However, since the variance seems to increase with the fitted values, there may be slight heteroskedasticity. The Q-Q plot on the right shows that the residuals are roughly distributed along the diagonal (r = 0.95), indicating approximate normality. However, the slight deviation in the tails shows that there may be slight non-normality. The residual density plot in the middle is centered at zero and has a single peak, which is similar to the residual density plot of a good model fit, and is bell-shaped, but a slight skewness is observed.

4.2 Lasso Regression

To mitigate multicollinearity and enforce sparsity in our predictor set, we fit a Lasso model using the **glmnet** package. In matrix notation, letting x_i denote the p-dimensional predictor vector for observation i and y_i its quality score, the Lasso estimate solves

$$\min_{\beta_0,\beta} \frac{1}{N} \sum_{i=1}^{N} \left(y_i - \beta_0 - x_i^\top \beta\right)^2 \; + \; \lambda \sum_{j=1}^{p} |\beta_j|, \quad \lambda \geq 0. \tag{2}$$

Prior to fitting, all predictors were standardized and an intercept term retained.

Lambda Selection via 10-Fold Cross-Validation. Using the glmnet() function, with the penalty parameter α set to 1, we can get a complete Lasso model. We continue to set the response to the quality of the wine, and the dependent variable is the remaining 11 variables. We hope to use the Lasso model to delete as many unimportant variables as possible, making the model parameters simpler - that is, to get a sparse model. This way, we can more clearly understand which physical and chemical properties (variables) have a more significant impact on the quality of the wine.We then output a plot of $Log(\lambda)$ and Mean Square Error to help determine the optimal λ value.

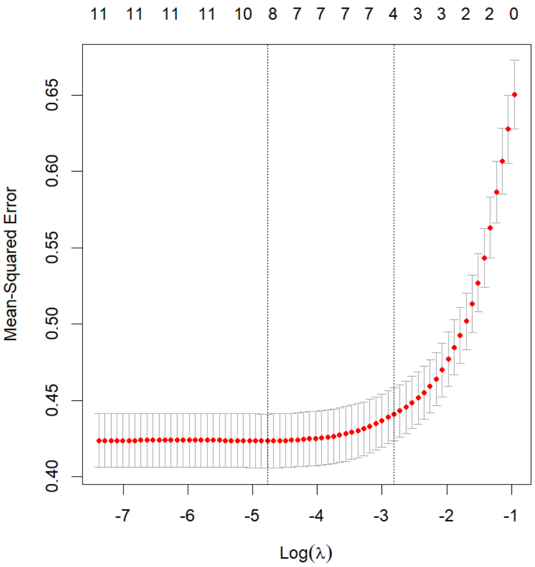

Figure 5. Plot of CV_Lasso

The figure shows that as $Log(\lambda)$ increases from -7 to -1, the MSE of the model also increases. To prevent overfitting, we believe that $Log(\lambda)$ = -3 is the best value, because reducing $Log(\lambda)$ further will penalize the model too much and delete some necessary parameters. The number at the top (11 to 0) indicates the number of factors with non-zero coefficients. As λ increases, fewer predictors are selected, thereby reducing the complexity of the model. At this optimal λ, Lasso retains the following 8 non-zero coefficients.

Compute the cross-validation MSE for a sequence of λ values. The value (denoted as λ_{min}) that minimizes the CV error is found to be. $\hat{\lambda}_{min} = 0.00848$

Table 4

Non-zero Lasso coefficients at $\hat{\lambda}_{\text{min}}$

term	estimate
(Intercept)	4.1628
VolAcid	-1.0243
Sugar	0.0013
Chlor	-1.7093
FSO2	0.0024
TSO2	-0.0027
pH	-0.3825
Sulph	0.8196
Alc	0.2852

Three predictors fixed acidity, citric acid, and density are driven exactly to zero, indicating minimal marginal contribution once penalization is applied. The retained coefficients largely mirror OLS signs and magnitudes, reaffirming volatile acidity and chlorides as the negative drivers of quality and sulphates and alcohol content as positive drivers.

Afterwards, I used 10-fold cross validation to compare the prediction performance of OLS and Lasso models.

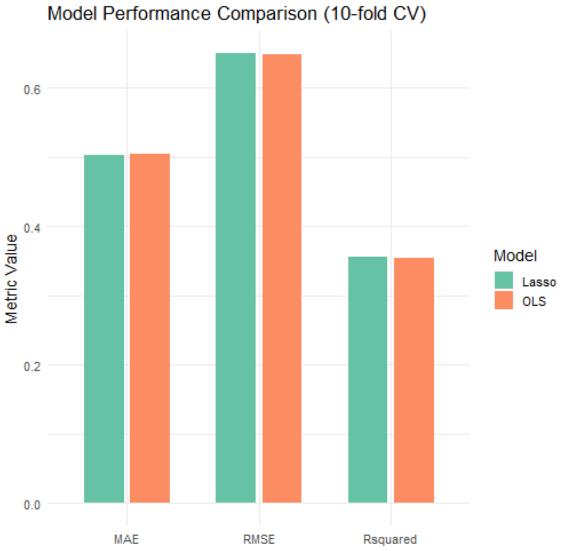

Figure 6. OLS and Lasso Model Performance Comparison

From the comparison of the 10-fold cross validation of the two models, it can be seen that LASSO and OLS perform similarly in predicting wine quality, with little difference in MAE, RMSE, and R^2.

5.3 Bayesian Cumulative Logit Regression

To account for the ordinal nature of our quality scores (3–8), we replace the Gaussian assumption with a Bayesian cumulative logit model, using the brms package. This framework estimates cutpoints (intercepts) that partition the latent propensity into the seven observed quality levels, and slopes for each predictor, with full posterior uncertainty.
Model specification

$$\Pr(\mathrm{Qual}_i \le k \mid x_i) = \mathrm{logit}^{-1}(\gamma_k - x_i^\top \beta), \quad k = 1, \dots, 7,$$

where $\gamma_1 < \gamma_2 < \dots < \gamma_7$ are the cutpoints and β the vector of slopes. Priors:

$$\beta_j \sim \mathrm{Normal}(0, 1), \quad \gamma_k \sim \mathrm{Normal}(0, 5).$$

The model was fit with four NUTS chains, 2,000 iterations each (1,000 warmup), yielding 4,000 post–warmup draws.

Posterior summary & convergence. Table 5 reports posterior medians, standard deviations, and 95% credible intervals for each slope. Convergence diagnostics are excellent ($\widehat{R} = 1.00$, Bulk ESS & Tail ESS >1,500), though 83 divergent transitions suggest increasing `adapt_delta` in future work.

Table 5
Posterior medians, standard deviations, and 95\% CIs for the Bayesian cumulative logit model

	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
Intercept[1]	-4.01	2.77	-9.71	1.21	1.00	1,624.69	1,479.60
Intercept[2]	-2.29	2.40	-7.02	2.25	1.00	2,843.95	2,430.04
Intercept[3]	0.97	2.05	-3.04	5.01	1.00	3,344.90	2,709.81
Intercept[4]	2.86	2.02	-1.12	6.77	1.00	3,322.67	2,507.98
Intercept[5]	6.52	2.02	2.57	10.44	1.00	3,294.96	2,520.03
Intercept[6]	9.32	2.04	5.34	13.23	1.00	3,263.36	2,585.26
Intercept[7]	12.31	2.05	8.31	16.23	1.00	3,268.48	2,550.20
FixAcid	0.08	0.05	-0.01	0.17	1.00	2,419.76	2,664.06
VolAcid	-3.17	0.36	-3.88	-2.46	1.00	2,724.61	1,781.00
CitAcid	-0.61	0.40	-1.39	0.16	1.00	2,408.26	2,762.07
Sugar	0.04	0.04	-0.04	0.12	1.00	4,439.33	2,766.81
Chlor	-1.92	0.77	-3.40	-0.40	1.00	3,424.30	2,619.14
FSO2	0.02	0.01	0.00	0.03	1.00	3,819.52	2,975.51
TSO2	-0.01	0.00	-0.02	-0.01	1.00	3,851.32	3,572.76
Dens	-0.02	1.03	-2.07	1.94	1.00	4,693.23	2,546.20
pH	-0.80	0.42	-1.63	0.02	1.00	2,904.98	2,664.17
Sulph	2.24	0.32	1.62	2.88	1.00	3,841.10	2,925.55
Alc	0.92	0.06	0.81	1.03	1.00	3,259.26	2,778.33

Then do the Posterior predictive check for the Bayesian cumulative logit regression model. And use pp_check() to output the diagnosis figure:

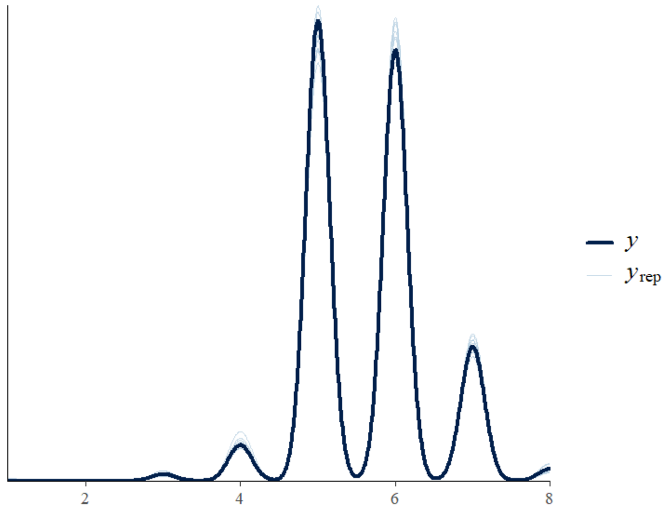

Figure 7. Posterior predictive check: observed vs. replicated quality distributions

Figure 7 overlays the kernel density of the observed quality distribution (y, dark blue) against 10 replicated datasets (y_{rep}, light blue). The replicated densities closely track the empirical peaks at scores 5, 6 and 7, demonstrating that the model captures the overall shape and modal structure of the data, with only extreme slight under-coverage of the low (3–4) tail and large(7-8) tail.

Judging from the above analysis and diagnosis results, the ordered regression model has good convergence and the diagnosis results are relatively ideal. Volatile acid, chloride, and total SO are negative influencing factors, free SO , sulfate, and alcohol content are positive influencing factors, and other physical and chemical characteristic indicators do not show significant effects. The Bayesian Cumulative Logit Regression model shows that the low-impact indicators that were not excluded by the previous OLS model and Lasso model only left the last 6 indicators. These indicators are considered to have a significant impact on the quality score of red wine in the three models, indicating that the Bayesian model further helps reduce unimportant indicators from a more numerically based aspect.

5.4 Model results verification

Based on the above three models, the six variables that mainly affect the quality score of red wine are given, namely volatile acid, chloride, total SO , free SO , sulfate, and alcohol content. In order to verify the discriminative ability of these six variables in evaluating the binary classification model at different thresholds, we use Logistic regression to binarize the quality of red wine. We define samples with a red wine quality score greater than or equal to 6 as “high quality” (HighQual=1), and the rest as “non-high quality” (HighQual=0). Then, a linear Logistics model is fitted and the odds ratio of each variable is calculated to further detect the positive/negative effect of each variable on the red wine quality score, as well as the intensity of the effect.

Table 6
Odds Ratios for Logistic Regression Predictors

Term	OR
(Intercept)	0.00
Alc	2.36
Sulph	14.97
TSO2	0.98
Chlor	0.01
VolAcid	0.06
FSO2	1.02

Output table (odds table of each variable) The model results show that alcohol and sulfate have a strong positive effect on the probability of “high quality”. The regression coefficients of the two are significantly positive, and the corresponding odds ratios are approximately 2.36 and 14.97, respectively, indicating that under the same other conditions, the odds of high quality increase by about 136% for every 1 percentage point increase in alcohol; the odds of high quality increase by nearly 15 times for every unit increase in sulfate. Although the marginal effect of free sulfur dioxide is small, its coefficient is also positive and significant, indicating that an appropriate amount of free SO can also help improve quality evaluation. On the contrary, total sulfur dioxide, chloride and volatile acidity all show a significant inhibitory effect on the probability of high quality. The odds ratios of the three are approximately 0.98, 0.01 and 0.06, respectively, which means that their increase will correspondingly significantly reduce the probability of wine being rated as high quality, among which chloride and volatile acidity have particularly obvious negative effects.

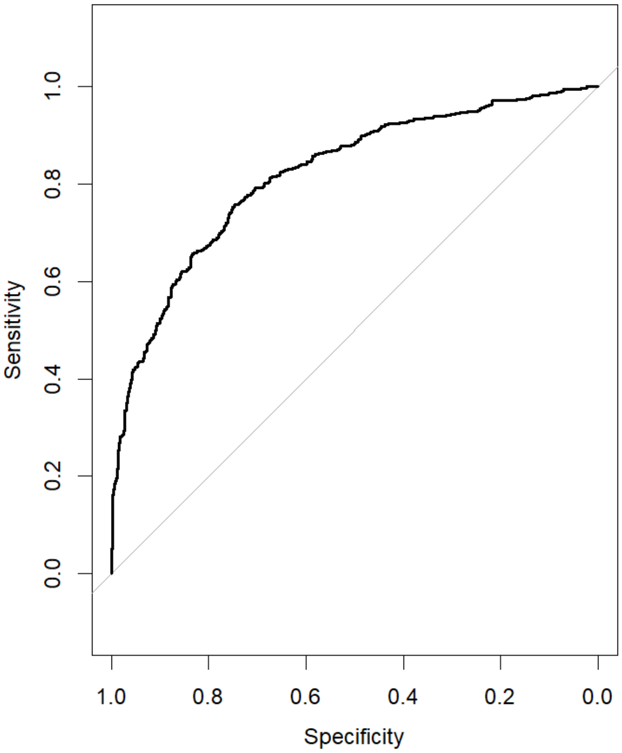

Figure 8. ROC curve for logistic regression

Figure 8 shows the ROC curve output by the detection model. The curve shows a classic “S-shaped” trajectory from the lower left corner to the upper left corner and then to the upper right corner. It can achieve a high true positive rate (True Positive Rate > 0.7) in a low false positive rate range (False Positive Rate < 0.2), which shows that the model’s ability to identify high-quality wines is already very good at a small error cost. The area under the curve (AUC) is 0.8188, which is greater than 0.8 and belongs to the evaluation standard of the “good” classification level, indicating that under all possible decision thresholds, the logistic regression model has a high stability in distinguishing the binary red wine quality scores. It should be noted that although the AUC value reflects the overall performance, it does not directly guide the selection of the optimal threshold, so in practical applications, it may be necessary to further use a more complete threshold selection method.

6 Conclusion & Future Directions

6.1 Conclusion & Discussion

The comparison results of the three models studied above consistently show that volatile acidity, chloride, total SO , free SO , sulfate, and alcohol are the main influencing factors of red wine quality scores. In the OLS, Lasso, and Bayesian Cumulative Logit

Regression models, higher alcohol is associated with an increased probability of obtaining a high-quality score, while increased volatile acidity seriously reduces the quality of red wine. Sulfate and chloride are closely followed in importance (positive values for sulfate and negative values for chloride), and total/free sulfur dioxide shows a small but credible non-zero effect in the Bayesian estimation. Other predictors (fixed acidity, citric acid, density, sugar) always produce credible intervals or confidence intervals that overlap with zero, indicating that their marginal contributions are negligible once the key drivers are considered.

The three methods differed in their handling of multicollinearity and variable selection. OLS retains all 11 slopes, but inflates standard errors when predictors are correlated (e.g., fixed acid and density, free sulfur dioxide, and total sulfur dioxide). Lasso's L1 penalty automatically shrinks small coefficients to zero, dropping fixed acid, citric acid, and density to produce a more interpretable, parsimonious model with nearly identical prediction errors (CV RMSE 0.65). In contrast, the Bayesian Cumulative Logit Regression model retains all slopes but quantifies uncertainty with 95% credible intervals, allowing us to see which effects are credibly nonzero in an ordinal framework. This full a posteriori approach also allows for direct probabilistic descriptions of how a unit change in each predictor changes the probability of obtaining a higher quality grade.

From a production perspective, these findings have practical implications:

1. Fermentation control: Perceived quality can be improved by selecting yeast strains or adjusting sugars to target slightly higher alcohol by volume (ABV).

2. Acidity management: Minimizing volatile acidity during fermentation and aging (e.g., by adding sulfur dioxide early or controlling temperature) can prevent off-flavors from disproportionately damaging sensory scores.

3. Finishing additives: Fine-tuning sulfate and chloride concentrations can further improve balance and mouthfeel. Producers can therefore focus their resources on a few factors that most reliably improve wine quality and contribute to consumers' drinking experience.

6.2 Limited of this study

Limitations of this study include reliance on a single red wine dataset (UCI "winequality-red.csv"), a cross-sectional design lacking time or regional stratification, and only discussing linear (or cumulative linear) relationships. This study did not explore potential interactions (e.g., alcohol × pH), nonlinear dose-response curves, or external sensory covariates that might further refine quality predictions.

6.3 Future Directions

Looking ahead, this study can be further deepened and expanded from the following aspects: On the one hand, interaction terms and nonlinear basis expansion can be introduced into the existing model to capture the complex coupling effects and nonlinear responses between chemical properties, thereby improving the prediction accuracy; On the other hand, a horizontal comparative analysis can be conducted on red wine samples from different production areas or years to test the versatility and robustness of the model under differences in regions, climates and cultivation conditions; In addition, with the help of a hierarchical Bayesian framework, grouping information such as producers and production areas can be incorporated into the model hierarchy, and parameter estimation can be improved through information aggregation and “learning by leveraging”, and the uncertainty at each level can be quantified; At the same time, the chemical measurement data and the subjective scores obtained from the sensory evaluation panel can be organically combined to construct a chemical-sensory mapping model to further reveal the intrinsic connection between chemical indicators and human tasting experience; Finally, other potential influencing factors (such as brewing technology, microbial community characteristics, etc.) can be introduced and modeled together with existing chemical variables, so as to comprehensively deepen the understanding of the driving mechanism of red wine quality and provide more accurate decision-making support for theoretical research and industrial practice.

References

- Montgomery, D. C., Peck, E. A., & Vining, G. G. (2012). Introduction to linear regression analysis. Hoboken, NJ: Wiley.
- Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological), 58(1), 267–288. Wiley.
- Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., & Rubin, D. B. (2013). Bayesian data analysis (3rd ed.). Boca Raton, FL: Chapman & Hall/CRC.
- Polson, N. G., Scott, J. G., & Windle, J. (2013). Bayesian inference for logistic models using Polya–Gamma latent variables. Journal of the American Statistical Association, 108(504), 1339–1349. Taylor & Francis.

```r
# Loading necessary library
library(tidyverse)
library(here)
library(dplyr)
library(tidyr)
library(ds4ling)
library(lme4)
library(lmerTest)
library(GGally)
library(corrplot)
library(ggcorrplot)
library(rstanarm)
library(brms)
library(glmnet)
library(caret)
library(tibble)

# Read the raw data
data_wine_raw <- read.csv(
  here("data_raw", "winequality-red.csv"),
  header = TRUE,
  sep = ";",
  stringsAsFactors = FALSE
)

# Test if there has any NA in data_wine_raw
anyNA(data_wine_raw)
colSums(is.na(data_wine_raw))

# View data structure
glimpse(data_wine_raw)

summary(data_wine_raw)

data_wine_tidy <- data_wine_raw %>%
  rename(
    FixAcid = fixed.acidity,
    VolAcid = volatile.acidity,
    CitAcid = citric.acid,
    Sugar   = residual.sugar,
    Chlor   = chlorides,
    FSO2    = free.sulfur.dioxide,
    TSO2    = total.sulfur.dioxide,
    Dens    = density,
    pH      = pH,
    Sulph   = sulphates,
    Alc     = alcohol,
    Qual    = quality
  )|>
  write_csv("./data_tidy/data_wine_tidy.csv")


# dplyr summary
data_wine_tidy |>
  summarise(across(
    .cols = everything(),
    .fns = list(
      mean   = ~ mean(.x, na.rm = TRUE),
      median = ~ median(.x, na.rm = TRUE),
      Q1     = ~ quantile(.x, 0.25, na.rm = TRUE),
      Q3     = ~ quantile(.x, 0.75, na.rm = TRUE)
    ),
    .names = "{.col}_{.fn}"
  )) |>
  pivot_longer(everything(),
               names_to  = c("variable", "stat"),
               names_sep = "_",
               values_to = "value") |>
  arrange(variable, stat)


five_num_summary <- data_wine_tidy |>
  summarise(across(
    .cols = everything(),
    .fns = list(
      Min  = ~ min(.x, na.rm = TRUE),
      Q1   = ~ quantile(.x, 0.25, na.rm = TRUE),
      Median = ~ median(.x, na.rm = TRUE),
      Q3   = ~ quantile(.x, 0.75, na.rm = TRUE),
      Max  = ~ max(.x, na.rm = TRUE)
    ),
    .names = "{.col}_{.fn}"
  )) |>
  pivot_longer(everything(),
               names_to = c("variable", "stat"),
               names_sep = "_",
               values_to = "value") %>%
  pivot_wider(names_from = stat, values_from = value)
```

```
# Display the Five-Number Summary
print(five_num_summary)

# Create Boxplots for all variables
data_long <- data_wine_tidy %>%
  pivot_longer(
    cols = everything(),
    names_to = "Variable",
    values_to = "Value"
  )

ggplot(data_long, aes(x = Variable, y = Value, fill = Variable)) +
  geom_boxplot(outlier.color = "darkred", outlier.size = 1.5) +
  labs(
    title = "Boxplots of All Variables",
    x = "Variables",
    y = "Value"
  ) +
  theme_minimal(base_size = 12) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_brewer(palette = "Pastel1") +
  theme(legend.position = "none")


# Select the core variables and pivot to long format
data_core <- data_wine_raw %>%
  select(fixed.acidity, volatile.acidity, alcohol, quality) %>%
  pivot_longer(
    cols = everything(),
    names_to  = "Variable",
    values_to = "Value"
  )

# Plot boxplots for core variables
ggplot(data_core, aes(x = Variable, y = Value)) +
  geom_boxplot(fill = "lightblue", outlier.color = "darkred") +
  labs(
    title = "Distribution of Part Variables (Boxplots)",
    x     = NULL,
    y     = "Value"
  ) +
  theme_minimal(base_size = 14)

# Variable correlation heatmap
# Calculate correlation matrix
corr_mat <- cor(data_wine_tidy, use = "pairwise.complete.obs")

# Generate a gradient color from blue to white to red
my_col <- colorRampPalette(c("#2166AC", "white", "#B2182B"))(200)

corrplot(
  corr_mat,
  method      = "color",
  col         = my_col,
  type        = "upper",
  order       = "hclust",
  tl.col      = "black",
  tl.srt      = 45,
  tl.cex      = 0.8,
  addCoef.col = "black",
  number.cex  = 0.7,
  diag        = FALSE,
  mar         = c(0,0,1,0)
)
title("Wine Quality Variable correlation heatmap", line = 0.5, cex.main = 1.2)


#Pairwise scatterplot matrix

GGally::ggpairs(
  data_wine_tidy,
  lower = list(continuous = wrap("points", alpha = 0.3, size = 0.5)),
  upper = list(continuous = wrap("cor", size = 3)),
  diag  = list(continuous = wrap("densityDiag"))
)

# 6. Ordinary Least Squares Regression (OLS)
model_ols <- lm(Qual ~ ., data = data_wine_tidy)
summary(model_ols)


diagnosis(model_ols)

# Fit the Lasso regression
x <- model.matrix(Qual ~ ., data = data_wine_tidy)[, -1]
y <- data_wine_tidy$Qual

# 10-fold cross-validation to find optimal
```

```
set.seed(2025)
cv_lasso <- cv.glmnet(x, y, alpha = 1, family = "gaussian", standardize = TRUE, nfolds = 10
plot(cv_lasso)
best_lambda <- cv_lasso$lambda.min
best_lambda

# Extract coefficients
lasso_coef <- coef(cv_lasso, s = "lambda.min")
print(lasso_coef)

# Model performance comparison
set.seed(2025)
train_ctrl <- trainControl(method = "cv", number = 10)

# OLS CV
ols_cv <- train(
  Qual ~ .,
  data      = data_wine_tidy,
  method    = "lm",
  trControl = train_ctrl
)

# Lasso CV
lasso_cv <- train(
  Qual ~ .,
  data      = data_wine_tidy,
  method    = "glmnet",
  trControl = train_ctrl,
  tuneGrid  = expand.grid(alpha = 1, lambda = cv_lasso$lambda)
)

# Output model performance
ols_cv
lasso_cv



#Fit the Bayes model
priors <- c(
  set_prior("normal(0, 1)", class = "b"),
  set_prior("normal(0, 5)", class = "Intercept")
)

model_bayes <- brm(
  Qual ~ .,
  data            = data_wine_tidy,
  family          = cumulative(link = "logit"),
  prior           = priors,
  chains          = 4,
  cores           = parallel::detectCores(),
  iter            = 2000,
  seed            = 2025
)

print(model_bayes, digits = 2)

# 9.2 Posterior intervals
posterior_interval(model_bayes, prob = 0.95)

# 9.3 Posterior predictive check
pp_check(model_bayes)
```

```
# Binarize: Qual >= 6 as 1, otherwise 0
data_wine_tidy <- data_wine_tidy |>
  mutate(HighQual = if_else(Qual >= 6, 1, 0))

# Logistic regression
model_logit <- glm(
  formula = HighQual ~ Alc + Sulph + TSO2 + Chlor + VolAcid + FSO2,
  data    = data_wine_tidy,
  family  = binomial(link = "logit")
)

summary(model_logit)

# Calculate odds ratios
exp(coef(model_logit))

# Predict probabilities & ROC
library(pROC)
pred_prob <- predict(model_logit, type = "response")
roc_obj   <- roc(data_wine_tidy$HighQual, pred_prob)
plot(roc_obj); auc(roc_obj)
```