STSCI 4740 Final Project

Flavia Jiang (yj472), Chunyu Wu (cw577)

I. Introduction

In the winemaking industry, the quality of the final product is influenced by a symphony of various physicochemical elements. Therefore, the ability to predict the quality of wine from these measurable attributes holds significant commercial value. We applied and evaluated several machine learning methods to discover how various physicochemical properties were associated with the quality of red and white wines. This report presents our methods, analysis, and results to predict wine quality.

II. Data Description

The datasets came from the UCI machine learning repository: https://archive.ics.uci.edu/ml/datasets/Wine+Quality. Our team used the combined red and white wine dataset (wine-quality-white-and-red.csv) provided on the course website. The dataset contains 6,497 observations. There are 11 continuous variables describing the physicochemical properties of the wine (fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol), one categorical variable for the quality rating (on a scale of 0 to 10), and one categorical variable for the wine type (either red or white). The response variable is wine quality; the 11 continuous variables for physicochemical properties are potential predictors.

III. Methods

We separated the dataset into white and red wine datasets, examined descriptive statistics, trained various regression and classification models, and applied repeated 5-fold cross-validation to evaluate model performance. Specifically, we assessed model performance based on test accuracy, Area Under Curve (AUC), and Mean Absolute Error (MAE).

i. Performance Evaluation Metrics and Methods

Accuracy, AUC, and MAE

Accuracy is the ratio of correctly predicted instances to the total number of instances, which measures the overall correctness in predicting class labels.

AUC summarizes the trade-off between true and false positive rates. Because the distribution of response classes is unbalanced (there are many more normal wines than excellent or poor ones), knowing the specificity and sensitivity of the fitted models rather than solely accuracy is helpful. There are multiple response classes, so the AUC here refers to multiclass AUC.

MAE is the average absolute difference between predicted and actual values, which also measures the model's overall prediction accuracy. Given the unbalanced classes, we used MAE rather than MSE to avoid giving more weight to outlining predictions and, therefore, to avoid over-penalizing models that predicted higher or lower wine quality rather than medium ones.

Repeated 5-Fold Cross-Validation

To better understand the models' performances on new data, we used repeated 5-fold cross-validation to calculate the average test accuracy, test AUC, and test MAE for the fitted models. Repeated 5-fold cross-validation involves partitioning the dataset into five subsets, performing cross-validation, and repeating this process multiple times to obtain more robust and reliable performance estimates.

ii. Regression Methods

We used several regression methods for the red and white wine datasets separately. To calculate test accuracy and AUC, we rounded the fitted values to their nearest integers and compared them to the actual rating classes.

Multiple Linear Regression (MLR)

Multiple linear regression is represented by the equation $Y = b_0 + b_1X_1 + \ldots + b_pX_p + \varepsilon$, where Y is the response variable "quality"; $X_1 \ldots X_p$ are values for predictors; ε is the error term. The intercept b_0 and the regression coefficients $b_1 \ldots b_p$ are chosen to minimize the training Residual Sum of Squares (RSS). As MLR assumes linearity and is parametric, it is a relatively inflexible model, which is easier to interpret and is likely to give lower variance but higher bias than more flexible models.
We started with a full model containing all 11 explanatory variables and calculated its cross-validation accuracy, AUC, and MAE. Then we performed the best subset selection based on RSS to choose the best k-predictor model (k ranged from 0 to 11). Then we compared the resulting 12 models based on their cross-validation accuracy to identify the best reduced model.

Ridge and Lasso Regression

Ridge and Lasso regressions are extensions of MLR. Ridge regression chooses the values of the coefficients to minimize training RSS + L_2, where $L_2 = \lambda*(b_1^2 + \ldots + b_p^2)$. Lasso regression chooses the values of the coefficients to minimize training RSS + L_1, where $L_1 = \lambda*(|b_1| + \ldots + |b_p|)$. λ is a non-negative tuning parameter, which controls the relative importance between fitting the data well and keeping the model simple. Ridge and Lasso regressions are used to shrink the coefficients towards zeros in the hope of significantly reducing variance with a minor increase in bias for high-dimensional settings. Though we only had 11 predictors (and $n >> p$), we thought it was worth trying to see how the resulting accuracy compared to MLR's. So, we performed ridge and lasso regressions, with the best lambda chosen

from cross-validation based on MAE. Same as above, we repeated 5-fold cross-validation to get a robust measure of the test accuracy, AUC, and MAE.

Principal Component Regression

Principal Component Regression (PCR) combines Principal Component Analysis (PCA) with linear regression. In PCR, the predictor variables are transformed into a set of principal components, and the regression is performed on these components. The advantages of PCR include its ability to handle multicollinearity among predictor variables by creating uncorrelated main components and reducing the dimensionality of the data, which can enhance model interpretability and mitigate overfitting issues. Again, we used repeated 5-fold cross-validation to evaluate model performance. For each fold under each repetition, we used cross-validation to choose the optimized number of components to be included in PCR based on the mean squared error of prediction (MSEP).

Boosted Trees

Boosted Trees involve sequentially building models, each focusing on errors made by the previous one, thereby progressively improving accuracy. It can handle complex relationships in data and mitigate overfitting through the gradual learning process. We implemented the Gradient Boosting Machine (GBM) approach with the Gaussian distribution, 5000 trees, an interaction depth of 4, and a shrinkage rate of 0.05. Similarly, the cross-validation was structured to repeat over five random seed values to evaluate the reliability of the results.

iii. Classification Methods

Classification Tree

Trees utilize a tree-like model of decisions and their possible consequences. It breaks down the dataset into smaller subsets while, at the same time, an associated decision tree is incrementally developed. The result is a tree with decision nodes and leaf nodes, where a leaf node corresponds to a prediction outcome. It can handle numerical and categorical data and set the foundation for more tree-based ensemble methods. Again, we employed a 5-fold cross-validation approach to evaluate the model's performance, repeated over five seed settings.

Random Forest

A Random Forest consists of numerous trees during training and outputting the class of individual trees. Each tree in the forest is built from a sample drawn with replacement from the training set. When splitting a node during the tree's construction, the best split is found from all input features or a random subset. This process will reduce the variance of the model and improve generalizability. In our implementation,

each tree was trained on a randomly selected subset of the data, and each tree's decision at a split was based on a random selection of three features (mtry = 3).

Bagging

Similar to random forest, bagging builds multiple decision trees on varied dataset samples and averages their predictions. The final output prediction is averaged across the individual predictions to improve accuracy. Bagging helps to reduce variance and avoid overfitting. We set mtry = 11 to utilize all predictors at each split and evaluated its performance using repeated 5-fold cross-validation.

k-Nearest Neighbors (kNN)

kNN is a non-parametric method that classifies samples by identifying the nearest data points in the feature space. It has advantages in handling multi-class prediction tasks by classifying samples based on the closest training examples. We implemented kNN (k=1) to predict the quality of the wine based on the closest training examples and our predictions that wines with similar physicochemical profiles likely share identical quality ratings. To assess the performance, we implemented repeated 5-fold cross-validation.

Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis (LDA) is a classification method for finding linear combinations of features that best separate different classes in a dataset. LDA assumes normally distributed features in every class and equal covariance matrices across classes. However, LDA can perform well even if the assumptions are violated. Therefore, we fitted the LDA models under repeated 5-fold cross-validation without carefully checking the assumptions.

Naive Bayes

Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem, which assumes independence between features given the class label. Despite its simplicity and the naive assumption of feature independence, Naive Bayes has the potential to perform well in practice, and it is computationally efficient, particularly with high-dimensional datasets, and requires a relatively small amount of training data to estimate the parameters. So we also performed Naive Bayes.

Quadratic Discriminant Analysis (QDA)

We considered using Quadratic Discriminant Analysis (QDA). However, it was excluded from the analysis since the dataset contains classes with too few observations to estimate the necessary parameters – we got an error while attempting to implement it ("some group is too small for 'qda'). This was the case with our data with unbalanced class distributions.

IV. Results

	Red Wine			White Wine		
Method	**Accuracy**	**AUC**	**MAE**	**Accuracy**	**AUC**	**MAE**
Regression methods						
MLR - full model	59.23%	77.28%	0.504	51.89%	73.95%	0.585
MLR - reduced model	59.29%	77.43%	0.504	52.00%	74.03%	0.585
Ridge regression	59.26%	77.28%	0.504	51.74%	73.91%	0.585
Lasso regression	59.25%	77.28%	0.504	51.81%	73.95%	0.585
PCR	59.13%	77.22%	0.504	51.84%	73.49%	0.585
Boosted trees	65.49%	80.19%	0.449	63.65%	80.31%	0.477
Classification methods						
Classification tree	56.61%	75.80%	0.476	50.70%	65.24%	0.563
Random Forest	69.53%	79.33%	0.338	68.87%	79.97%	0.348
Bagging	68.90%	78.97%	0.345	68.40%	80.35%	0.352
kNN (k=1)	69.07%	84.21%	0.334	64.48%	82.35%	0.410
LDA	59.15%	77.69%	0.455	53.05%	75.05%	0.531
Naive Bayes	54.81%	80.65%	0.522	44.30%	73.54%	0.672

V. Discussion: Comparing Methods

i. Red Wine

Among all the methods, Random Forest and kNN demonstrated the highest cross-validation accuracies, surpassing 69%. They also yielded noticeable AUC values, meaning they successfully reached a good balance between sensitivity and specificity. These non-parametric methods outperformed other techniques due to their ability to handle the complex, non-linear relationships in the data. For other classification methods, bagged trees also did reasonably well. The two parametric classification methods, LDA and Naive Bayes, resulted in relatively low accuracies compared to the non-parametric models. However, Naive Bayes yielded a relatively high multiclass AUC, suggesting the model is good at balancing the recall and precision of its predictions.

The regression methods, including MLR (full and reduced), Ridge Regression, Lasso Regression, and PCR, were consistent in accuracy (around 59%) for red wine. The MAE values across these regression methods also remained constant, suggesting the linearity assumption did not help predict red wine quality.

Specifically, the reduced MLR model given by best subset selection contains seven predictors – volatile acidity, chlorides, free sulfur dioxide, total sulfur dioxide, pH, sulphates, and alcohol. This model gives a slightly higher CV accuracy than the full MLR model (59.29% > 59.23%), so overfitting might be the case for the full model. If we were to choose between the reduced MLR model and the full MLR model, the reduced model would be better because it is easier to interpret and yields higher test accuracy. For Ridge and Lasso regressions, cross-validation chose very small lambdas (mostly 0.0001), failing to perform feature selection and making them similar to MLR. For PCR, the optimized numbers of principal components to be included in the model chosen by cross-validation were constantly 9 or 10, indicating dimensionality reduction was barely performed. The boosted tree was the only non-parametric method in the 6 regression methods we used. It outperformed all the other regression methods, which demonstrated that the relatively strong assumptions of parametric methods might not be suitable in this context. Rather, a non-parametric method is more flexible and yields more accurate predictions.

ii. White Wine

For white wine, Random Forest and Bagging were the top performers regarding accuracy and MAE, with accuracy rates of over 68% and MAE lower than 0.36, reflecting their ability to predict non-linear patterns without making assumptions about data distribution. The regression models exhibited similar accuracy, around 51%-52%, suggesting a limited predictive performance when using linear approaches on this dataset. Notably, all regression methods shared an MAE of 0.585, and their AUC scores were closely clustered around 73%-74%, indicating a similar capacity for class differentiation.

The reduced MLR model contains 9 predictors and yielded an accuracy of 0.11% higher than the full model. For the Lasso regression, the tuning parameter lambdas chosen by cross-validation were tiny – around 0.0001, so it was similar to MLR in predicting white wine quality. For the Ridge regression, the lambdas ranged from 0.0001 to 0.0069, indicating that shrinkage was in place for some interactions, but overall it did not help improve the model performance. For PCR, cross-validation mainly chose 11 as the optimized number of principal components for model fitting, indicating that dimensionality reduction was barely performed. Again, boosted trees outperformed other regression methods due to their non-parametric nature, with a significantly higher accuracy of 63.65% and the best AUC of 80.31%. This suggested the ability of boosted trees to capture complex relationships in white wine data more effectively with iteratively correcting mistakes of prior trees.

VI. Limitations

Dataset Limitation

Our dataset contains only 6,000+ entries, limiting the full spectrum of variability in wine characteristics for predictive tasks. Additionally, some response classes are underrepresented. An unbalanced dataset in classification poses challenges, such as biased model training towards the majority class, leading to poor recognition of minority classes. This imbalance can result in suboptimal generalization to new data and increased noise sensitivity, impacting the classification model's overall robustness and effectiveness. Moreover, the dataset only contains variables measuring the physicochemical properties of wine. Other predictors, such as the place of origin and the winemaking technique, might be significant in predicting wine quality. Excluding them from the models might result in biased estimations.

Method Limitation

Although non-parametric models perform well in our case, there is always a risk, especially if the models are excessively complex or the data needs to be sufficiently diverse. The models need to generalize better to the real-world new data, particularly if those datasets have different feature distributions or scales of quality ratings.

We also acknowledge that there are various methods we did not use here, such as support vector machines, which might perform well for this task. Additionally, techniques such as neural networks and deep learning could uncover complex patterns that simpler models may overlook, given sufficient data. Including these methods in future work could offer a more comprehensive understanding of the factors influencing wine quality and lead to more accurate predictive models.

VII. Conclusion

In assessing the performance of various machine learning methods on this dataset, we found that Random Forest and Bagging were most effective in predicting wine quality, evidenced by their high accuracy and AUC values for red and white wines, outperforming simpler linear models and other classification approaches. Overall, non-parametric methods outperform parametric models in predicting wine quality due to their ability to handle the complex, non-linear relationships in the data. The necessity for continuous model evaluation and alternative modeling strategies is highlighted for future studies.