Final Project Report: Statistical Learning (UNL STAT 983)

by Alison Kleffner, Sarah Aurit, Emily Robinson

April 17, 2021

1 Introduction

Successful marketing campaigns and productive selling strategies are directly linked to communication about key indicators of quality; hence, objective measurements of quality are essential. Within the wine industry, there are two types of quality assessment: physiochemical and sensory tests. Sensory tests require a human expert to assess the quality of wine based on visual, taste, and smell [Hu et al., 2016]. Hiring human experts to conduct sensory tests can take time and be expensive [Gupta, 2018]. In addition, taste is the least understood of all human senses [Cortez et al., 2009]. Unlike sensory tests, laboratory tests for measuring the physiochemical characteristics of wine such as acidity and alcohol content do not require a human expert. The relationship between physiochemcial and sensory analysis is not well understood. Recently, research in the food industry has utilized statistical learning techniques to evaluate widely available characteristics of wine. It is desirable to classify wines using physicochemical properties since this does not involve human bias that would come into play with human tasters. This type of evaluation allows the automation of quality assessment processes by minimizing the need of human experts [Gupta, 2018]. These techniques also have the advantage of identifying important the physiochemical characteristics that have an impact on the quality of wine as determined by a sensory test.

The goal of this paper was to generate models that would utilize the objective physiochemical characteristics well to classify wines into three categories: poor quality, normal quality and high quality.

2 Methods

2.1 Data Source; Variables and Outcomes of Interest

We applied classication and resampling techniques to the "Wine Quality Data Set" found on the UCI Data Repository at https://archive.ics.uci.edu/ml/datasets/Wine+Quality. [UCI, Cortez et al., 2009]. The data consist of information from samples of vinho verde, which is a product from the northwest region of Portugal. The data were collected from May, 2004 to February, 2007, and consisted of 1599 red wines and 4898 white wines for a total of 6493 observations [Cortez, 2009]. For each of the wines in the dataset, 11 physiochemical variables were recorded: fxed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, and alcohol. Additionally, we classified the combined red and white wine data sets, and included an additional predictor variable categorizing the given observation as red or white wine. Our response was a quality rating based on a sensory test carried out by at least three sommeliers, where 0 was considered very bad and 10 was excellent [Gupta, 2018]. Following Hu et al. [2016], we separated the wine into three classes: Low Quality (≤ 4), Normal (5-7), and High quality (≥ 8).

3 Statistical Methods

3.1 Statistical Methods: Resampling

The categorization of quality was imbalanced as there were many more normal quality wines than low or high quality wines (Figure 1), a significant challenge. Typically, classifier algorithms assume a relatively balanced distribution; therefore, imbalanced data tend to be biased towards the majority class [Yanminsun, 2011]. Classification rules for

the minority classes tend to be undiscovered or ignored, so the minority class is misclassified more often than the majority class. To address this, we evaluated different resampling techniques in order to determine if resampling improved performance and to identify which resampling method was best. The goal with these methods is to create a data set that has close to a balanced class distribution, so the classifying algorithms will have better predictions, so it will predict to the minority class more accurately [Chawla, 2013]. We examined the data set from a binary standpoint where "Low" and "High" were classified as the minority (rare), and "Normal" was classified as the majority (not rare). When creating the resampled data sets, the binary classification of "rare" and "not rare" was used due to the resampling algorithms needing a binary class.

To see how valuable resampling was in our data set, we ran the classifying algorithms with the original imbalanced data set, a Synthetic Minority Over-Sampling Technique (SMOTE), which is an oversampling method, and a random undersampling method. We initially evaluated the classifier performance with the original imbalanced data set (without resampling), and hypothesized that it would be associated with low performance, however, we felt it provided an appropriate baseline.

Next, we handled imbalanced data through resampling the original data set by oversampling the majority class. We applied a resampling method using SMOTE, an algorithm where the minority class (in this case low and high quality) are oversampled. SMOTE, an oversampling technique uses interpolation of the minority class to create synthetic data. The process begins by finding the k nearest neighbors of each observation of the minority class based on some distance measure. Then a point between the minority class observation and one of its nearest neighbors is randomly picked by first finding the difference between the observation and its nearest neighbor. This difference is then multiplied by a random number between 0 and 1. This is then added to the observation, which becomes the new synthetic data point that is then added to the data set [Chawla, 2002]. In Hu et al, they used k=5, and we mirrored that to create our resampled data set [2016]. Oversampling tends to have an overfitting problem since now the minority class extends into the majority space, however, this generally poses less of an issue with SMOTE [Luego, 2010]. This was ran using the SMOTE function in the smotefamily package in R.

The final resampling method we applied was a random under sampler. In this method, instances of the majority class are discarded at random until reaches balanced with the minority class [Chawla, 2013]. For example, say there are 1000 observations in the majority class, and 100 in the minority class, observations in the majority class will be randomly discarded until this class is also 100. The benefit of this method is that since the data frame is being reduced, it is less costly. In contrast, since data are being removed, potentially valuable information is not considered [Hu et al, 2016]. We hypothesized that we would be impacted by this loss of information in that when information is discarded, there may be a less clear decision boundary between the majority and minority class, causing a decrease in prediction performance [Chawla, 2013].

Accuracy of each classification method and resampling technique was evaluated by the overall correct classification rate and each individual group (low quality, normal quality, and high quality) correct classification rate.

3.2 Statistical Methods: Classification

We evaluated two classification techniques, which included Random Forest and eXtreme Boosting (XGBoost) methodology, both tree based ensemble methods. Prior investigation of the white wine data showed the Random Forests technique performed well; we wanted to test this method using both white and red wine. Previous papers on the wine data set have also applied different versions of gradient boosting such as adaptive boost; we utilized an alternative gradient boosting technique XGBoost.

The first classficiation technique utilized the Random Forest algorithm and the associated selection of independent variables for the Random Forest with the randomForest and tuneRF functions. In general, a Random Forest algorithm involves the creation of many classification trees, which are each grown through recursive partioning of independent variable space. During the growth phase for each classification tree, split points of variable space are investigated to maximize the reduction of heterogeneity, which is measured with the Gini impurity index (a reflection of the probability of a variable being wrongly chosen through a stochastic process). Each tree is then grown to a phase where node(s) represent a delineation of data per split point of each variable, and leaves are the terminal points that represenet the final classification of data. After the growth phase, a leaf is potentially pruned back to a node to maximize the tradeoff between complexity and rate of misclassification. A random forest incorporates many trees and subsequently aggregates the predictions made by each tree. We initially selected 500 trees and four variables that

could be randomly sampled from to generate a split point at each node. Finally, a stepwise function was created to iteratively conduct the Random Forest algorithm for the purposes of selection of the optimal number of variables to sample from.

We also applied eXtreme Gradient Boosting (XGBoost) using the xgboost function with the multiclass classification using the softmax objective function with three classes maximizing the multiclass misclassfication error evaluation metric in the xgboost library [Chen et al., 2021]. This method combines the tree-based model approach by implementing recursive partitioning and the boosting algorithm which repeatedly optimizes classification methods on the training set. In repeated optimization, a weak classifier is fit on the original data set with each observation having equal weight, the weight is then calculated for the current model based on the error rate and observations are assigned new weights used to fit the next weak classifier. This process is repeated for a final boosted classifier given by the weighted sum of our weak classifiers. XGBoost allows for a variety of evaluation metrics providing a benefit over other boosting methods. Tuning parameters for maximum tree depth, step size shrinkage to prevent overfitting (eta) were selected by conducting a hyperparameter grid search with MCMC.

3.3 Statistical Methods: Cross-Validation

Monte Carlo Cross-Validation (MCMC) was used to select tuning parameters and evaluate model performance. The MCMC algorithm repeatedly split the data into training and testing sets (B = 50) by randomly selecting the designated proportions (70% training; 30% testing) of the overall data set for each iteration. The undersampling and oversampling techniques described below were then applied to the training set to obtain a final resampled training data set. For each split, the final resampled training data set was used to build the model and the accuracy was evaluated on the corresponding testing data set. The mean, 5th-quantile, and 95th-quantile of the correct classification rates are presented as a performance metric for the B splits.

All analyses were completed with R version 3.6.1.

4 Results

The response values were imbalanced as seen in Figure 1, with most of the wines having a response of "Normal."

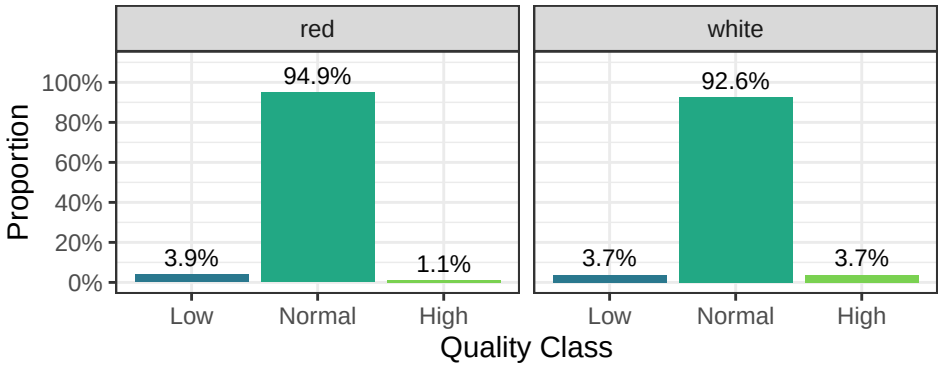

Figure 1: Wine quality class imbalance.

It was found that the majority class had 6053 observations whereas the minority class had 444 observations, a ratio of about 13:1. Due to this imbalanced nature of these data, classifying algorithms were biased towards classifying wines as Normal, and causing poor predictions for the Low and High classes.

4.1 Preliminaries; Sarah put in table and explanation in this section? SJA: sounds great

In Figure 2, boxplots of the 11 physiochemical variables are given. As can be see in this plot, for most of the predictor variables there are points that would be considered outliers, which are represented by black dots. We did not consider removing outliers, so no data points were removed for our final analysis. Looking at the boxplots, one can use these to give an idea of which of these predictor variables may be helpful in helping to determine the classification of the wines. For example, for alcohol, the Interquartile of the High category is above that of the Normal and Low class. Figure 3, gives the correlation plot for the 11 physiochemical variables, with also the 0-10 scale for the quality of the wines. Correlations that are not considered to be statistically significant are not shown. Most of the correlations seem to be on the lower end of the spectrum, so multicollinearity does not seem to be a huge issue that needs to be addressed among the predictor variables.It also shows that all of the predictor variables have a statistically significant correlation with the response of quality, so they should be helpful for classification.

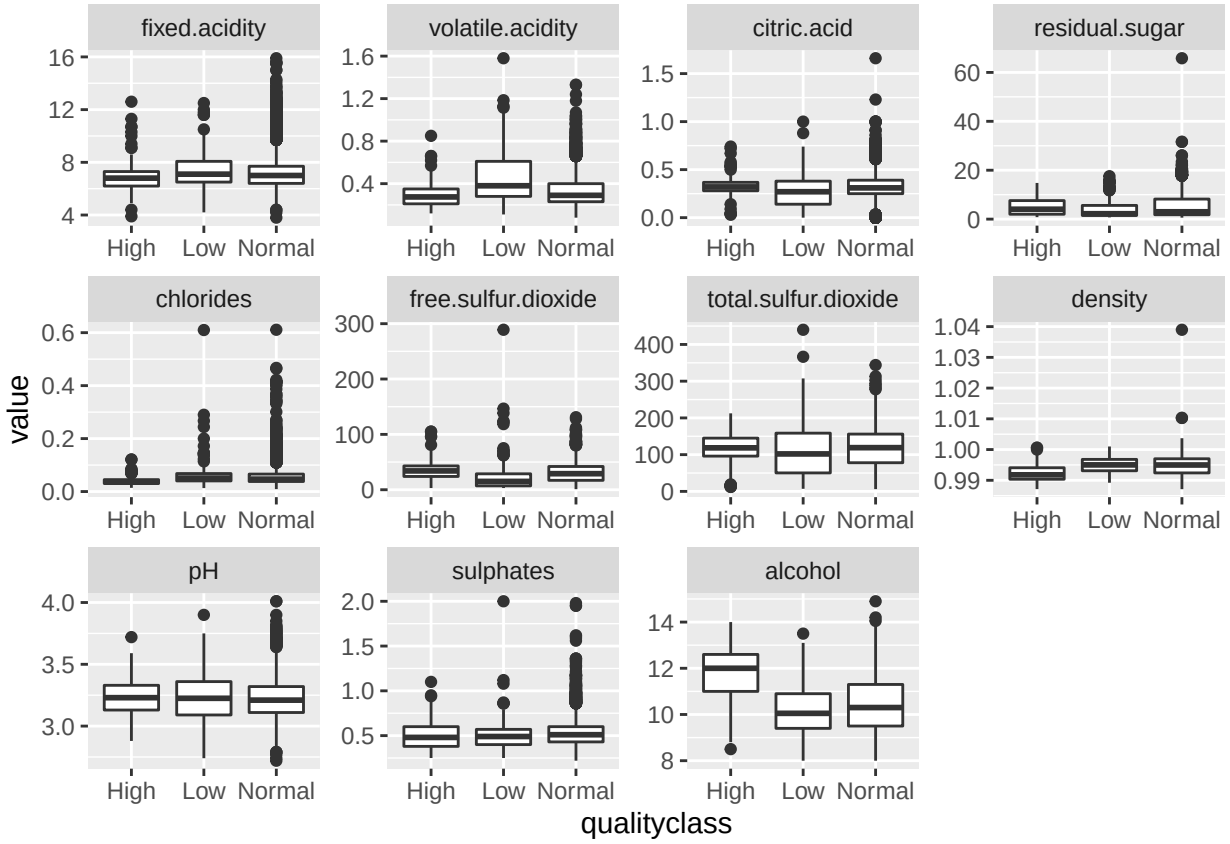

Figure 2: Distribution of the Predictor Variables

5 Discussion

Place references concerning performance of each technique here and contrast with our results

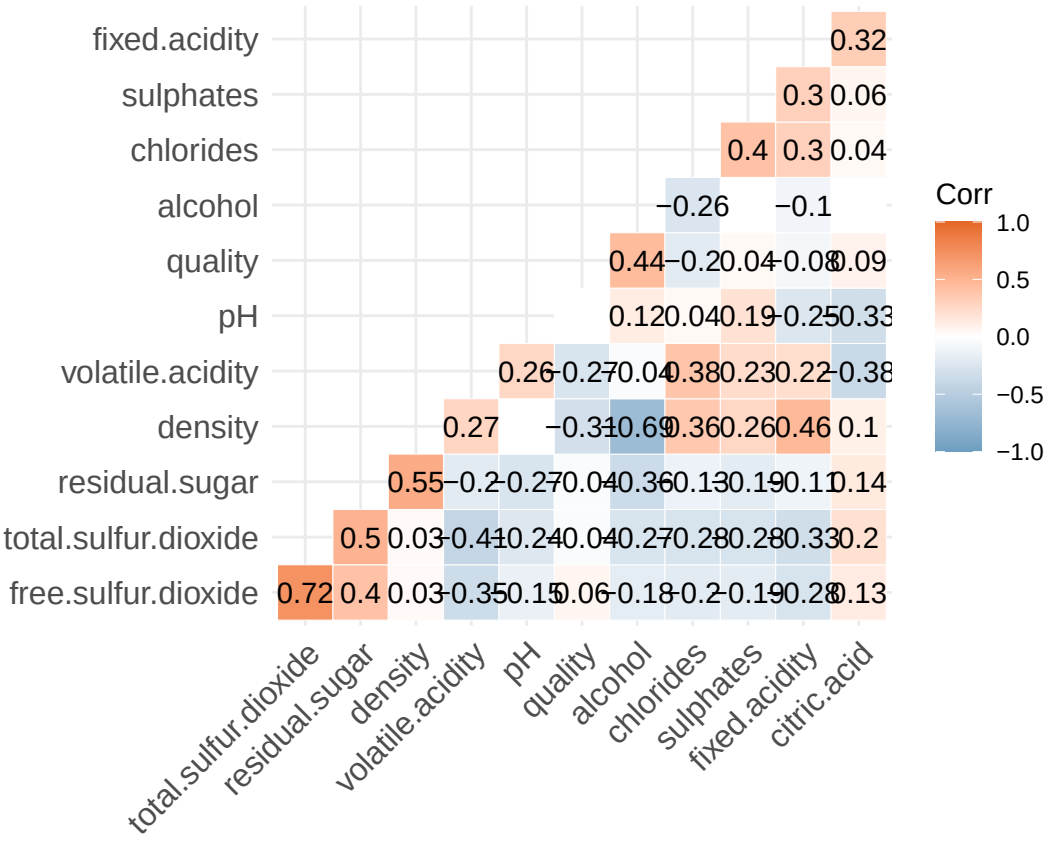

Figure 3: Correlation Plot

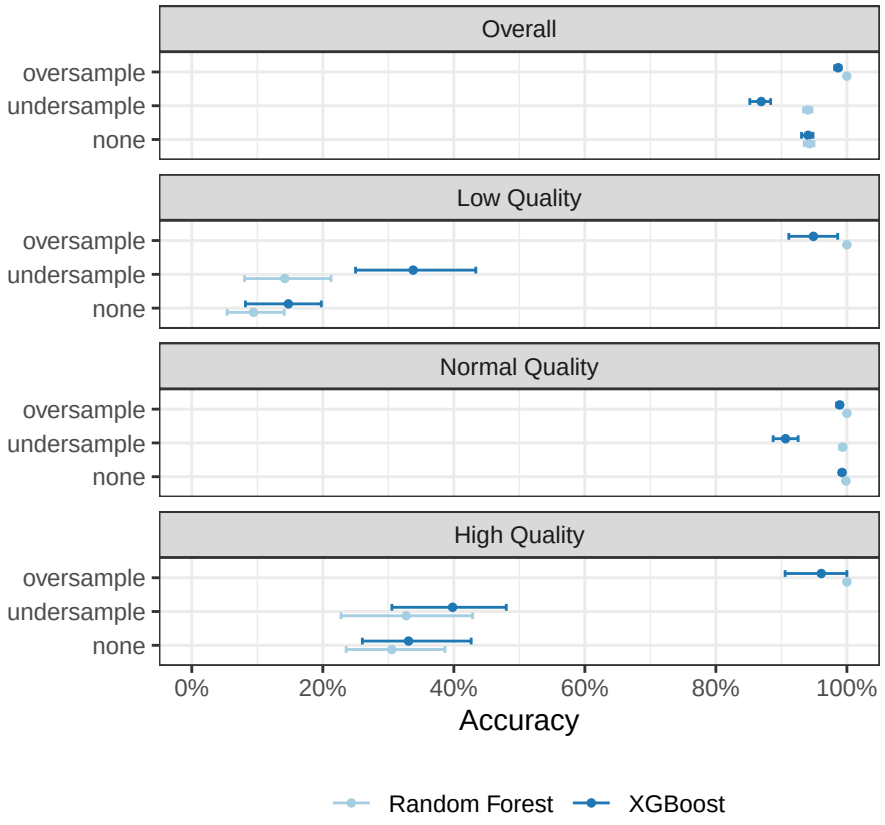

Figure 4: MCMC accuracy results for each classifier applied to each resampling technique.

6 Conclusion, Lessons Learned, and Future Directions

References

T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, K. Chen, R. Mitchell, I. Cano, T. Zhou, M. Li, J. Xie, M. Lin, Y. Geng, and Y. Li. *xgboost: Extreme Gradient Boosting*, 2021. URL https://CRAN.R-project.org/package=xgboost. R package version 1.3.2.1.

P. Cortez, J. Teixeira, A. Cerdeira, F. Almeida, T. Matos, and J. Reis. Using data mining for wine quality assessment. In *International Conference on Discovery Science*, pages 66–79. Springer, 2009.

Y. Gupta. Selection of important features and predicting wine quality using machine learning techniques. *Procedia Computer Science*, 125:305–312, 2018.

G. Hu, T. Xi, F. Mohammed, and H. Miao. Classification of wine quality with imbalanced data. In *2016 IEEE International Conference on Industrial Technology (ICIT)*, pages 1712–1217. IEEE, 2016.