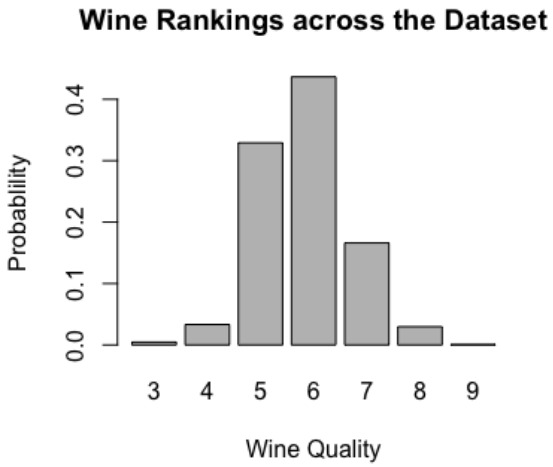
Exploring the Effect of Physicochemical Properties on Wine Ranking

Introduction and Exploratory Analysis

Vihno verde, literally "green wine," is a Portuguese wine from the Mihno region of Portugal. Contrary to its name, this wine has both red and white varieties. Our dataset is reasonably large with 6497 observations. A majority of these samples, 4898, are white wine observations and the remaining 1599 are red wine observations.

Currently, the Portuguese government requires that the wine undergo both physicochemical and sensory tests. During the sensory tests, wine experts assign each wine a quality on a scale from 3-9 where higher quality wines obtain a higher score. We began by examining the distribution of these rankings in the dataset. As shown in Figure 1, the data appears to be roughly normally distributed. Both high and low rankings are rare, and the median ranking is 6.

Figure 1: Distribution of Wine Rankings



The authors of the original study sought to predict the quality of the wine based on the physicochemical properties measured during the physicochemical tests, and thus eliminate the need for sensory tests. Human testers, even experts, are not immune to their own subjective preferences and so the wine industry could benefit from implementing a more objective procedure such as a machine learning model. Eliminating the second phase of testing could also save time and money by creating a streamlined process. However, such a model would have to be accurate and maintain the same standards for wine quality in the industry.

As shown in Table 1, white wine appears to receive a higher quality ranking on average; however, since over 75% of observations are white wine, it may be unfair to directly compare the scores. The authors chose to fit their models on the red and white wine observations separately. We will attempt to develop a model for the entire dataset using Type as a categorical predictor where Type=1 if the sample was a white wine and Type=0 if it was red wine. In doing so we will look at both regression and classification methods.

Table 1: Descriptive Statistics for Wine Quality

Variable	Type	Total (N, %)	Quality: 3	Quality: 4	Quality: 5	Quality: 6	Quality: 7	Quality: 8	Quality: 9
Wine Type	White	4898 (.754)	20 (.004)	163 (.033)	1457 (.297)	2198 (.449)	880 (.180)	175 (.034)	5 (.001)
	Red	1599 (.246)	10 (.006)	53 (.033)	681 (.426)	638 (.400)	199 (.124)	18 (.011)	0 (.000)

The dataset contains information about 12 physicochemical properties. Descriptive statistics were computed for each one and are displayed below in Table 2. In doing so, we identified several potential outliers. As for data cleaning, observation 2782 has a residual sugar content of 65.80 grams which is nearly 13 standard deviations above the mean. Observation 4745 has a free sulfur dioxide value of 289, which is more than 14 standard deviations above the mean. Given the improbability of observing these values, we chose to remove them from the dataset. Interestingly, none of the variables are highly correlated with quality.

Table 2: Descriptive statistics for numerical predictors

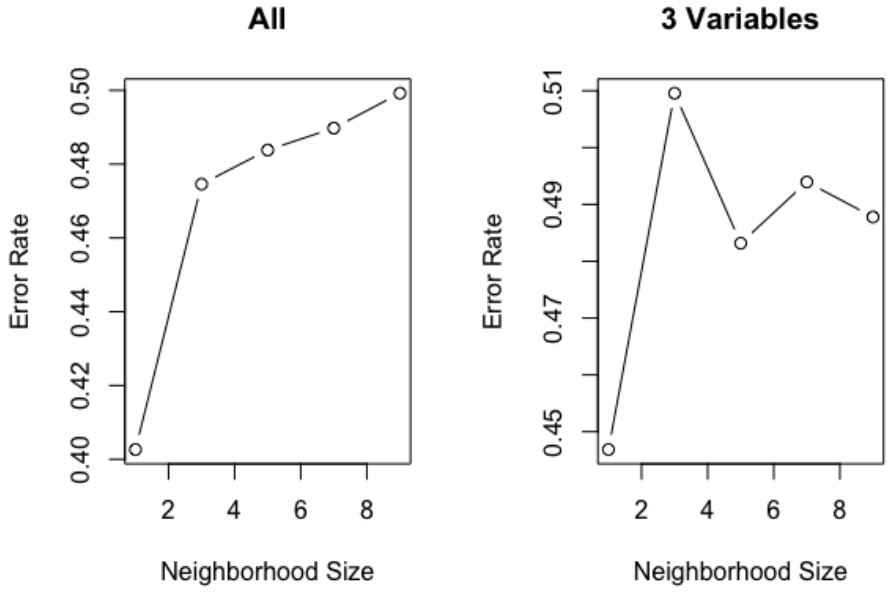
Variable	**Minimum**	**Mean**	**Maximum**	**Standard Deviation**	**Correlation with Quality**
Fixed Acidity	3.80	7.22	15.90	1.30	-0.07
Volatile Acidity	0.08	0.34	1.58	0.16	-0.27
Citric Acid	0.00	0.32	1.66	0.15	0.07
Residual Sugar	0.60	5.44	31.60	4.70	-0.04
Chlorides	0.01	0.06	0.61	0.04	-0.20
Free Sulfur Dioxide	1.00	30.53	146.50	17.46	0.06
Total Sulfur Dioxide	6.00	115.69	336.50	56.38	-0.04
Density	0.99	0.99	1.01	0.00	-0.31

pH	2.72	3.22	4.01	0.16	0.02
Sulphates	0.22	0.53	2.00	0.15	0.03
Alcohol	8.00	10.49	14.90	1.19	0.44

Classification Methods

Since the outcome variable is discrete, we first considered several classification methods. We began with k-nearest neighbors (KNN) with all 12 predictors, since this non-parametric method is easy to fit and understand. Utilizing 5-fold cross-validation we determined that the optimal neighborhood size to be k=1, which has a misclassification rate of .448. While this model performed significantly better than random guessing (which would have an error rate of 6/7=.857), it still struggled to classify many of the observations correctly. We theorized this was due to the large number of predictors and worried that the so-called "curse of dimensionality" meant there were not enough observations in each neighborhood. To address this issue, we referred to the correlations computed for Table 1 and identified Alcohol, Density and Volatile Acidity and the variables most correlated with quality. We re-fit the KNN model using these three predictors. Surprisingly, 5-fold cross-validation once again identified k=1 to be the optimal neighborhood size with a slightly better misclassification rate, .429, then before.

Figure 2: 5-fold Cross-Validation



While logistic regression can be adapted for use with multiple classes, we felt the model would become too complicated and the probabilities more difficult to obtain. Linear discriminant analysis is better suited for this situation. However, LDA performed slightly worse than k-nearest

neighbors, with a misclassification rate of .458. The results from all 3 models are summarized below.

Table 3: Summary of Classification Methods

Method	Variables Included	Test Error Rate
KNN	All	.448
KNN	Alcohol, Density, Volatile Acidity	.429
LDA	All	.458

The best classification model we identified is k-nearest neighbors with 3 predictors and a neighborhood size of k=1; however, the error rate is still relatively high. The confusion matrix is displayed below. Many of the misclassification errors are clustered along the diagonal, indicating that our model was *close* to predicting the correct class. This speaks to the ordinal nature of the data and indicates that classification methods such as KNN and LDA may not be the optimal approach.

Figure 3: Confusion matrix

		Actual					
Predicted	3	4	5	6	7	8	9
3	0	1	0	0	0	0	0
4	1	9	10	19	3	1	0
5	2	23	397	160	37	5	0
6	1	23	207	529	113	19	2
7	2	7	38	113	153	10	0
8	1	0	2	18	18	23	1
9	0	0	0	1	0	0	0

Regression Methods

To address this issue, we chose to treat quality as a continuous variable and fit several regression models to the dataset, beginning with multiple linear regression. We utilized forward selection to determine the best model at each level of complexity and then computed the AIC, BIC and Adjusted R^2 values of each of the 12 best models (including 1-12 variables, respectively). All of the models performed similarly, with high AIC/BIC values and low

Adjusted R^2 values. The model with all 12 variables had the lowest AIC and highest Adjusted R^2 values of 10107.67 and .291, respectively. The model with 9 variables had the lowest BIC value of 10182.09.

We chose the latter model as the difference in AIC and Adjusted R^2 is small compared to the best model and we prefer a more parsimonious model as it is less likely to overfit the data. This model had a test MSE of .534. The results from the two models are summarized in the table below:

Table 4: Forward Selection

Number of Variables	**Variables Included**	**AIC**	**BIC**	**Adjusted R^2**
9	Alcohol, Volatile Acidity, Total Sulfur Dioxide, Sulphates, Residual Sugar, Type, Free Sulfur Dioxide, Density, Chlorides	10107.67	10191.16	0.291
12	All	10111.45	10182.09	0.290

All variables from the chosen model are highly significant. The coefficients and p-values are displayed below.

Table 5: Multiple Linear Regression with 9 Predictors

Variable	**Intercept**	**Alcohol**	**Volatile Acidity**	**Total Sulfur Dioxide**	**Sulphates**	**Residual Sugar**	**Type**	**Free Sulfur Dioxide**	**Density**	**Chloride**
$\hat{\beta}$	47.777	0.028	-1.515	-0.002	0.064	0.038	-0.036	0.006	-44.84	-1.312
p-value	1.39e-05	< 2e-16	< 2e-16	3.00e-05	6.98e-12	2.94e-15	1.50e-07	5.72e-10	4.05e-05	0.00065

Next, we utilized ridge and lasso regression which can improve over least squares. With 10-fold cross-validation, we identified that the optimal tuning parameter for ridge regression to be λ=0.034. This model results in a test MSE of 0.532, which is only a small improvement on the linear regression. Next, we repeated the 10-fold cross validation with the L1 penalty, to determine that the optimal tuning parameter for lasso regression is λ=0.0006. With the lasso regression, we have a test MSE of 0.531.

Indeed, in the table below we can see that the coefficients in both models are very similar. Most notably, the coefficients for Type appear in the table as 0.000. While Lasso regression can perform variable selection and minimize the coefficients to zero, ridge regression cannot so the

coefficient for Type is, in reality, a very small value. However, the variable Type appears to be insignificant in both models.

Table 6: Ridge and Lasso

Variable	Ridge Coefficient	Lasso Coefficient
Intercept	32.462	37.899
Type	0.000	0.000
Fixed Acidity	0.034	0.048
Volatile Acidity	-1.278	-1.363
Citric Acid	-0.154	-0.220
Residual Sugar	0.032	0.038
Chlorides	-0.740	-0.475
Free Sulfur Dioxide	0.005	0.005
Total Sulfur Dioxide	-0.002	-0.002
Density	-30.69	-36.539
pH	0.221	0.273
Sulphates	0.754	0.796
Alcohol	0.281	0.29

The final method we tested was regression trees as this model is very different from what we tried previously and we were curious if it could improve our results. We first fit a simple tree using all the predictors which resulted in a tree that had 5 terminal nodes and included alcohol and volatile acidity. The residual mean deviance was quite high at 0.57. This served as our baseline. We then utilized pruning, random forests, boosting and BART to see if we could improve upon this result. Using cross-validation we determined the optimal level of complexity was 6. This is the same number of nodes selected in our first tree model, so pruning will not improve the test MSE.

The next method we tried was random forest, considering the default value of $p/3 = 4$ variables at every split. This resulted in a significant decrease in the test MSE to .379, which was the lowest test MSE obtained thus far. We then tried boosting with 5000 trees, which resulted in a slightly lower test MSE, but could not outperform Random Forest. BART resulted in similar

results to Boosting. The results from all of the regression methods are summarized in the table below. Random Forest is the best regression method we identified.

Table 7: Summary of Regression Methods

Method	**Test MSE**
Multiple Linear Regression	0.534
Ridge	0.532
Lasso	0.531
Regression Tree	0.566
Random Forests	0.379
Boosting	0.519
BART	0.474

Conclusions

The original authors of this study utilized Multiple Regression (MR), Support Vector Machines (SVM) and neural networks (NN) to predict wine quality for this dataset. The authors found that SVM was the most successful model, and had an accuracy of 62.4% for white wine and 64.4% for red wine, and thus had misclassification rates of 37.6% and 35.6%, respectively. This is only an improvement on our best classification method, k nearest neighbors with neighborhood size k=1, which had an overall misclassification rate of .429. For our regression methods, we have the most success with Random Forest.

The authors found sulphates, pH and total sulfur dioxide to be the most important predictors in the SVM model for red wine and sulphates, alcohol and residual sugar to be the most important predictors for white wine. We chose to use volatile acidity, density and alcohol as the 3 most important factors in our KKN model. To see if we could improve upon this result, we decided to re-do the KNN with three variables on the red and white wine data separately, using the “best” variables identified by the authors; however, it did not lead to a reduction in test error.

Clearly, the methods proposed here, and by the original authors, are not yet robust enough for commercial use. The author’s SVM model achieved less than 65% accuracy and less than 90% accuracy when the two nearest classes were also treated as a correct response. Our best classification model had even less success. The author’s model performed especially poorly for high and low responses, with all of the observations in the highest and lowest categories (3 and

9) being classified incorrectly for both red and white wine. One observation with a true quality of 3 was classified as high as a 6 and another observation with a true quality of 9 was classified as low as a five. As is clear in the confusion matrix, our model suffered with the same issue. The small number of observations in these categories is likely to blame. Before undertaking further research, it would be beneficial to obtain more samples in the two highest and lowest categories so that Figure 1 has a more uniform distribution. Identifying especially high or low quality wine is an important feature of any successful model.

If the wine industry wishes to replace their human testers, more research into how the physicochemical properties affect the taste, and therefore the quality of wine, is needed to construct a more well-informed model. Perhaps there are additional physicochemical properties not captured in this dataset that play a role in our perception of the wine. Of course, tastes and wine preference are highly subjective so it may also be that the rankings of multiple tasters over a multiple year period are too complex to be captured by a single model. However, the author's SVM model showed some promising results and even our more simple KNN and Random Forest methods had reasonable accuracy for intermediate rankings. It is our belief that with further study, the wine industry can successfully implement a machine learning model to determine wine quality.

References

Cortez, Paulo, Cerdeira, A., Almeida, F., Matos, T., and Reis, J.. (2009). Wine Quality. UCI Machine Learning Repository. https://doi.org/10.24432/C56S3T.