Predicting Wine Quality Score (Group 18)

Sid Ahuja, Xander Dawson, Zackarya Hamza

Table of contents

Summary

In this report we attempt to build a k-nearest neighbors (k-nn) classification model which predicts the quality of a Portuguese white wine based on its chemical components and physical properties. The dataset classified the wine qualities on a 10-point scale which we transformed to a binary classification problem where wines with scores of 0-5 are considered low-quality and scores of 6-10 are considered high-quality. Our final model had an accuracy of 0.77, correctly predicting 77% of the test set samples. It did a better job at correctly predicting good quality wines than bad quality wines with a recall score of 0.89. While this model can definitely be improved upon, the implications of incorrect predictions are not very harmful. Additionally, it is likely that this model will not be used solely to make decisions about wine quality and production, but rather be used alongside with other tools and rankings by professional sommeliers as well as personal preferences of consumers. With this, we believe this model can be used to make predictions about Portuguese white wines but will require further training to be used on other wines.

Introduction

Portugal is internationally recognized for its exceptional wines and booming wine industry. This distinction is rooted in the country's rich viniculture history and its diverse climatic conditions, which contribute to the production of wines with unique flavors and aromas. However, with the wine market becoming increasingly saturated and competitive, the ability to accurately assess the quality of wine based on objective measurements has become highly valuable. The quality of a wine is heavily influenced by its various chemical components and physical properties and such features can be used to predict the quality of a wine (Fernandes Ferreira Madureira and Simões de Sousa Nunes 2013).

In this report, we aim to explore the application of machine learning algorithms in predicting the quality of Portuguese white wines, based on their chemical compositions and physical properties. Our goal is to develop a predictive model that can distinguish between high and low-quality wines with a high degree of accuracy. The significance of such a model lies in its potential to provide consumers with quality predictions prior to purchase as well as provide produced with information on ways to improve their wines; our model should be particularly good at identifying good wines to provide such information to manufacturers. Through the application of machine learning, this study contributes to the growing field of data-driven approaches in food science and quality assurance, marking a step towards the integration of technology and quality wine production.

Methods

Data

In order to explore and build a wine quality classification model, we are using the wine quality data set sourced from the UCI Machine Learning Repository and created by P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis from the University of Minho in Portugal (Cortez and Reis 2009). Specifically, we are interested in predicting white wine quality based on the chemical composition of the wine. Each row represents a white wine and the chemical measurements taken from the wine and there are 4898 samples in the dataset. The target value (integer wine quality score) was determined by the Vinho Verde Wine Commission (CVRVV) of Portugal (*Vinho Verde* 2024).

Table 1: Column Descriptions

Feature	Type	Description
Fixed Acitity	Coninuous	Concentration (g/L) of tartaric acid.Impacts the tartness of wines.

Feature	Type	Description
Volatile Acitity	Coninuous	Concentration (g/L) of acetic acid.Impacts the vinegar-like taste in wines.
Citric Acid	Coninuous	Concentration (g/L) of citric acid.Impacts the freshness of wines.
Residual Sugar	Coninuous	Concentration (g/L) of sugar remaining after fermentation.Impacts the sweetness of wines.
Chlorides	Coninuous	Concentration (g/L) of chlorides.Impacts the saltiness of wines.
Free Sulfur Dioxide	Coninuous	Concentration (mg/L) of unbound SO2.Prevents microbial growth.
Total Sulfur Dioxide	Coninuous	Concentration (mg/L) of total SO2.Prevents microbial growth and impacts aroma/taste.
Density	Coninuous	Density (g/mL) measurement.Relates alcohol to sugar content.
pH	Coninuous	Measurement of wine acidity.
Suphates	Coninuous	Concentration (mg/L) of total sulphates.
Alcohol	Coninuous	Percentage (%) of alcohol content.

Analysis

To predict the wine quality, we utilized the k-nearest neighbors (k-nn) algorithm and built a classification model based on certain features within the dataset (specifically alcohol, volatile acidity, total sulfur dioxide content, density, chlorides, and residual sugar of the wines). First we converted the quality_score target column into a quality_class column where scores 0-5 were considered bad and scores 6-10 were considered good. We did this to reduce the number of target classes (creating a binary classification problem) as well as to allow for more examples within each class. Then we split the data into train (70%) and test splits (30%). All selected features were scaled prior to model training. We selected the features based on a qualitative analysis of their distribution for each class; features that greatly overlapped across classes were

dropped. Then, the best value for hyperparameter K was determined using a 10-fold cross-validation test. For this model, we determined accuracy to be the best measurement/metric for assessing our model as there are a similar number of samples within each class. For the confusion matrix metrics, we consider good to be the positive category and bad to be the negative category. The R programming language (R Core Team 2019) and the following packages were used to perform the analysis: tidyverse (Wickham 2017), tidymodels (Kuhn and Wickham 2020), repr (Angerer, Kluyver, and Schulz 2023), psych (William Revelle 2024), kknn (Schliep and Hechenbichler 2016), and knitr(Xie 2014).

Results

We start by loading in the raw data as seen in Table 2. Then we processed the data and generated a summary table describing the features within the dataset, shown in Table 3.

Table 2: Raw Wine Data

fixed_acidity	volatile_acidity	citric_acid	residual_sugar	chlorides	free_sulfur_dioxide	total_sulfur_dioxide	density	pH	sulphates	alcohol	quality_score
7.0	0.27	0.36	20.7	0.045	45	170	1.0010	3.00	0.45	8.8	6
6.3	0.30	0.34	1.6	0.049	14	132	0.9940	3.30	0.49	9.5	6
8.1	0.28	0.40	6.9	0.050	30	97	0.9951	3.26	0.44	10.1	6
7.2	0.23	0.32	8.5	0.058	47	186	0.9956	3.19	0.40	9.9	6
7.2	0.23	0.32	8.5	0.058	47	186	0.9956	3.19	0.40	9.9	6
8.1	0.28	0.40	6.9	0.050	30	97	0.9951	3.26	0.44	10.1	6

Table 3: Summary statistics of each column in the dataset

n	mean	sd	median	min	max	range
3428	6.8647170	0.8350113	6.8000	3.90000	14.20000	10.30000
3428	0.2772943	0.0996217	0.2600	0.08000	0.96500	0.88500
3428	0.3327100	0.1176027	0.3100	0.00000	1.00000	1.00000
3428	6.4113040	5.0873601	5.3000	0.60000	65.80000	65.20000
3428	0.0457611	0.0219342	0.0430	0.00900	0.34600	0.33700
3428	35.3327013	17.0846500	34.0000	2.00000	289.00000	287.00000
3428	138.0439032	42.2618066	134.0000	9.00000	440.00000	431.00000
3428	0.9940724	0.0029959	0.9938	0.98713	1.03898	0.05185
3428	3.1875875	0.1502339	3.1800	2.74000	3.81000	1.07000
3428	0.4889673	0.1139145	0.4700	0.22000	1.08000	0.86000
3428	10.4881145	1.2175065	10.3000	8.00000	14.20000	6.20000
3428	1.6651109	0.4720206	2.0000	1.00000	2.00000	1.00000

We can see in (**summary-stats?**) that there are no missing values as well as the summary metrics for each column. This table is generated using unscaled data so that we can use out intuition and recall the specific units of each column, gaining a better understanding of the column characteristics.

Table 4: Summary statistics of each column by wine class

quality_category	count	percentage	fixed_acidity_avg	volatile_acidity_avg	citric_acid_avg	residual_sugar_avg	chlorides_avg	free_sulfur_dioxide_avg	total_sulfur_dioxide_avg	density_avg	pH_avg	sulphates_avg	alcohol_avg
bad	1148	33.488[illegible]	[illegible]984408	[illegible].3080662	[illegible].33367[illegible]	[illegible]897822	[illegible].05058[illegible]	[illegible]95122	146.8214	0.9950[illegible]	[illegible]	[illegible]	[illegible]59117
good	2280	66.511[illegible]	[illegible]804452	[illegible].2618004	[illegible].33222[illegible]	[illegible]766338	[illegible].04333[illegible]	[illegible]52478	133.6243	0.9935[illegible]	[illegible]	[illegible]	[illegible]804820

From Table 4, we can see that about two-thirds of the dataset are wines under the good category, and the remaining one-third are bad wines (based on our definition of good/bad). Immediately we can see some features have similar averages between both categories and thus, those features may not be good to add in the model as they do a poor job discerning the class. However we still must consider the distributions of these features.

In the Figure 1 above, alcohol, volatile acidity, total sulfur dioxide content, density, chlorides, and residual sugar of the wines all seem to have distinct distributions for both classes of wine; the distributions have non-overlapping peaks and regions. Such features are good to add in the model because they can be used to identify one type of wine from the other.

Next, we perform hyperparamter optimization and make the train and fit the model using cross-validation to find the optimal K value for this classifier.

Table 5: Cross-validations scores for different K values

neighbors	.metric	.estimator	mean	n	std_err	.config
1	accuracy	binary	0.7710117	10	0.0064373	Preprocessor1_Model01
6	accuracy	binary	0.7678098	10	0.0064423	Preprocessor1_Model02
11	accuracy	binary	0.7683921	10	0.0051562	Preprocessor1_Model03
16	accuracy	binary	0.7689777	10	0.0055471	Preprocessor1_Model04
21	accuracy	binary	0.7730628	10	0.0055958	Preprocessor1_Model05
26	accuracy	binary	0.7707304	10	0.0061132	Preprocessor1_Model06

Figure 2 shows us that as K becomes larger, the accuracy of the model decreases. The model is overfitted at low K values and tends toward underfitting as K increases. The ideal K value for this problem seems to be around 20-25. Specifically, the best value for K is 21.

Finally, we use our test set to evaluate the classifier. We use several metrics to assess our model as seen below.

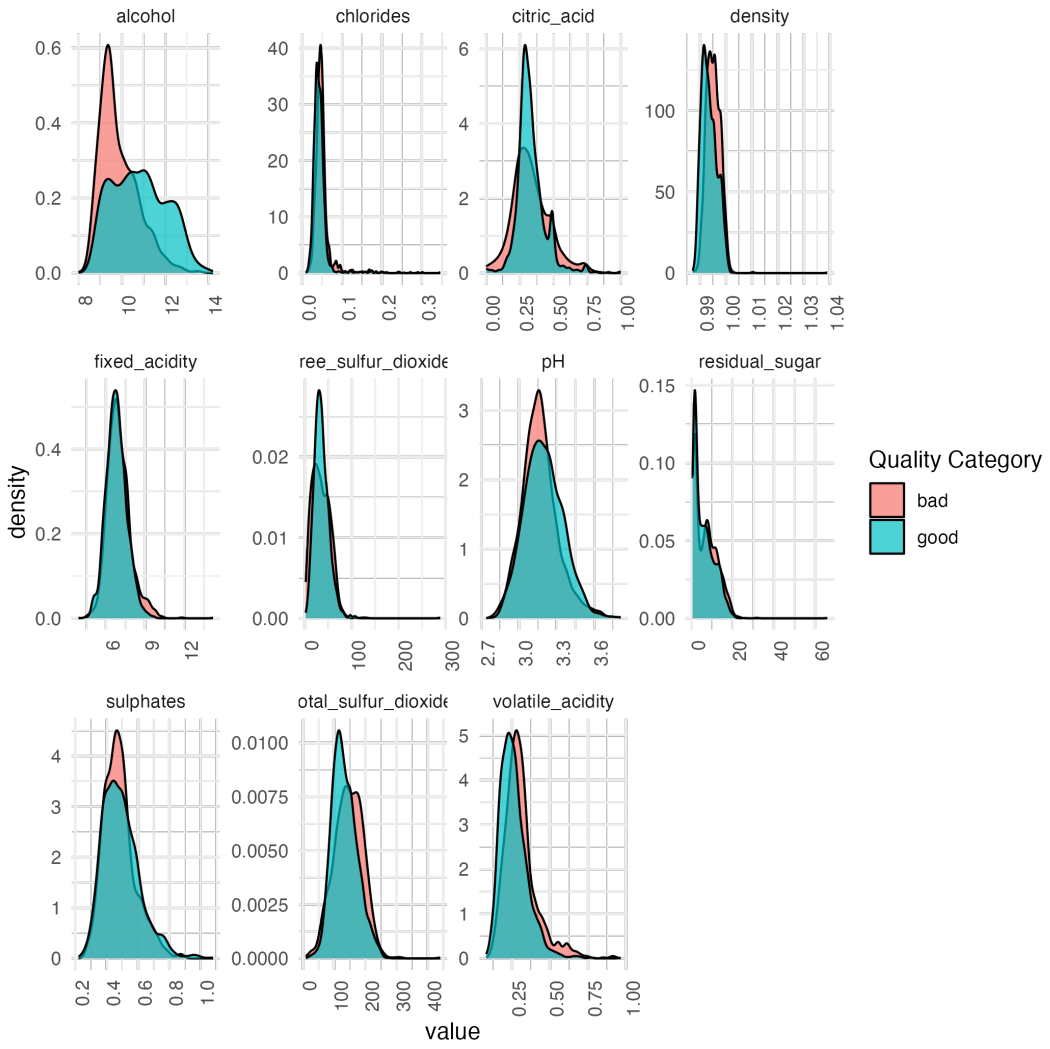

Figure 1: Distributions of feature values between both classes of wine.

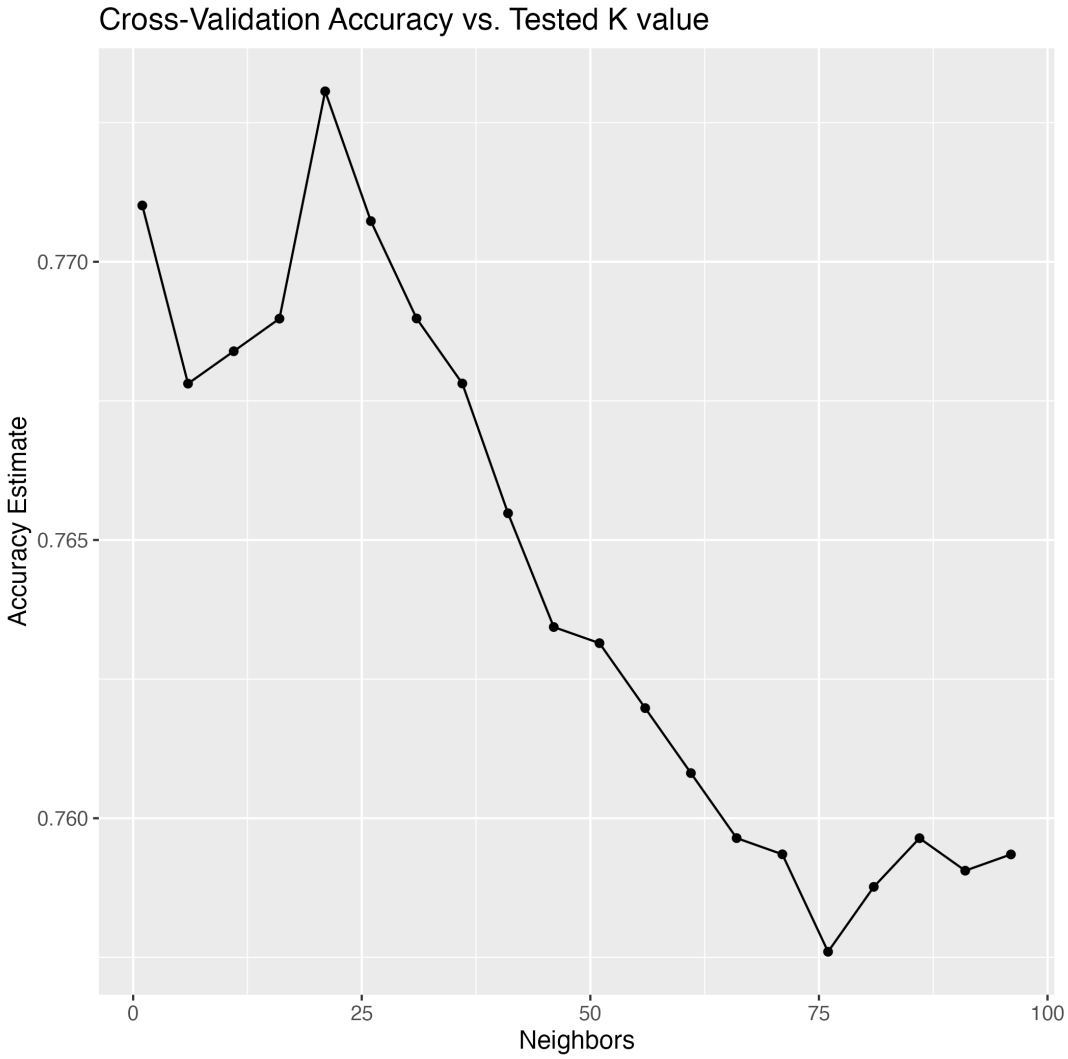

Figure 2: Accuracy scores for different values of K.

Table 6: Accuracy and other metrics for evaluating the model

.metric	.estimator	.estimate
accuracy	binary	0.7734694
precision	binary	0.7178082
recall	binary	0.5325203

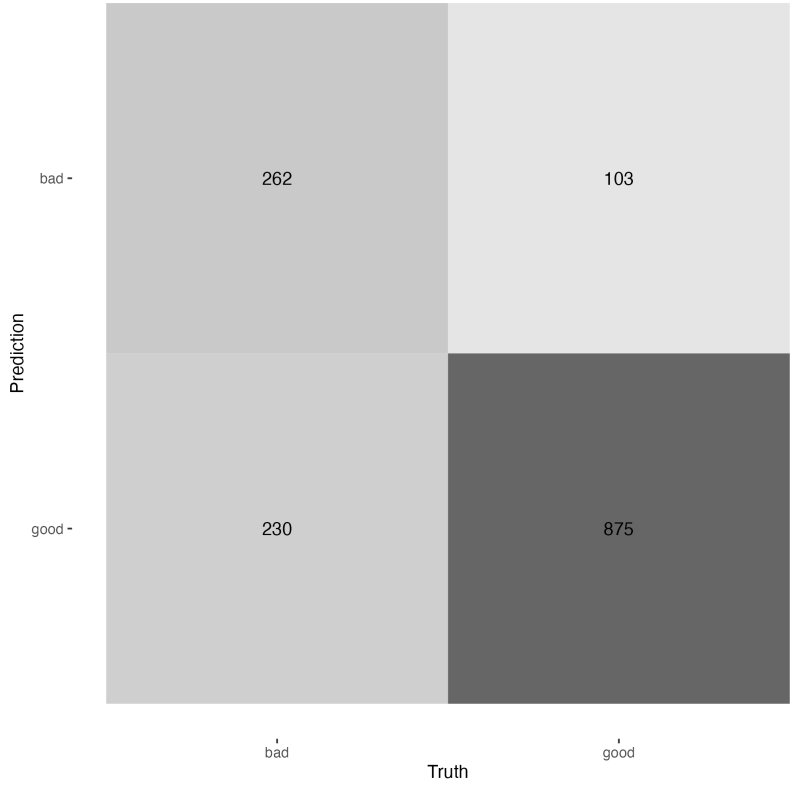

Figure 3: Confusion Matrix.

Table 6 above present the accuracy, precision, and recall of our model on the test set. With an accuracy of 0.77, out model is good but can clearly be improved upon. Additionally, for the recall and precision tests, the good wine category is considered to be the positive class. We can see that the recall is high, meaning that the model has a high true positive rate (TPR). Figure 3 shows the confusion matrix, further emphasizing the model assessment.

Discussion

The wine-quality prediction model seems to do okay with the test data, having an accuracy of 0.77. It does a decent job at classifying good wines as good, where ~90% of true good wines were predicted to be good-quality. However, the model seems to not have a high true negative rate; only ~50% of true bad wines were predicted to be bad quality (as seen in Table 6 and Figure 3). We could try to increase the sensitivity of the model or further optimize it, but seeing as wine quality tends to be quite subjective and that the implications of an incorrect prediction are not severe, this model is passable as a predictor. To improve this model, we could use a more concrete and quantitative approach to feature selection and choose a metric that is suited for a 1:2 class ratio within the dataset. We could also use a different classification strategy such as SVM or Random Forest Classifier. In its current state, this model is best used as a reference where wine producers and consumers can predict wine qualities while determining the quality through other means as well.

References

Angerer, Philipp, Thomas Kluyver, and Jan Schulz. 2023. *Repr: Serializable Representations*. https://CRAN.R-project.org/package=repr.

Cortez, Cerdeira, P., and J.. Reis. 2009. *Wine Quality*. UCI Machine Learning Repository. https://doi.org/10.24432/C56S3T.

Fernandes Ferreira Madureira, T.C., and F. J. Simões de Sousa Nunes. 2013. *Relevant Attributes of Portuguese Wines: Matching Regions and Consumer's Involvement Level*. International Journal of Wine Business Research. ttps://doi.org/10.1108/17511061311317318.

Kuhn, Max, and Hadley Wickham. 2020. *Tidymodels: A Collection of Packages for Modeling and Machine Learning Using Tidyverse Principles*. https://www.tidymodels.org.

R Core Team. 2019. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Schliep, Klaus, and Klaus Hechenbichler. 2016. *Kknn: Weighted k-Nearest Neighbors*. https://CRAN.R-project.org/package=kknn.

Vinho Verde. 2024. CVRVV. https://www.vinhoverde.pt/en/homepage.

Wickham, Hadley. 2017. *Tidyverse: Easily Install and Load the 'Tidyverse'*. https://CRAN.R-project.org/package=tidyverse.

William Revelle. 2024. *Psych: Procedures for Psychological, Psychometric, and Personality Research*. Evanston, Illinois: Northwestern University. https://CRAN.R-project.org/package=psych.

Xie, Yihui. 2014. "Knitr: A Comprehensive Tool for Reproducible Research in R." In *Implementing Reproducible Computational Research*, edited by Victoria Stodden, Friedrich Leisch, and Roger D. Peng. Chapman; Hall/CRC. http://www.crcpress.com/product/isbn/9781466561595.