Predicting Wine Quality Using Machine Learning Techniques

Kaylynn Mosier

9 August 2024

Introduction

Everyone loves a good bottle of wine but choosing that bottle can be a difficult task. Any wine store worth its weight will have recommendations on deck for their customers. However, these recommendations are often based on the store owners' opinions and not hard facts. Customer alienation occurs when a business has practices that make the customer experience less enjoyable. In this case, recommending a good wine that turns out to not be so good could alienate customers and make them patronize other stores. On the other side of that, consistently and accurately recommending good wine to customers will increase sales, customer loyalty, and retention.

In this evaluation, I struck out to find a way to predict which wines are good in order to provide customers with the best wine recommendations. Using two datasets from the UC Irving Machine Learning Repository, I built a supervised machine learning model to pair measurable physiochemical properties with qualitative sensory data. The ultimate goal of this model is to remove human opinion from the mix and build a model that can accurately predict the quality of a bottle of wine using only the provided physiochemical properties.

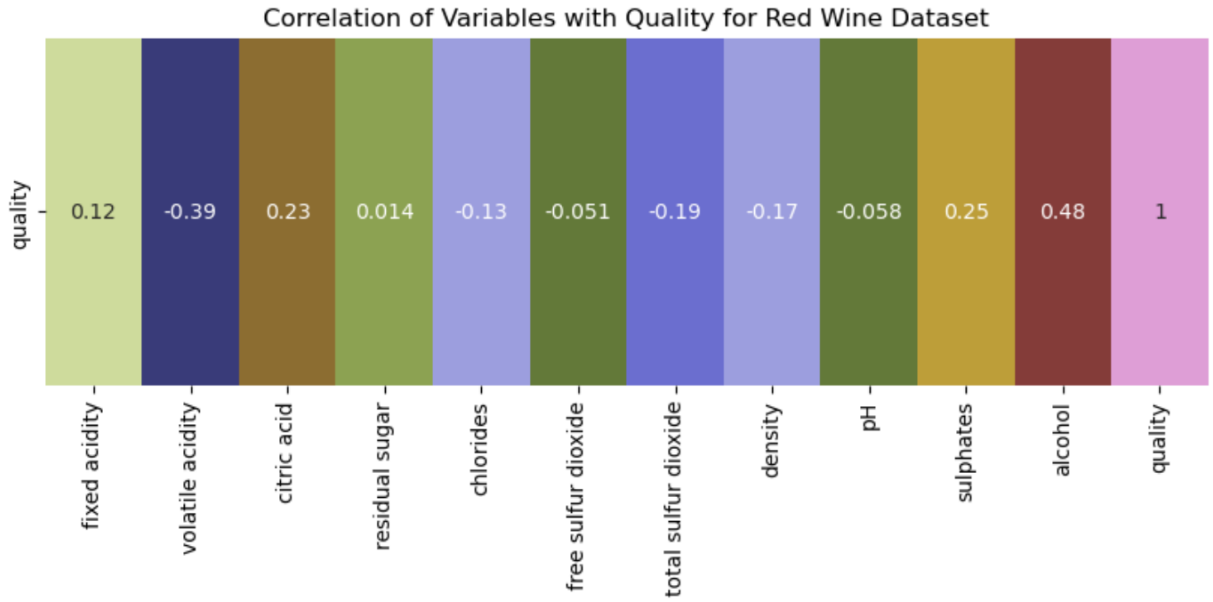
Exploratory Data Analysis

All analysis described in the following sections was done on two datasets, one that contained physiochemical and sensory data about red wine and the other on white wine. To keep analysis within the scope of this project, I chose to keep the datasets separate from each other. Both datasets contain the same feature variables: fixed acidity, volatile acidity, citric acid, residual sugars, chloride, free sulfur dioxide, total sulfur dioxide, density, pH, sulfates and

alcohol. The target variable for both datasets is quality, which was initially reported on a scale from 1-10 where 10 was the best quality rating. The datasets were both free from missing values without any transformations on my part. It is important to note that these datasets are not balanced; there are many more observations of average wines than very good or very poor wines.

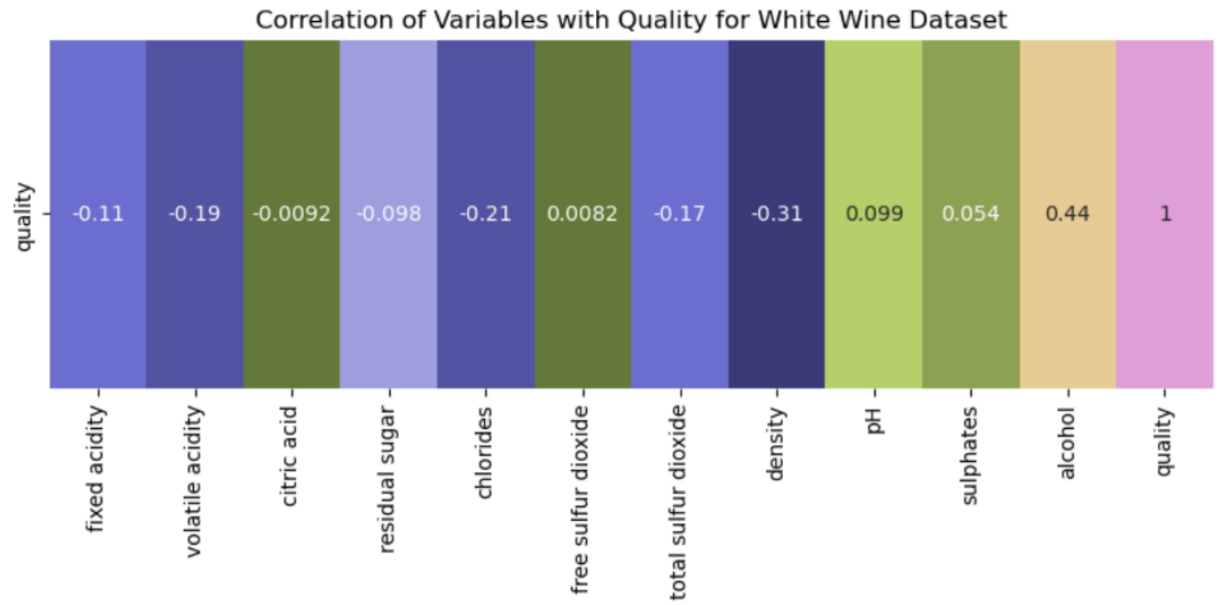
Red Wine

I began exploratory analysis on this dataset by plotting histograms of each variable. These histograms revealed pH and density are roughly normally distributed while all other features display right skews. The target variable, quality, displays a slight left skew. Boxplots displayed outliers in all variables. Plotting a correlation matrix of all variables with the target variable (shown below) reveals no feature has a strong correlation with quality. Further analysis showed moderately strong correlations between a few features which could result indicate multicollinearity. Due to this possibility, it will be advantageous to explore dimensionality reduction during model building.



White Wine

Similar to during the exploratory analysis above, I began analysis of this dataset by plotting histograms of each variable. In this case, pH was the only variable with a roughly normal distribution while the other variables displayed a slight right skew. Boxplots of each variable displayed outliers for all variables except alcohol. A correlation matrix of all variables with the target variable (shown below), showed no features had a strong correlation with quality. An expanded correlation matrix revealed a strong positive correlation between density and residual sugar as well as a strong negative correlation between density and alcohol. Additionally, there was a moderately strong correlation between total sulfur dioxide and free sulfur dioxide.



Data Preparation

The main concern during data preparation is usually transforming missing values and removing or transforming outliers. Neither dataset used in this analysis contains any missing

values. Both datasets do contain outliers, however I chose to retain them in the datasets. The sheer number of outliers made me hesitate to remove them outright. Instead, during the model building stage I tested a model with a minmax scaler applied to minimize the noise from outliers.

The only major change I made to the datasets was to recode the target variable. For the red wine dataset quality values range between 3 and 8 while for the white wine dataset quality ranges between 3 and 9. I am only interested in if a wine is good or bad, I am not interested in how good a good wine is (an 8 vs 9 rating is arbitrary at this stage). For that reason, I recoded the quality column to contain only good or bad ratings. All quality values of 7 or greater were considered ‘good’ and all other values were considered ‘bad’. I chose to be conservative when labeling wine as ‘good’, because I want to reduce type II error as much as possible. A wine store that consistently recommends wines that are not actually good will gain a poor reputation. For that reason, the best model will be one that minimizes type II error. The only downside to this approach is it results in an imbalanced dataset; there are many more bad wines than good wines in both datasets.

Model Building and Evaluation

Both datasets contained labeled data; I already knew what the quality rating of each wine was. For this reason, I employed supervised models by splitting each dataset into two sets: one training set and one test set. This is a classification problem, so I had a few choices when deciding what model to go with. To begin, I created a cross-validation pipeline using sklearn to test which supervised learning model was the best fit for my data. I chose to test logistic regression, random forest classifier, gaussian naïve bayes, and decision tree classifiers. After hyperparameter tuning, I checked which classifier and hyperparameters produced the most

accurate model. For both datasets, the decision tree classifier with 1000 estimators was the best model.

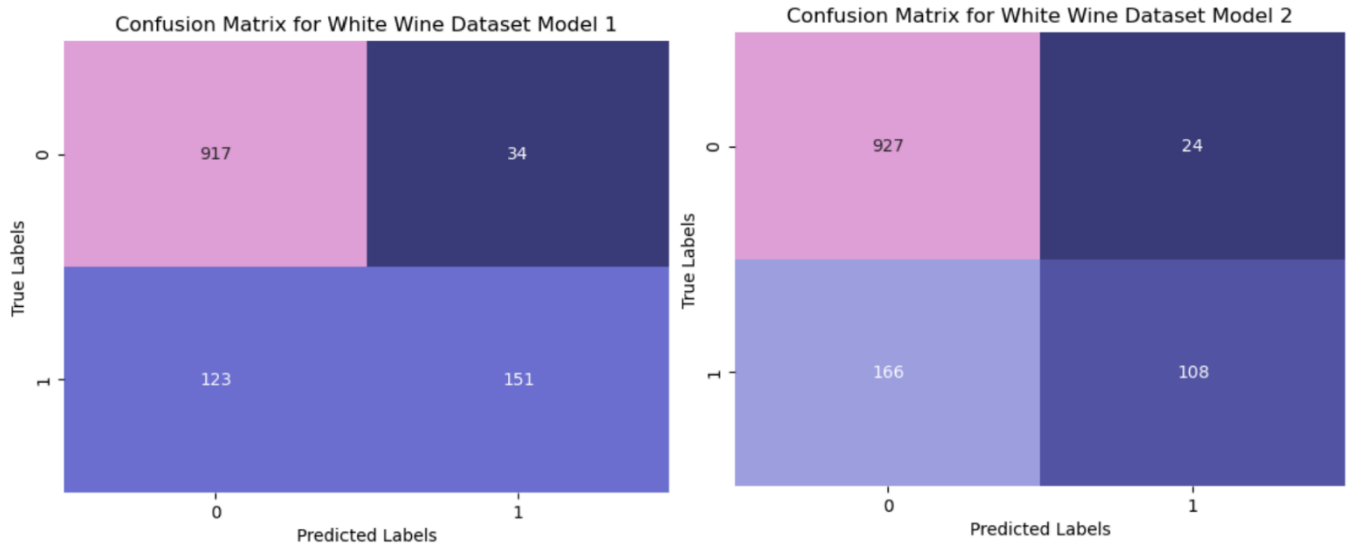
With the best model found, I fit the model to each training set, made predictions using the model and evaluated the model using accuracy, F1 score, and confusion matrices. The accuracy of a model is the number of correct predictions divided by the total number of predictions. The F1 score is a way to incorporate precision and recall into one metric. The closer the value is to 1.0, the better the model is performing. The values for accuracy and F1-score are outlined in the table below in the sections labeled 'Model 1'. Both models have relatively high accuracy values, but moderate F1 scores which may indicate imbalances in type I and type II error.

Although this initial model is accurate for both datasets, I explored a few other areas of improvement. First, I chose not to remove outliers during the data preparation step. To minimize the influence of outliers, I applied a minmax scaler to the datasets and scaled all feature values between 0 and 1. Second, none of the features had a strong correlation with the target variable, but a few of the variables were highly correlated with other variables. To correct for this, I employed PCA as a dimensionality reduction technique in model 2. The accuracy and F1 scores of this model are outlined in the table below in the section marked 'Model 2'. In both the red wine and white wine dataset, there was a reduction in accuracy between model 1 and model 2. In the red wine dataset there was a negligible changes in F1 score and in the white wine dataset there was a decrease in the F1 score.

	Accuracy	F1
Red Wine Model 1	92.5	0.55
Red Wine Model 2	92.0	0.567
White Wine Model 1	87.2	0.658
White Wine Model 2	84.5	0.532

These metrics would seem to indicate the addition of a minmax scaler and PCA are not useful additions to the model. However, that may not be the case for the white wine dataset.

As stated previously, reducing type I error is of utmost importance in this model. On the white wine dataset, the addition of a scaler and use of dimensionality reduction reduces type I error (top right box in the images below). In exchange, type II error is greatly increased. Type II error results in labeling wines 'bad' when they are supposed to be labeled as 'good'. In most circumstances, an increase in either error type would not be favorable. However, a type II error would be much more costly to business.



Conclusion

Analysis of the dataset and model reveals it is likely possible to build a model to predict the quality of wine given enough datapoints. This dataset has only captured a very small corner of the wine market, so before this model is deployed, I recommend training the model on a more robust variety of wines. This has not been an exhaustive effort in model building. There are additional opportunities for model improvement by reducing outliers, fine-tuning hyperparameters, and even exploring additional classifiers.

Works Cited

UCI Machine Learning Repository. (n.d.). https://archive.ics.uci.edu/dataset/186/wine+quality