For this analysis I will be using the classification models decision tree, and random forest to see which factors will make wine “good” or “bad” quality. The reason I chose this dataset is because I have always heard that drinking a glass of wine a day is healthy. So, if I were to invest into a wine to drink a day, it should be wine that is of good quality and not bad quality.

To start off, I imported all the necessary libraries required to make build these two models and then uploading the dataset into my notebook. Then, I took a look at the dataset to see if there are any null values and to see if standardization is necessary. After that, I made a bar graph of the count of good quality wines and bad quality wines in the dataset. We can see from this graph that there is a higher number of good quality wines than bad ones. Next, I made a histogram to see the spread of the alcohol content in this data and found that a majority of the wines are on the lower end of the alcohol content spectrum (mainly around 9% – 10%). I also made a histogram to see the distribution of the pH levels of the wines. I found that all the pH levels are on the acidic side with the highest being 3.3. I also made a correlation heatmap to see if there are any strong correlations between the columns in the dataset. Then, I one hot coded the quality column to make it 0 for bad and 1 for good quality wine and replaced the original column with the new one hot coded column.

It was time to set up the models. First, I separated the X values (all other columns beside the quality column) and the y value (quality column). Then, I split the data with test being 30% of the dataset and a random state of 20. After that, I fit the data into a decision tree model. I also performed a 5-fold cross validation for the decision tree and found that the average accuracy of the model is around .73. After that I made a classification report for the decision tree. It predicted the good wine with the precision of .70 of the time and the bad wine .79 of the time. The accuracy of this model is .75. F1 score is .73 for good wine and .76 for bad wine. Overall, this model does decently well at predicting the quality of the wine based on the 11 attributes presented in the data.

Now to make the random forest. First, I created the random forest model setting n_estimators at 10, criterion as ‘gini’, and random state to 1. After that, I fitted the data that we had split earlier for the decision tree into the random forest model. I also ran a 5-fold cross validation for the random forest model and got an average accuracy of .77. The classification report shows that the random forest has a precision rate of .71 for classifying the good wine and .81 for bad wine. F1 score is .75 for good wine and .77 for bad wine. The accuracy for this model is stated to be .76. Overall, the random forest is slightly better than the decision tree for classifying the quality of wine based on the 11 attributes.