Analysis of Supervised Learning Techniques on Red Wine Quality and Breast Cancer Diagnosis

Jay Patel
CS 4641 Machine Learning

1. Abstract

Using supervised learning algorithms for the evaluation and prediction of modern datasets are really common. This paper is an analysis of five supervised learning algorithms used for determining red wine quality and determining breast cancer diagnosis. The list of algorithms are Decision Trees With Pruning, Boosted Decision Trees, Neural Networks, Support Vector Machines, and K-Nearest Neighbor. Cross-validation is used to determine the accuracy and the dataset is split into an 80/20 ratio of training dataset and testing dataset. The choices of hyperparameters and their outcomes and the choices that lead to overfitting and overall accuracy are very significant.

2. Red Wine Quality

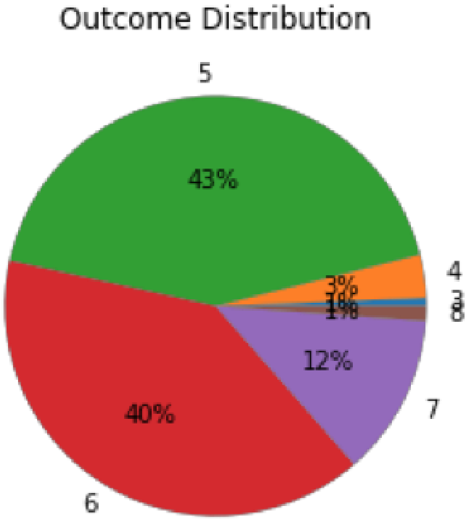
This dataset was retrieved from Kaggle.com and is related to red variants of the Portuguese "Vinho Verde" wine. The goal of the dataset is to predict the quality of wine using a pre-determined set of measurements. The dataset has 1599 instances and 11 attributes. The attributes are:

fixed acidity	Most acids involved with wine or fixed or non-volatile (do not evaporate readily)
volatile acidity	The amount of acetic acid in wine, which at too high of levels can lead to an unpleasant, vinegar taste
citric acid	found in small quantities, citric acid can add 'freshness' and flavor to wines
residual sugar	the amount of sugar remaining after fermentation stops.
chlorides	the amount of salt in the wine
free sulfur dioxide	the free form of SO2 prevents microbial growth and the oxidation of wine

total sulfur dioxide	The amount of SO2 becomes evident in the nose and taste of wine
density	the density of wine is close to that of water depending on the percent alcohol and sugar content
pH	describes how acidic or basic a wine is on a scale from 0 (acidic) to 14 (basic)
sulphates	a wine additive which can contribute to sulfur dioxide gas (S02) levels, which acts as an antimicrobial and antioxidant
alcohol	the percent alcohol content of the wine
quality	output variable (based on sensory data, score between 0 and 10)

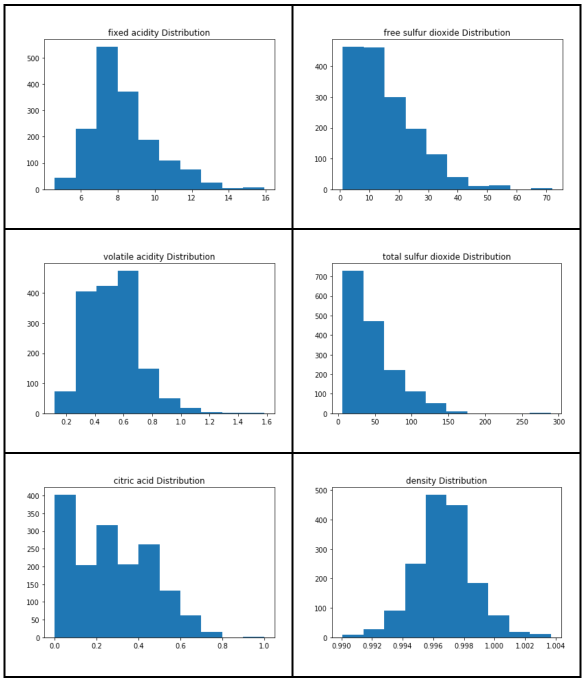
Outcome Distribution

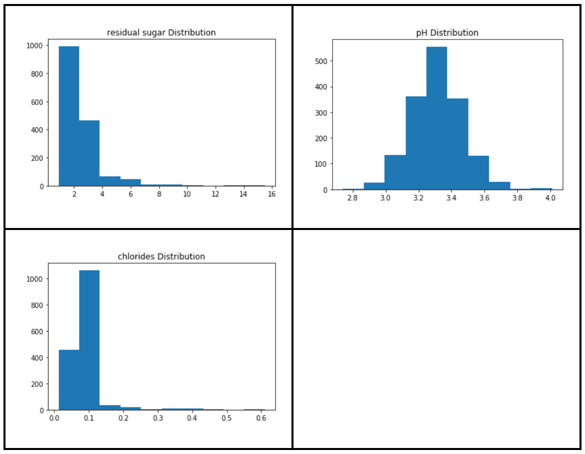
43% of the data had a quality rating of 5, 40% had a quality rating of 6 and 12% had a quality rating of 7. The remaining 5% had all other ratings. Due to the unbalanced nature of the dataset, our predictive models can expect an accuracy of at least 43%.



Distribution of Attributes

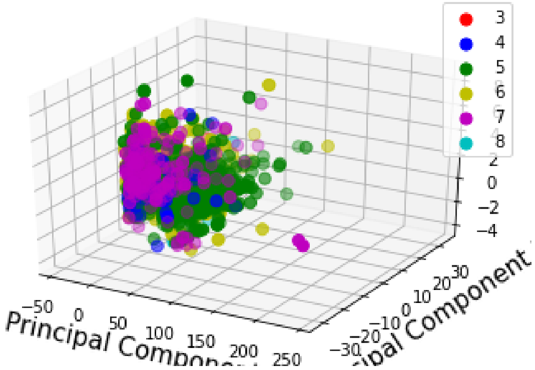
For the distribution of the inputs: fixed acidity, volatile acidity, density, and pH level had normal distributions, while the remaining features were all skewed to the left.





Priniciapal Componet Analysis

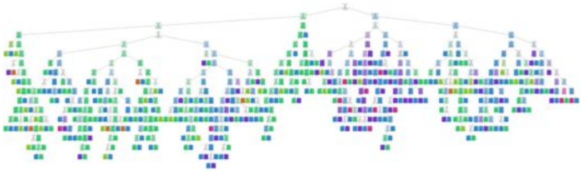
Using PCA, we can reduce the dimensions from eleven down to three so we are able to visualize the data in a 3-D graph. We can see the clustering in the graph and determine that it is a good dataset to be classifed



Decision Tree with Pruning

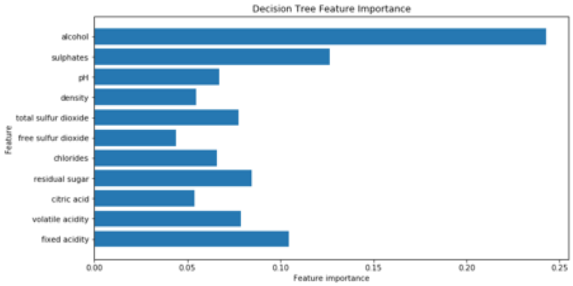
A decision tree is an intuitive flowchart-like structure that passes observations through a series of rules in the form of if-else statements. The rules are based on the training data Overfitting is when a model is not able to generalize beyond its training data. Pruning is a way of reducing the size of the decision trees by removing parts of the tree that do not provide power to classify instances and the goal is to acquire a target value. Decision trees might turn very complex leading to overfitting.

The decision tree without any form of pruning, see below, gives us an accuracy of 57.81% accuracy rate on our given dataset. Due to the complexity of the tree, it is likely that overfitting is occurring.



If we use pruning, where we cap the max depth of the tree at 10 and use a minimum sample split of 10, accuracy jumps to 61.8%, an increase of 4%. We used entropy as the criterion for measuring the quality of the decision split. We chose entropy over Gini due to the fact that while logarithmic functions are computationally intensive, our dataset is small enough and entropy would provide an accuracy with greater detail as it uses the logarithmic function.

In order for us to see what the algorithm is primarily making its decisions on, we can observe the feature importance on a plot. Alcohol and sulfates were the two attributes that had the greatest impact on determining the quality of the wine.



Boosted Decision Trees with Pruning

For the purpose of boosting our decision tree, we can experiment with adaptive boosting, or adaboost. adaboost, which is just a modified decision tree algorithm which limits the maximum number of estimators allowed before boosting ends. This is mainly done by using a hyperparameter in the number of estimators. This is done by iteratively accounting for the incorrectly classified training set instances.

Using adaboost on the decision tree without any form of pruning, and using a maximum number of estimators of 200, there was a slight change in the accuracy of the model. It rose to 58.44%. And when we experiment on the number of maximum number of estimators on our unpruned decision tree, there was no substantial difference in the accuracy among different sizes of estimators.

Doing the same process on our pruned decision tree, the model yielded an accuracy jump to 70.00%. And just like before the number of estimators made little to no difference in the accuracy of the model.

The reason for the difference in accuracy jumps between the two decision trees is likely due to the fact that the unpruned decision tree was already complex and over-fitted, and thus adding more complexity had no significant increase in accuracy. While the pruned decision tree was simpler in complexity, using adaboost on top of it gave it the much needed complexity to more accurately classify the data.

Neural Network

Neural Networks is a powerful supervised learning algorithm based on the neurons in the brain. We used scikit's multilayer perceptron for the basis of our analysis. MLP is a class of feedforward artificial neural network with three layers of nodes. The three are input, output and a varying number of hidden layers in between. The reason why there is a varying amount of hidden layers is because it allows us to fine-tune our algorithm.

As the first approach of our neural network, we can try two hidden layers with the number of nodes equal to the average of the number of input nodes and the number of output nodes. In our case, we will use eight nodes for the hidden layers. This led to an accuracy of 62.81% and as presumed the accuracy rises and then hits a ceiling as the number of epochs raises.

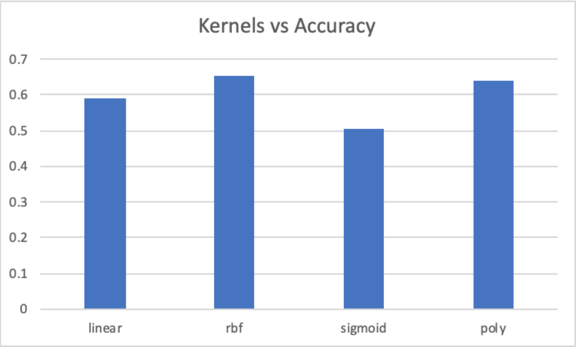
We can experiment with the number of nodes in the hidden layers. And just like the number of epochs, the accuracy rose to 62.81% at 8 nodes in each hidden layer and then it hit a ceiling.

Support Vector Machine

Support Vector Machine categorizes the training data by finding the best fit parameters for a specifically chosen kernel function that will try to find the highest accuracy in classification.

The four types of kernels we tested were liner, rbf, sigmoid and poly. After experimenting with different types of kernels on the scikit's SVC function, we got the following accuracy rates.

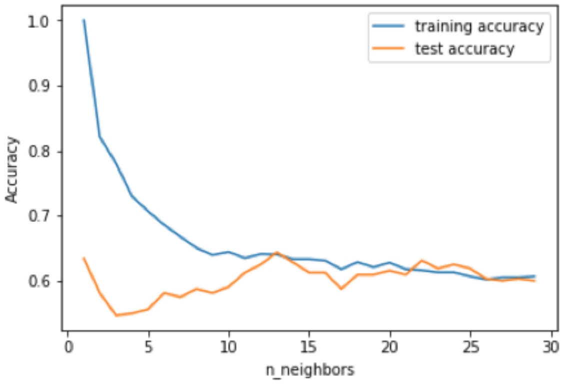
RBF, radial basis function, gave us the highest accuracy, an accuracy of 65.31%. This was evident because of the complexity level of the dataset and due to rbf having flexibility in its boundaries it can fit around data without overfitting. Varying other parameters of the function yielded no change in accuracy.



K-Nearest Neighbors

K- Nearest Neighbors, KNN, is a simpler algorithm compared to the rest. KNN works by making a prediction by comparing an input to the closest k data points to it and assigning the input to the class with the highest count.

We used scikit's KNeighborsClassifier function and experimented with the number of neighbors ranging from 1 to 30.



Varying the number of neighbors led to testing accuracy converging to the rate of 61.56% with k = 13 being the lowest k value where changes in accuracy could not be distinguished any more. Just like before, varying other parameters in the function did not lead to maximizing the accuracy.

Conclusion for Red Wine Quality

For the dataset on red wine quality, these are the highest test accuracies for each of the different classifiers.

Classifier	Accuracy Rate
Decision Tree w/ Pruning	62.80%

Support Vector Machine	65.31%
K-Nearest Neighbor	61.56%
Boosted Decision Tree w/ Pruning	70.00%
Neural Networks	62.81%

Adaptive Boosting overwhelmingly achieved the best results on the dataset at 70.00%. K-Nearest Neighbors performed the worst on this dataset at 61.56%. The reason KNN performed poorly is likely due to the feature set having eleven dimensions, and the higher the dimensions, the more the equidistant the data points become thus impacting performance.

3. Breast Cancer

This dataset was retrieved from Kaggle.com and was collected at the University of Wisconsin. The aim of the dataset is to predict whether the cancer is benign or malignant using a pre-determined set of measurements. The dataset has 569 instances and 30 attributes. The attributes are:

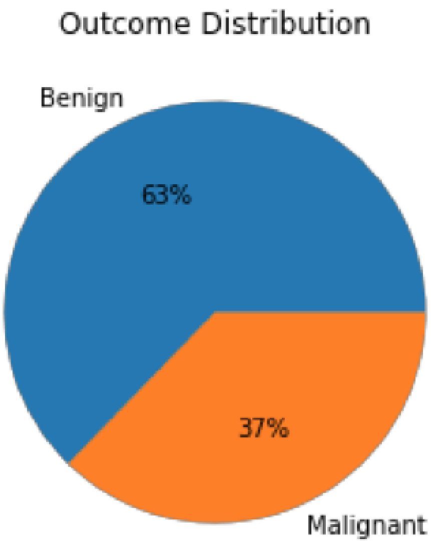
radius_mean	mean of distances from the center to points on the perimeter
texture_mean	the standard deviation of gray-scale values
perimeter_mean	mean size of the core tumor
area_mean	
smoothness_mean	mean of local variation in radius lengths
compactness_mean	mean of perimeter^2 / area - 1.0
concavity_mean	mean of severity of concave portions of the contour
concave points_mean	mean for number of concave portions of the contour
symmetry_mean	
fractal_dimension_mean	mean for "coastline approximation" - 1
radius_se	standard error for

	the mean of distances from center to points on the perimete
texture_se	standard error for standard deviation of gray-scale values
perimeter_se	
area_se	
smoothness_se	standard error for local variation in radius lengths
compactness_se	standard error for perimeter^2 / area - 1.0
concavity_se	standard error for severity of concave portions of the contour
concave points_se	standard error for number of concave portions of the contour
symmetry_se	
fractal_dimension_se	standard error for "coastline approximation" - 1
radius_worst	"worst" or largest mean value for mean of distances from center to points on the perimeter
texture_worst	"worst" or largest mean value for standard deviation of gray-scale values
perimeter_worst	
area_worst	
smoothness_worst	"worst" or largest mean value for local variation in radius lengths
compactness_worst	"worst" or largest mean value for perimeter^2 / area - 1.0
concavity_worst	"worst" or largest mean value for severity of concave portions of the contour
concave points_worst	"worst" or largest mean value for number of concave portions of the contour
symmetry_worst	
fractal_dimension_worst	"worst" or largest mean value for "coastline approximation" - 1

diagnosis	The diagnosis of breast tissues (M = malignant, B = benign)

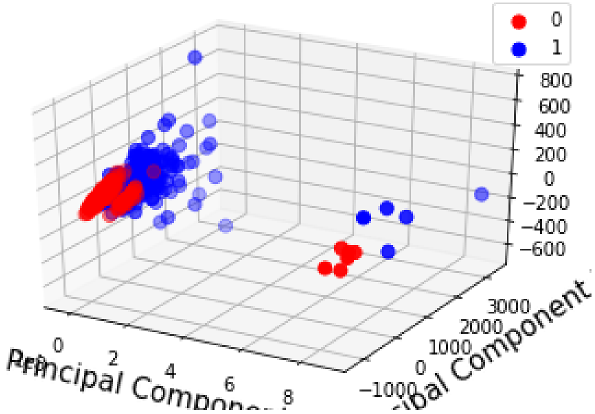
Outcome Distribution

The outcome of the dataset shows the diagnosis of patients. 63% of patients were diagnosed with the cancer being benign, while 37% were diagnsed the cancer is malignant. Due to the unbalanced nature of the dataset, our predictive models can expect an accuracy of at least 63%.



Principal Component Analysis

Just like before, by using PCA, we can reduce the dimensions from thirty down to three so we are able to visualize the data in a 3-D graph. We can clearly see the clustering in the graph and determine that it is a good enough dataset to be classified.

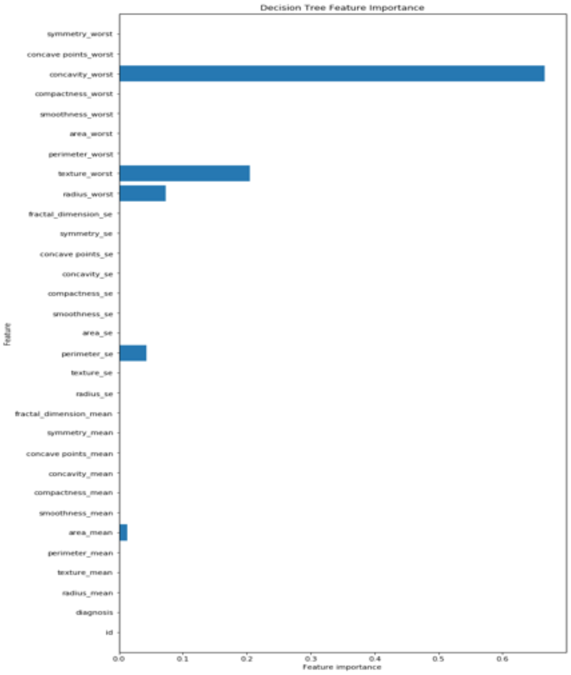


Decision Tree With Pruning

The decision tree without any form of pruning, see below, gives us an accuracy of 92.10% accuracy rate on our given dataset. Considering that our dataset has a single class of 63% the of the dataset, this shows that the model is very predictive. The decision tree for this is:

If we use pruning, where we cap the maximum depth of the tree at 5 accuracy jumps to 94.73%. The reason we decided to impose a maximum depth is due to the tendency for decision trees to get very specific as the depth increases. This leads to overfitting the training data. Although there was no difference in results between Gini and entropy, we used entropy as the criterion for measuring the quality of the decision split.

In order for us to see what the algorithm is primarily making its decisions on, we can observe the feature importance on a plot. Concavity_worst, texture_worst, and radius_worst of the tumors were the three attributes that had the greatest impact on determining the diagnosis of cancer.



Boosted Decision Trees with Pruning

Using adaboost on the decision tree without any form of pruning, and using a maximum number of estimators of 200, there was no change in the accuracy of the model. It remained at 92.20%. And when we experiment on the number of maximum number of estimators on our unpruned decision tree, there was still no difference in the accuracy among different sizes of estimators.

Doing the same process on our pruned decision tree, the model yielded an accuracy jump to 97.37%. Just like the gain of accuracy in the red wine quality dataset, there was again as well. And it is also likely due to the same reason. Also just like before the number of estimators made little to no difference in the accuracy of the model.

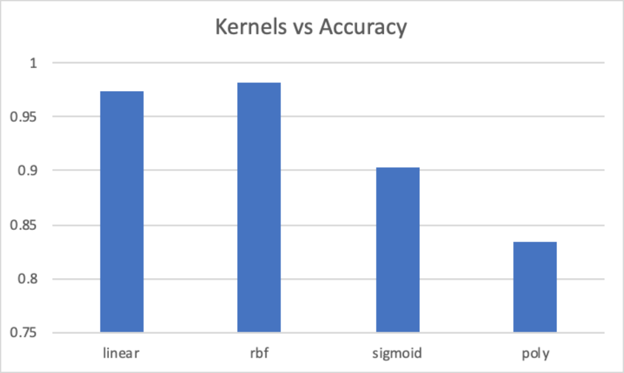
Neural Network

As the first approach of our neural network we can use the same approach for the architecture of the neural network as the red wine quality dataset. We gave two hidden layers nodes equal to the average of number of inputs and number of outputs (sixteen nodes). This led to an accuracy of 98.24%.

We can experiment with the number of epochs and the number of nodes in the hidden layers. And just like the red wine dataset, the accuracy rose to 98.24% at 16 nodes in each hidden layer and then it hit a ceiling.

Support Vector Machine

Once again, we tested 4 kernel types for the SVM models. And once again, rbf performed the best in test accuracy with 98.24%. Just like before, this likely due to rbf being liberal in the shape of its boundaries leading to a better fit for the dataset.



K- Nearest Neighbors

We used scikit's KNeighborsClassifier function and experimented with the number of neighbors ranging from 1 to 90.

The value of K where accuracy was maximized was k = 8. The accuracy rate was 98.24% . For k > 20, accuracy hovers around 96.5% and 97.5%. For computing the nearest neighbors, algorithims such as ball_tree, kd_tree, brute and auto were used, yet they all had an accuracy of 98.24%.

Conclusion for Breast Cancer Diagnosis

For the dataset on breast cancer diagnosis, these are the highest test accuracies for each of the different classifiers.

Classifier	Accuracy Rate
Decision Tree w/ Pruning	94.73%
Support Vector Machine	98.24%

K-Nearest Neighbor	98.24%
Boosted Decision Tree w/ Pruning	97.37%
Neural Networks	98.24%

This time there was a three-way tie in highest accuracy. K-nearest neighbors, support vector machine and neural network all performed with a 98.24% test accuracy. The lowest performing model was decision tree with pruning. KNN performed well despite higher dimensionality. All the models had results with very high testing accuracies and all were very predictive.

4. Conclusion

In brief, all models for both datasets had high test accuracies. The model that tested the highest, if not near the highest for both datasets was Adaptive Boosting. If the models had to be reworked, improving on the details of inclusion of certain hyperparameters could be looked at again.

5. References

Red Wine Quality Dataset: https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009

Breast Cancer Dataset: https://www.kaggle.com/uciml/breast-cancer-wisconsin-data