HW2 Technical Report

Bryce Grover

Introduction

The data set is a collection of data of seven attributes of apples. These attributes include size, weight, sweetness, crunchiness, juiciness, ripeness, acidity, and quality. All features but quality are continuous data, and quality is a cetegorical variable with "good" and "bad" as the two posssible values. These continuous data were used to predict the quality of apples.

Two models were developed to predict apple quality - a feed forward neural network model and a decision tree model. After the neural network was created and fitted, the decision tree model was created and fitted on the same data to predict the same outcome (quality). Performance metrics were calculated for each and performances were compared.

Analysis

Cleaning and Pre-processing

In preparing the apple data for analysis, a few important steps were taken to ensure the information was ready to use. First, any missing or incomplete data was removed to keep the analysis accurate. Next, the features of the data set were z-scored so the values of every feature were on the same scale. Finally, the values of the predicted varible 'Quality' were convereted to be ones and zeros since the models require numeric input. This preparation helped make the models perform better because they received clear, regularized data to learn from.

Exploratory Analysis

Before building the models to predict apple quality using apple features, summary statistics were generated alongside several visualizations including histograms, box pots, and a correlation matrix to find any surface level pattern in the data.

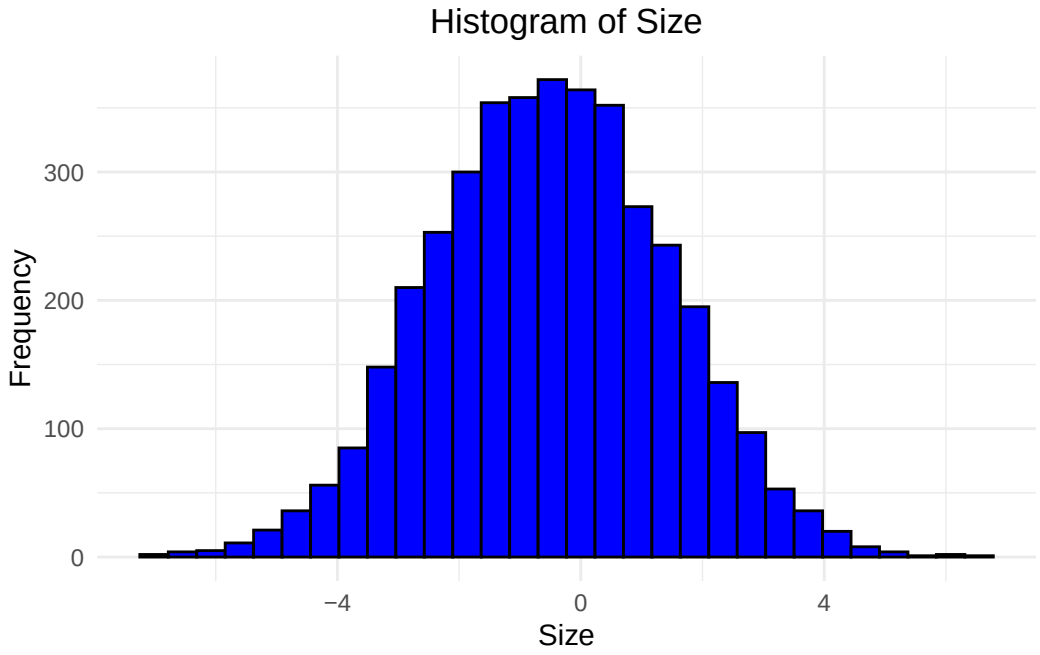
Below are the summary statistics.

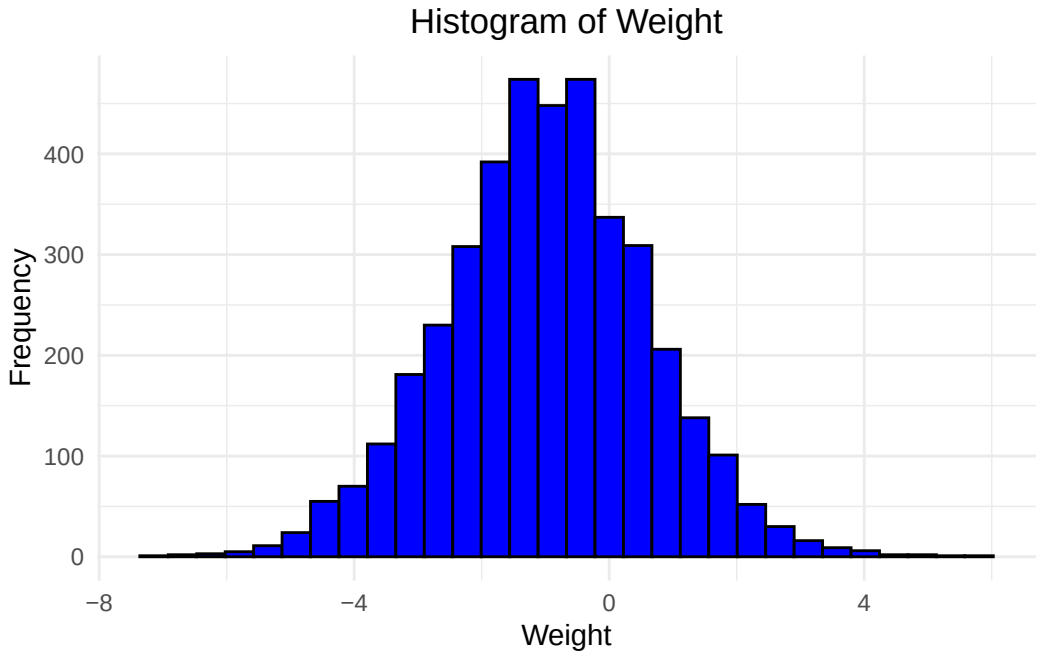
```
     A_id              Size              Weight            Sweetness
 Min.   :   0.0   Min.   :-7.1517   Min.   :-7.14985   Min.   :-6.8945
 1st Qu.: 999.8   1st Qu.:-1.8168   1st Qu.:-2.01177   1st Qu.:-1.7384
 Median :1999.5   Median :-0.5137   Median :-0.98474   Median :-0.5048
 Mean   :1999.5   Mean   :-0.5030   Mean   :-0.98955   Mean   :-0.4705
 3rd Qu.:2999.2   3rd Qu.: 0.8055   3rd Qu.: 0.03098   3rd Qu.: 0.8019
 Max.   :3999.0   Max.   : 6.4064   Max.   : 5.79071   Max.   : 6.3749
  Crunchiness         Juiciness          Ripeness          Acidity
 Min.   :-6.05506   Min.   :-5.9619   Min.   :-5.8646   Min.   :-7.01054
 1st Qu.: 0.06276   1st Qu.:-0.8013   1st Qu.:-0.7717   1st Qu.:-1.37742
 Median : 0.99825   Median : 0.5342   Median : 0.5034   Median : 0.02261
 Mean   : 0.98548   Mean   : 0.5121   Mean   : 0.4983   Mean   : 0.07688
 3rd Qu.: 1.89423   3rd Qu.: 1.8360   3rd Qu.: 1.7662   3rd Qu.: 1.51049
 Max.   : 7.61985   Max.   : 7.3644   Max.   : 7.2378   Max.   : 7.40474
   Quality
 Length:4000
 Class :character
 Mode  :character
```

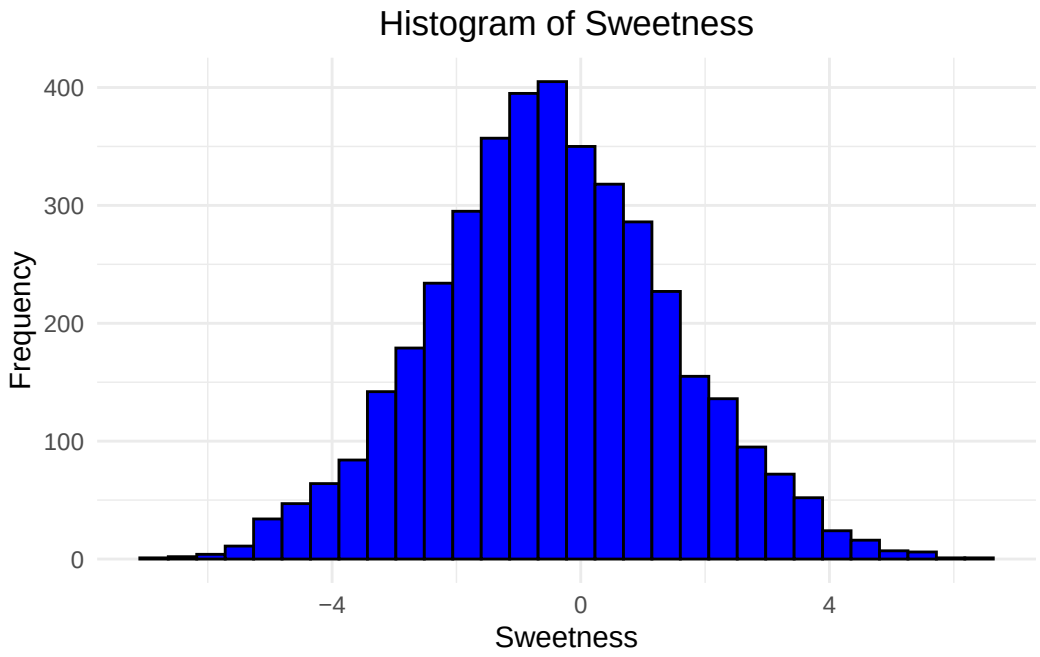
Visualizations

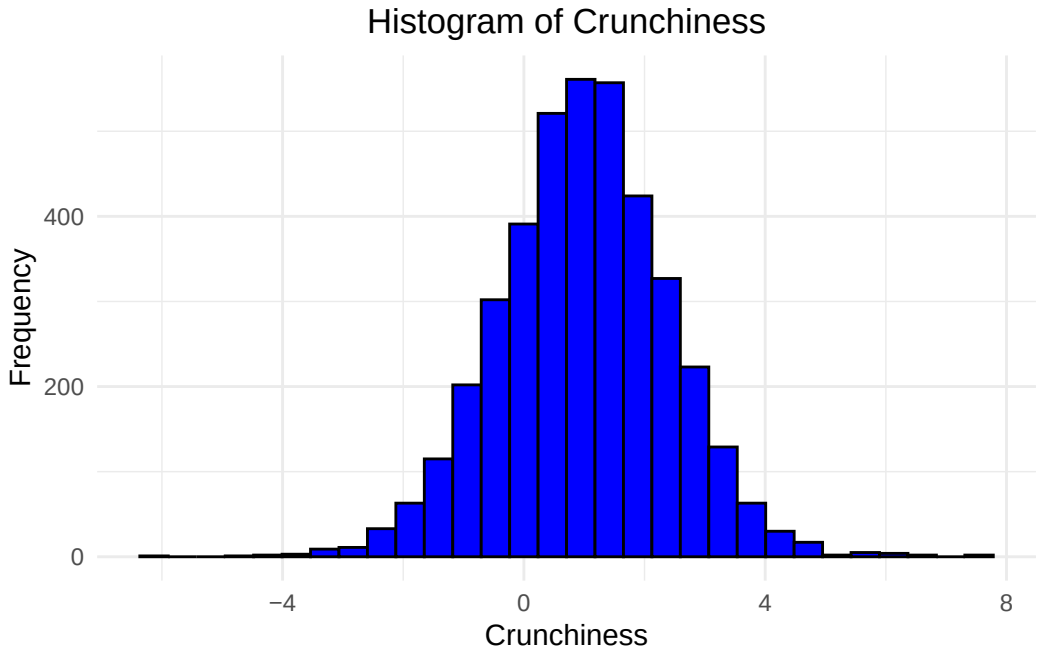
Histograms

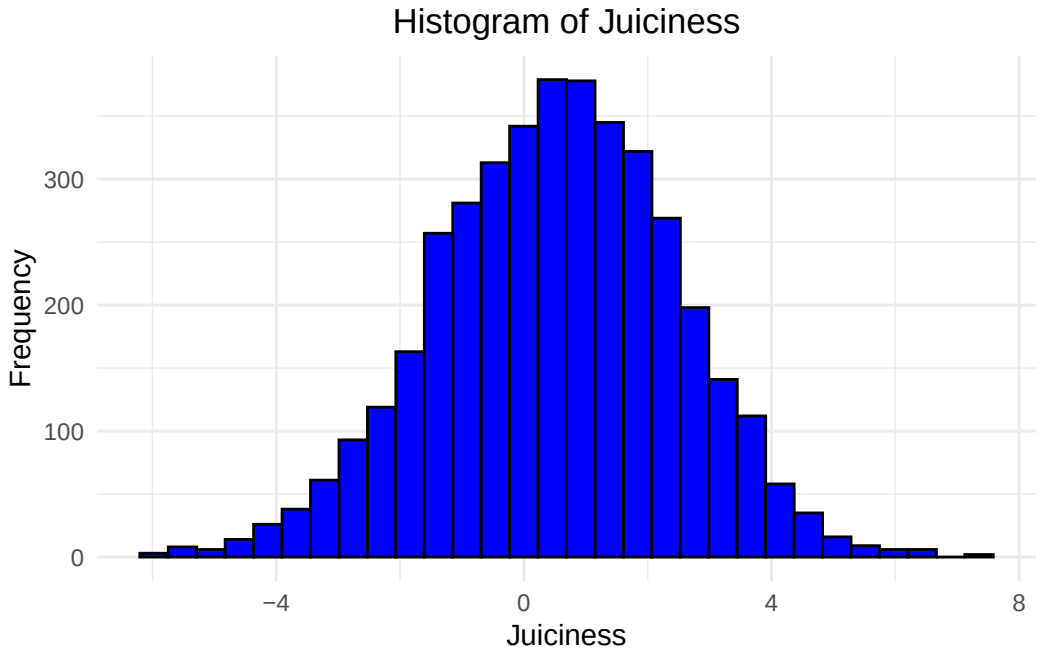
A histogram of every feature and its frequency was generated. By looking at the plots, one can see that the features are all unimodal normal distributions.

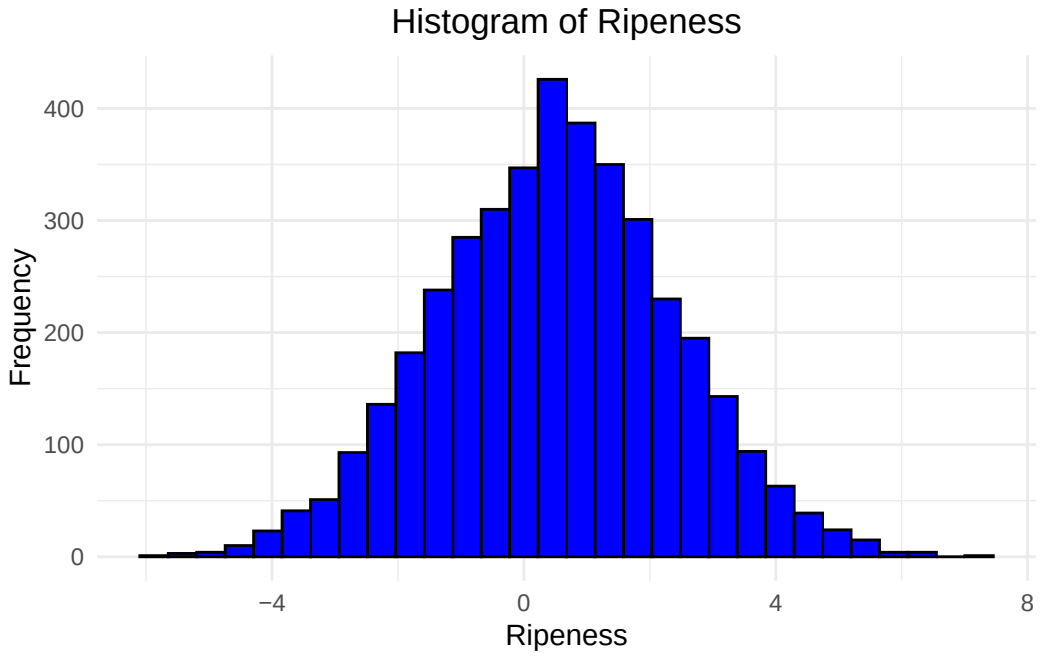

Histogram of Size
Frequency
300
200
100
0
−4
0
4
Size

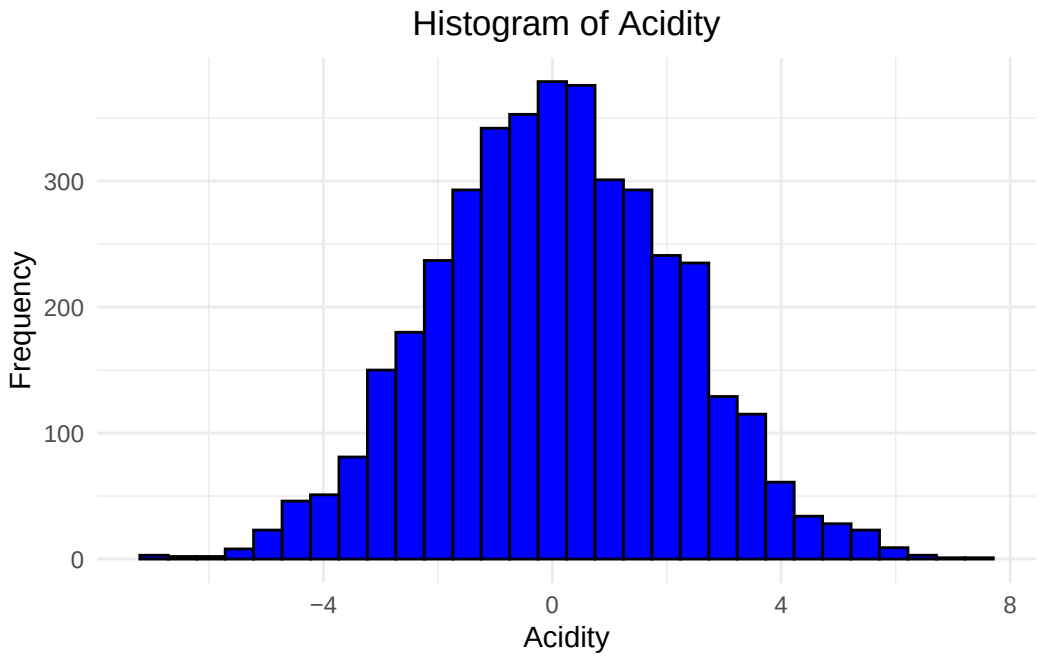

Histogram of Weight
Frequency
400
300
200
100
0
−8
−4
0
4
Weight


Histogram of Sweetness
Frequency
400
300
200
100
0
−4
0
4
Sweetness


Histogram of Crunchiness
Frequency
400
200
0
−4
0
4
8
Crunchiness

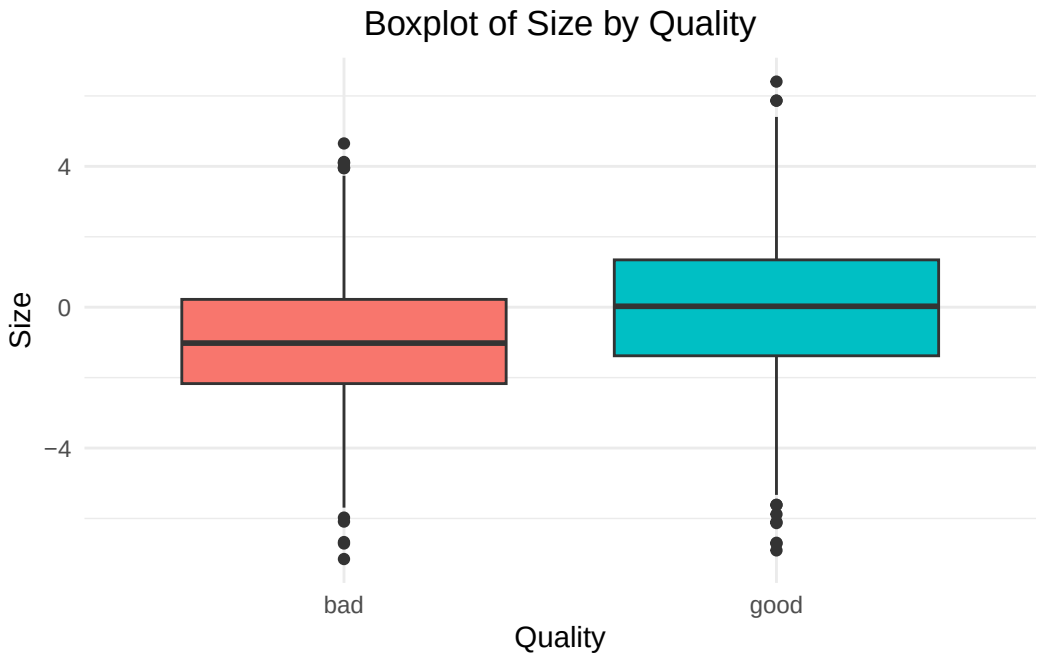

Histogram of Juiciness
Frequency
300
200
100
0
−4
0
4
8
Juiciness

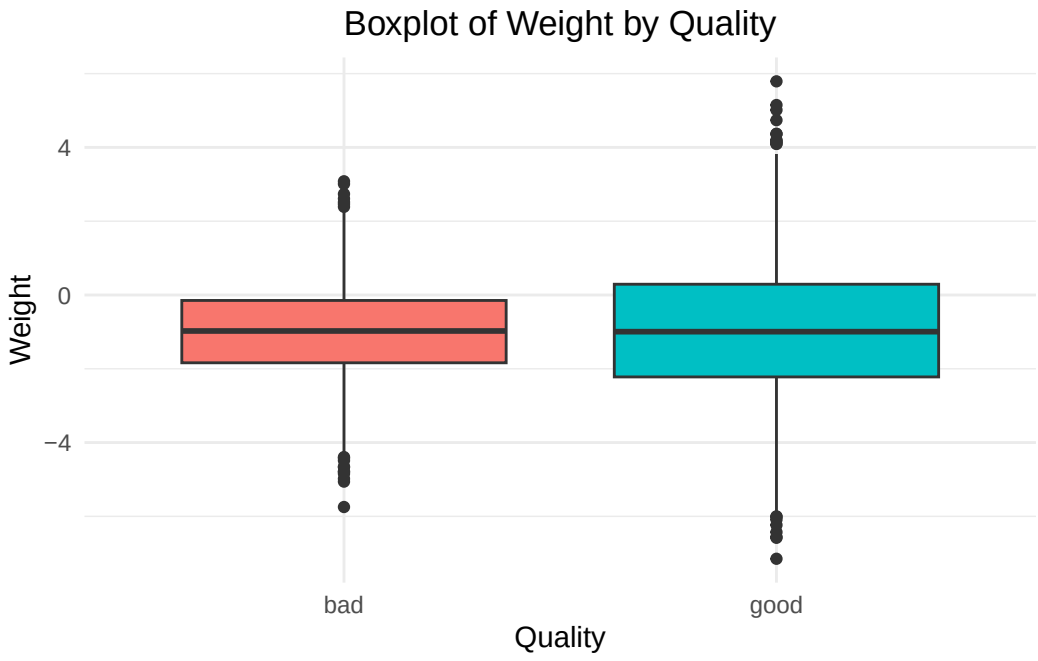

Histogram of Ripeness
Frequency
400
300
200
100
0
−4
0
4
8
Ripeness

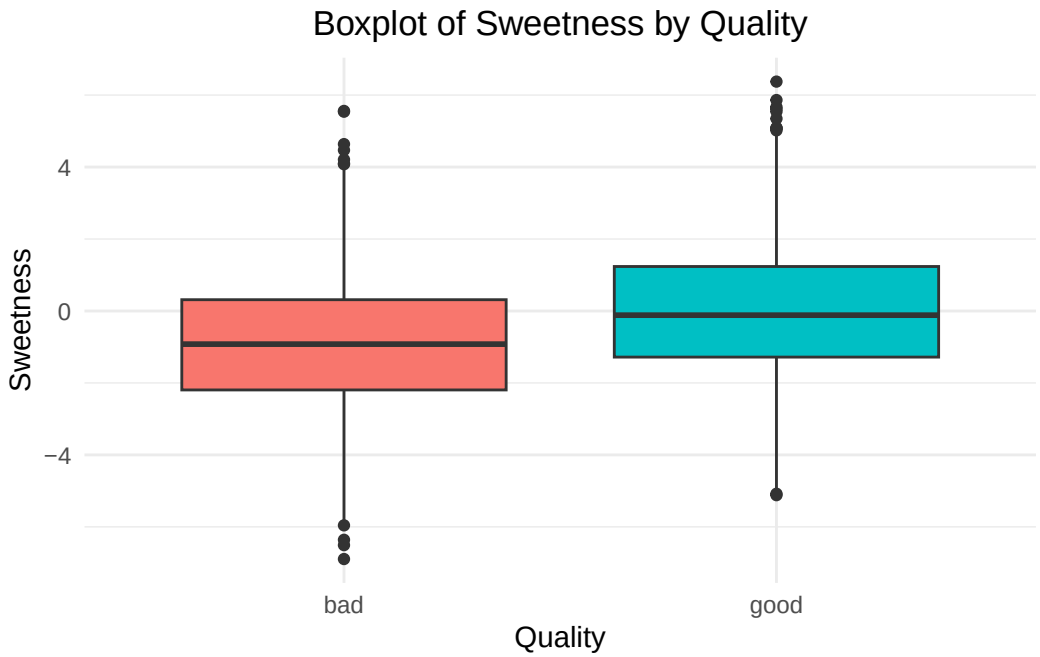


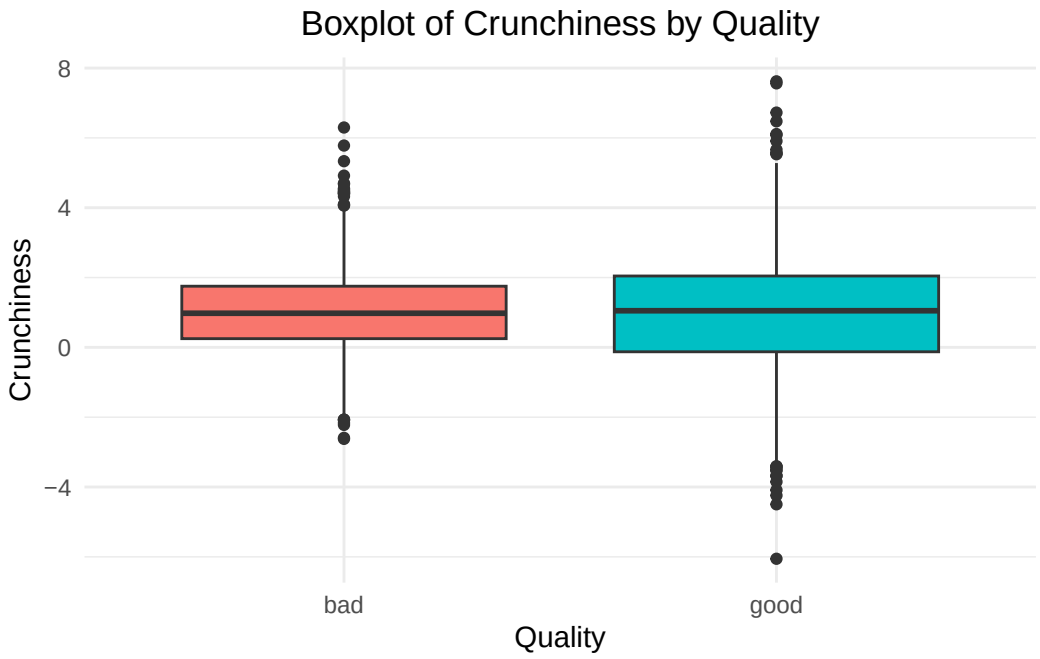
Box Plots

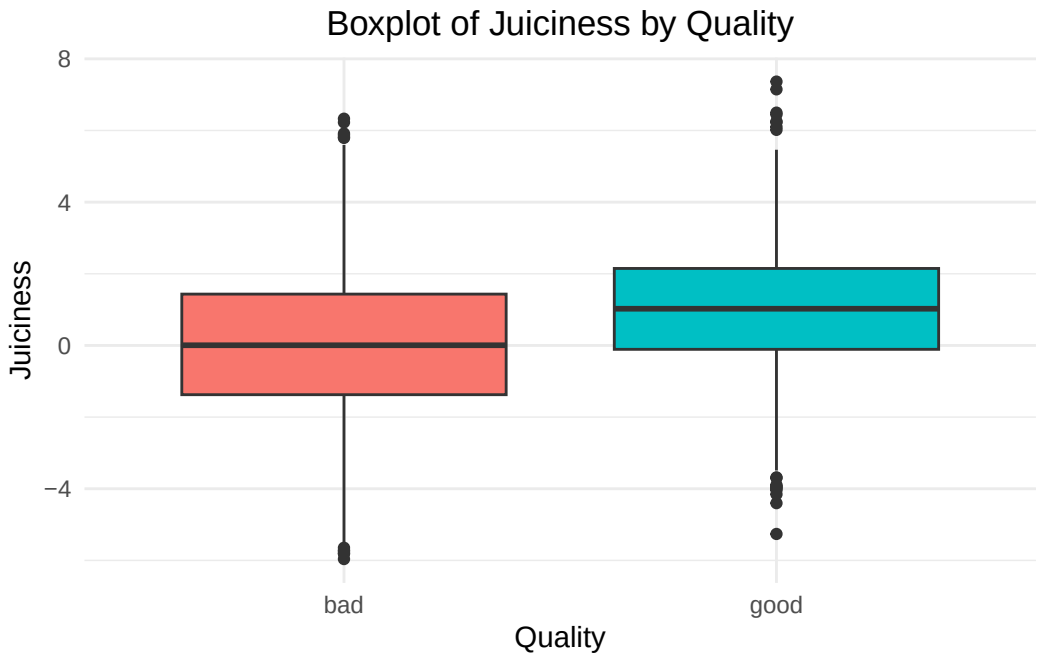
In addition to histograms, box plots where generated for each feature by our predicted value, 'Quality'. Generally, the different features did not vary when comparing 'good' quality and 'bad' quality apples. This suggests it is difficult to predict the quality of an apple by one feature alone and that if a relationship exists between apple features and apple quality, it is one that is subtle, complex, and/or spans across all features.

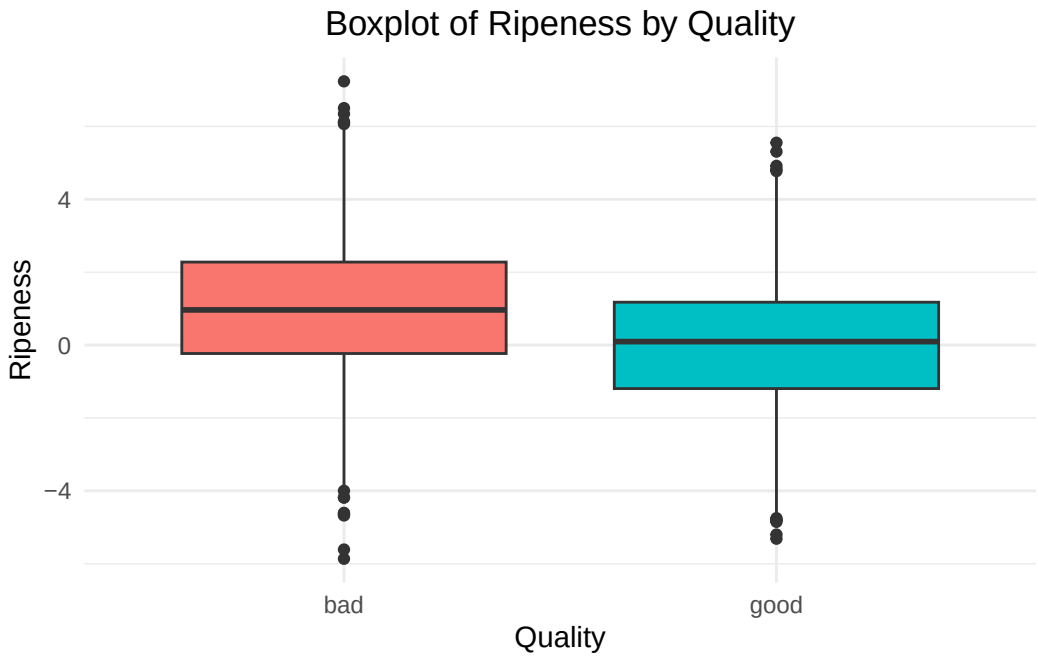

Boxplot of Size by Quality
Size
4
0
−4
bad
good
Quality

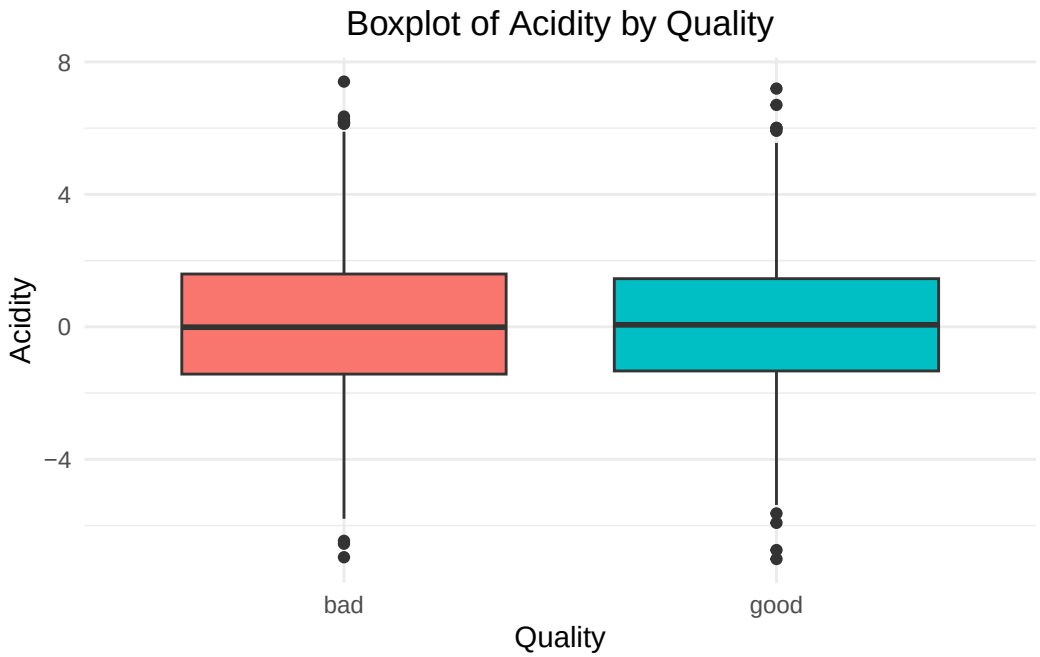

Boxplot of Weight by Quality
Weight
4
0
−4
bad
good
Quality


Boxplot of Sweetness by Quality
Sweetness
4
0
−4
bad
good
Quality


Boxplot of Crunchiness by Quality
Crunchiness
8
4
0
−4
bad
good
Quality

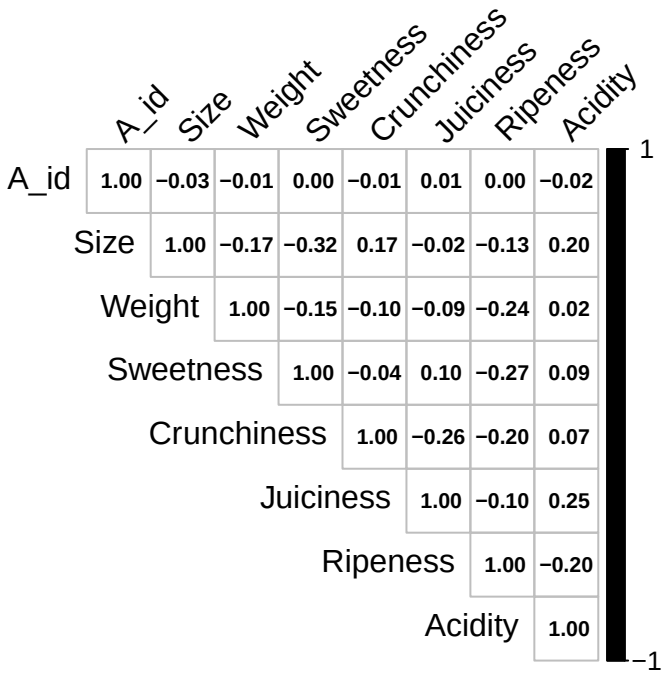

Boxplot of Juiciness by Quality
8
4
0
−4
Juiciness
bad
good
Quality


Boxplot of Ripeness by Quality
4
0
−4
Ripeness
bad
good
Quality



Correlation Matrix

The existence of a complex relationship between apple features and apple quality is further evidenced by a lack of an obvious correlation between individual features and apple quality.



Methods

Building a Feed Forward Neural Network

Given the complexity of the relationship between apple features and apple quality, which likely involves intricate patterns spanning multiple features, constructing a neural network to predict apple quality requires a somewhat deep architecture. The deep architecture is necessary for detecting and learning the patterns and interconnections present in the data.

The neural network architecture is made up of an input layer that taken in the seven features, followed by a series of dense layers. The first dense layer contains 50 nodes, which are likely to capture a broad representation of the input features. This is followed by a batch normalization layer, which stabilizes the learning process by normalizing the layer inputs. The following layers include a 75-node dense layer with ReLU activation to introduce non-linearity, allowing for the modeling of complex relationships. An important aspect of the model is the increase in node count in the middle layers, from 50 to 75 nodes. This increase allows a deeper, more complex representation of the data, improving the model's ability to decipher patterns that predict apple quality. To reduce overfitting since since the model is finding intricate patterns in the training data, a dropout layer with a rate of 0.3 is implemented which randomly omits a fraction of the nodes during training so no few nodes become too weighty in the model when it is given unseen data and attempts to make a prediction.

The architecture concludes with a dense layer for output, using a softmax activation function to categorize the predicted quality. The model was compiled with an Adam optimizer and categorical crossentropy as the loss function, because the prediction the model is making is whether an apple is 'good' or 'bad' - a categorical output.

The neural network was trained on an 80/20 split for training and validation, with standard scaling applied to the feature inputs to normalize the data distribution. Early stopping was also utilized to halt training if the loss function did not improve for three consecutive training epochs.

Building a Decision Tree

A decision tree model was developed to classify apple quality as 'good' or 'bad', utilizing the same features as those in the neural network model. To mitigate overfitting and ensure a well-regulated model, a grid search was employed to identify the optimal hyperparameters. The values obtained through cross-validation were used to make a decision tree with a maximum depth of 10, a minimum of 10 samples per leaf, and a minimum of 2 samples required to split a node. This decision tree model, similarly to the neural network, was created to accurately classify apple quality, while maintaining effectiveness when using unseen data.

The decision tree model was trained on the same 80/20 split as the neural network to ensure consistency when comparing the models' performance.

Results

The evaluation of the neural network and decision tree models focused on three metrics: accuracy, precision, and recall.

Accuracy reflects the overall correctness of the models. The neural network had an accuracy of 0.86, meaning that 86% of its apple quality classifications were correct. The decision tree model showed a slightly lower accuracy of 0.81.

Precision measures the models' ability to identify 'good' apples. The decision tree model achieved a precision of 0.84, suggesting just as good at predicting 'good' apples than the neural network which had a precision of 0.84 as well.

Recall measures how well the models can identify all actual 'good' apples. It is the true positive rate. The neural network had a recall of 0.89, meaning it could correctly identify 89% of the 'good' apples in the data set. The decision tree's recall was 0.76, meaning it missed more of the 'good' apples compared to the neural network.

These results indicate that while the neural network generally performs better in terms of overall accuracy and recall, the decision tree model was better suited in precision.

Table 1: Performance Comparison of Neural Network and Decision Tree Models

Model	Accuracy	Precision	Recall
Neural Network	0.86	0.84	0.89
Decision Tree	0.81	0.84	0.76

Utilization of Deep Learning

When looking at the use of deep learning for classifying apples as 'good' or 'bad', the comparison between the neural network and decision tree models shows that despite the neural network's advanced capabilities and slightly superior performance metrics, the improvement does not always justify the increase in computational cost and complexity. Particularly for tasks with relatively straightforward classification objectives, such as the task performed this assignment, the efficiency and simplicity of a decision tree model present a comparable alternative. Even when the exploratory analysis revealed that if a relationhip existed between apple features and apple quality it was intricate and complex, the neural netowork was only slighlty better at discerning that relationship. A cost-benefit analysis, in this context, might suggest a simpler model be used. The point being, while deep learning offers great utility and capability, its use is not needed in all circumstances. The decision to use a more complicated and advanced model should be made with the marginal gains in performance in mind, especially when simpler models can achieve results that are almost as good with less resource expenditure.

Reflection

After having read the assignment instructions, I was genuinely unsure as to whether the lesson of the assignment was that Neural Nets always perform better because they utilize Deep Learning or if the lesson was the inverse where Deep Learning model are not always the shining-star choice. In doing this assignment I learned that the latter is true. Deep Learning, while very capable, is not always the *only* capable method.