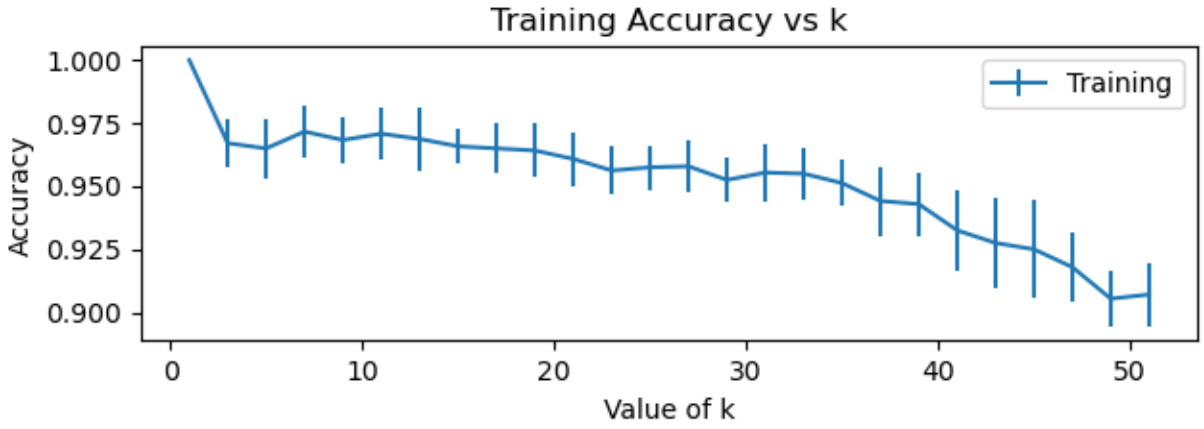
CS589 HW1

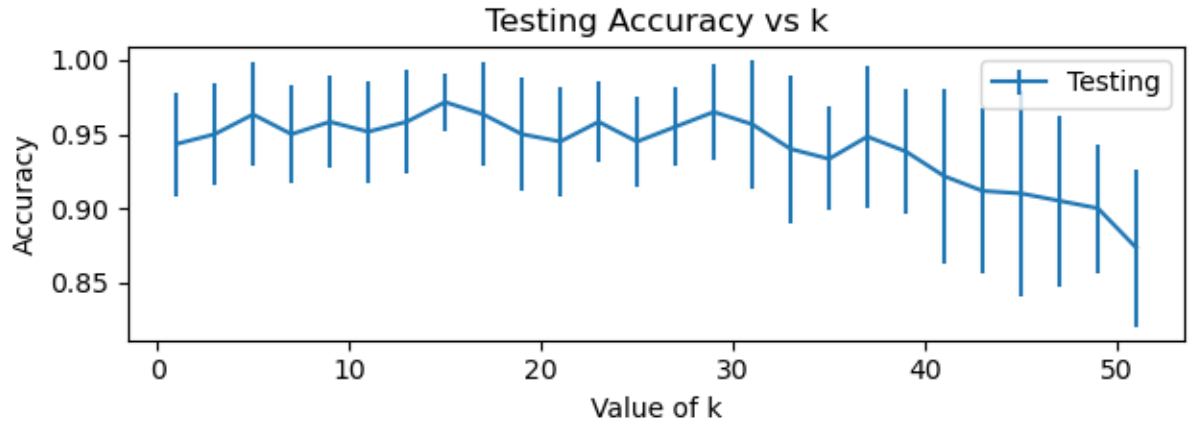
Aryan Tipnis

March 2, 2024

Part 1: KNN Algorithm

Q1.1 and Q1.2:





Q1.3:

On the training set, we can see that as the value of K increases from 1 to 51, the accuracy of predictions over the training set decreases.

- When k=1, the model only considers its nearest neighbor, since the model was trained on the same values of training data, it leads to a very high accuracy of 100%.
- As k increases, the model considers more neighbors which leads to a improvement in the generalization of the model and it is able to capture underlying patterns due to which it results in a high accuracy of 95% to 97.5%.
- For very high values of k, the decision boundaries become less distinct due to which the model might miss the important patterns in the data which decreases accuracy.
- The standard deviation for training data is less than testing data, since the model has been trained on the same dataset leading to lower variability in the accuracy.

On the testing set, we can see that as the value of K increases from 1 to 51, the accuracy of predictions over the training set increases at first, then decreases with some peaks in between.

- When k=1, the model only considers its nearest neighbor which can lead to poor generalization of data and overfitting and thus result in lower accuracies on unseen data than on training data.
- As k increases, we can see that the accuracy increases at first due to consideration of more neighbors and improvement in the generalization of the model since it is correctly able to capture the underlying patterns.
- The peaks in the graph could be due to the fact that some values of k are better than the other values in capturing the patterns in the dataset.
- For very high values of k, like in the training data, the decision boundaries become less distinct due to which the model might miss the important patterns in the data which decreases accuracy.
- The standard deviation on testing data varies more than training data since accuracy varies more when predicting unseen data.

Q1.4:

The k-NN algorithm seems to underfitting for the values of K = 41 to 51. This can be due to the fact that for extremely large k values, the decision boundaries become much smoother, due to which the k nearest neighbors might not be able to give a correct estimate and thus

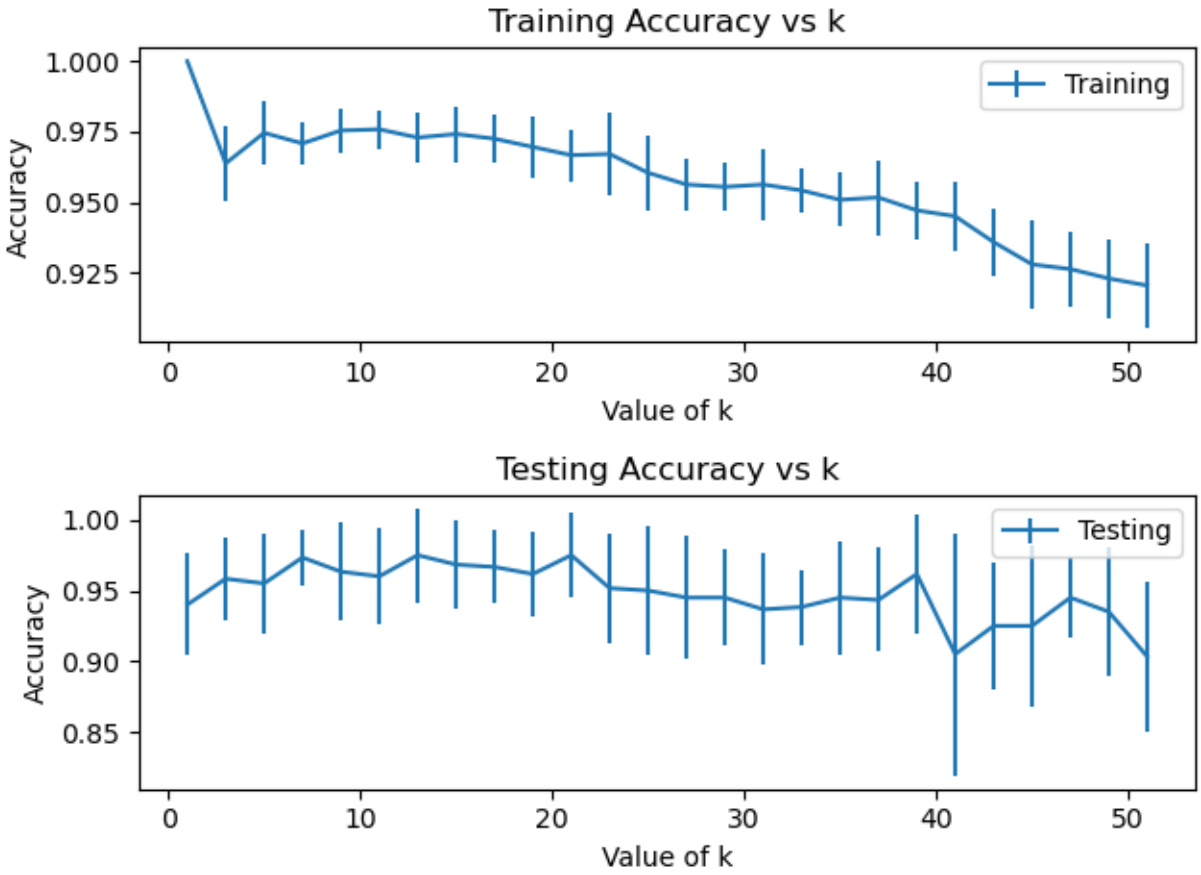
miss important patterns in the data which reduces the accuracy for both training and testing set.

The k-NN algorithm seems to be overfitting when K= 1 to 5. This is because for smaller k values, very less nearest neighbors are taken into account. In the training set, the model nearly memorises the dataset for these values, since it has been trained on the same set, which can lead to very high accuracy. However, in the testing set, these k values lead to poor generalization of data since the model is very sensitive to outliers which can lead to lower accuracies as compared to training data.

Q1.5:

I would choose k = 15 to fine-tune this algorithm so that it worked in real life. This is because when k = 15, the testing set has an accuracy of 97% which is the highest out of all k values and the training set also has an accuracy of 96.5%. This k value minimizes underfitting since it performs well on both training and testing set. Additionally, it also minimizes overfitting since it does not memorize the training dataset but rather finds a good generalization pattern on unseen data. The standard deviation of accuracies when k = 15 is also low which proves the the accuracies are stable.

Q1.6:



(a)

(b) K = 13 results in k-NN performing the best on the testing set.

(c) In this particular dataset, there is not much difference in the graph of the testing dataset with and without normalizing the data.

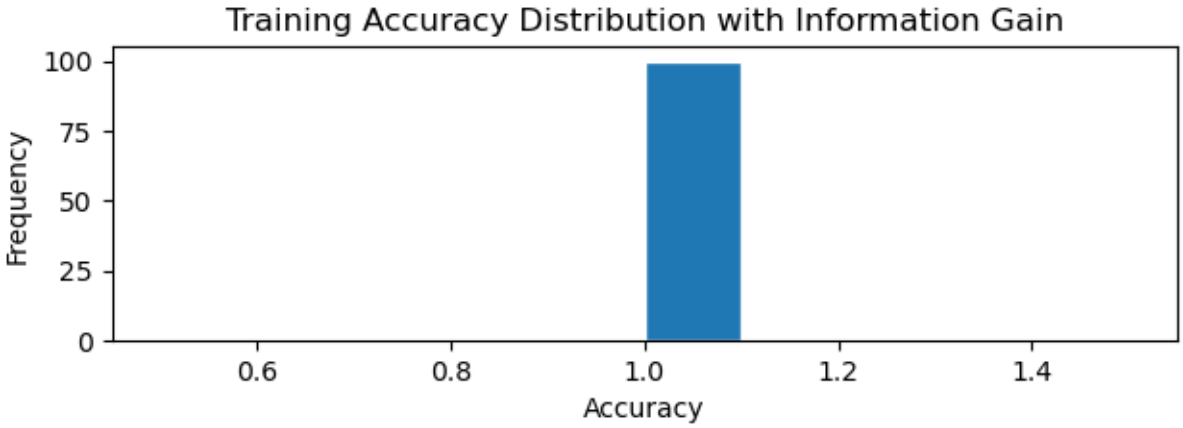
Even the mean accuracies of both training set and testing set in both cases are similar. This can be due to some characteristics that may occur in this particular dataset like: similar magnitude differences may occur between data points for all columns or the dataset itself can be simplistic and easily classifiable. Another reason could be since the attributes in this data are already uniformly scaled due to which normalization might not have a significant impact on the performance of the k-NN algorithm.

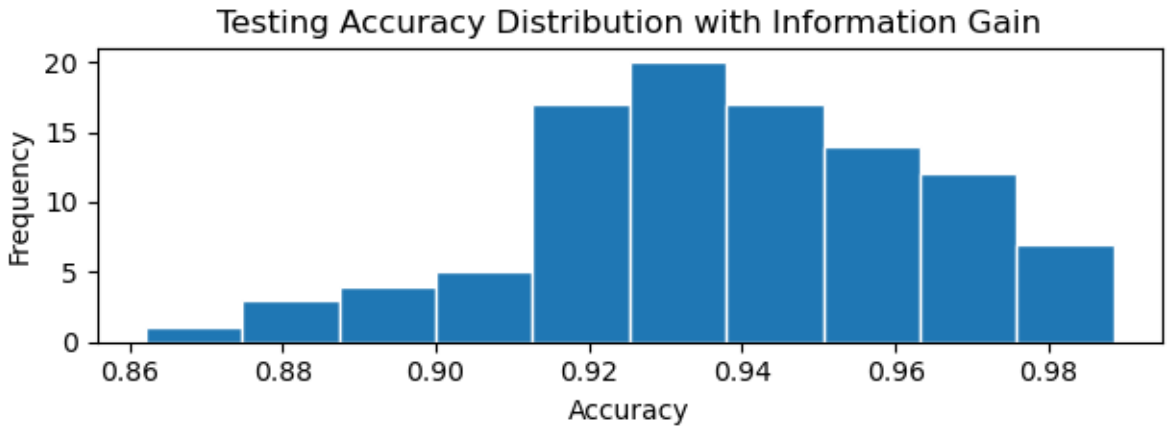
However, in the testing graph without normalization, there are less peaks which can be due to the fact that features with larger scales tend to dominate the distance calculations and overshadow the contributions of other features.

Part 2: Decision Tree Algorithm

Q2.1 and Q2.2:

On training set, the mean accuracy is 1.0 and standard deviation is 0.0 while on testing set, the mean accuracy is 0.93 and standard deviation is 0.022.





Q2.3:

The histogram of the training dataset has an accuracy of 100% since the model perfectly memorizes the dataset rather than generalizing it, leading to overfitting. This can also be due to the fact that the classes in the dataset are perfectly separable due to which the model can easily create splits in data.

The histogram of the testing dataset has a pretty high mean accuracy of 93% because the model is performing resonably well in generalizing data and capturing underlying patterns. However, randomness in real life can lead to outliers which reduce the accuracy.

The variance of testing dataset is more than the training dataset while the the mean accuracy of training dataset is more than the testing dataset. This is because while training, the

model memorizes the data perfectly and thus giving an accuracy of 100% every time, while on unseen testing data, the model generalizes the data and finds patterns which leads to different values of accuracy which leads to higher variance.

Q2.4:

Based on the graphs, I feel that algorithm overall is performing well with little bit of overfitting. The perfect accuracy achieved on the training set suggests that the model might be overfitting to the training data, as it memorizes the data rather than generalizing patterns. The high mean accuracy of 93% on the testing set indicates that the model generalizes reasonably well to unseen data. Overall, the model is likely performing reasonably well but might be slightly overfitting to the training data.

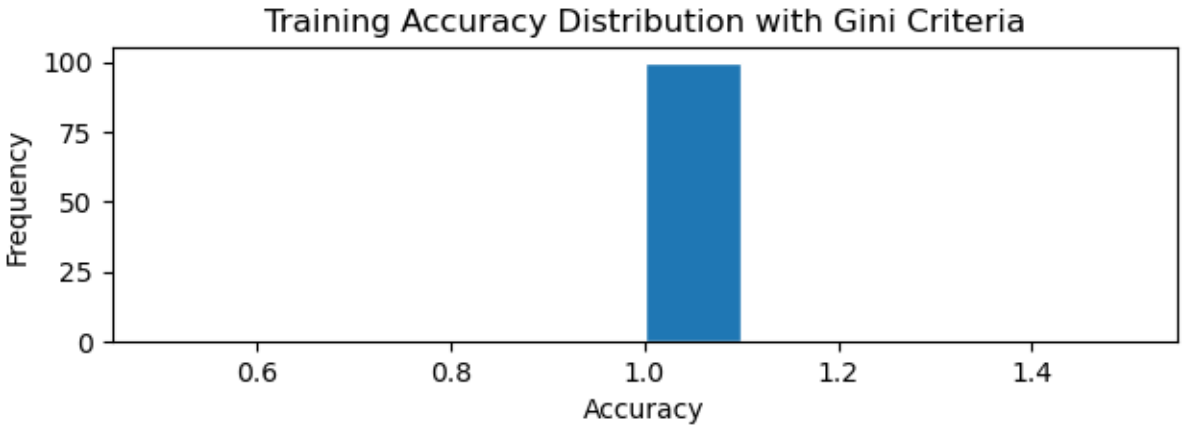
Q2.5:

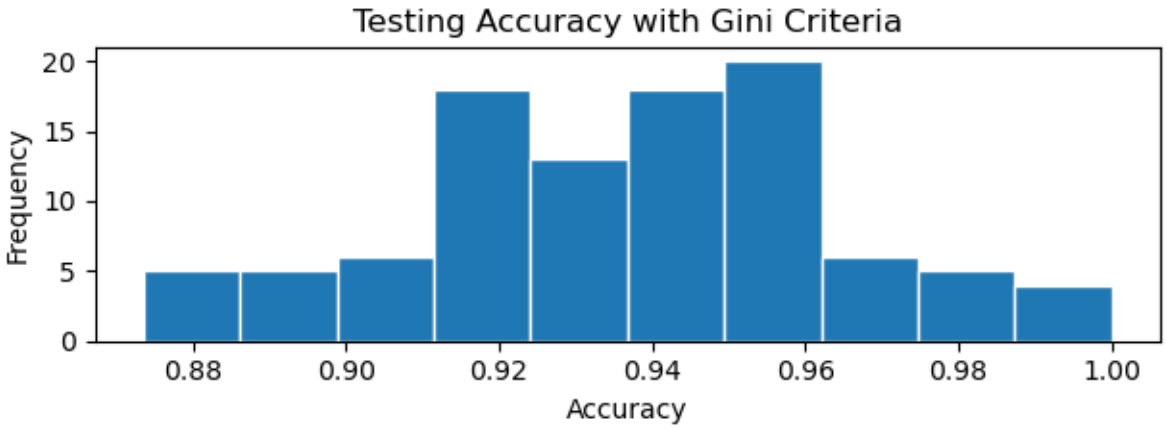
Yes, it's possible to experimentally confirm the non-robustness of Decision Trees by analyzing the histograms and their corresponding average accuracies and standard deviations. By observing the standard deviation of the accuracy values in the testing set, we can determine the degree of variability in the model's performance. A higher standard deviation indicates greater variability, which is a characteristic of non-robust models. Additionally, on comparing the mean accuracy between the training and testing sets, if the training set accuracy is significantly higher than the testing set accuracy, it suggests overfitting, which is a form of non-robustness.

Part 3: Extra Credit 1

Q3.1 and Q3.2:

On training set, the mean accuracy is 1.0 and standard deviation is 0.0 while on testing set, the mean accuracy is 0.94 and standard deviation is 0.024.





Q3.3

The histogram of the training dataset has an accuracy of 100% since the model perfectly memorizes the dataset rather than generalizing it, leading to overfitting. This can also be due to the fact that the classes in the dataset are perfectly separable due to which the model can easily create splits in data.

The histogram of the testing dataset has a high mean accuracy 94% which can be because the model is performing resonably well in generalizing the data. However, randomness in the real life can lead to outliers which reduce the accuracy.

This is very similar to the histograms of information gain criterion.

The variance of testing dataset is more than the training dataset while the the mean accuracy of training dataset is more than the testing dataset. This is because while training, the model memorizes the data perfectly and thus giving an accuracy of 100% every time, while on unseen testing data, the model generalizes the data and finds patterns which leads to different values of accuracy which leads to higher variance.

Q3.4:

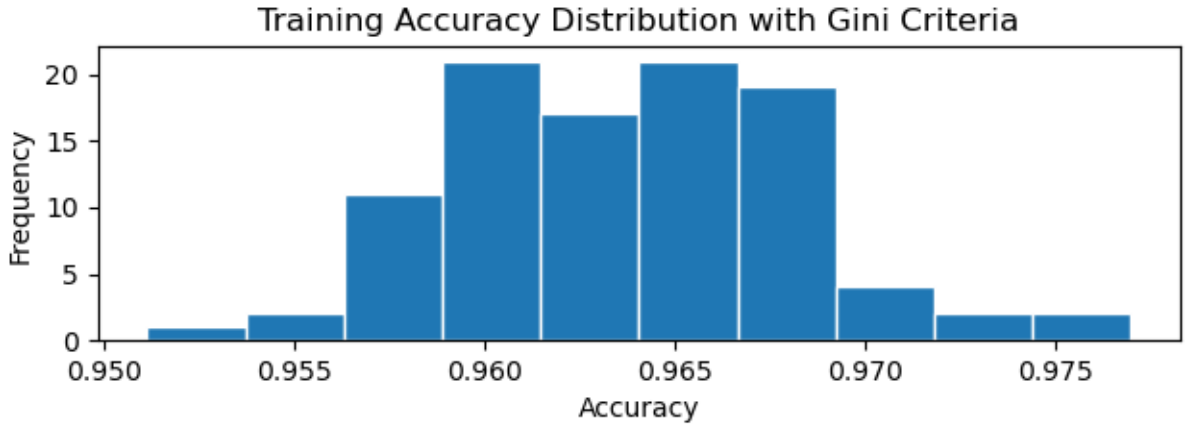
Based on the graphs, I feel that algorithm overall is performing well with little bit of overfitting. The perfect accuracy achieved on the training set suggests that the model might be overfitting to the training data, as it memorizes the data rather than generalizing patterns. The high mean accuracy of 0.94 on the testing set indicates that the model generalizes reasonably well to unseen data. Overall, the model is likely performing reasonably well but might be slightly overfitting to the training data.

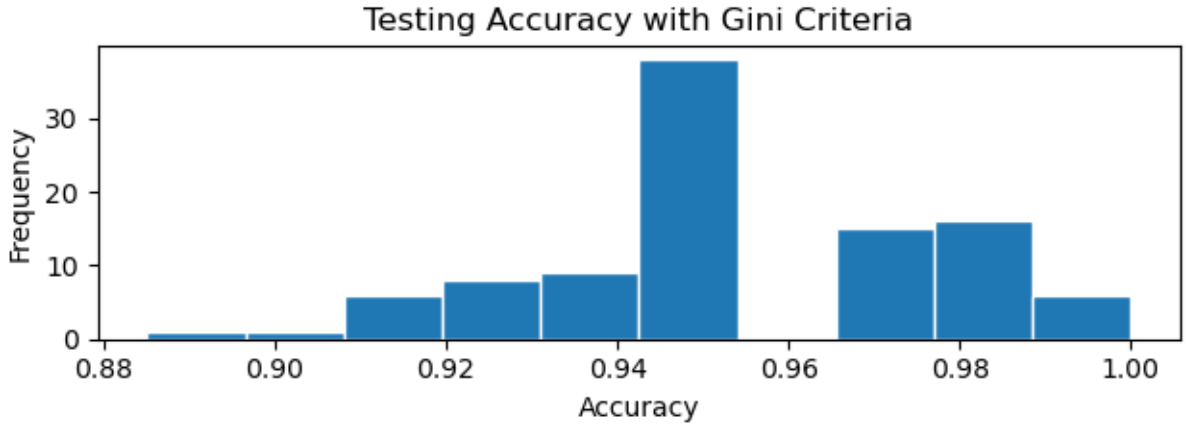
Part 4: Extra Credit 2

We use Gini Criterion while performing this.

Q4.1 and Q4.2:

On training set, the mean accuracy is 0.962 and standard deviation is 0.0047 while on testing set, the mean accuracy is 0.955 and standard deviation is 0.021.





Q4.3:

The histogram of the training dataset has an accuracy of 96% and testing dataset has an accuracy of 95% which shows that the model has a strong performance on the both datasets data. This is because after we execute a max depth of 85% as another stopping criteria, it prevents the model from making overly complex decision boundaries and thus generalize data and not memorize it. This model is also less prone to overfitting and captures the underlying patterns more effectively.

Q4.4:

Based on the graphs, I feel that algorithm overall is performing well. The high mean accuracies on both the training and testing sets indicate that the model is performing well and effectively captures the underlying patterns in the data. The relatively low standard deviations suggest that the model's performance is consistent across random subsets of the data, indicating robustness and good generalization ability.