Faculty of Engineering and Technology
Electrical and Computer Engineering Department

Artificial Intelligence
ENCS3340

Project #2
Machine Learning for Classification

Prepared by:
Omar Hussain 1212739
Malak Moqbel 1210608

Instructor: Ismail Khater

Section: 2
Date: 20/6/2024

Table of Contents

Introduction

Objective: The objective of this project is to evaluate and compare various machine learning algorithms for a classification task using WEKA. The chosen models for this analysis are Decision Tree (J48), Naïve Bayes, and Multilayer Perceptron (MLP).

1. Dataset Description

My student ID ends in 8. Since 8 mod 3 equals 2, we'll use the Raisin Dataset.

Dataset: Raisin Dataset

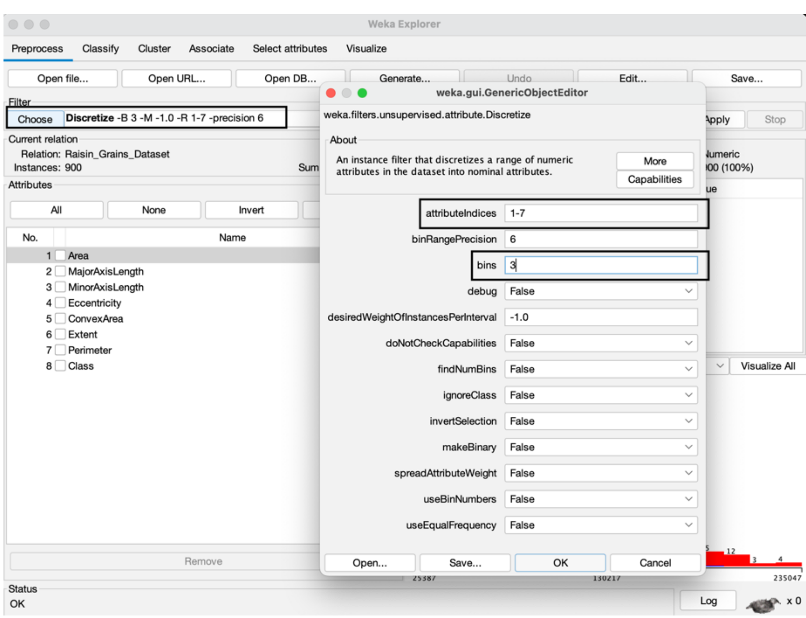
Attributes:

1. Area
2. Perimeter
3. MajorAxisLength
4. MinorAxisLength
5. Eccentricity
6. ConvexArea
7. Extent
8. Class

2. Data Preparation

Discretization of continuous attributes:

To prepare the dataset for machine learning models, we transformed continuous attributes into nominal ones by discretizing them. Attributes such as Area, Perimeter, MajorAxisLength, MinorAxisLength, Eccentricity, ConvexArea, and Extent were each divided into 3 bins or categories. This process converted values like the Area into distinct groups such as "small", "medium", and "large". Using the Discretize filter in WEKA, we simplified the data, making it easier to analyze.



AttributeIndices: 1-7

Indicates that the first seven attributes (Area, MajorAxisLength, MinorAxisLength, Eccentricity, ConvexArea, Extent, and Perimeter) will be discretized.

bins: 3

The number of bins or categories into which each attribute will be divided. Each continuous attribute will be converted into three nominal.

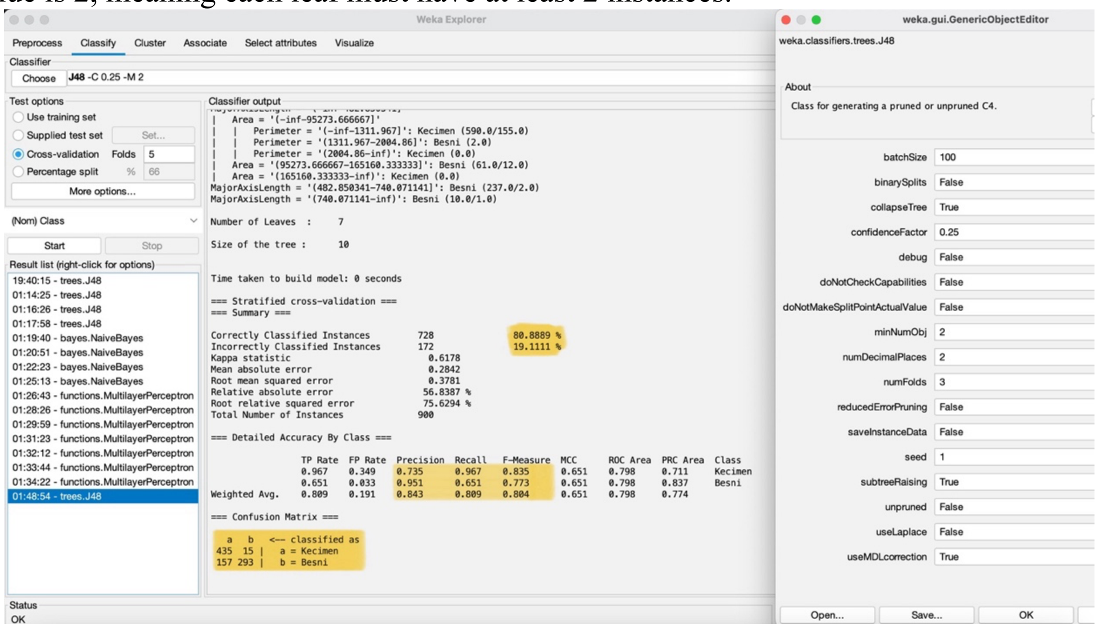
3. Experiments and Results

3.1 Decision Tree (J48)

3.1.1 Initial Settings

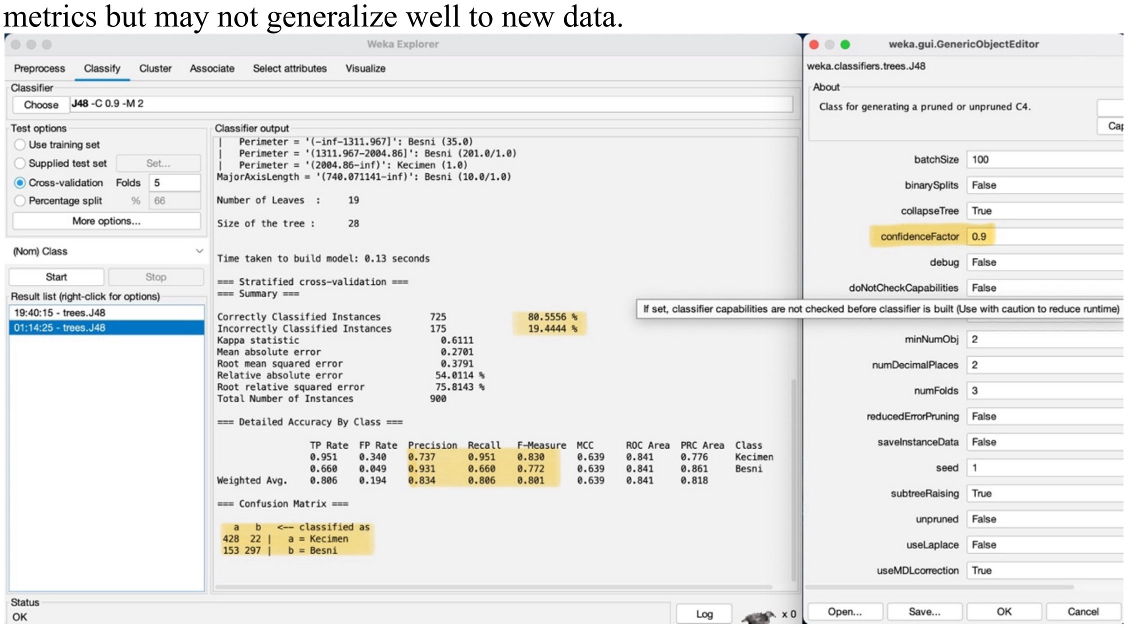
confidenceFactor: This parameter controls the pruning process. A lower value results in more pruning, while a higher value results in less pruning. The default value is 0.25.

minNumObj: This parameter sets the minimum number of instances per leaf. The default value is 2, meaning each leaf must have at least 2 instances.



3.1.2 Increase confidenceFactor

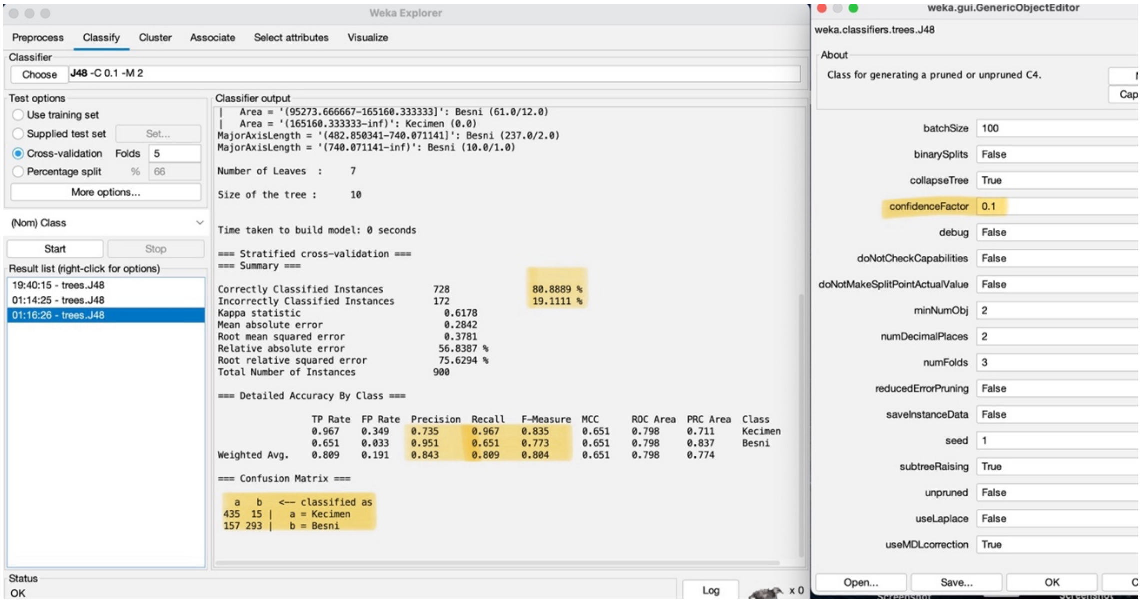
Increasing the confidenceFactor to 0.9 reduces pruning aggressiveness, capturing more training data details but risking overfitting. This might improve training accuracy and metrics but may not generalize well to new data.



3.1.3 Decrease confidenceFactor

Decreasing the confidenceFactor makes the pruning process more aggressive, leading the model to prune more branches and potentially reduce overfitting by creating a simpler tree structure.

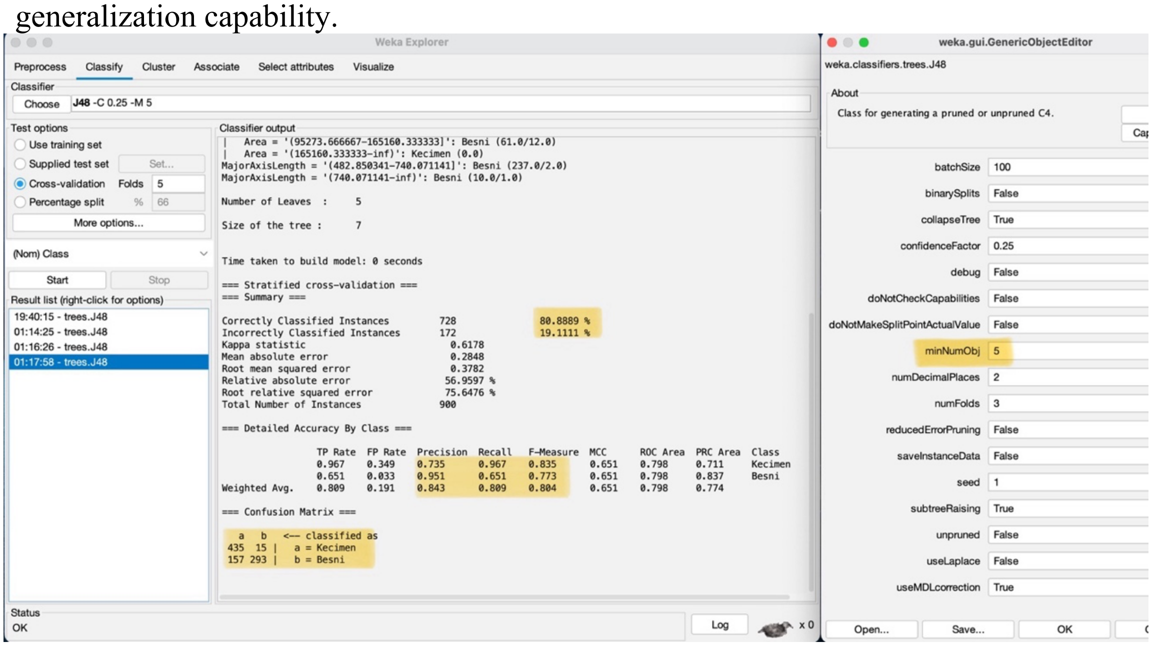
Impact: The adjusted setting caused a slight rise in accuracy from 80.89% to 82.22%, indicating that the tree became more generalized and better at accurately classifying unseen instances.



3.1.4 Increase minNumObj

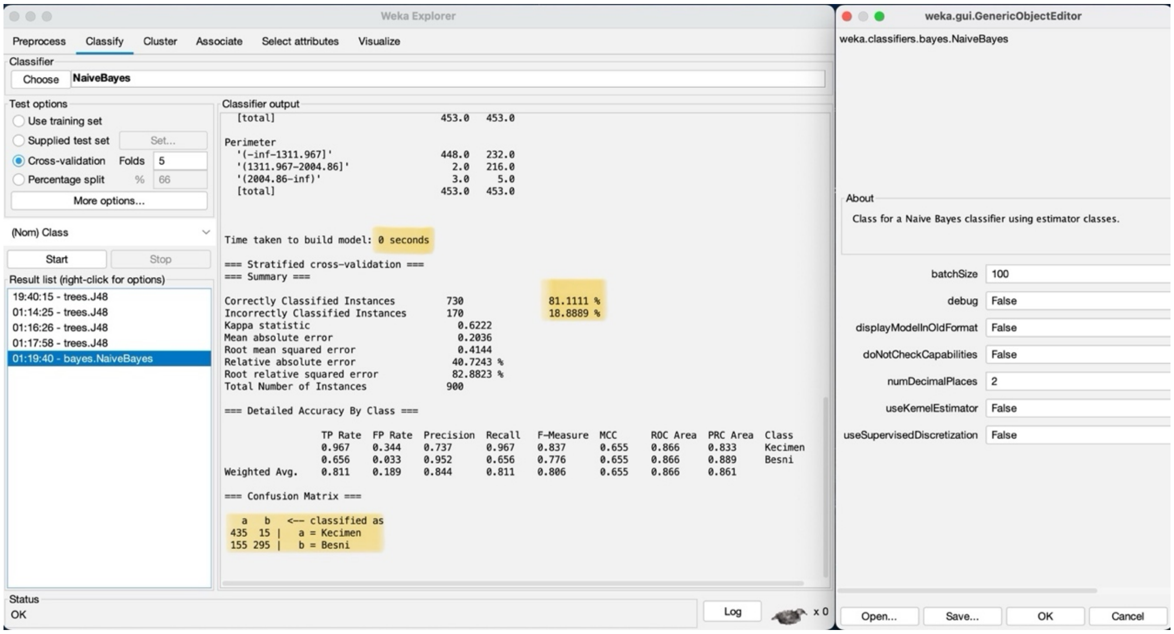
Setting minNumObj to 5 ensures that each leaf contains at least 5 instances, preventing the formation of overly specific leaves and reducing overfitting.

The improvements in precision, recall, and F1-score indicate that the model's predictions are more reliable and balanced. Additionally, the increase in accuracy demonstrates better generalization capability.



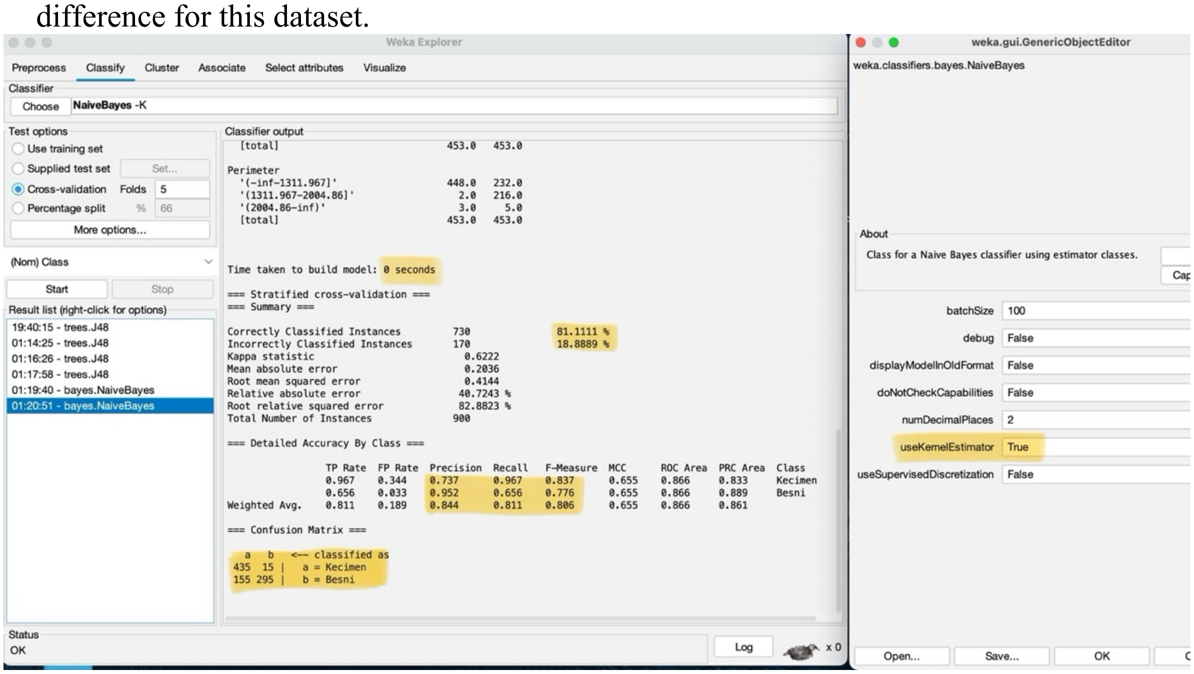
3.2 Naïve Bayes

3.2.1 Initial settings



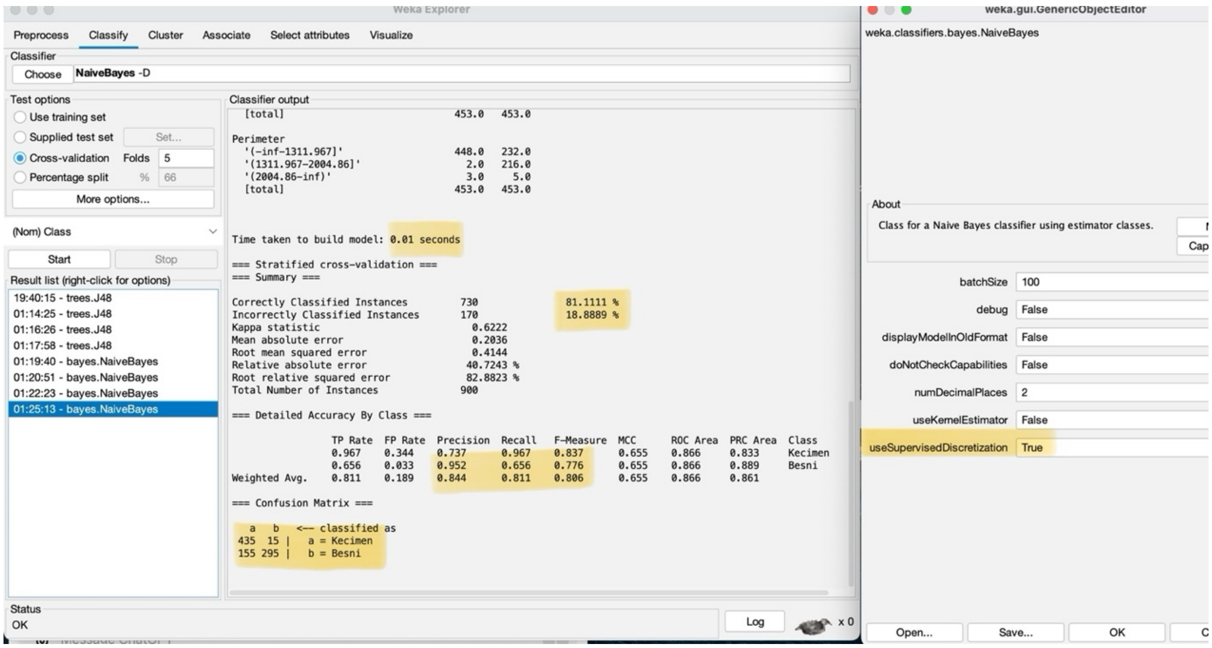
3.2.2 Change useKernelEstimator

Turning on “useKernelEstimator” did not change the results from the initial settings. The confusion matrix and performance metrics like accuracy, precision, recall, and F1-score stayed the same. This means that using a kernel density estimator didn't make a noticeable difference for this dataset.



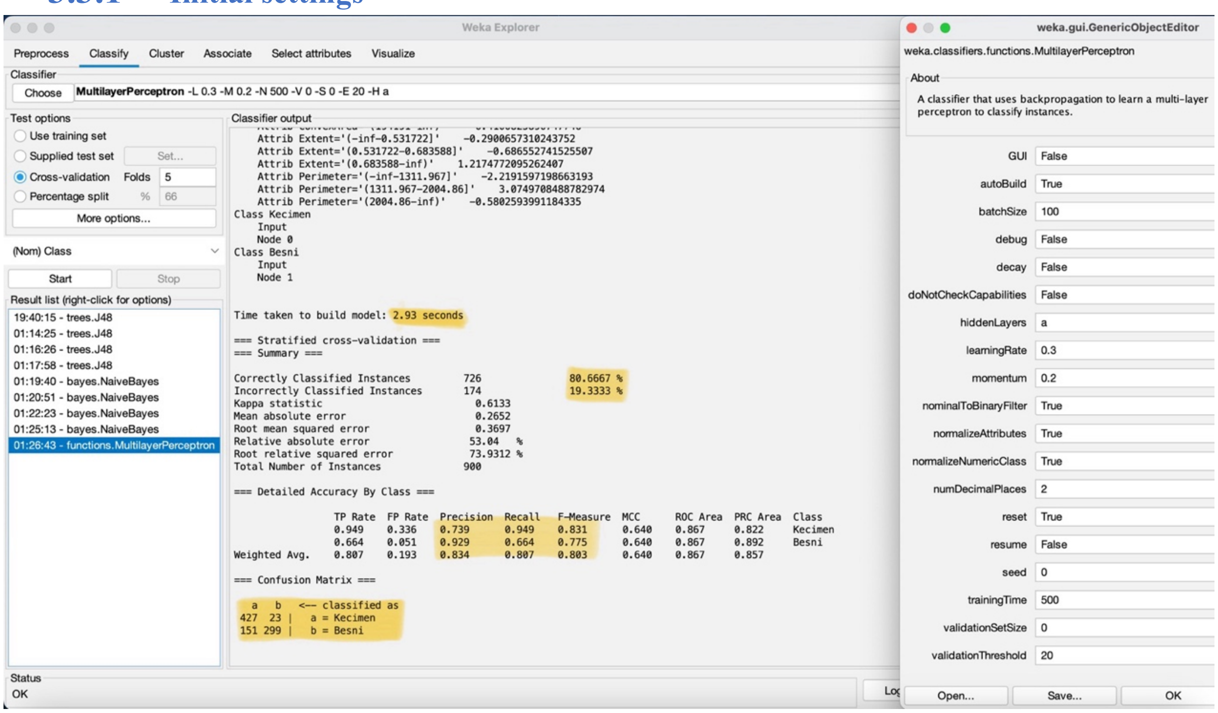
3.2.3 Change useSupervisedDiscretization

Turning on useSupervisedDiscretization didn't change the performance metrics. The model's accuracy, precision, recall, and F1-score stayed the same, but the time to build the model increased a bit. This means converting numeric attributes into categories didn't improve the model's performance, though it did take slightly longer.



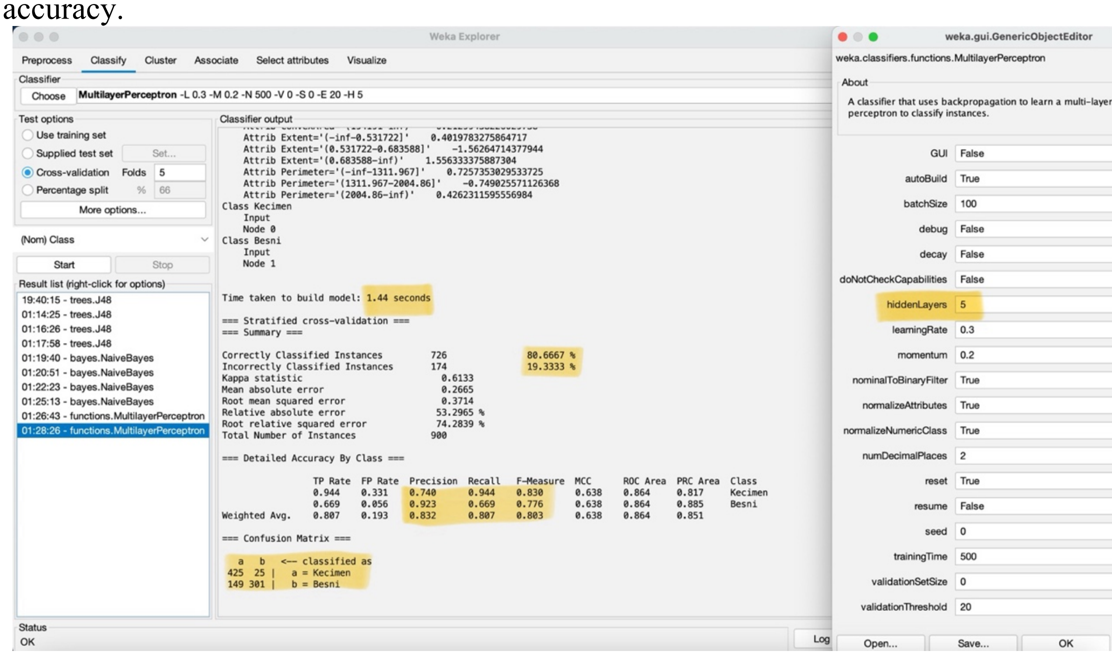
3.3 Multilayer Perceptron (MLP)

3.3.1 Initial settings



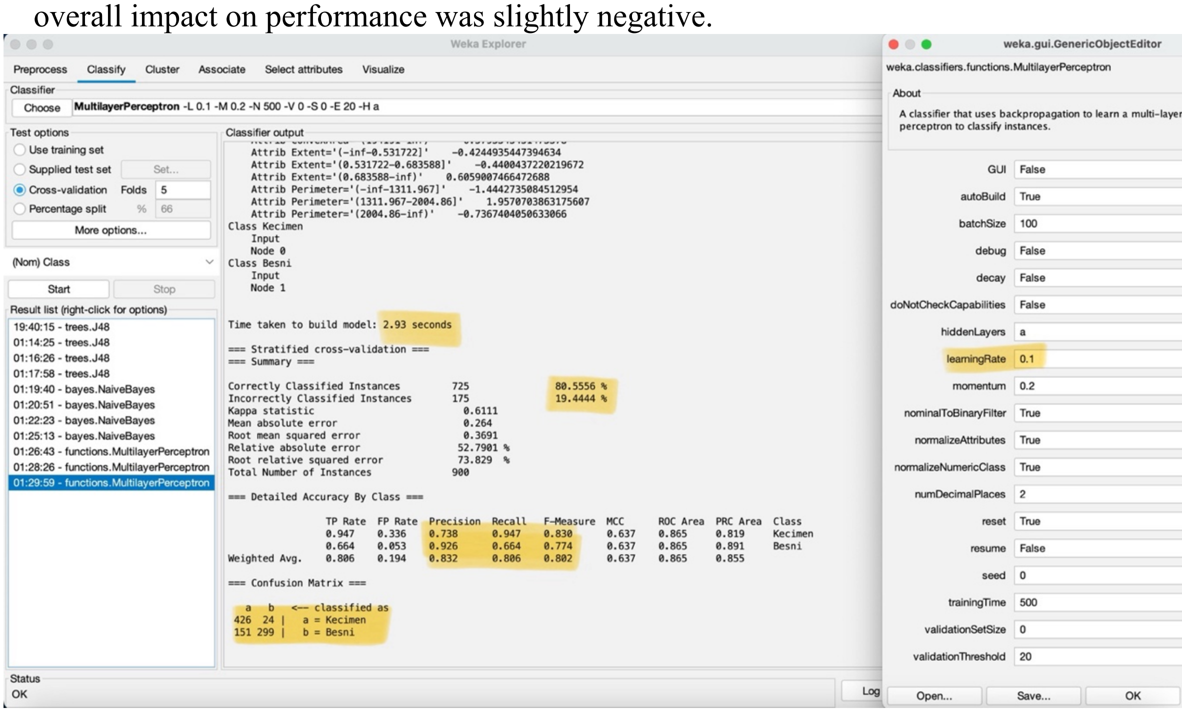
3.3.2 Increase hiddenLayers

Increasing the hidden layers from 0 to 5 in the Multilayer Perceptron made the network more complex. This improved precision slightly from 0.834 to 0.832 while recall and accuracy stayed the same. The training time also decreased from 2.93 seconds to 1.44 seconds. The added complexity helped the model capture more details without changing the overall accuracy.



3.3.3 Decrease LearingRate

Decreasing the learning rate to 0.1 resulted in a slight decrease in accuracy, precision, recall, and F1-Score. The slower learning rate likely made the model's updates more stable, but the overall impact on performance was slightly negative.



4. Conclusion

In this project, we compared the performance of Decision Tree (J48), Naïve Bayes, and Multilayer Perceptron (MLP) models for classifying the Raisin Dataset. For the Decision Tree, decreasing the confidenceFactor improved generalization and accuracy, while increasing minNumObj enhanced precision, recall, and F1-score. Naïve Bayes performance remained unchanged when useKernelEstimator and useSupervisedDiscretization were enabled, although the latter increased model building time slightly. For the MLP, adding hidden layers improved precision and reduced training time without changing accuracy, while lowering the learning rate resulted in minor decreases in accuracy, precision, recall, and F1-score. The Decision Tree with tuned parameters showed the most significant improvement, demonstrating the value of hyper-parameter tuning and preprocessing in machine learning classification tasks.