
Faculty of Engineering and Technology

Department of Electrical and Computer Engineering

ARTIFICIAL INTELLIGENCE | ENCS3340

Project II: Machine Learning for Classification

---

Prepared by:

Yara Hamed 1211269

Dana Assad 1211452

Instructor: Dr. Yazan Abu Farha

Section: 1

Date: 21/6/2024

## Abstract

This is a Machine Learning project using weka to compare different machine learning algorithms for a classification task, the machine learning algorithms are Decision Tree, Naïve Bayes, and MLP. And the model testing will use 5-fold cross-validation.

# Table of Contents

# Table of Figures

# Description

This is team ID's:

1211269

1211452

The first ID is less than the second one:

- ⇨ 9 % 3 = 0, so the dataset that will be used in this project is 0 "Speaker Accent Recognition Dataset".

The dataset contains from 13 attributes: language, X1, X2 up to X12.

We chose X1 to be our target class because the language class contains six bins so it's hard to calculate the values from its confutation matrix, and the Naïve Bayes algorithm didn't accept it as a target class.

# Discretization

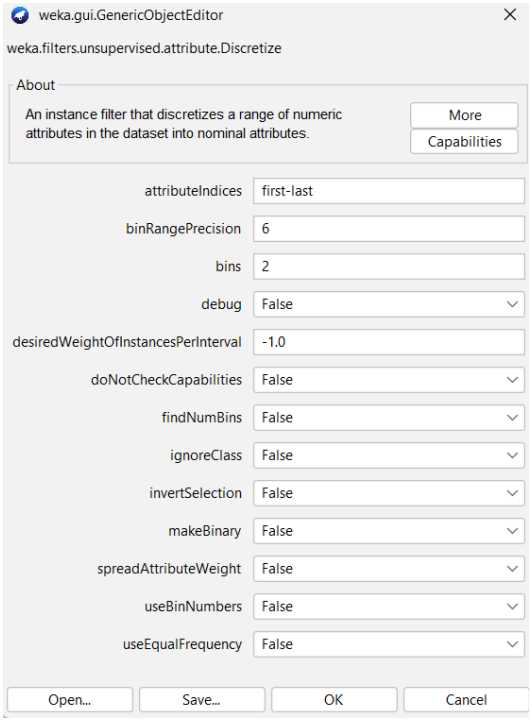


*Figure 1: Discretization setting.*

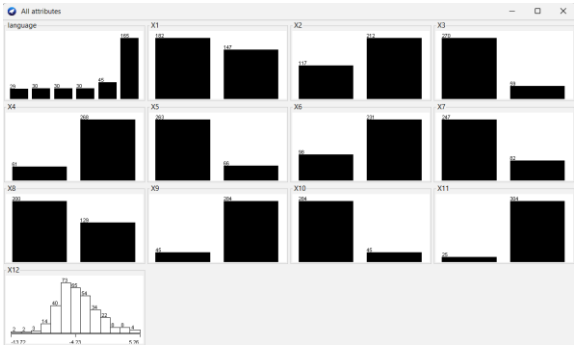

*Figure 2: Discretization result.*

As shown in the setting figure all attribute discretized with 2 bins. The language attribute already discrete "nominal" one so its didn't change.

Note:

we try to change the last att. But its didn't work, even if we try to change it alone with new file.

# 1. Naïve Bayes

## Model testing

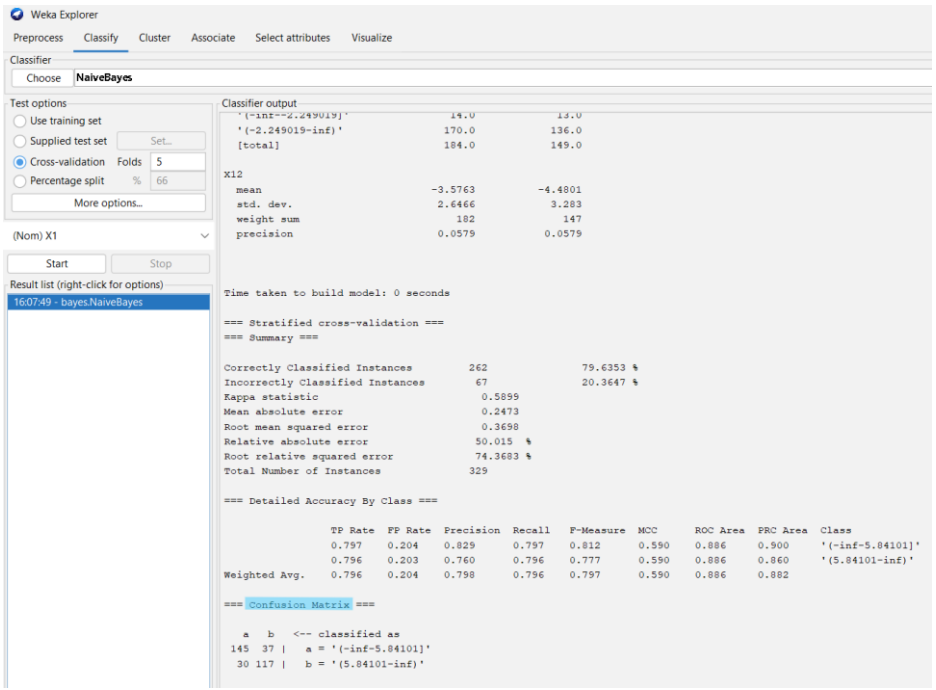


*Figure 3: NB-model testing.*

From the matrix:
TP = 145
FP = 30
TN = 117
FN = 37

Accuracy = 0.796353
= 79.6353%
Precision = 0.829
= 82.9%
Recall = 0.797
= 79.7%
F1-score = 0.813
= 81.3%

## Hyper-parameter changing

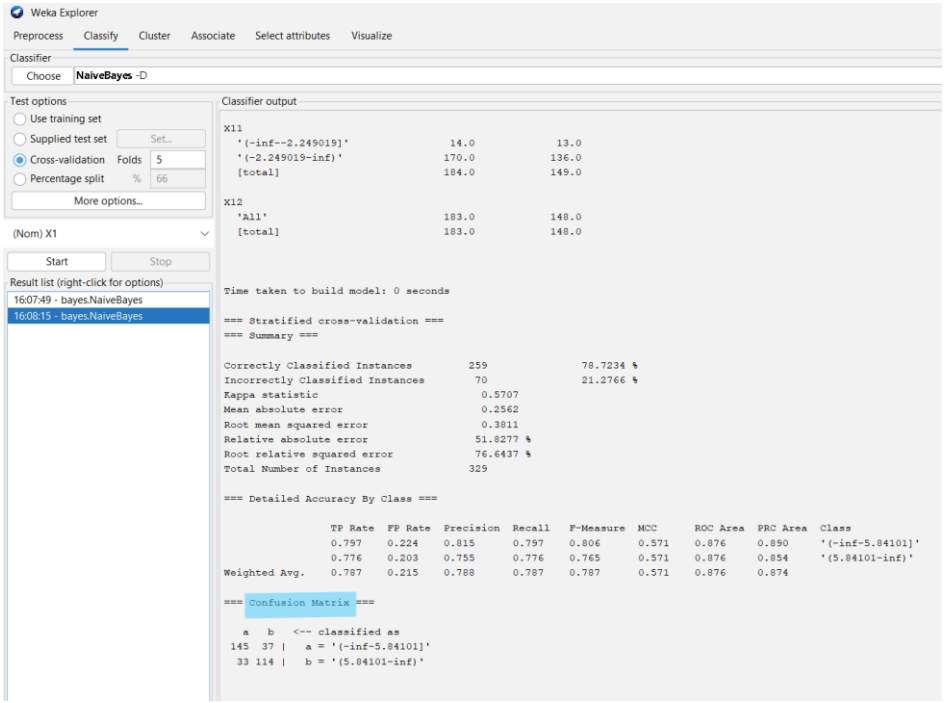


*Figure 4: NB-HP changing.*

From the matrix:
TP = 145
FP = 33
TN = 114
FN = 37

Accuracy = 0.787234
= 78.7234%
Precision = 0.815
= 81.5%
Recall = 0.797
= 79.7%
F1-score = 0.806
= 80.6%

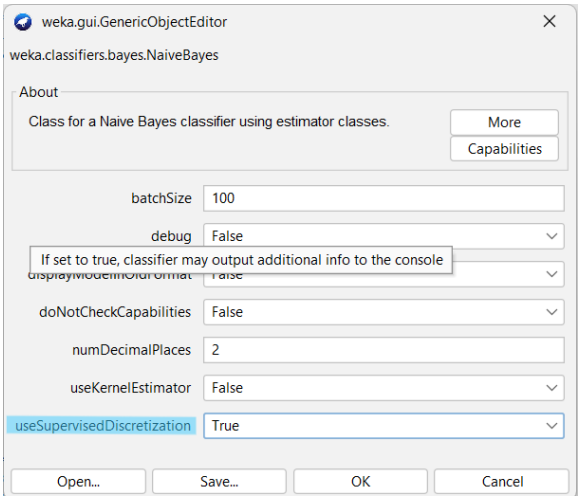


*Figure 5: NB-HP changing setting.*

Effect study:

When "useSupervisedDiscrtization" changed to true instead of false. The accuracy decreases from 79.6353% to 78.7234% and the F1-score decreases from 81.3% to 80.6%.

# 2. MLP

## Model testing

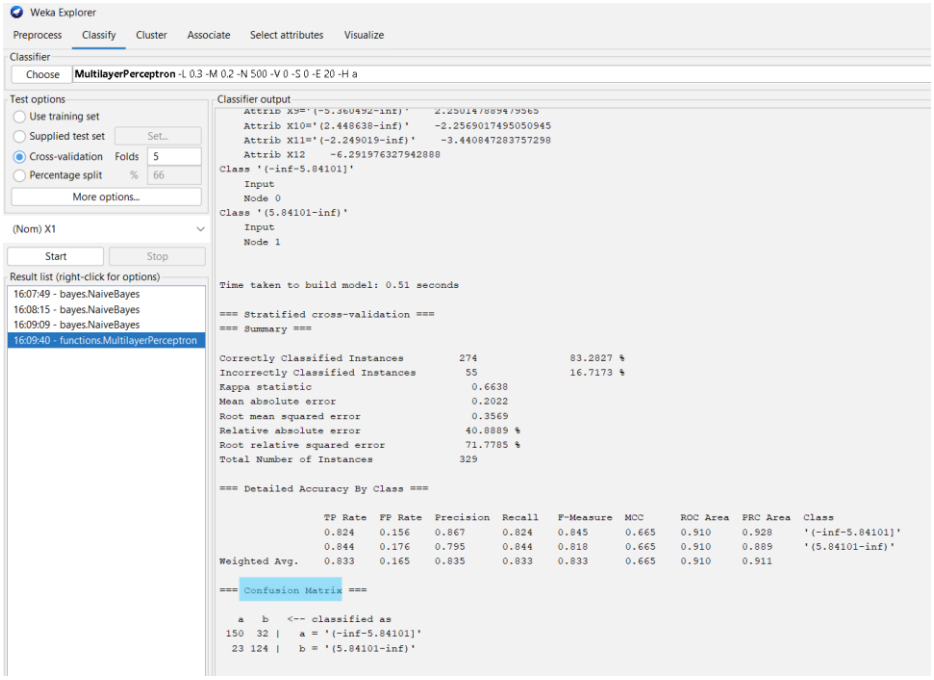


*Figure 6: MLP-model testing.*

From the matrix:
TP = 150
FP = 23
TN = 124
FN = 32

Accuracy = 0.832827
= 83.2827%
Precision = 0.867
= 86.7%
Recall = 0.824
= 82.4%
F1-score = 0.844
= 84.4%

## Hyper-parameter changing

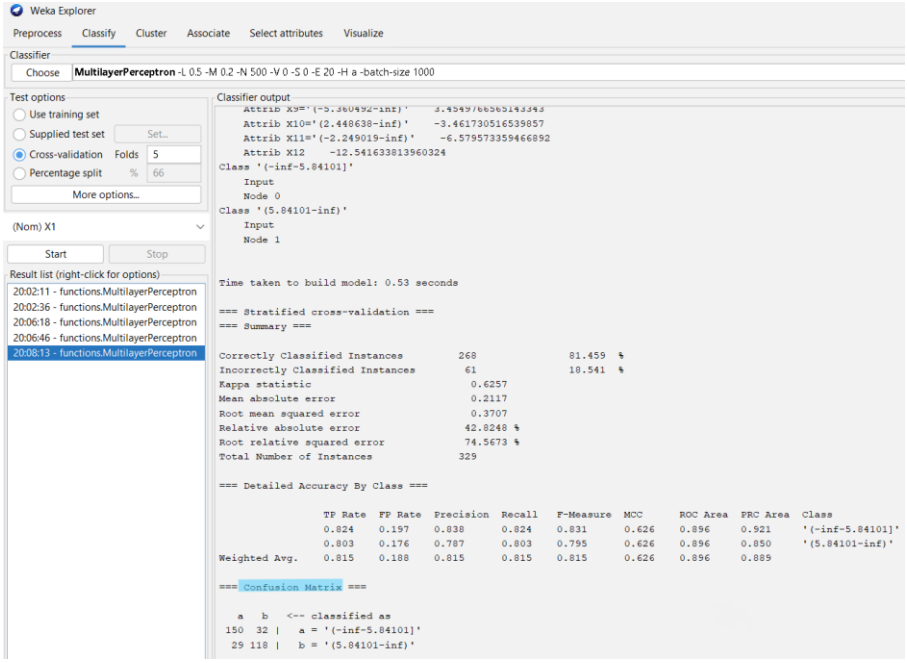


*Figure 7: MLP-HP changing.*

From the matrix:
TP = 150
FP = 29
TN = 118
FN = 32

Accuracy = 0.81459
= 81.459%
Precision = 0.838
= 83.8%
Recall = 0.824
= 82.4%
F1-score = 0.831
= 83.1%

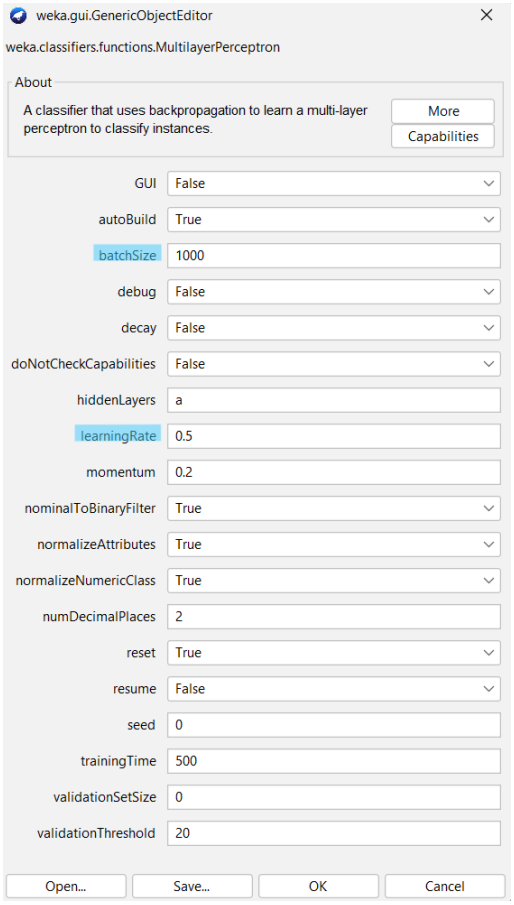


*Figure 8: MLP-HP changing setting.*

Effect study:

when "validationSetSize" changed to 30 insisted of 20 the accuracy didn't change. So, we try to change "batchSize" to 1000 instead of 100 and change "learningRate" to 0.5 instead of 0.3.
The accuracy decreased from 83.2827% to 81.459% and the F1 score decreased from 84.4% to 83.1%

# 3. Decision Tree

## Model testing

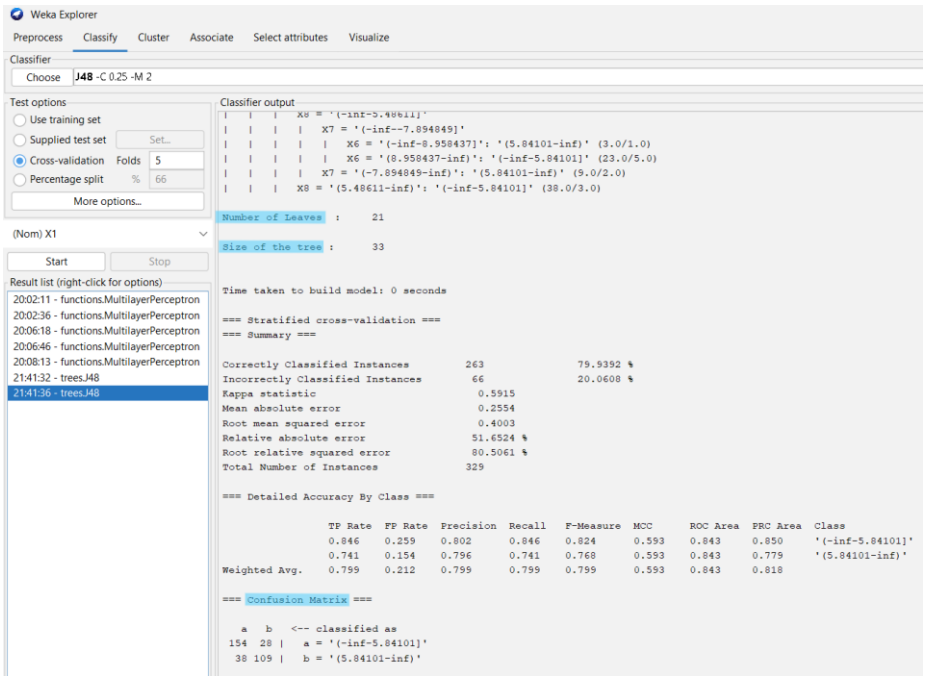


*Figure 9: DT-model testing.*

From the matrix:
TP = 154
FP = 38
TN = 109
FN = 28

Accuracy = 0.799392
= 79.9392%
Precision = 0.802
= 80.2%
Recall = 0.846
= 84.6%
F1-score = 0.823
= 82.3%
# of leaves: 21
Tree size: 33

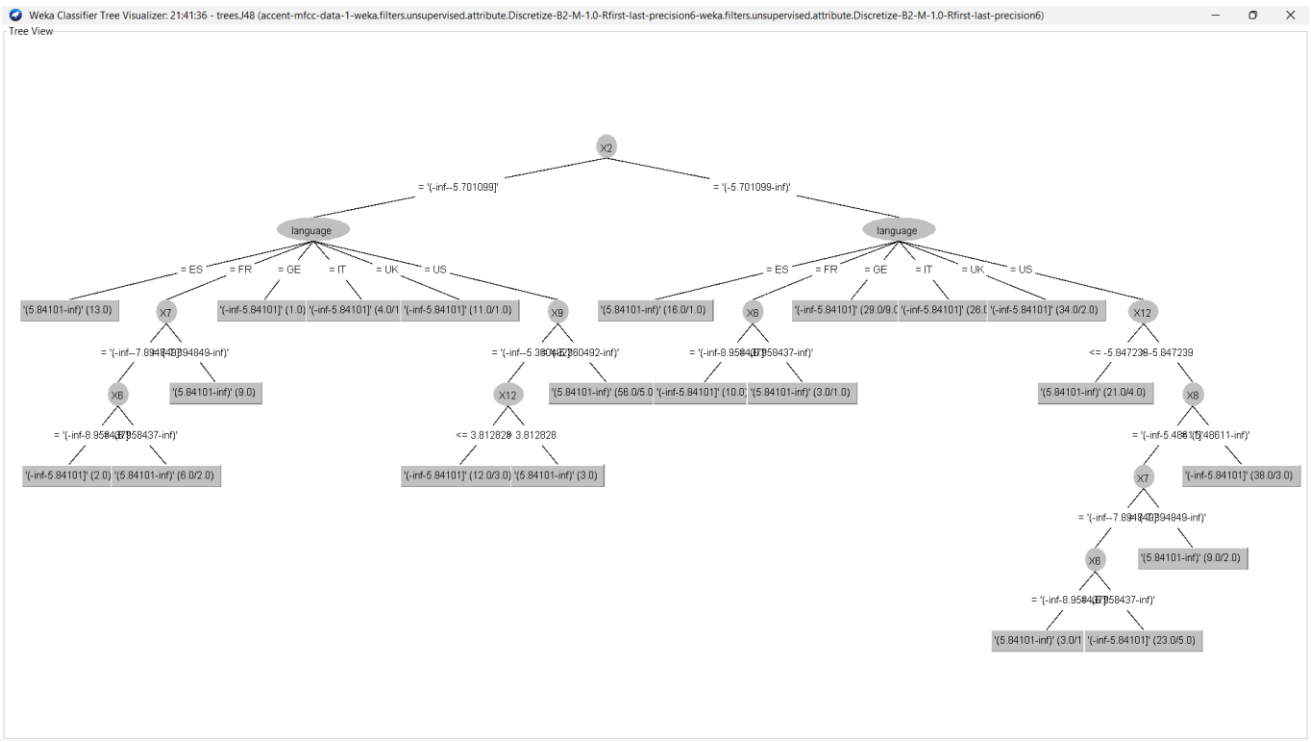


*Figure 10: DT result.*

## Hyper-parameter changing

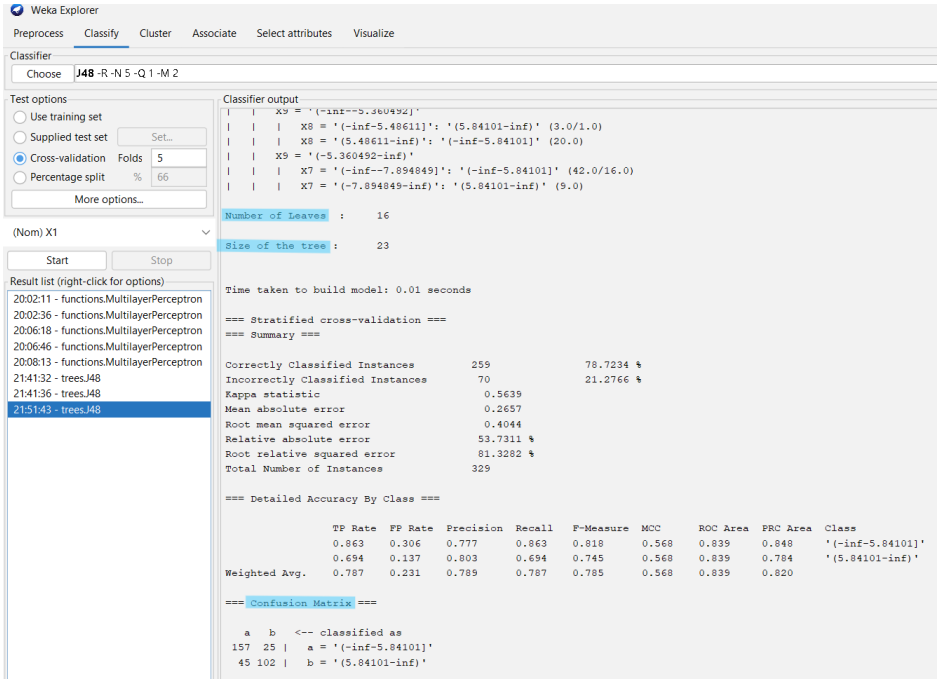


*Figure 11: DT-HP changing.*

From the matrix:
TP = 157
FP = 45
TN = 102
FN = 25

Accuracy = 0.787234
= 78.7234%
Precision = 0.777
= 77.7%
Recall = 0.863
= 86.3%
F1-score = 0.818
= 81.8%
# of leaves: 16
Tree size: 23

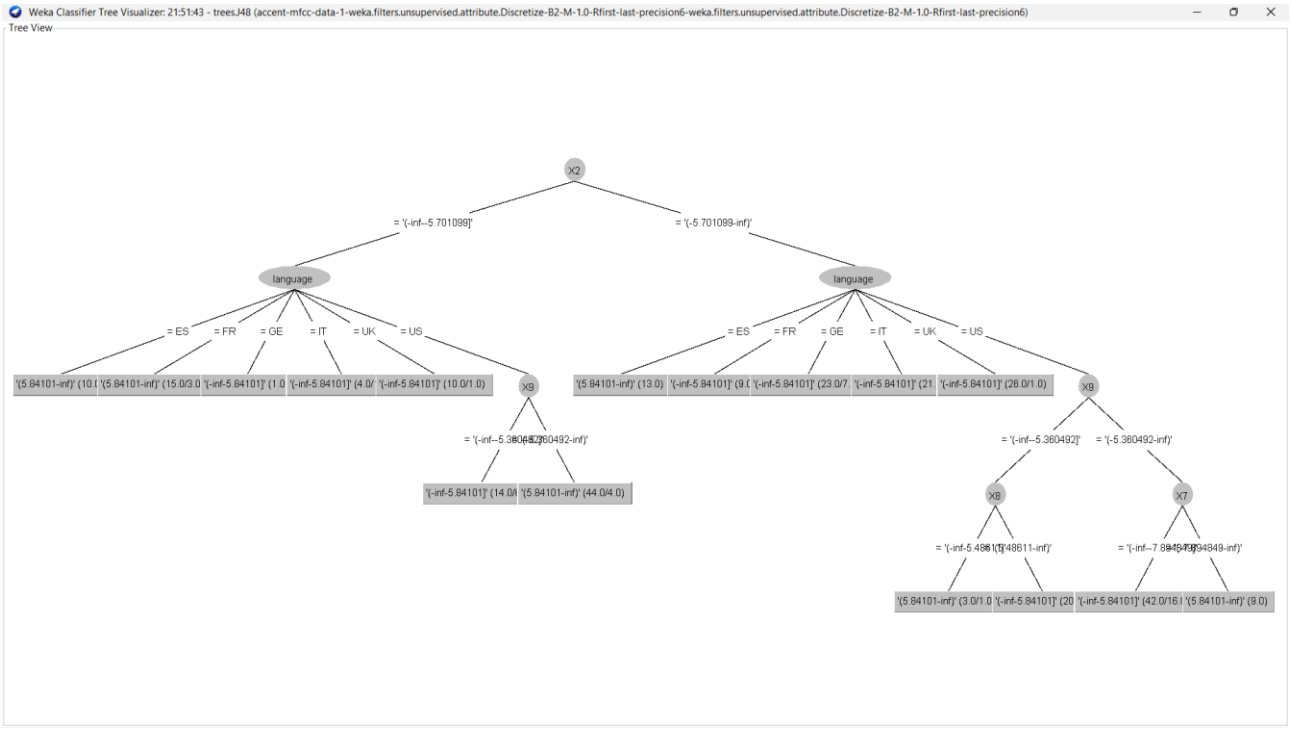


*Figure 12: DT-HP result.*

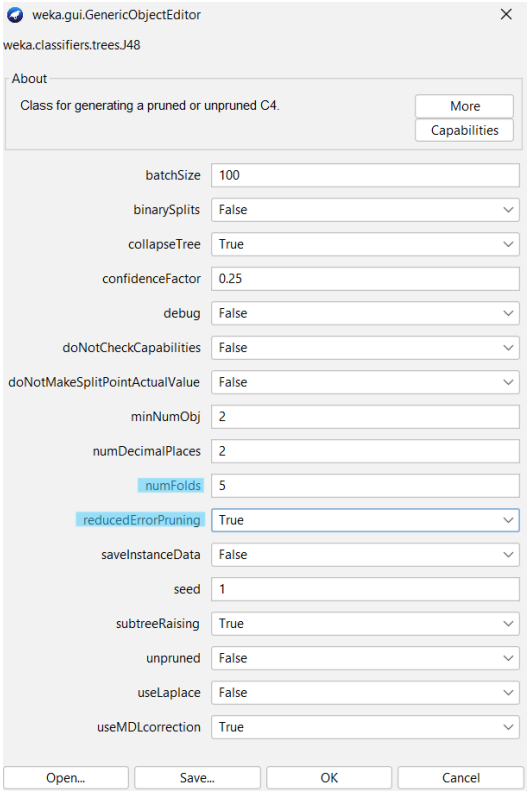


*Figure 13: DT-HP changing setting.*

## Effect study:

when "numFolds" changed to 5 instead of 3 and "reducedErrorPruning" changed to true instead of false.
The accuracy decrees from 79.9392% to 78.7234%, the F1-score decrees from 82.3% to 81.8%, the number of leaves decrease from 21 to 16, and the size of the tree decrease from 33 to 23.

## Note:

- ⇨ We took accuracy, precision, and recall from the results in the program as shown in the added figure. and calculate the F1-score from this rule: (2 * precision * recall) / (precision + recall).

```
=== Summary ===

Correctly Classified Instances         259               78.7234 %
Incorrectly Classified Instances        70               21.2766 %
Kappa statistic                          0.5639
Mean absolute error                      0.2657
Root mean squared error                  0.4044
Relative absolute error                 53.7311 %
Root relative squared error             81.3282 %
Total Number of Instances              329

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 0.863    0.306    0.777      0.863   0.818      0.568    0.839     0.848     '(-inf-5.84101]'
                 0.694    0.137    0.803      0.694   0.745      0.568    0.839     0.784     '(5.84101-inf)'
Weighted Avg.    0.787    0.231    0.789      0.787   0.785      0.568    0.839     0.820

=== Confusion Matrix ===

   a   b   <-- classified as
 157  25 |   a = '(-inf-5.84101]'
  45 102 |   b = '(5.84101-inf)'
```

*Figure 14: Note.*

## Conclusion

To sum up, this study showcases how three different machine learning methods - Naïve Bayes, MLP, and Decision Tree - perform in comparison to each other when applied to the "Speaker Accent Recognition Dataset" in Weka for classification purposes. Naïve Bayes demonstrated a well-rounded performance achieving an accuracy of 79.64% and an F1-score of 81.3%, albeit its performance dipped slightly after hyper-parameter tuning. The MLP algorithm showed strong performance for the classification task with an accuracy of 83.28% and an F1-score of 84.4%, showing its resilience despite slight decreases after tweaking hyper-parameters. Similarly to Naïve Bayes, Decision Tree also showed a decrease in performance when parameters were modified, despite achieving an accuracy of 79.94% and an F1-score of 82.3%. These findings demonstrate that although all three algorithms are suitable for classification, MLP typically delivers better results for this particular dataset. Future research could investigate additional parameter tuning and alternative machine learning algorithms to potentially improve classification accuracy and efficiency.