Faculty of Engineering & Technology
Electrical & Computer Engineering Department

ARTIFICAL INTELEGENT

ENCS3340

Second project report

MACHINE LEARNING

Prepared by:

Ibrahim Mahmoud - 1190747

Islam Jihad - 1191375

Instructor: Yazan Abu Farha & Adnan Yahya

Section: 2 & 3

Date: 11/June/2022

Abstract

We will use WEKA program to be preprocessing, classifying and simulate the testing results. According to the min ID in our group, we used test set number two which contains Early-stage diabetes risk prediction Dataset. We need to test these two algorithms: Decision Trees and Naïve Bayes. Also, there are one additional algorithm from our choice, we decided to use Random Forest and we will process them using 5-folds method.

Table of Contents

Table of figures

Theory

Machine learning

is a branch of research dedicated to understanding and developing 'learn' techniques, or methods that use data to enhance performance on a set of tasks. It is considered to be a kind of artificial intelligence. Machine learning algorithms create a model based on training data to make predictions or judgments without having to be explicitly programmed to do so. Machine learning algorithms are utilized in a broad range of applications, including medicine, email filtering, voice recognition, and computer vision, when developing traditional algorithms to do the required tasks is difficult or impossible.

However, not all machine learning is statistical learning. A subset of machine learning is strongly connected to computational statistics, which focuses on generating predictions using computers. The discipline of machine learning benefits from the study of mathematical optimization since it provides tools, theory, and application fields. Data mining is a similar branch of research that focuses on unsupervised learning for exploratory data analysis. [5] [6] Data and neural networks are used in certain machine learning implementations to replicate the functioning of a biological brain. [7][8] Machine learning is also known as predictive analytics when it is used to solve business challenges.

Machine learning models

A machine learning model is a mathematical representation of the training process's output. Machine learning is the study of various algorithms that can automatically develop and create a model based on experience and old data. A machine learning model is computer software that recognizes patterns or behaviors based on past experience or data. The learning algorithm finds patterns in the training data and generates a machine learning model that captures these patterns and predicts fresh data.

WEKA program

The companion program to the book "Data Mining: Practical Machine Learning Tools and Techniques," produced by the University of Waikato in New Zealand, is free software distributed under the GNU General Public License.

Data preparation, clustering, classification, regression, visualization, and feature selection are just a few of the usual data mining activities that Weka offers. Weka expects input to be prepared using the Attribute-Relational File Format (ARFF) and files with the.arff and.csv extensions. Weka's solutions are all based on the premise that the data is provided in a single flat file or relation, with each data item represented by a set of characteristics (normally, numeric or nominal attributes, but some other attribute types are also supported). Weka uses Java Database Connectivity to connect to SQL databases and can handle the results of a

database query. With Deeplearning4j, Weka gives you access to deep learning. [4] It does not support multi-relational data mining, but there is additional software that can be used to transform a collection of connected database tables into a single table appropriate for Weka processing. [5] Sequence modeling is another key field that is presently not addressed by the methods provided in the Weka package.

5-fold cross validation

WEKA is going to split the data into 5 parts one of the parts will be used as a test data and the others is for training, after finishing this it will change the test part and repeat the process for the 5 parts all. We use this method when there is a leek in data so we can perform better results. We used it in all the 3 methods down.

Procedure and Discussion

In our project the decision was on set 1 (Early-stage diabetes risk prediction Dataset)

Decision Tree method

In this method we set all values to be Nominal so we can understate it easily, all of attributes were nominal unless age so we translate it to be nominal and divide it to 10 stages, we notice that when we divide it into more pins the more accrue we get.

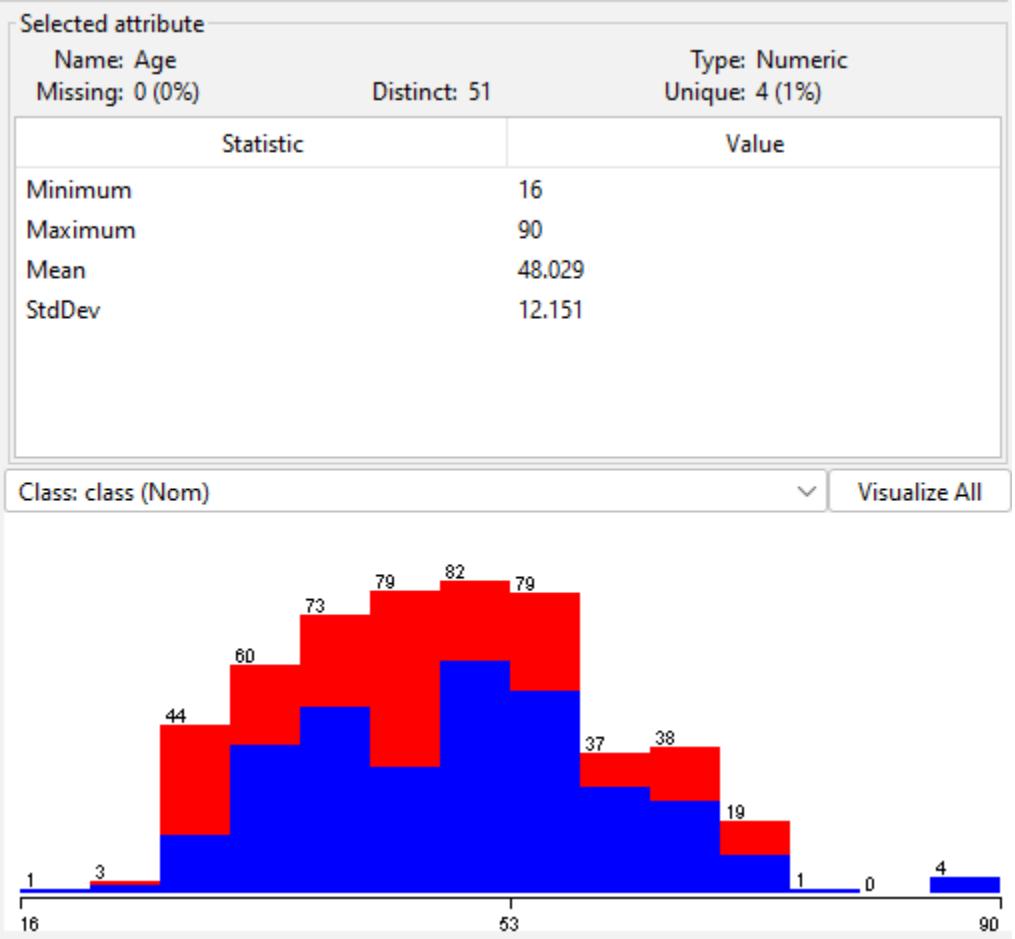


Figure 1 age before nominal

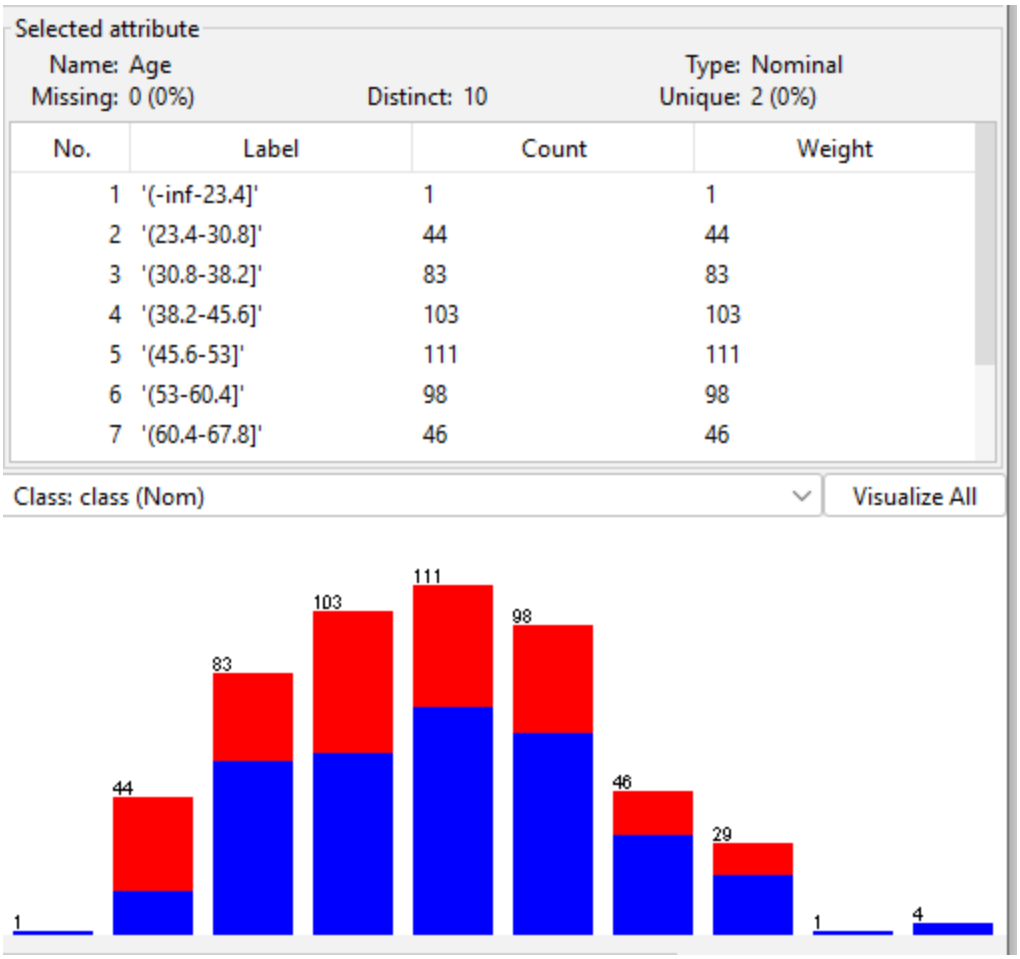


Figure 2 age after nominal

After making the classify of the dataset we got almost the same data analysis for tree decision choice when we turned the binary splits on. The change we got is on accuracy, and the tree hierarchical and number of nodes(which means more memory) so we can say when we used the binary splits, the results were better somehow, and the accuracy got little down, we can see that applying binary split reduced the accuracy but make the tree easier to understand . Attached the picture of the 2 trees before and after binary split:

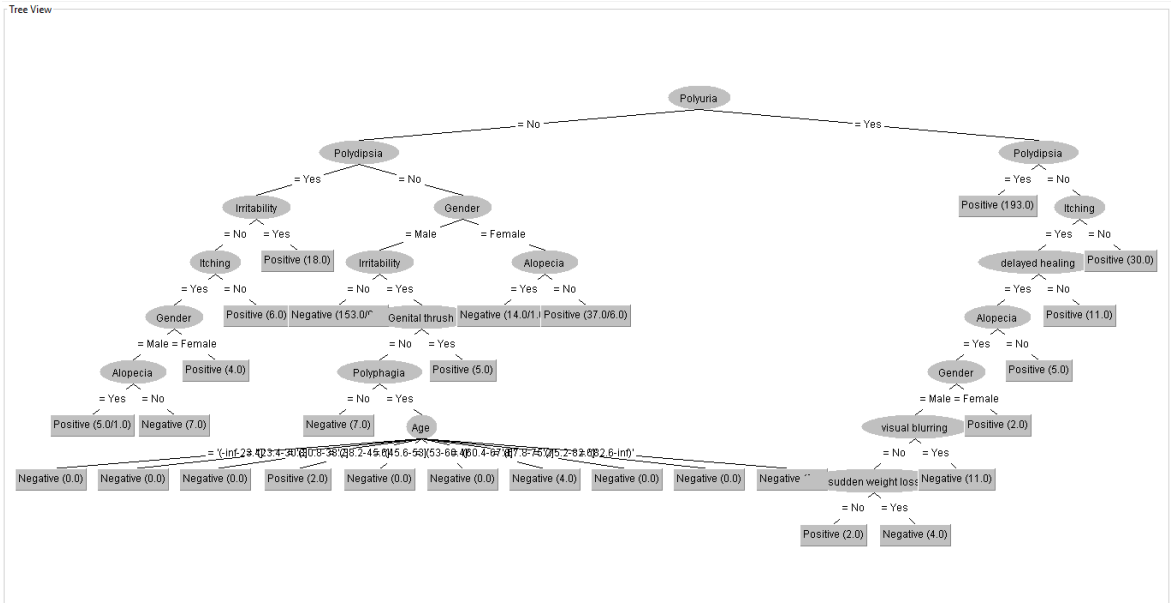


Figure 3 Decision Tree without binary split

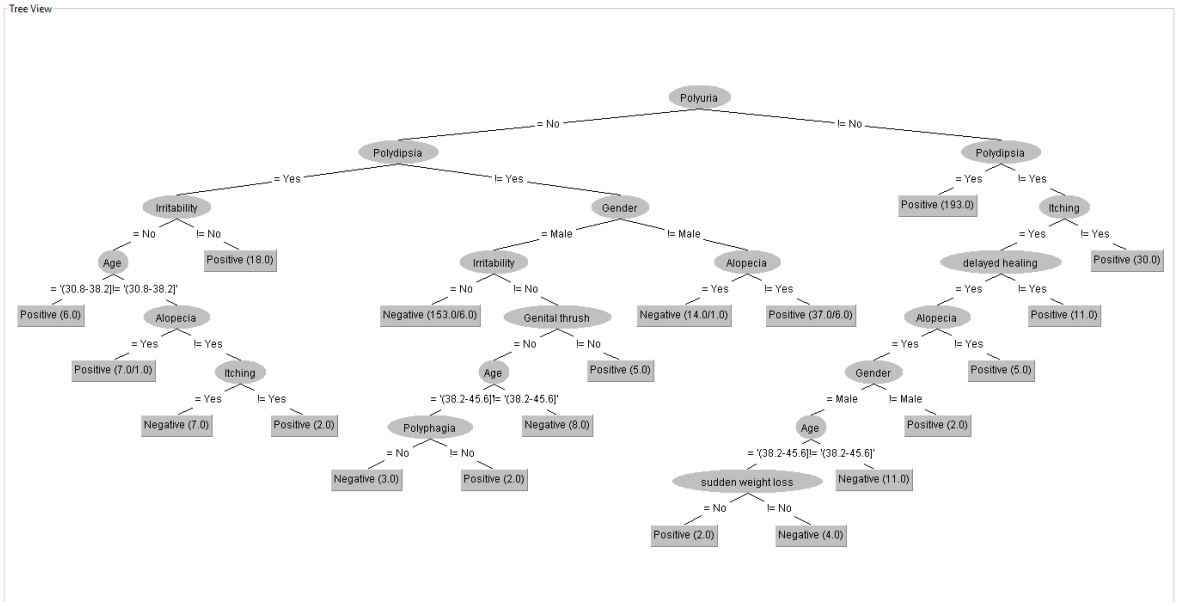


Figure 4 Decision Tree with binary split

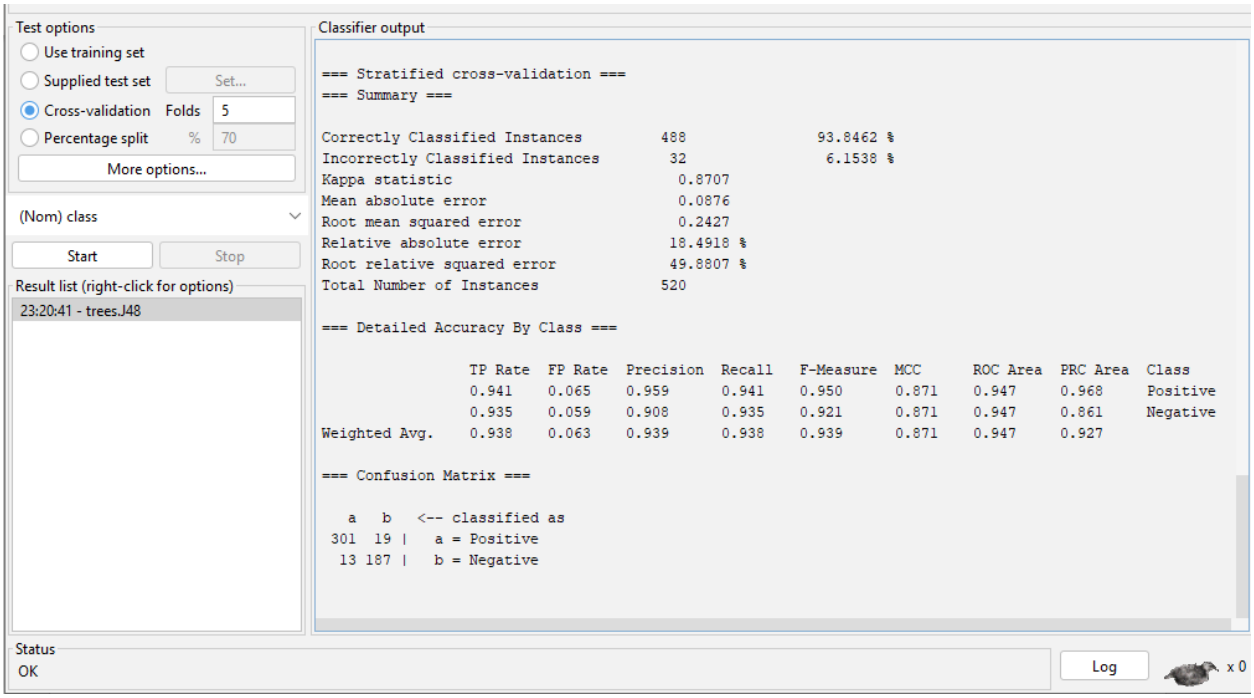


Figure 5 data without binary split

As seen in the picture the

True positive = 301 precision=TP/(TP+FP) = 301/(301+13) =0.959 same as in picture

True negative = 187 recall= TP/(TP+FN)= 301/(301+19)= 0.941 same as in picture

False positive = 13 accuracy= (TP+TN)/(TP+FP+TN+FN)= (301+187)/(301+187+13+19)=93.8%

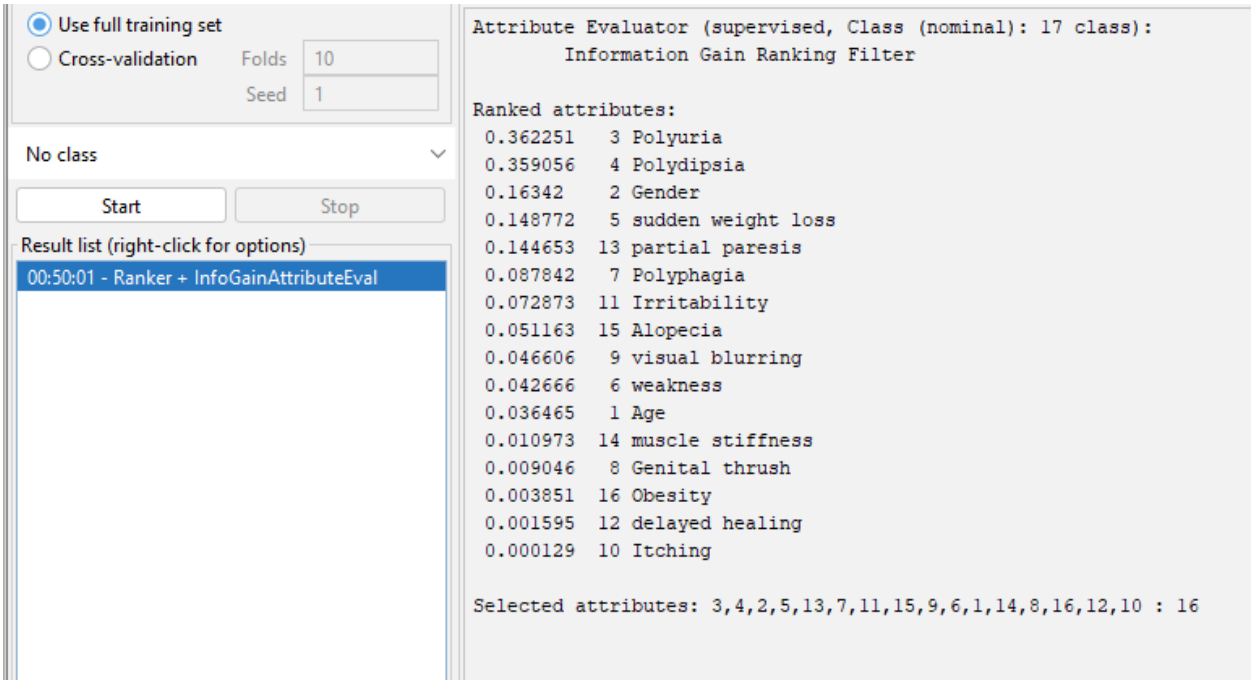
False negative = 19 F1= 2PR/(P+R) = 2*0.959*0.941/(0.959+0.941)=0.950

```
Correctly Classified Instances         484               93.0769 %
Incorrectly Classified Instances        36                6.9231 %
Kappa statistic                          0.8551
Mean absolute error                      0.0866
Root mean squared error                  0.2455
Relative absolute error                 18.2836 %
Root relative squared error             50.4541 %
Total Number of Instances              520

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.928    0.065    0.958      0.928    0.943      0.856    0.951     0.970     Positive
                 0.935    0.072    0.890      0.935    0.912      0.856    0.951     0.867     Negative
Weighted Avg.    0.931    0.068    0.932      0.931    0.931      0.856    0.951     0.930

=== Confusion Matrix ===

   a   b   <-- classified as
 297  23 |   a = Positive
  13 187 |   b = Negative
```

Figure 6 data with binary split

accuracy= (TP+TN)/(TP+FP+TN+FN)= (297+187)/(297+187+13+23)=93.07%

And here the data is different a bit and we can calculate them in the same way.



Information gain is 0.362 for Polyuria so that it was on the top of the decision tree.

Naïve Bayes method

According to the last digit of the least student id in the team mod3, the dataset we used was dataset number one (Early-stage diabetes risk prediction Dataset)

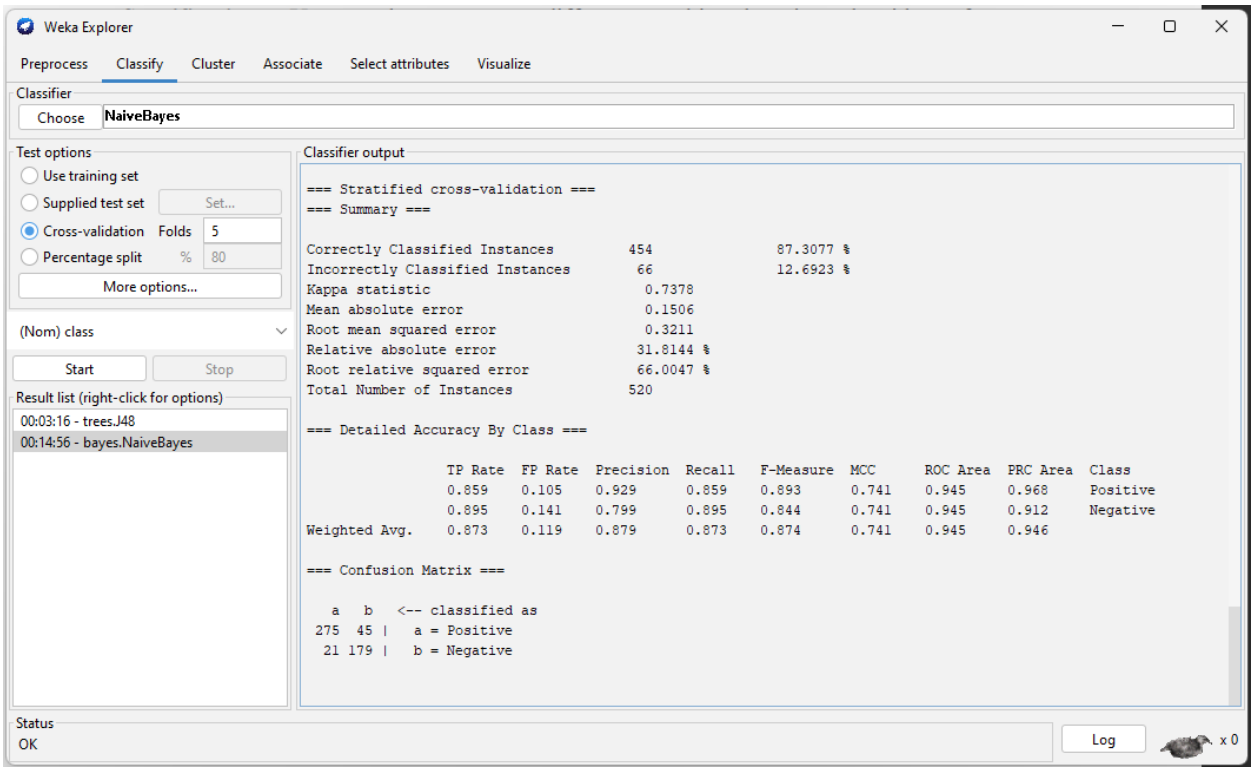


Figure 7 Naive Bayes data

From the figure above:

TP = 275, FP = 21, TN = 179, FN = 45

Precision = $\frac{TP}{TP + FP} = \frac{275}{275+21} = 0.929$

Recall = $\frac{TP}{TP+FN} = \frac{275}{275+45} = 0.859$

Accuracy = $\frac{TP+TN}{TP+FP+TN+FN} = \frac{275+179}{275+179+45+21} = 0.873 = 87.3\%$

F1-Score = $\frac{2pr}{p+r} = \frac{2*0.929*0.859}{0.929+0.859} = 0.893$

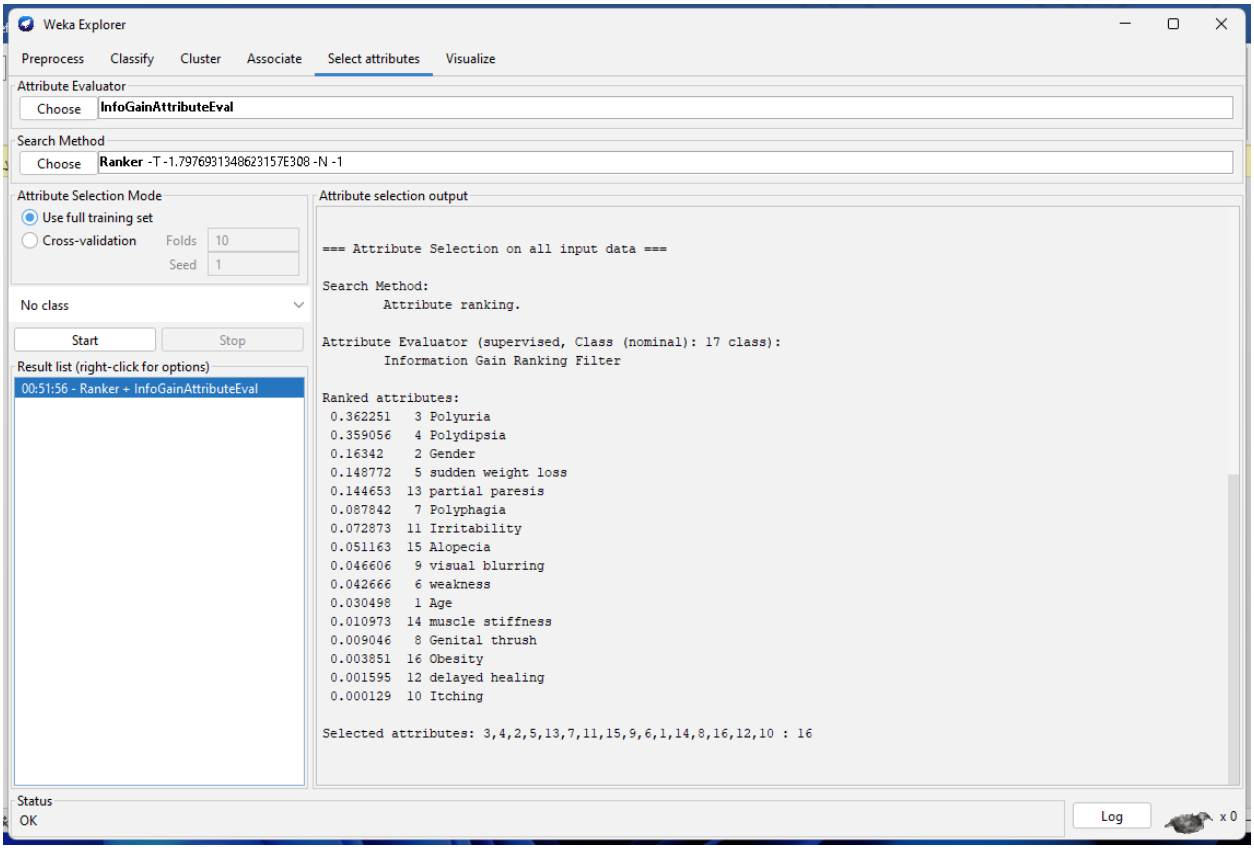


Figure 8 Naïve Bayes info gain

Information gain is 0.362 for Polyuria so that it was on the top of the decision tree.

Random Forest

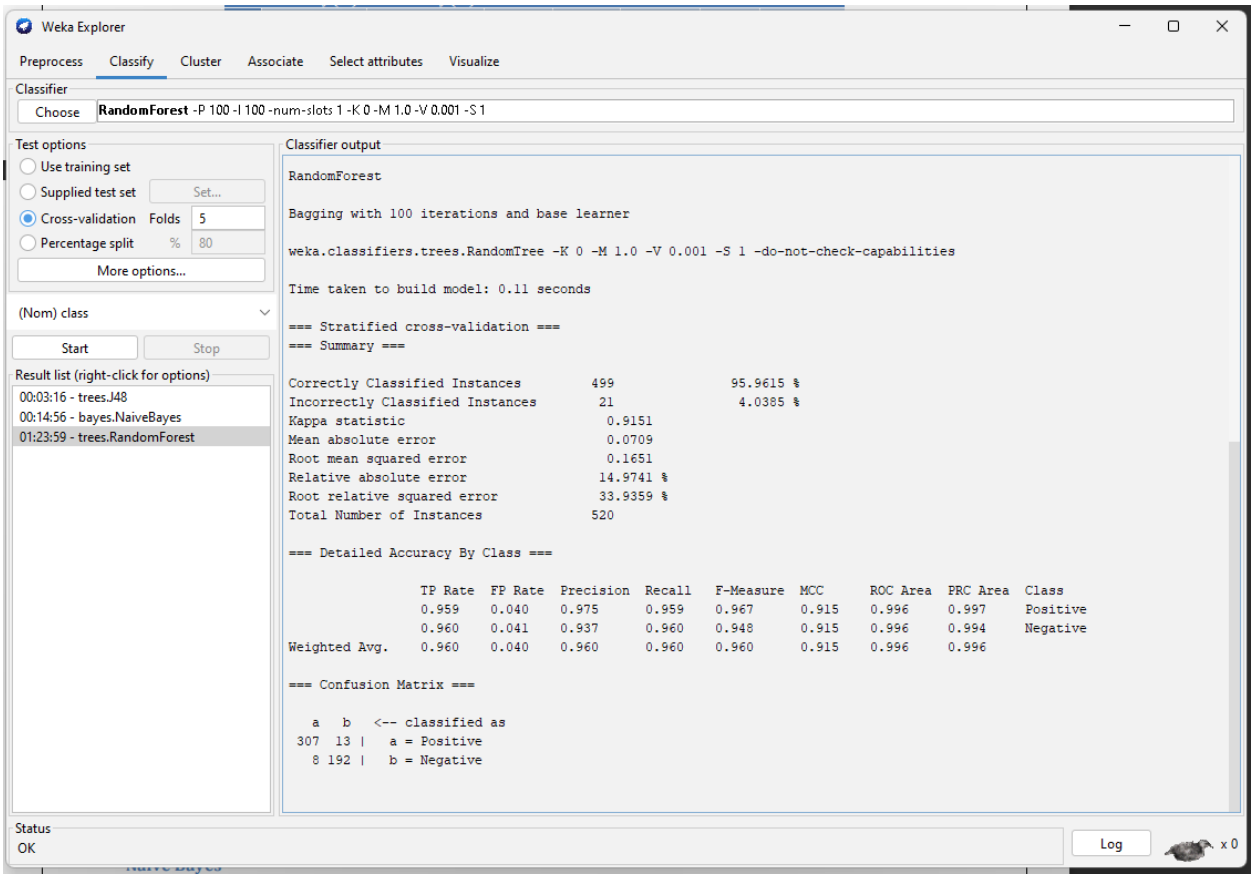


Figure 9 Random Forest data before break Ties Randomly

From the figure above:

TP = 307, FP = 8, TN = 192, FN = 13

Precision = $\frac{TP}{TP + FP} = \frac{307}{307+8} = 0.975$

Recall = $\frac{TP}{TP+FN} = \frac{307}{307+13} = 0.859$

Accuracy = $\frac{TP+TN}{TP+FP+TN+FN} = \frac{307+192}{307+192+13+8} = 0.959 = 95.96\%$

F1-Score = $\frac{2pr}{p+r} = \frac{2*0.975*0.959}{0.975+0.959} = 0.967$

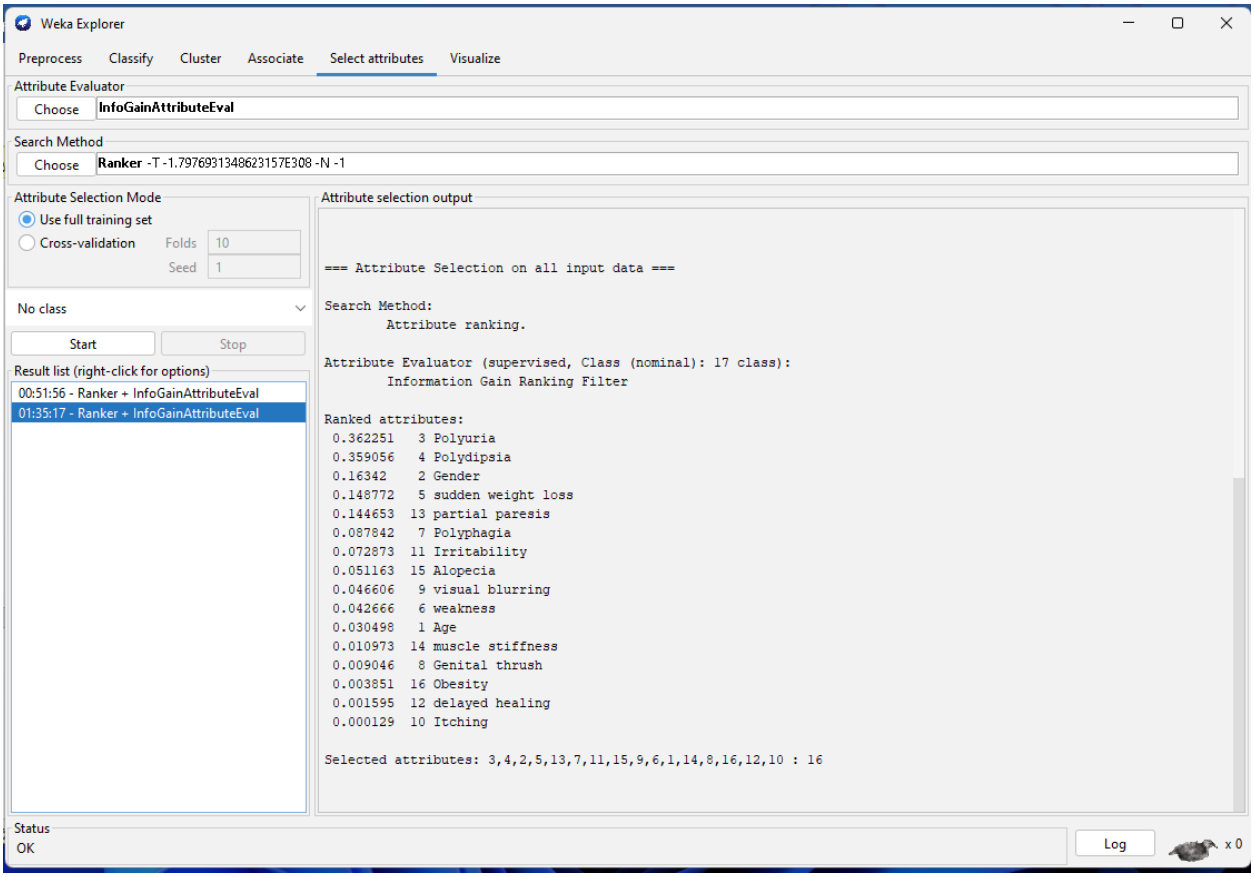


Figure 10 Random Forest info gain

Information gain is 0.362 for Polyuria so that it was on the top of the decision tree.

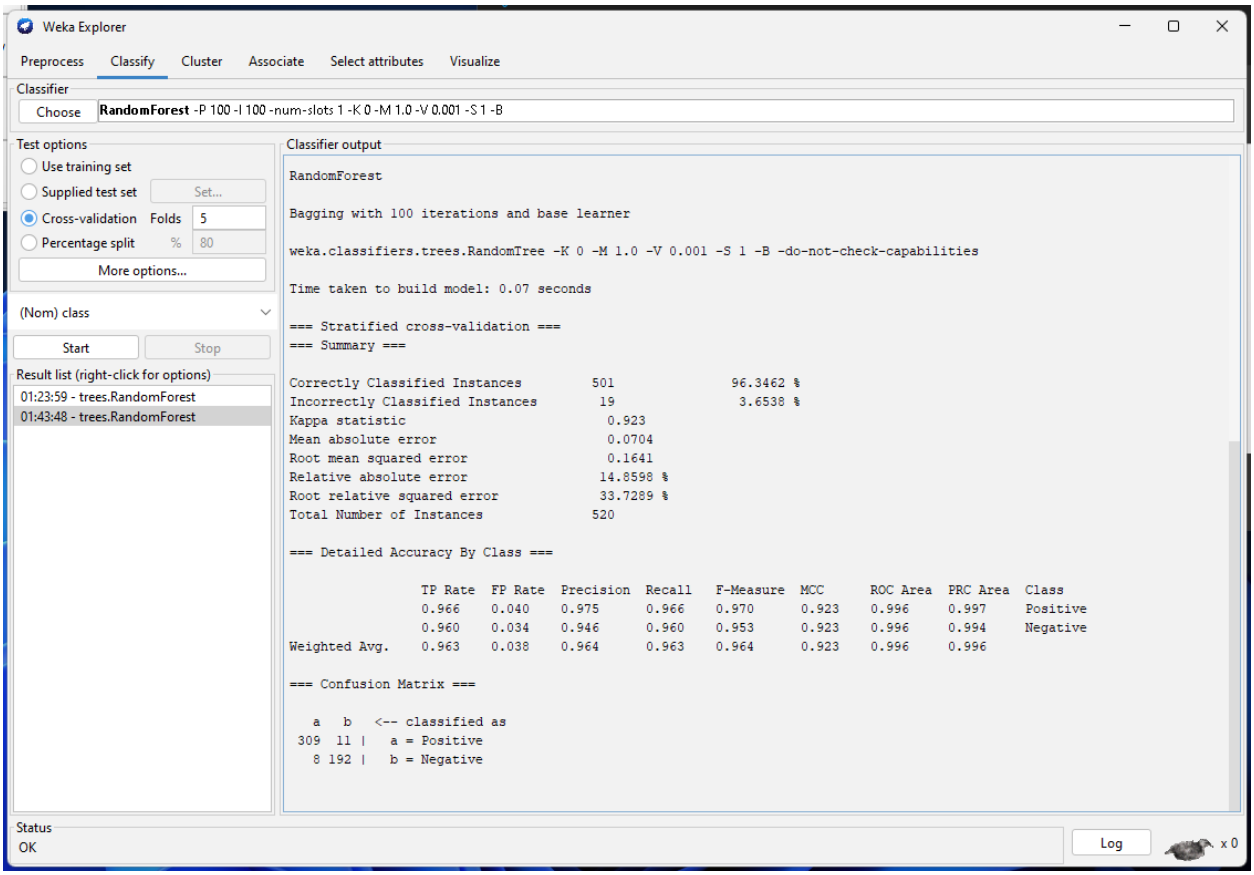


Figure 11 Random Forest data after break Ties Randomly

After changing the break Ties Randomly to true, the accuracy increased from 95.9% to 96.4%

Conclusion

When the training set is increased, the testing set produces significantly more good outcomes. Also, if we increase the number of examples, the outcomes improve, but the amount of space required grows. Consequently, the space in the ANN learning process would grow drastically, therefore we so we didn't use it in our project. We used decision tree, Naïve Bayes and Random Forest. In all test instances, the accuracy was greater than 87 percent and were fast to calculate and there attached data were good like recall, precision and F1.

Random forest algorithm avoids and prevents overfitting by using multiple trees rather than Decision Tree that always has a scope for overfitting. But Decision Tree is easier to visualize. The Random Forest classifier performed better than the Naïve Bayes method by reaching a 97.82% of accuracy. Though the Random Forest is the best of the 3 algorithms we used in this project

References

Doctor slides

Machine Learning Models - Javatpoint