Practical: 1

Aim: Installation of WEKA Tools.

Code:

Data Mining tools

- Data Mining is the set of techniques that utilize specific algorithms, statical analysis, artificial intelligence, and database systems to analyses data from different dimensionsand perspectives.

- Picturing data mining as a machine, the raw data becomes the input, the data mining activity becomes the task the machine is designed to do and the output from the machine is actionable data, in other words, data that can be used to take strategic or tactical decisions, positively impacting the bottom line.

- So, what does the machine itself in the below figure represent? The machine in this oversimplified model is thetool that is used to execute the various methods and techniques used in data mining.

- Different types of Data Mining Tools are:

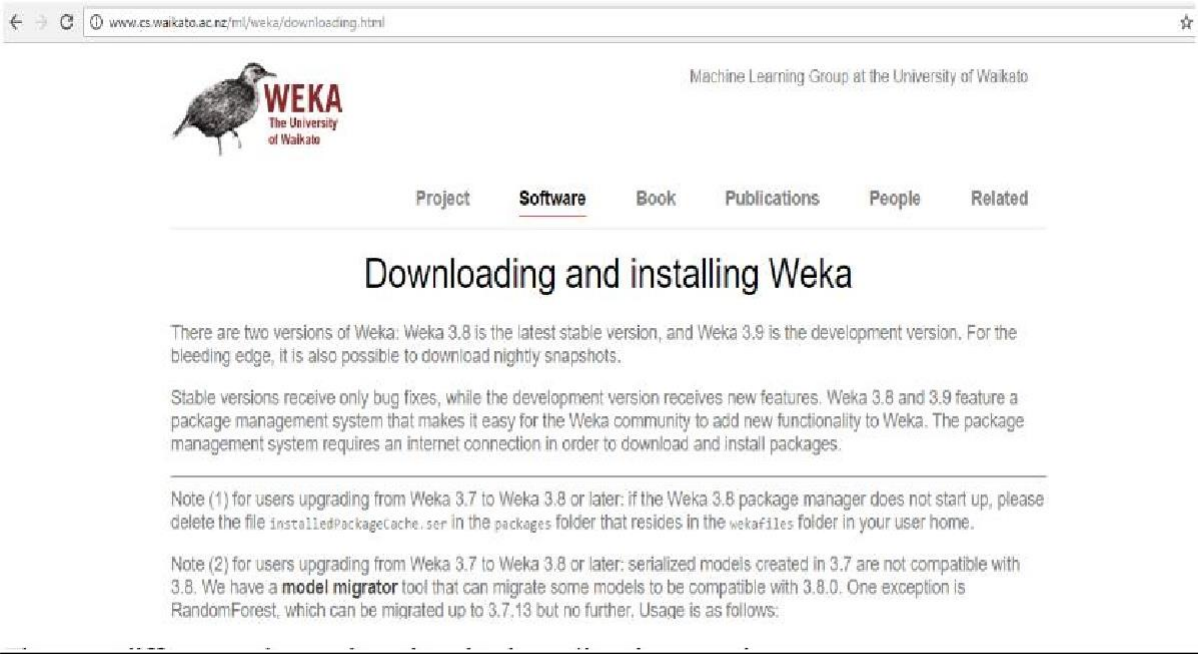
1. Rapid Miner
2. Oracle Data Mining
3. IBM SPSS Modeler
4. Knime
5. Python
6. Orange
7. Kaggle
8. Rattle
9. Weka
10. Teradata
11. H2O
12. Apache Spark
13. Sisense
14. Xplenty

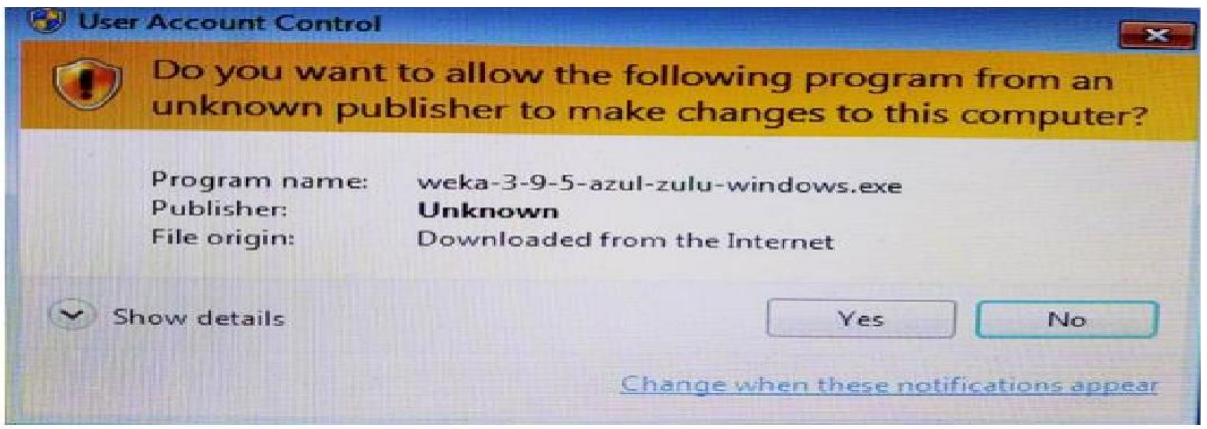
WEKA

(Waikato Environment for Knowledge Learning**)**

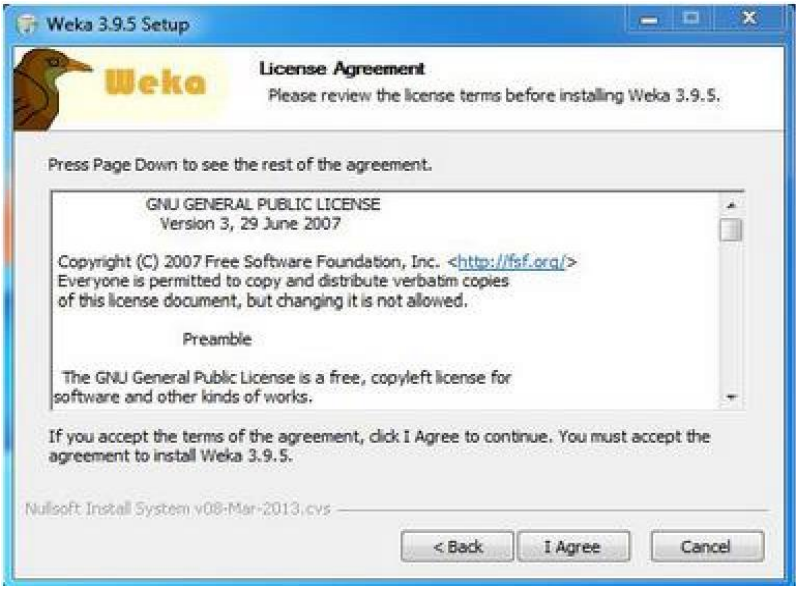
- Everybody talks about Data Mining and Big Data nowadays. Weka is a powerful, yet easy to use tool for machine learning and data mining.
- This course provides a deeper account of data mining tools and techniques.
- The emphasis is on principles and practical data mining using Weka, rather than mathematical theory or advanced details of particular algorithms.
- Students will work with multimillion-instance datasets, classify text, experiment with clustering, association rules, etc.

❖ Step to Install WEKA:

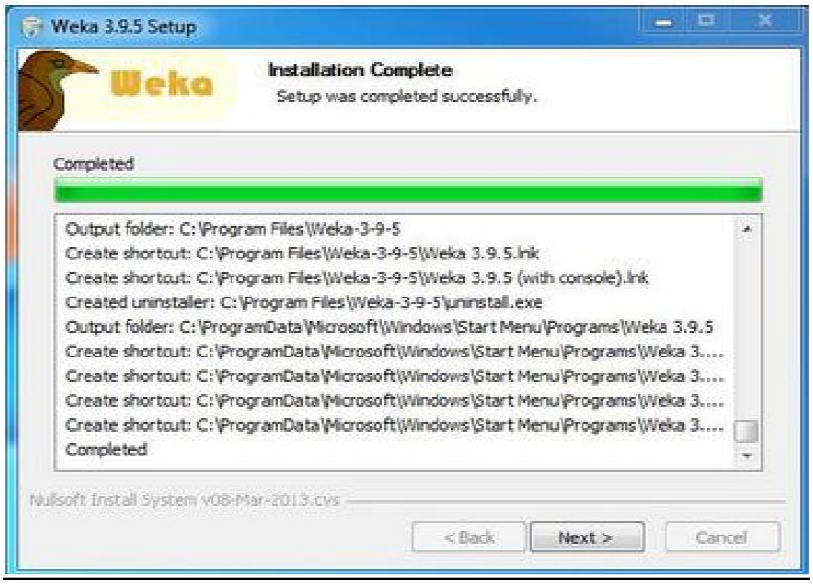





Weka 3.9.5 Setup
Welcome to the Weka 3.9.5 Setup Wizard
This wizard will guide you through the installation of Weka 3.9.5.
It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.
Click Next to continue.
Next >
Cancel


Weka 3.9.5 Setup
Weka
License Agreement
Please review the license terms before installing Weka 3.9.5.
Press Page Down to see the rest of the agreement.
GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007
Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.
Preamble
The GNU General Public License is a free, copyleft license for
software and other kinds of works.
If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install Weka 3.9.5.
Nullsoft Install System v08-Mar-2013.cvs
< Back
I Agree
Cancel

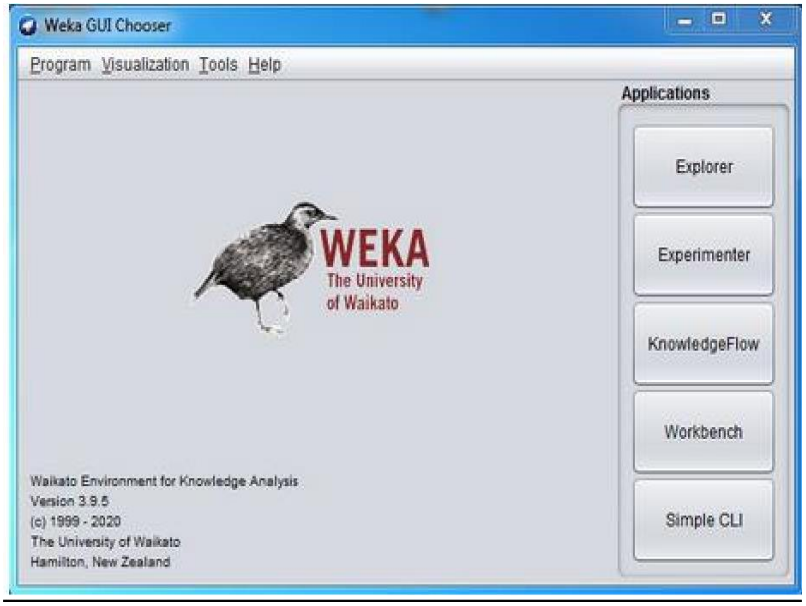

Weka 3.9.5 Setup
Weka
Choose Install Location
Choose the folder in which to install Weka 3.9.5.
Setup will install Weka 3.9.5 in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.
Destination Folder
C:\Program Files\Weka-3-9-5
Browse...
Space required: 301.0MB
Space available: 59.3GB
Nullsoft Install System v08-Mar-2013.cvs
< Back
Next >
Cancel


Weka 3.9.5 Setup
Weka
Installation Complete
Setup was completed successfully.
Completed
Output folder: C:\Program Files\Weka-3-9-5
Create shortcut: C:\Program Files\Weka-3-9-5\Weka 3.9.5.lnk
Create shortcut: C:\Program Files\Weka-3-9-5\Weka 3.9.5 (with console).lnk
Created uninstaller: C:\Program Files\Weka-3-9-5\uninstall.exe
Output folder: C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Weka 3.9.5
Create shortcut: C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Weka 3....
Create shortcut: C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Weka 3....
Create shortcut: C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Weka 3....
Create shortcut: C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Weka 3....
Completed
Nullsoft Install System v08-Mar-2013.cvs
< Back
Next >
Cancel


Weka 3.9.5 Setup
Weka
Choose Start Menu Folder
Choose a Start Menu folder for the Weka 3.9.5 shortcuts.
Select the Start Menu folder in which you would like to create the program's shortcuts. You can also enter a name to create a new folder.
Weka 3.9.5
Accessories
Administrative Tools
Adobe
AnyDesk
Flex Windows
Free Cam 8
Games
Intel PROSet Wireless
Internet Download Manager
IP Messenger for Win
Java
Do not create shortcuts
Nullsoft Install System v08-Mar-2013.cvs
< Back
Install
Cancel


Weka 3.9.5 Setup
Weka
Installing
Please wait while Weka 3.9.5 is being installed.
Extract: package-summary.html
Extract: MultiScheme.html... 100%
Extract: RandomCommittee.html... 100%
Extract: RandomSubSpace.html... 100%
Extract: RandomizableFilteredClassifier.html... 100%
Extract: RegressionByDiscretization.html... 100%
Extract: Stacking.html... 100%
Extract: Vote.html... 100%
Extract: WeightedInstancesHandlerWrapper.html... 100%
Extract: package-frame.html... 100%
Nullsoft Install System v08-Mar-2013.cvs
< Back
Next >
Cancel


Weka 3.9.5 Setup
Completing the Weka 3.9.5 Setup Wizard
Weka 3.9.5 has been installed on your computer.
Click Finish to close this wizard.
< Back
Finish
Cancel


Weka GUI Chooser
Program Visualization Tools Help
Applications
Explorer
Experimenter
KnowledgeFlow
Workbench
Simple CLI
WEKA
The University of Waikato
Waikato Environment for Knowledge Analysis
Version 3.9.5
(c) 1999 - 2020
The University of Waikato
Hamilton, New Zealand

❖ Introduction to WEKA:

Weka contains a collection of visualization tools and algorithms for data analysis and predictive modelling, together with graphical user interfaces for easy access to these functions.

The original non-Java version of Weka was a Tcl/Tk front-end to (mostly third-party) modelling algorithms implemented in other programming languages, plus data preprocessing utilities in C and a make file-based system for running machine learningexperiments.

This original version was primarily designed as a tool for analyzing data from agricultural domains. Still, the more recent fully Java-based version (Weka 3), developed in 1997, is now used in many different application areas, particularly for educational purposes and research. Weka has the following advantages, such as:

- Free availability under the GNU General Public License.
- Portability, since it is fully implemented in the Java programming language and thus runs on almost any modern computing platform.
- A comprehensive collection of data preprocessing and modelling techniques.
- Ease of use due to its graphical user interfaces.

Weka supports several standard data mining tasks, specifically, data preprocessing, clustering, classification, regression, visualization, and feature selection. Input to Weka is expected to be formatted according to the Attribute-Relational File Format and filename with the .arff extension.

❖ WEKA's Features

1. Preprocess

The preprocessing of data is a crucial task in data mining. Because most of the data is raw, there are chances that it may contain empty or duplicate values, have garbage values, outliers, extra columns, or have a different naming convention. All these things degrade the results.
To make data cleaner, better and comprehensive, WEKA comes up with a comprehensive set of options under the *filter* category. Here, the tool provides both supervised and unsupervised types of operations.

2. Classification

Classification is one of the essential functions in machine learning, where we assign classes or categories to items. The classic examples of classification are: declaring a brain tumor as "malignant" or "benign" or assigning an email to "spam" or "not_spam" class.
After the selection of the desired classifier, we select test options for the training set.

3. Cluster

In clustering, a dataset is arranged in different groups/clusters based on some similarities. In this case, the items within the same cluster are identical but different from other clusters. The examples of clustering include, but are not limited to, identifying customers with similar behaviors, organizing the regions according to homogenous land use.
When we apply the EM algorithm, the tool shows the mean and standard deviation of the clusters and attributes:

4. Associate

Association rules highlight all the associations and correlations between items of a dataset. In short, it is an *if-then* statement that depicts the probability of relationships between data items. A classic example of association refers to a connection between the sale of milk and bread.

5. Visualize

In the visualize tab, different plot matrix and graphs are available to show the trends and errors identified by the model.

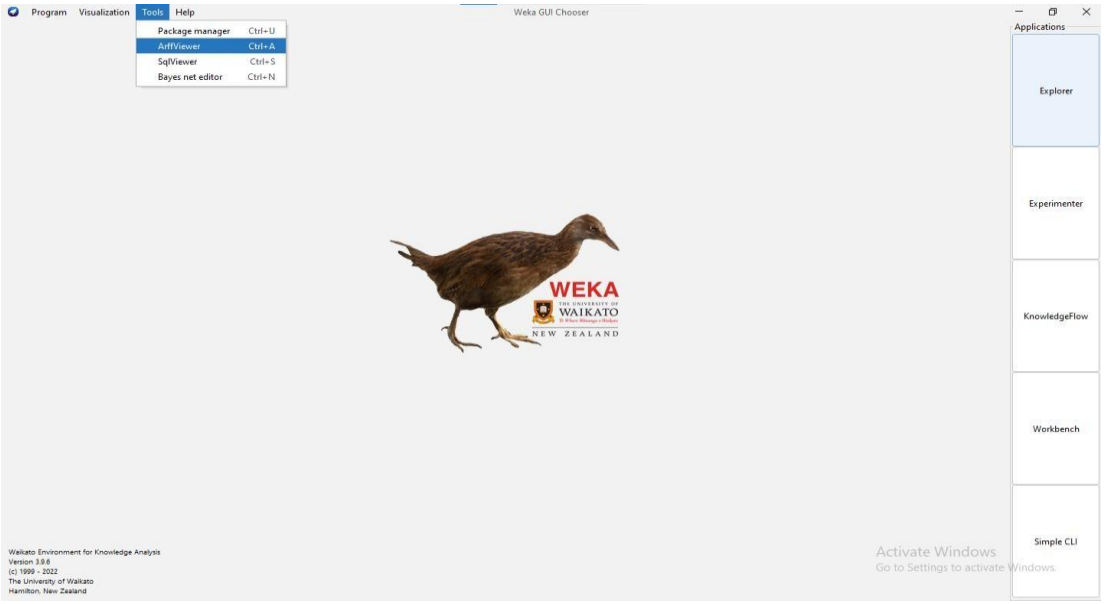
Practical: 2

Aim: Create your own excel file. Convert the excel file to .csv formatand prepare it as ARFF files.

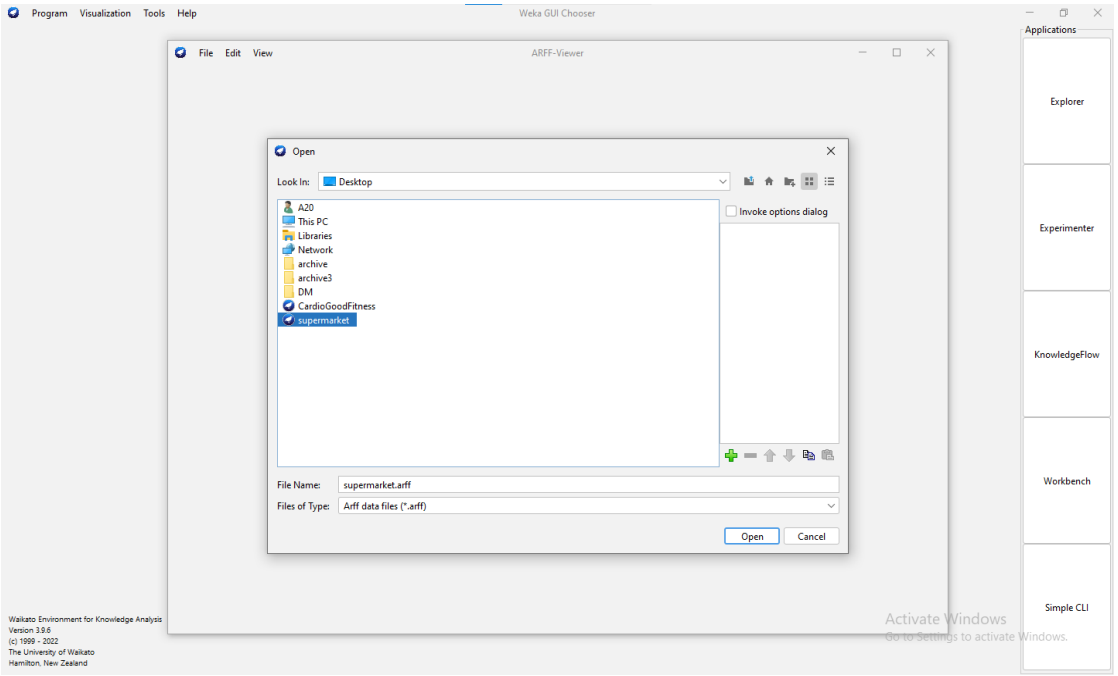
Code:

Steps:

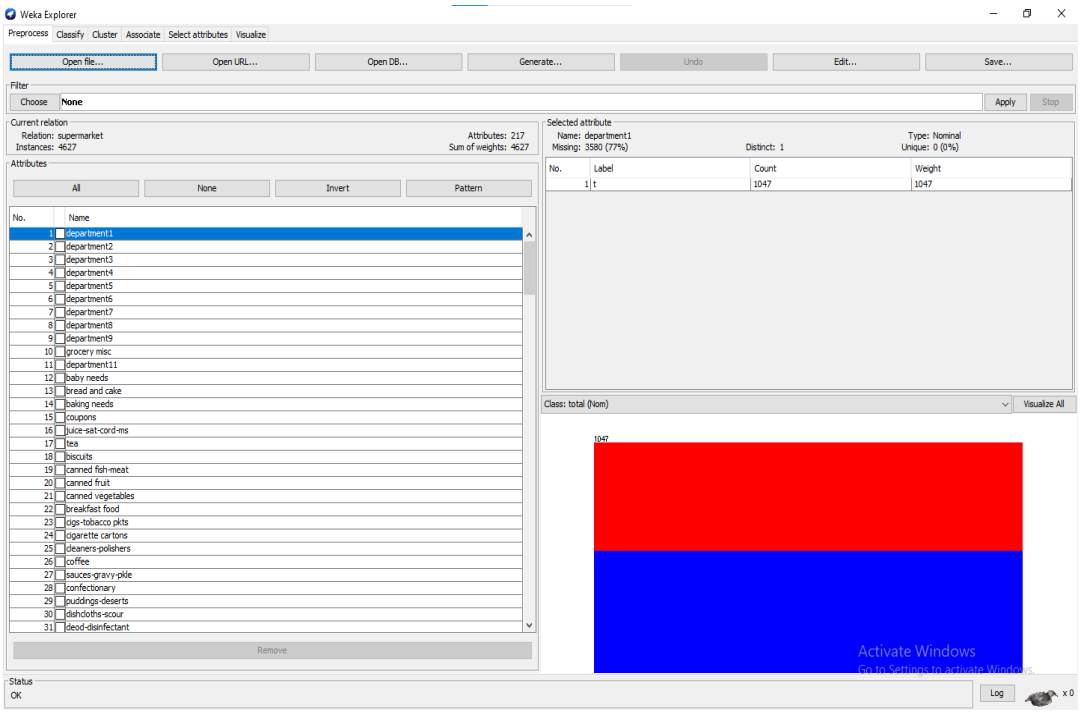
1) **Open WEKA**
   Select Tools and ARFF viewer



2) **Select the file**



3) Open the file in weka



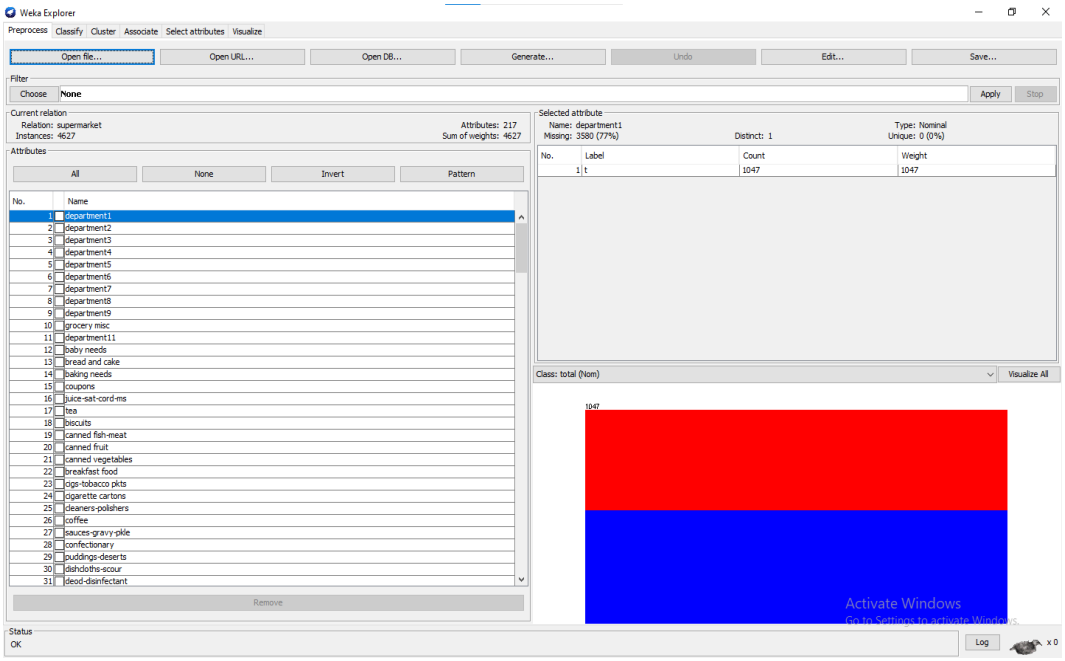
Practical: 3

Aim: Preprocess and classify your dataset.

Code:

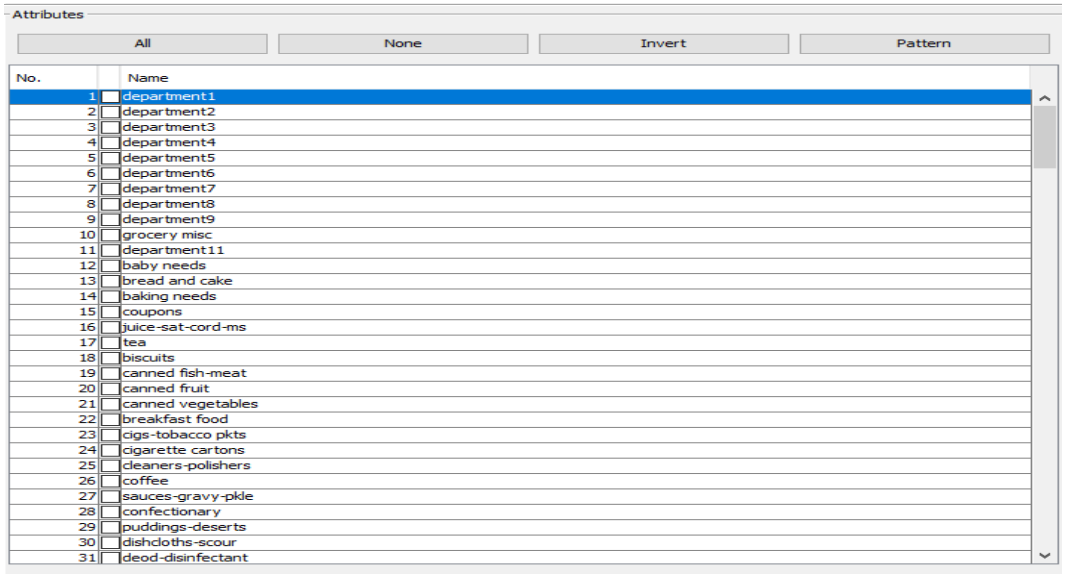
Steps:

1) **Open data in WEKA**

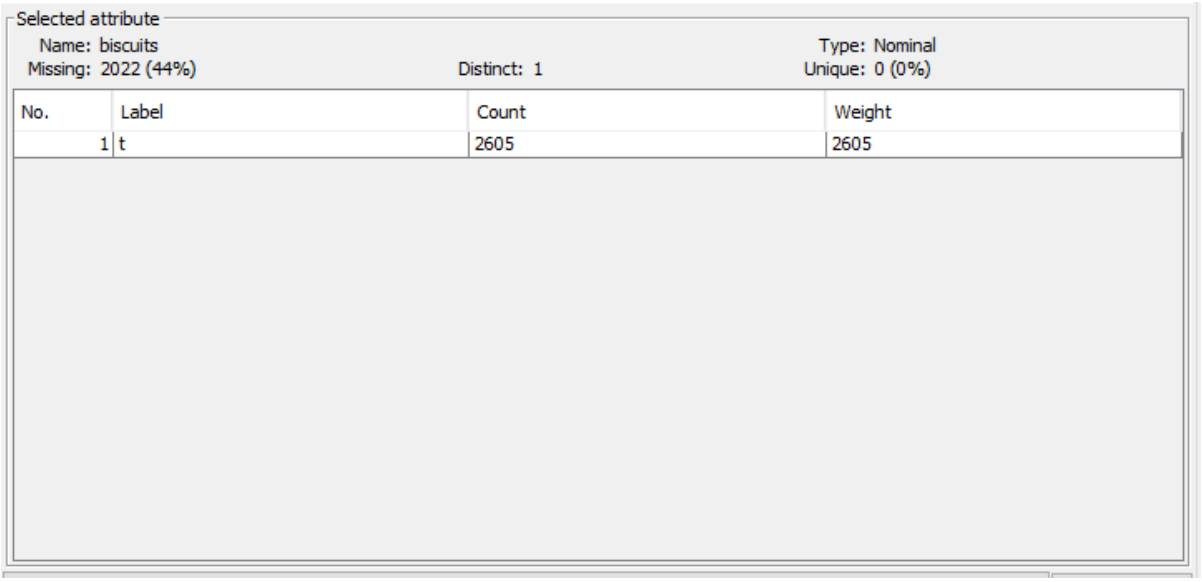


2) **Understanding Data**

Let us first look at the highlighted Current relation sub window. It shows thename of the database that is currently loaded. You can infer two points fromthis sub window −On the left side, notice the Attributes sub window that displays the various fields in the database.

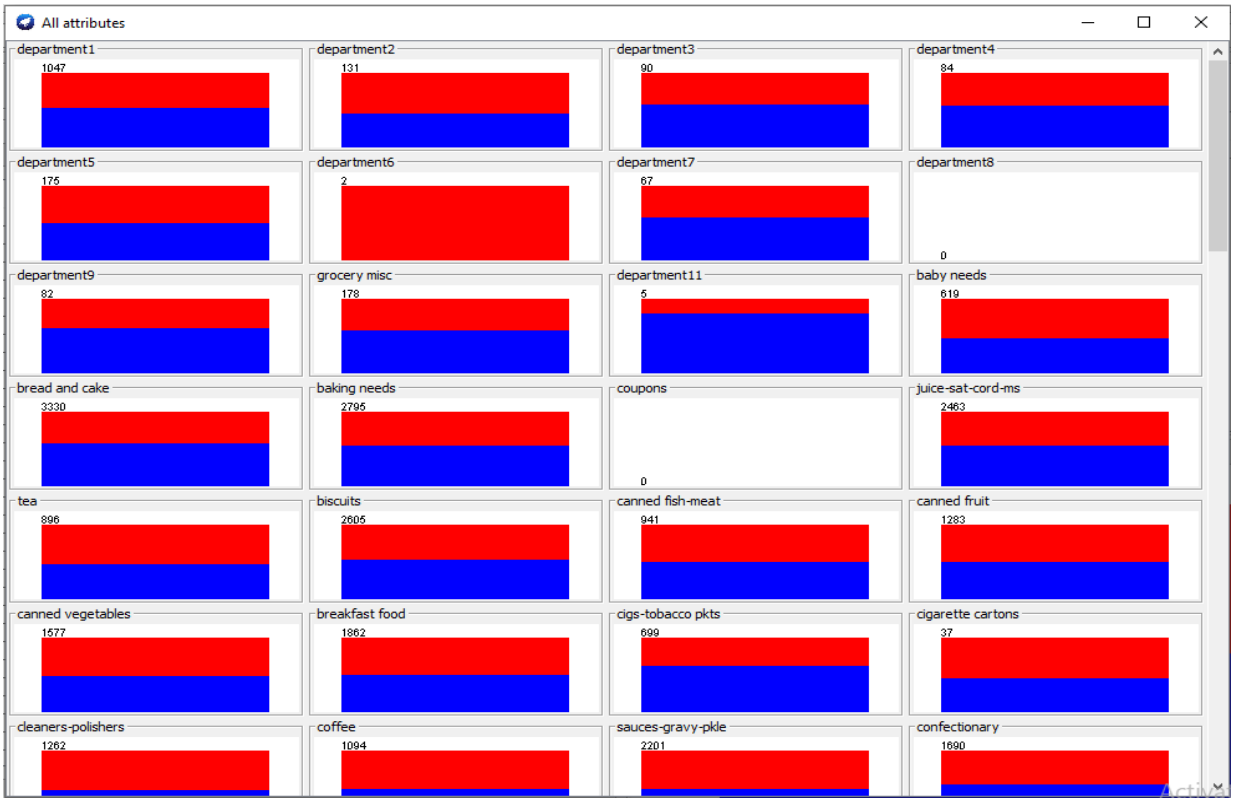


3) In the Selected Attribute sub window, you can observe the following −



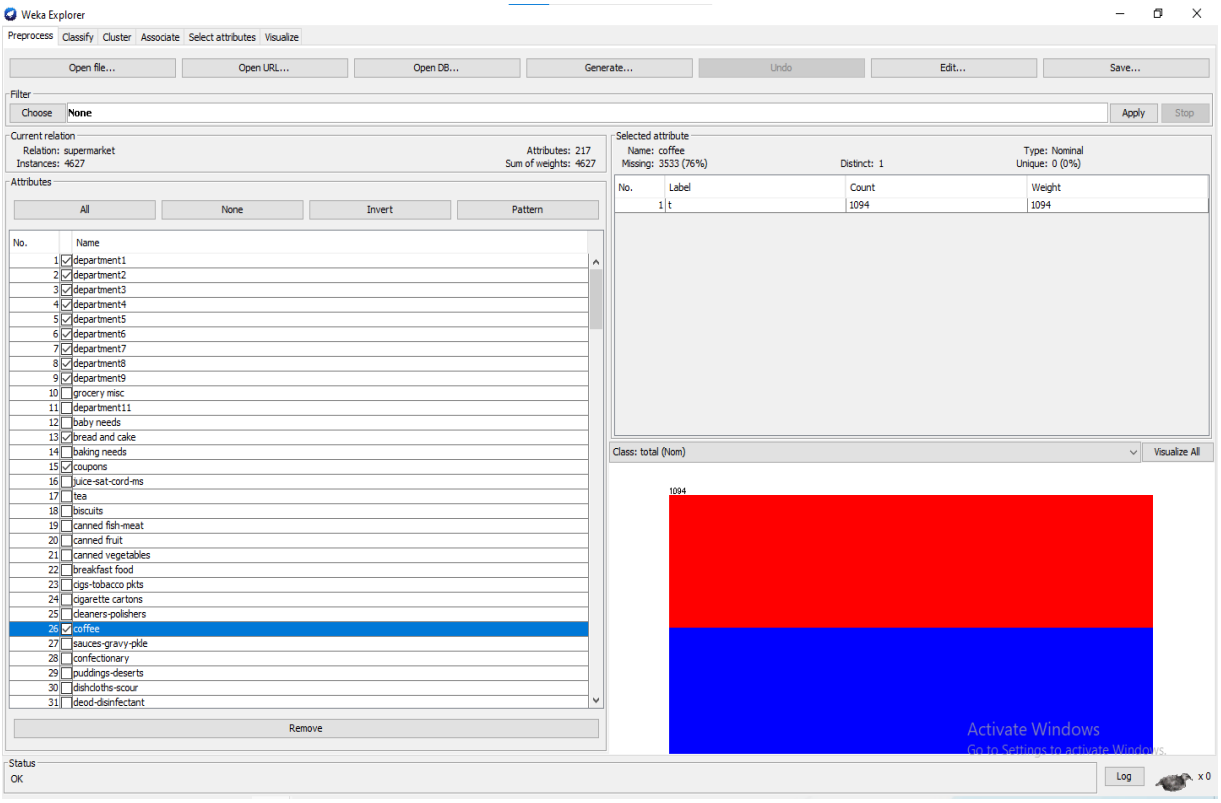
4) At the bottom of the window, you see the visual representation of the classvalues.

If you click on the Visualize All button, you will be able to see all features inone single window as shown here –



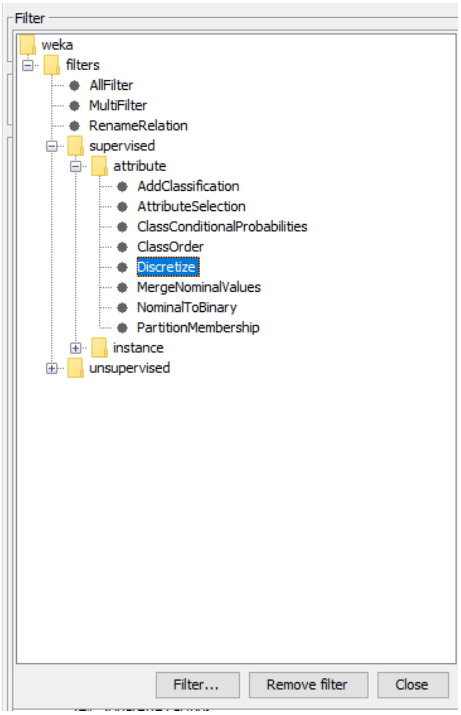
5) Removing Attributes

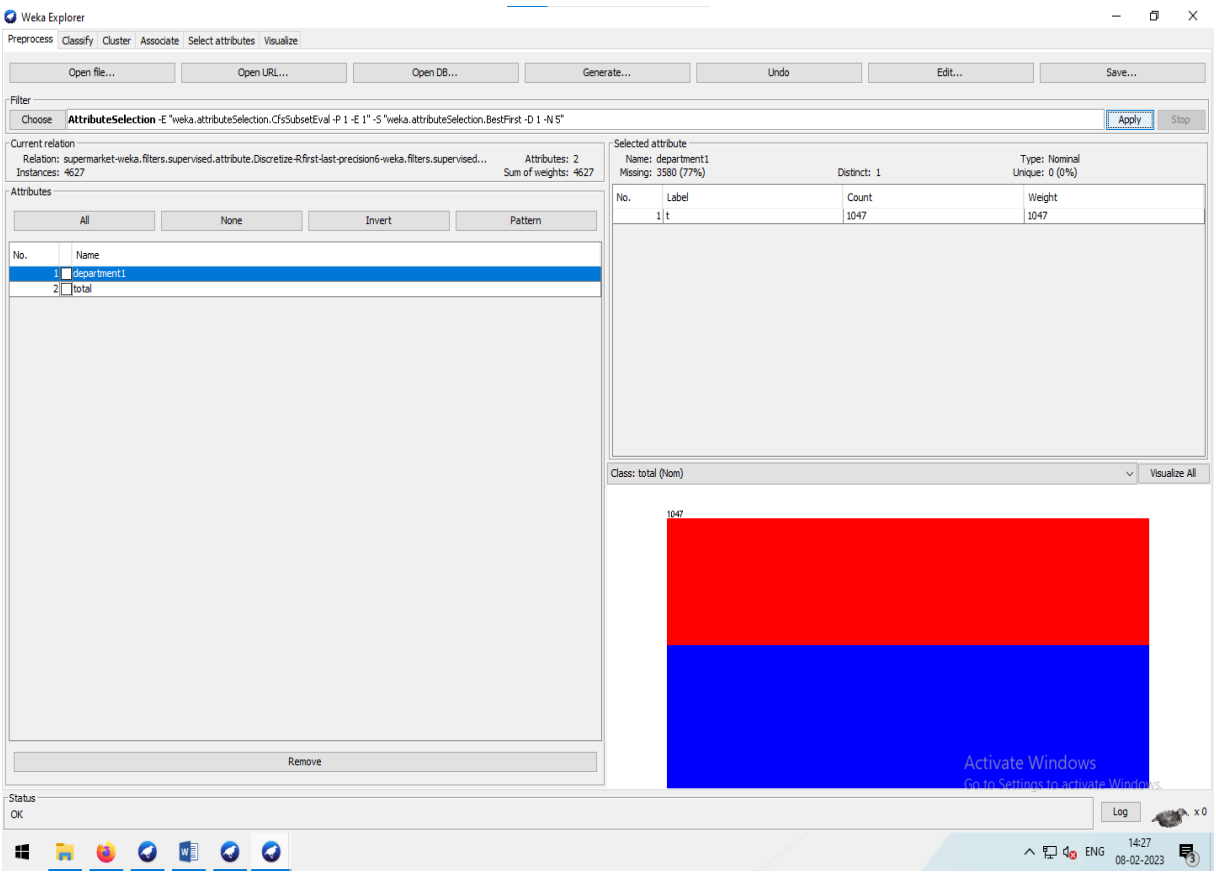
Many a time, the data that you want to use for model building comes with manyirrelevant fields. For example, the customer database may contain his mobile number which is relevant in analyzing his credit rating.



6) Applying Filters

Some of the machine learning techniques such as association rule mining requires categorical data. We will convert these to nominal by applying a filter onour raw data. Click on the Choose button in the Filter sub window and select the following filter –


Filter
weka
filters
AllFilter
MultiFilter
RenameRelation
supervised
attribute
AddClassification
AttributeSelection
ClassConditionalProbabilities
ClassOrder
Discretize
MergeNominalValues
NominalToBinary
PartitionMembership
instance
unsupervised
Filter...
Remove filter
Close

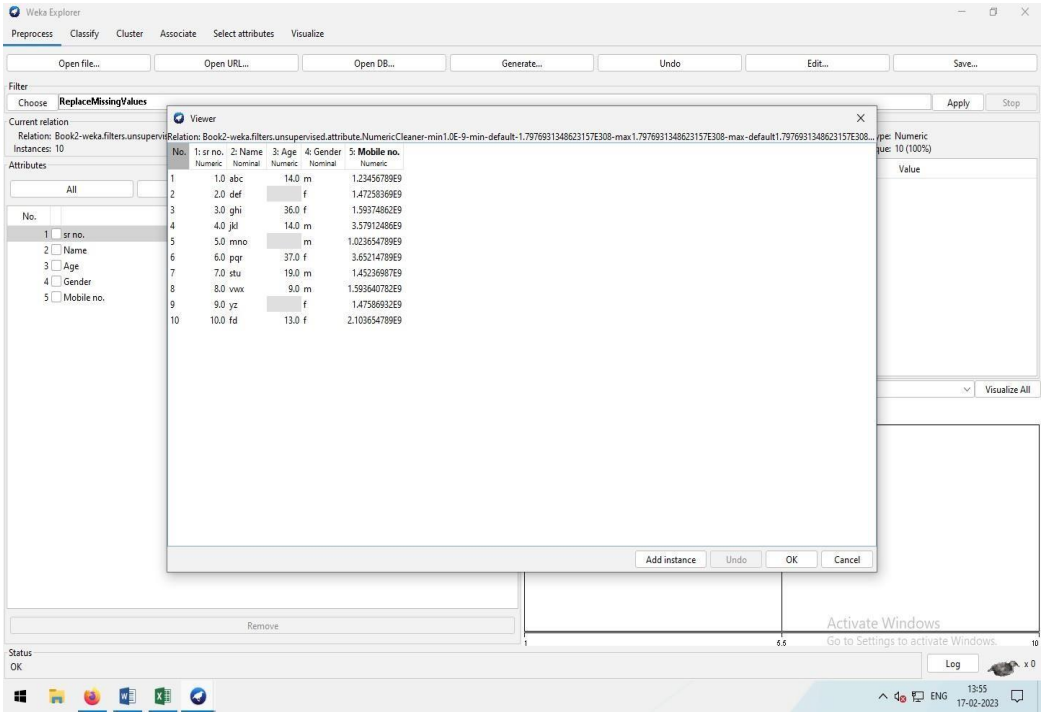

Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Open file...
Open URL...
Open DB...
Generate...
Undo
Edit...
Save...
Filter
Choose
AttributeSelection -E "weka.attributeSelection.CfsSubsetEval -P 1 -E 1" -S "weka.attributeSelection.BestFirst -D 1 -N 5"
Apply
Stop
Current relation
Relation: supermarket-weka.filters.supervised.attribute.Discretize-Rfirst-last-precision6-weka.filters.supervised...
Instances: 4627
Attributes: 2
Sum of weights: 4627
Selected attribute
Name: department1
Missing: 3580 (77%)
Distinct: 1
Type: Nominal
Unique: 0 (0%)
Attributes
All
None
Invert
Pattern
No.
Name
1 department1
2 total
No.
Label
Count
Weight
1 t
1047
1047
Class: total (Nom)
Visualize All
1047
Remove
Status
OK
Log
x 0
Activate Windows
Go to Settings to activate Windows.
ENG
14:27
08-02-2023

Practical: 4

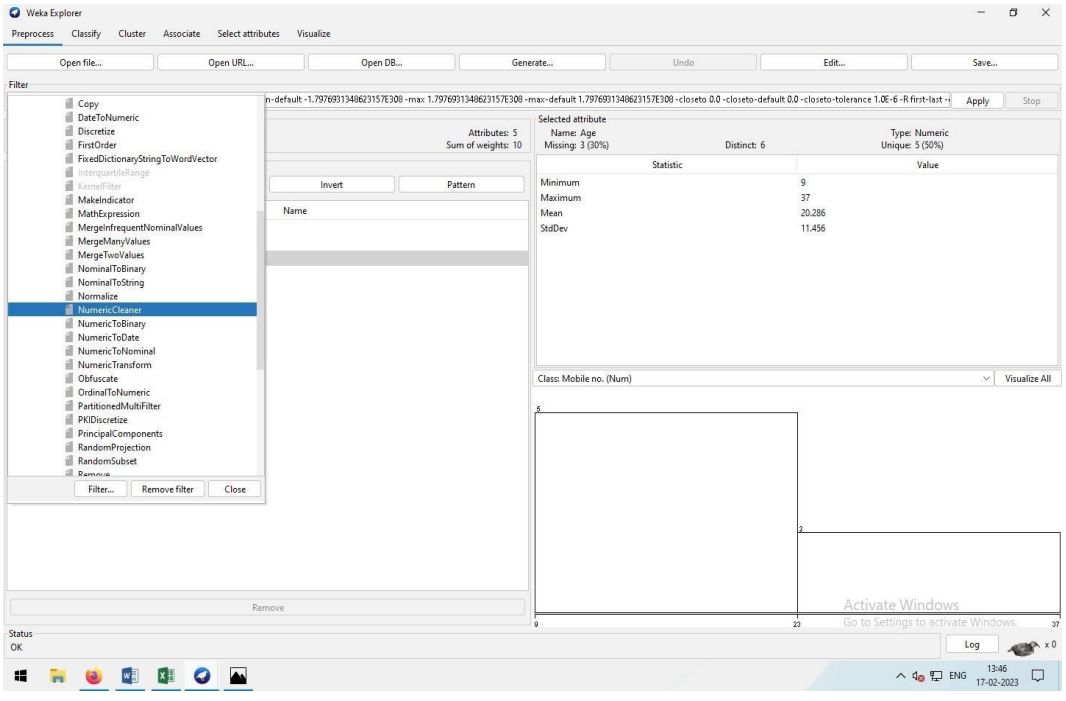
Aim: Pre-process a dataset based on handling missing values.

Code:

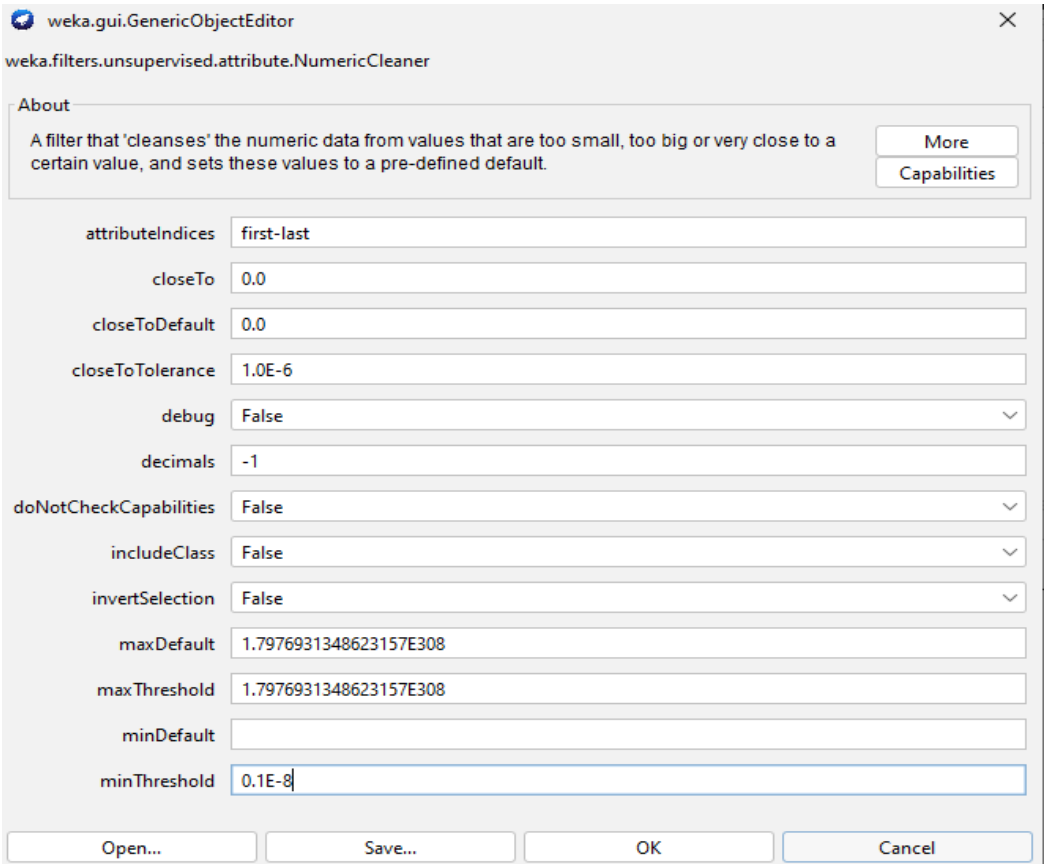
STEP:1- Load the weka onset of Book1 dataset.



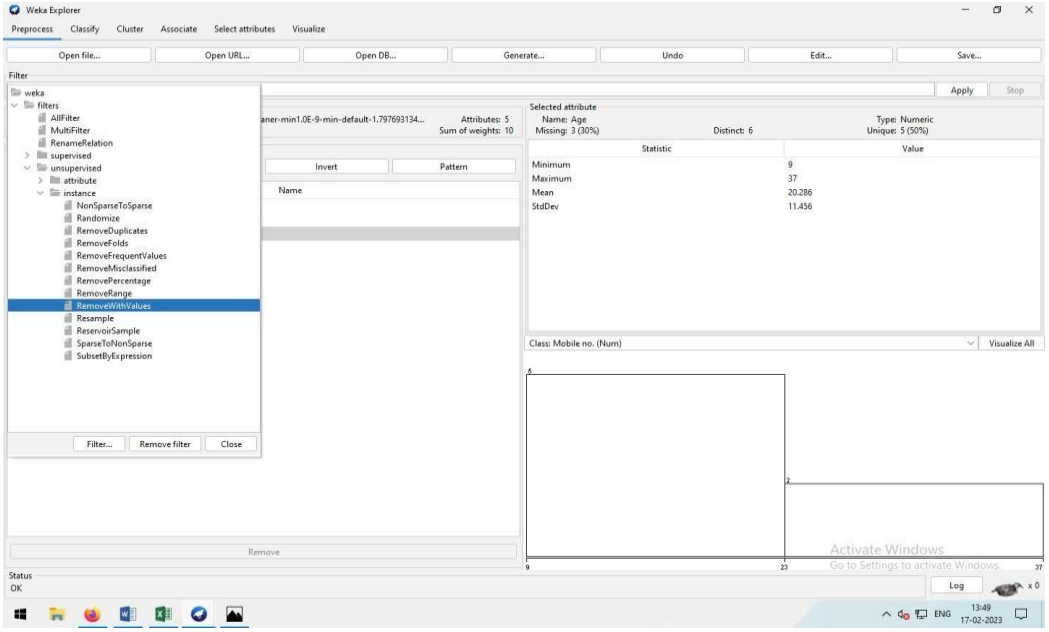
STEP:2- Click the "Choose" button for the Filter and select NumericalCleaner, it us under unsupervized.attribute.NumericalCleaner.



STEP:3- Set the *attributeIndicies* to 6, the index of the mass attribute. Set *minThreshold* to 0.1E-8 (close to zero), which is the minimum value allowed for the attribute.

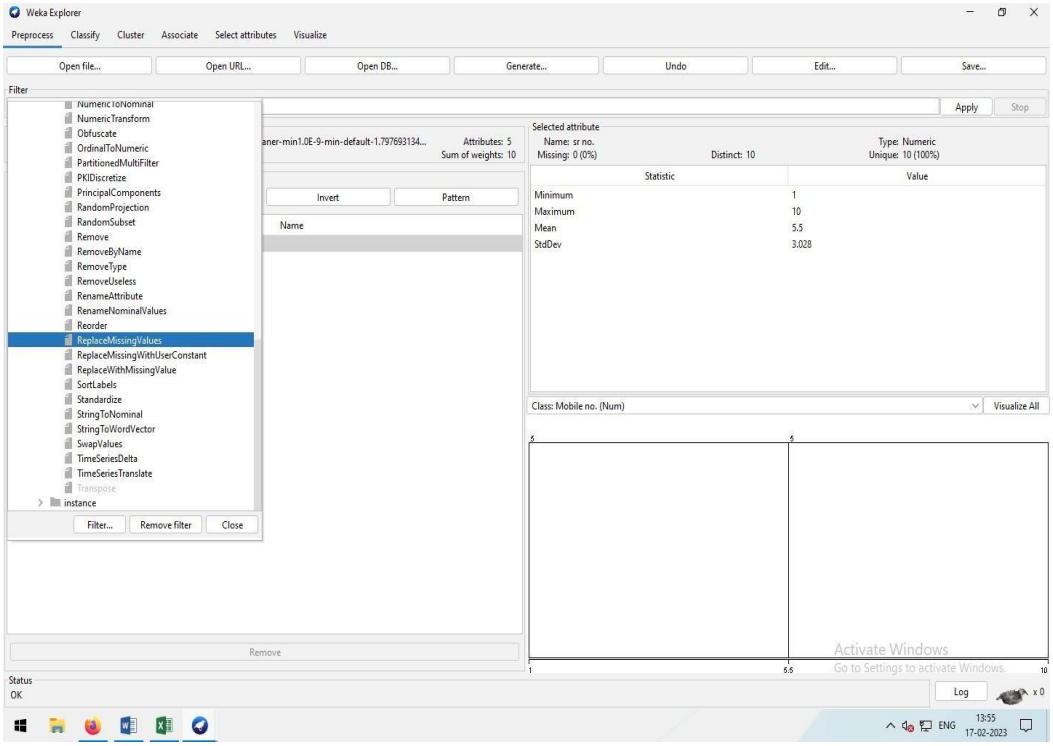


STEP:4- Click the "Choose" button for the Filter and select RemoveWithValues, it us under unsupervized.instance.RemoveWithValues.

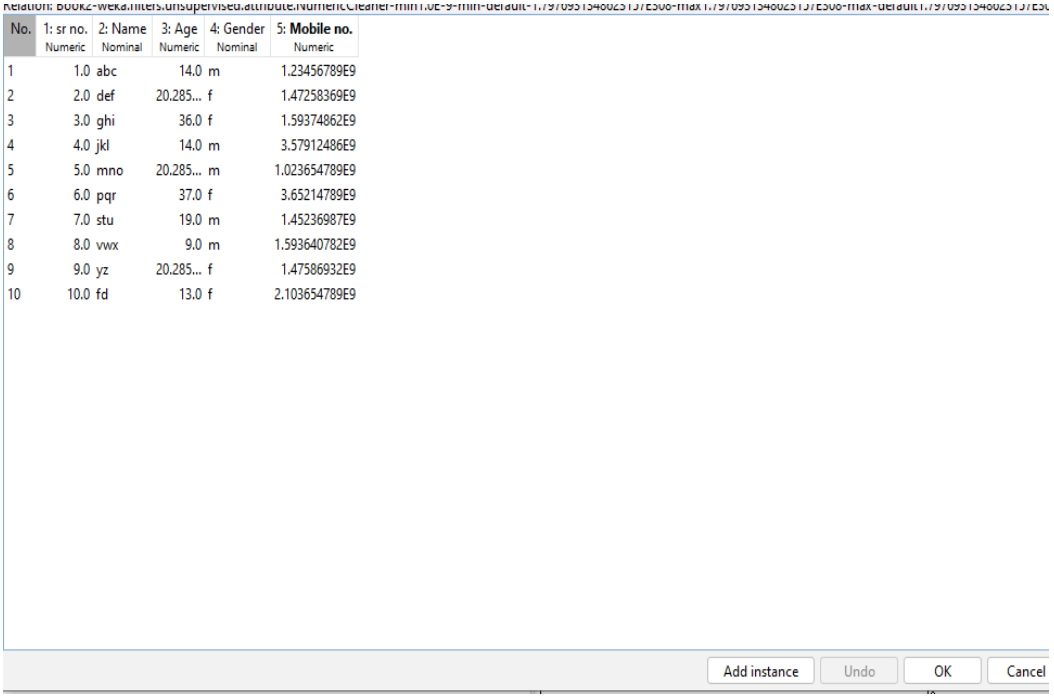


-

STEP:5- Click the "Choose" button for the Filter and select ReplaceMissingValues, it us under unsupervized.attribute.ReplaceMissingValues.



STEP:6- Click the "Apply" button to apply the filter.



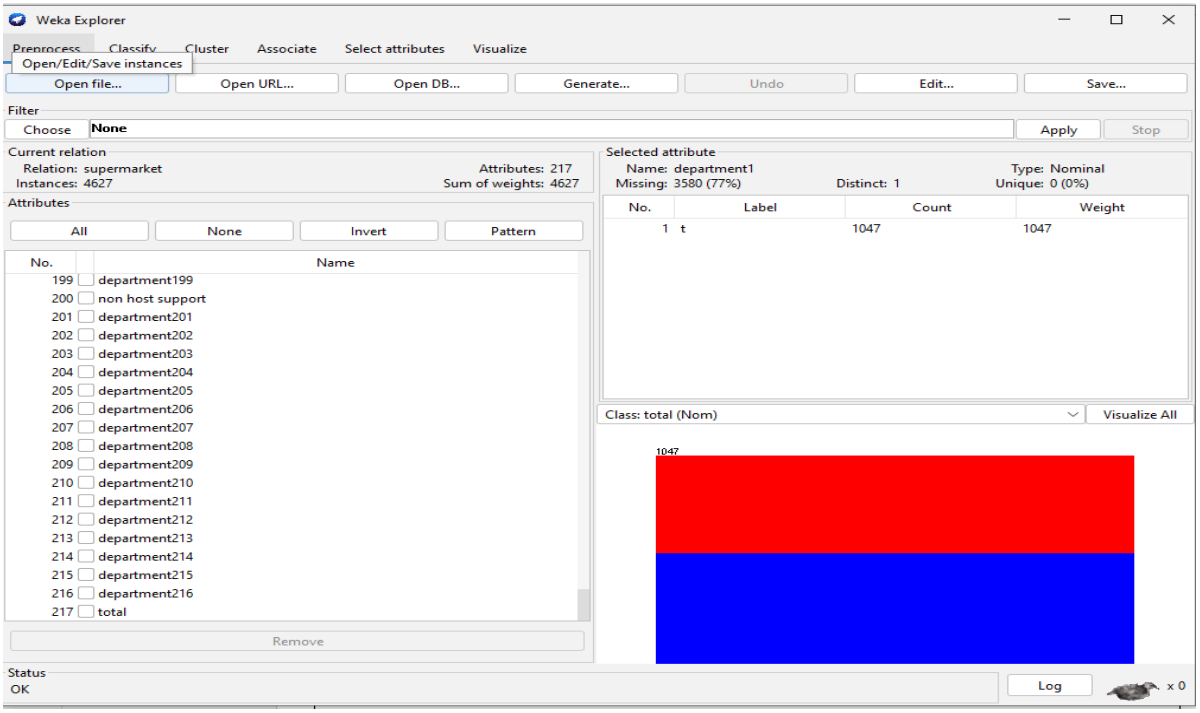
No.	1: sr no. Numeric	2: Name Nominal	3: Age Numeric	4: Gender Nominal	5: Mobile no. Numeric
1	1.0	abc	14.0	m	1.23456789E9
2	2.0	def	20.285...	f	1.47258369E9
3	3.0	ghi	36.0	f	1.59374862E9
4	4.0	jkl	14.0	m	3.57912486E9
5	5.0	mno	20.285...	m	1.023654789E9
6	6.0	pqr	37.0	f	3.65214789E9
7	7.0	stu	19.0	m	1.45236987E9
8	8.0	vwx	9.0	m	1.593640782E9
9	9.0	yz	20.285...	f	1.47586932E9
10	10.0	fd	13.0	f	2.103654789E9

Practical: 5

Aim: Generate association rules using the apriori algorithm.

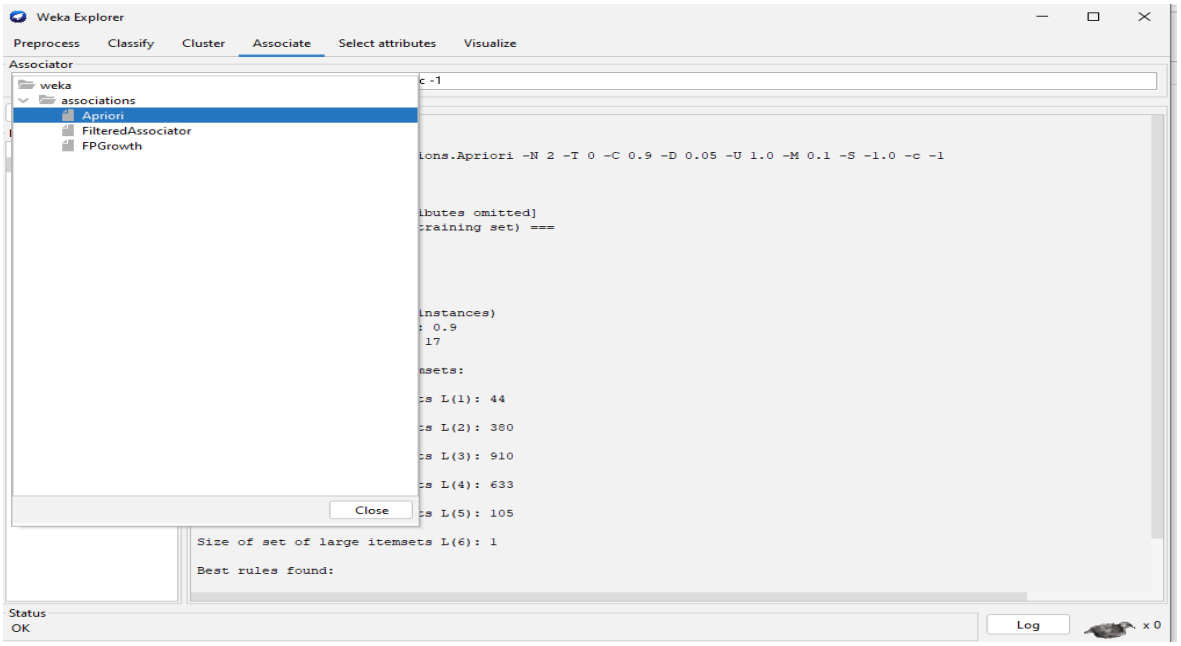
Code:

Step-1: In the WEKA open the Preprocess tab, click on the Open file button and select supermarket.arff dataset. After the data is loaded you will see the following screen

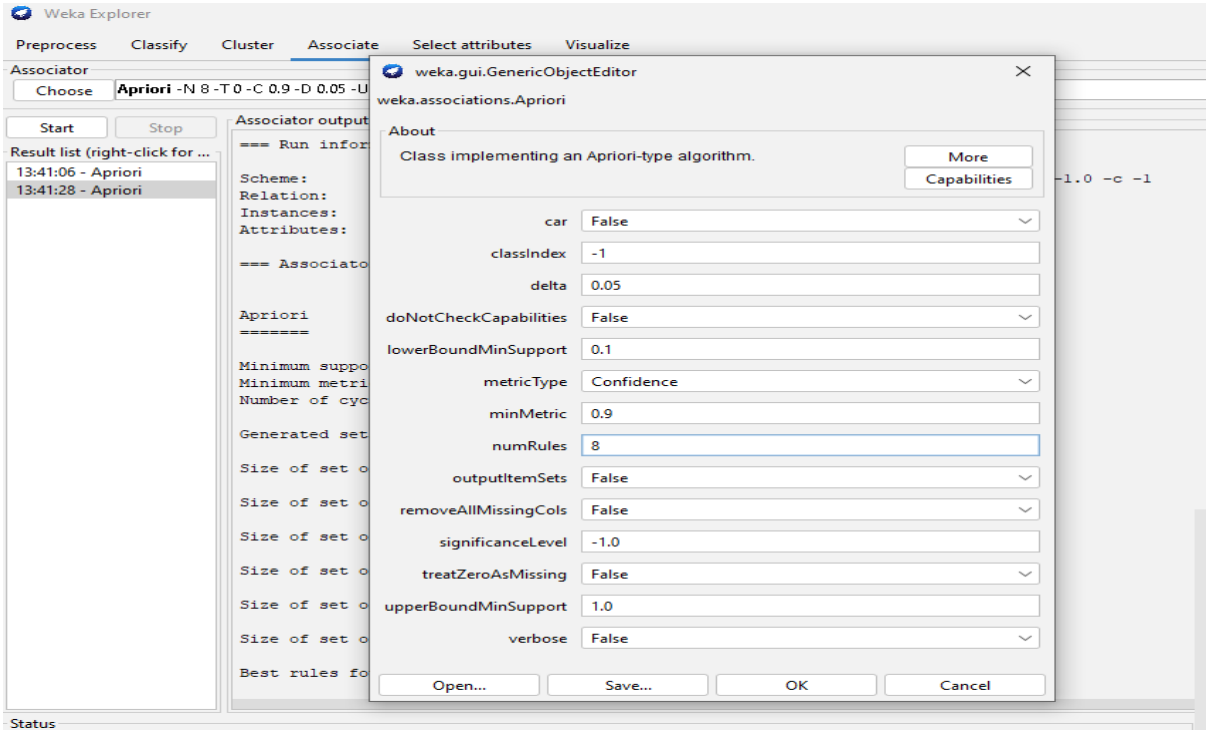


The database contains 4627 instances and 217 attributes

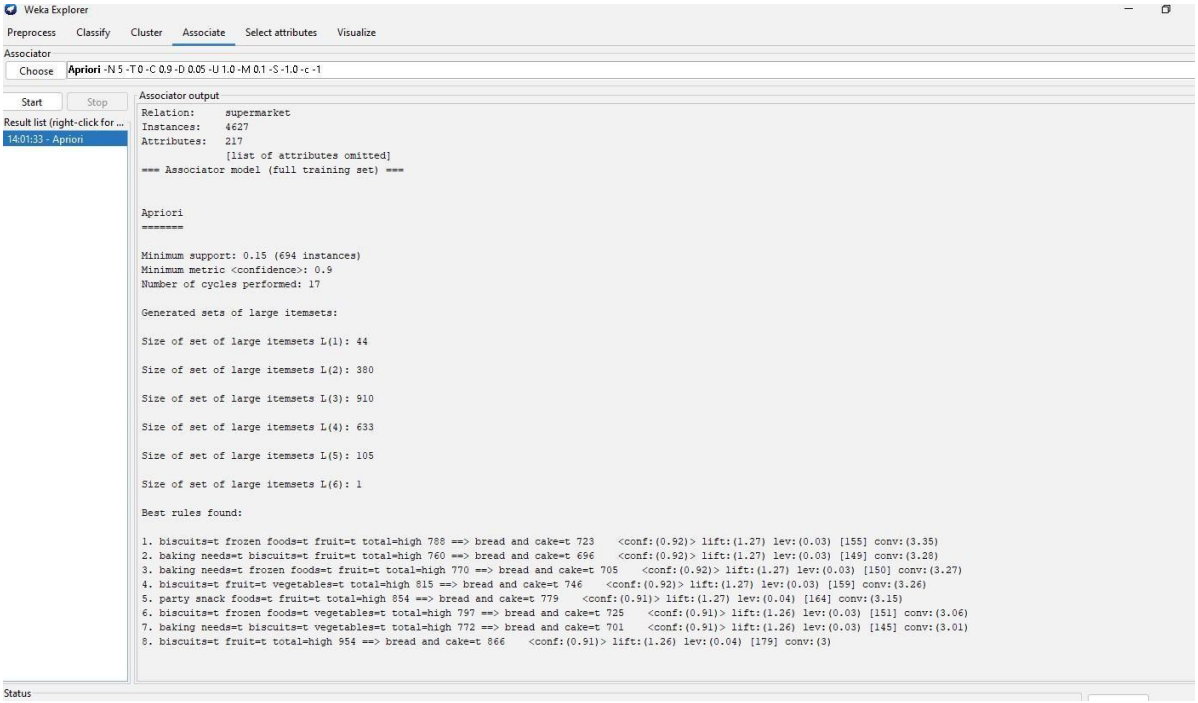
Step-2: Click on the Associate TAB and click on the Choose button. Select the Aprioriassociation as shown in below:



Step-3: set the parameters for the Apriori algorithm, click on its name, a window will pop up as shown below that allows you to set the parameters

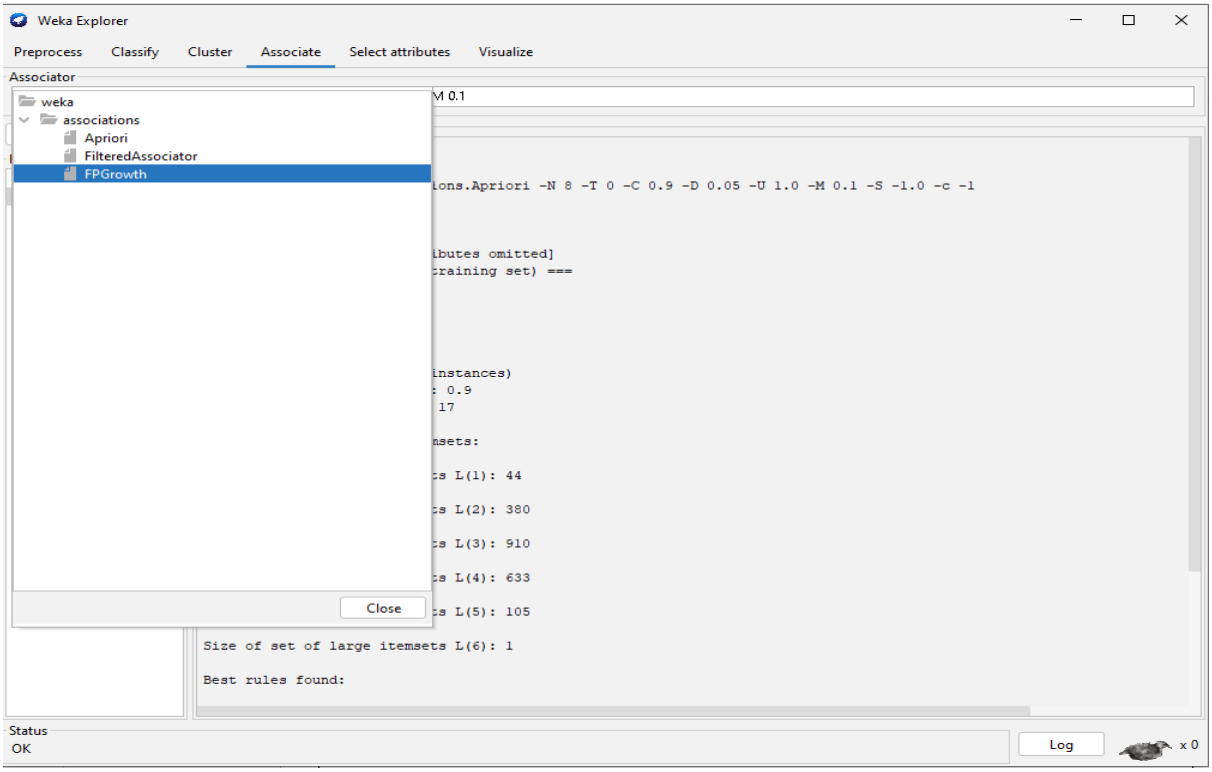


Step-4: set the parameters, click the Start button. After a while you will see the results as shown in the below.

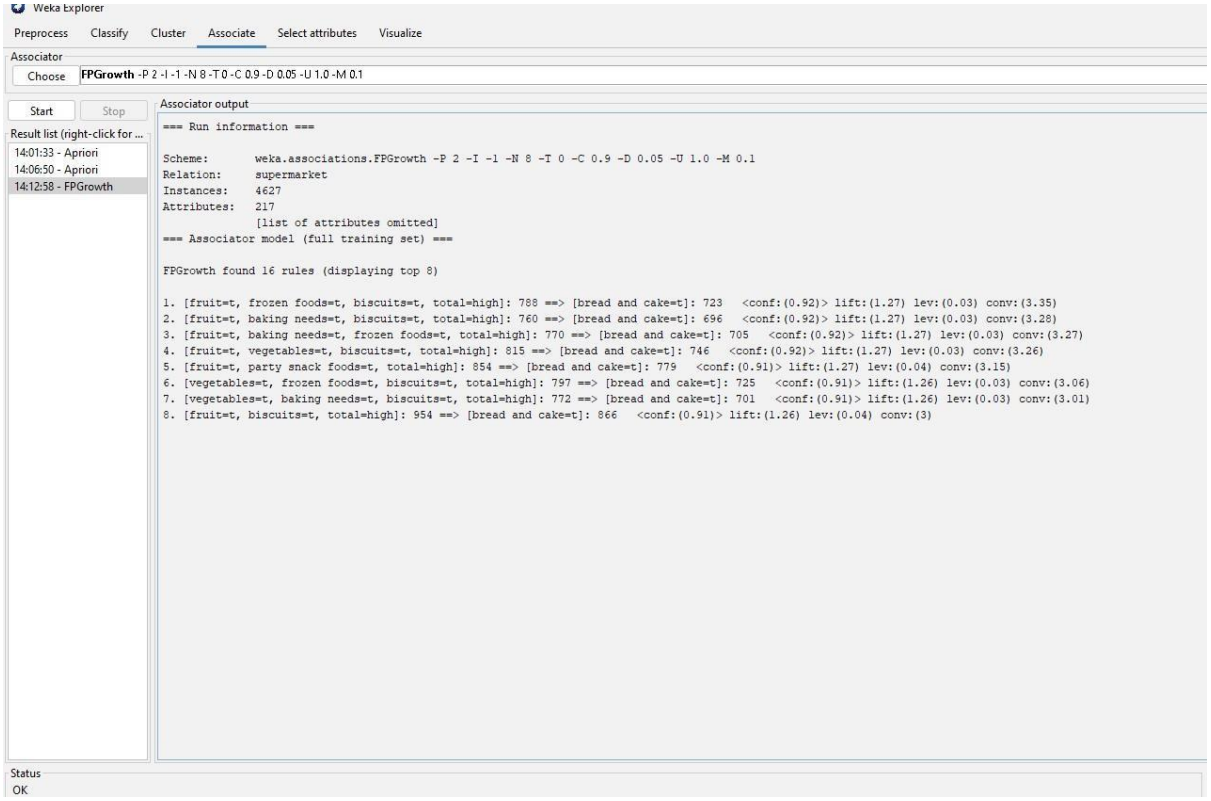


At the bottom, we will find the detected best rules of associations. This will help the supermarket in stocking their products in appropriate shelves.

Step-5: Click on the Associate TAB and click on the Choose button. Select the FPGrowthassociation as shown in below:



Step-6: set the parameters, click the Start button. After a while you will see the results as shown in the below.



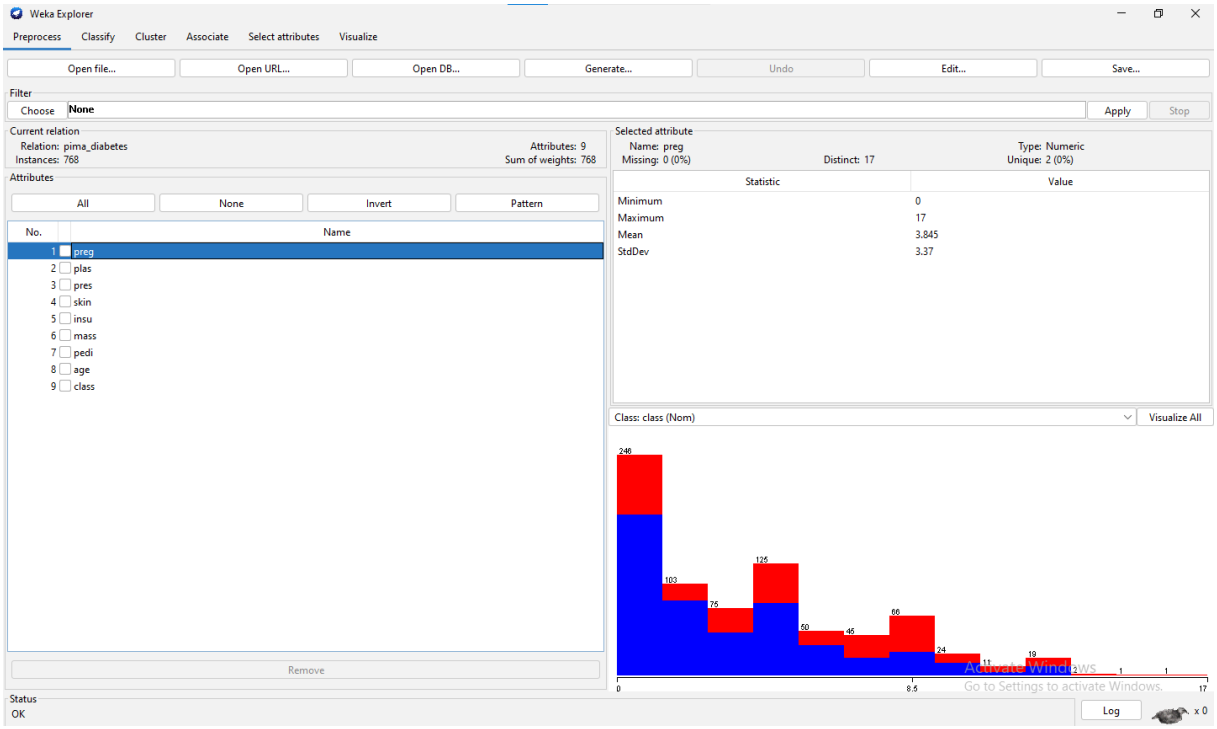
Practical: 6

Aim: Implement classifier decision tree on any two dataset.

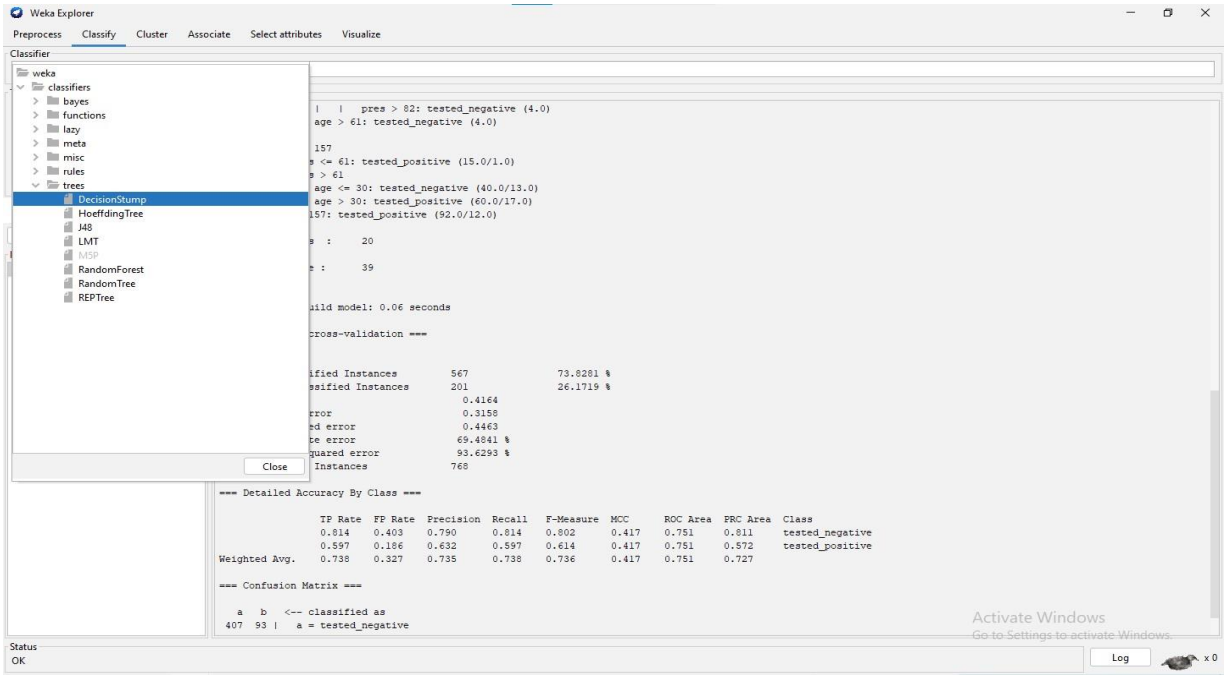
Code:

- Dataset 1

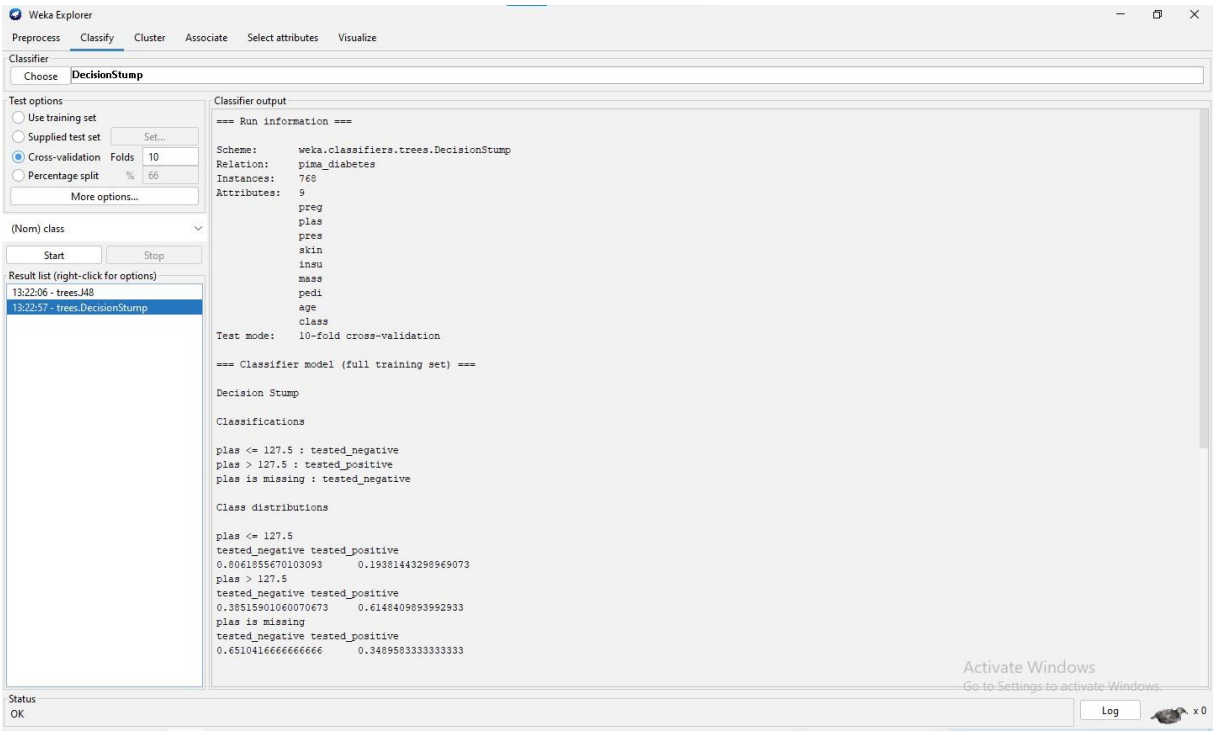
Step 1) Click on open file and load the data

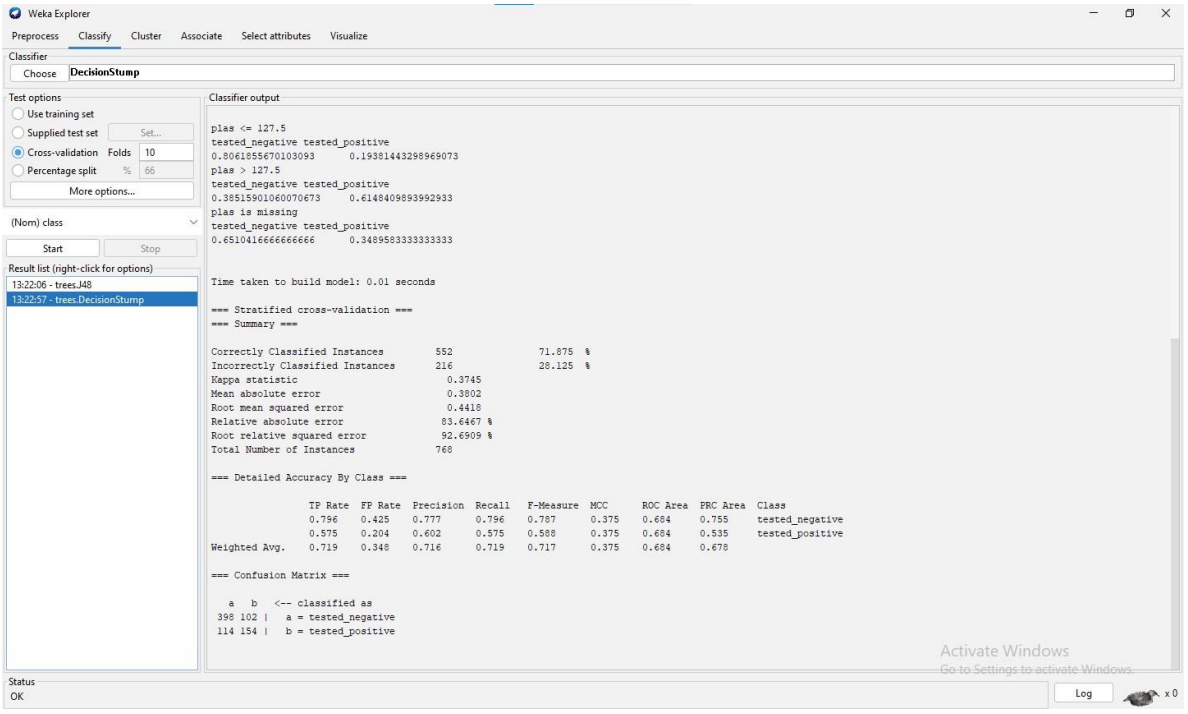


Step 2) Click on Classify and choose the algorithm of Decision tree.

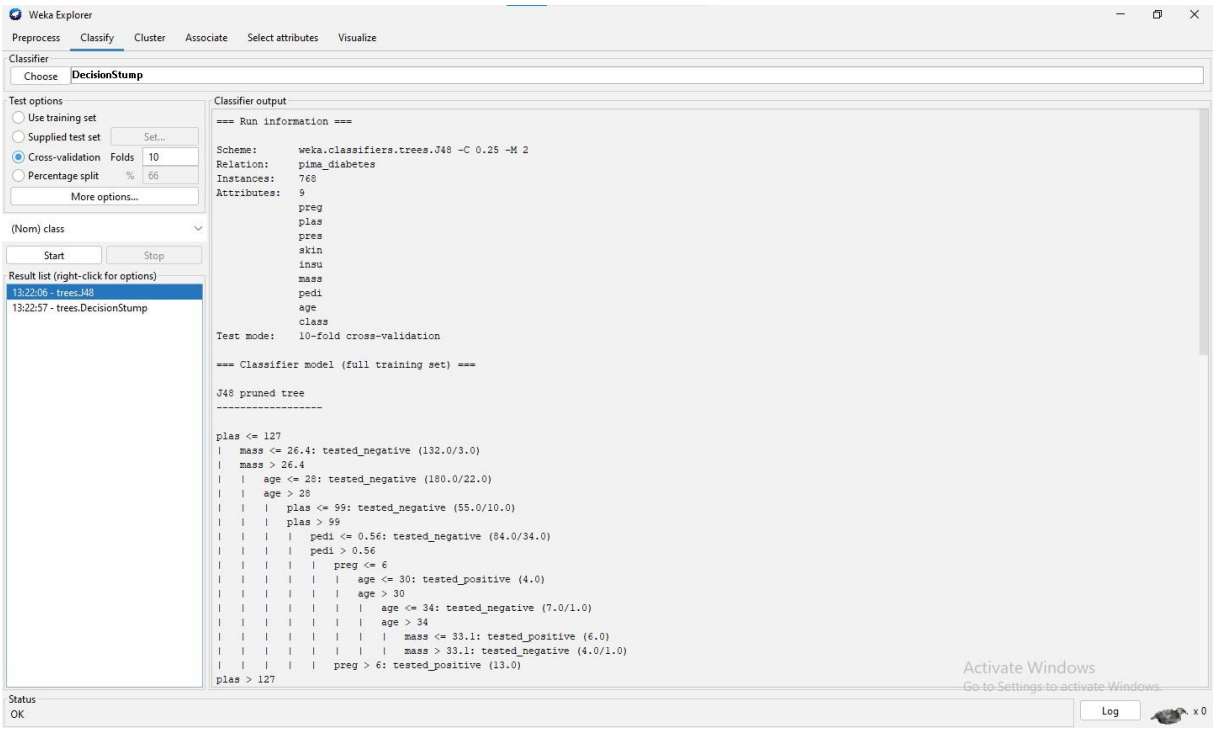


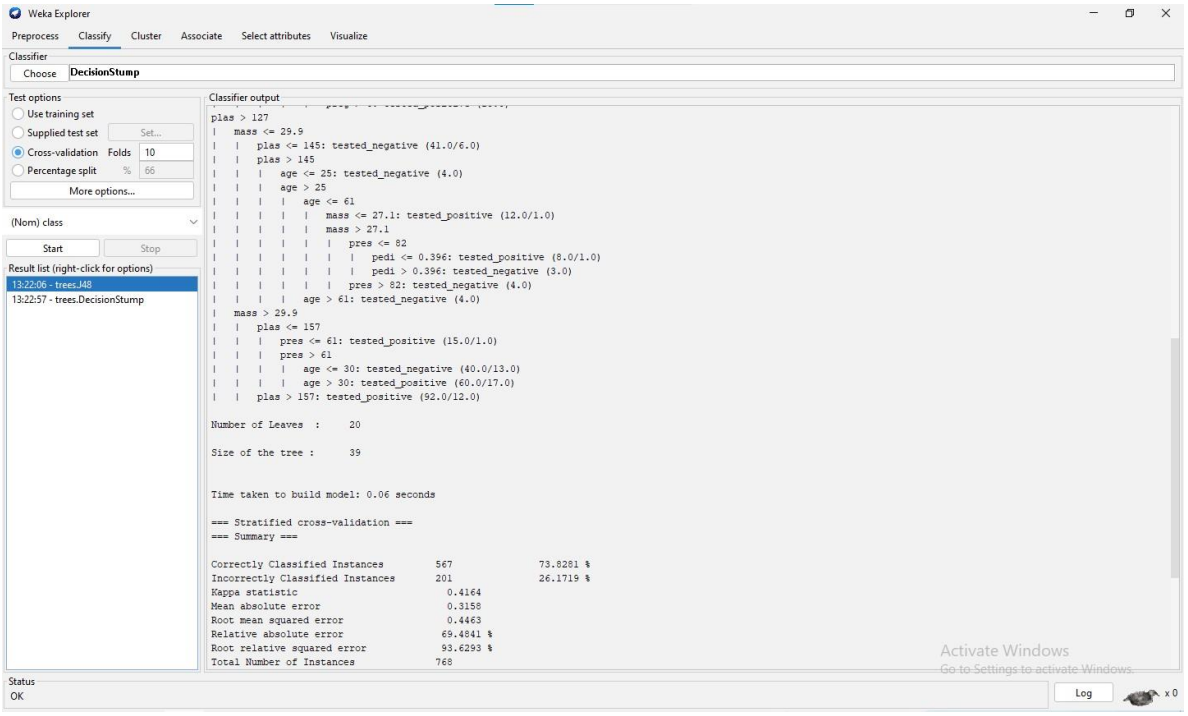
Step 3) DecisionStump Algorithm

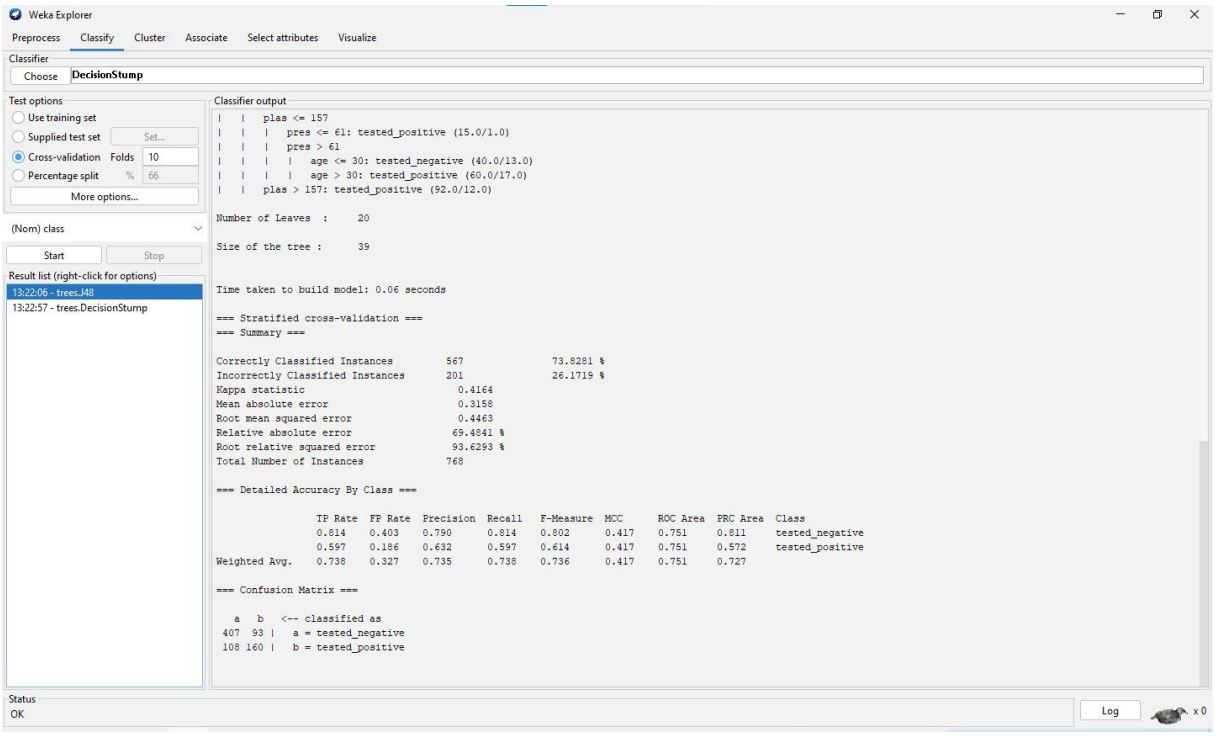




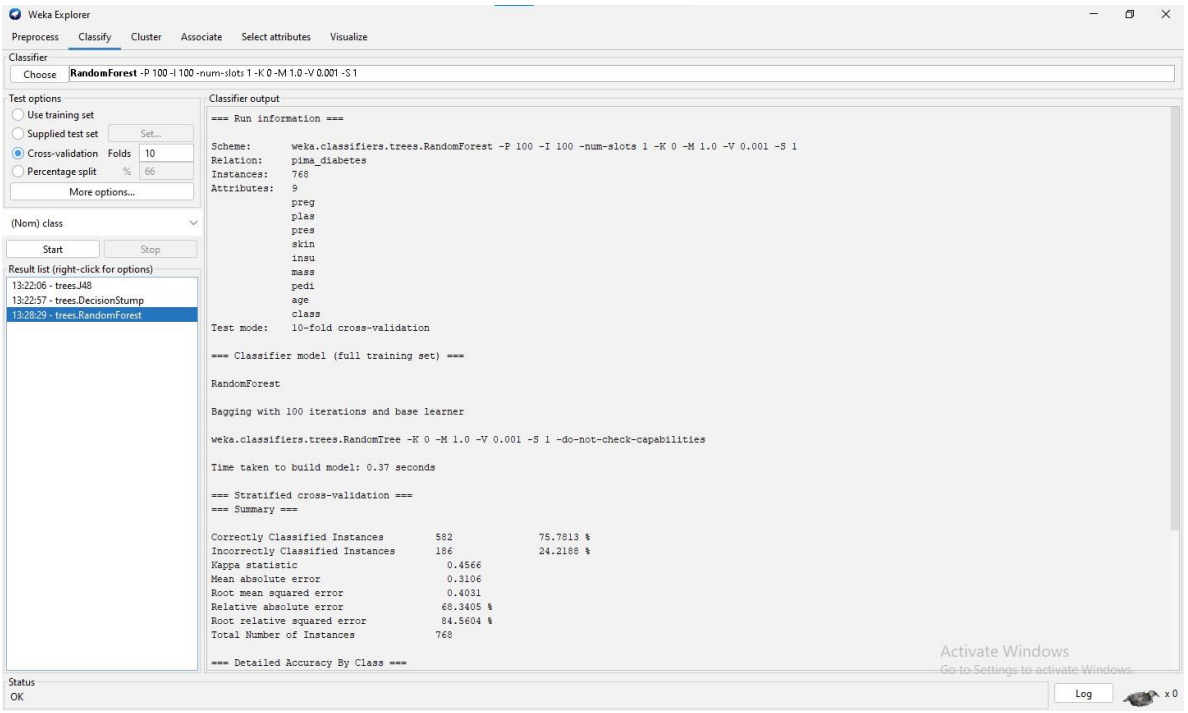
Step 4) tree.J48 Algorithm

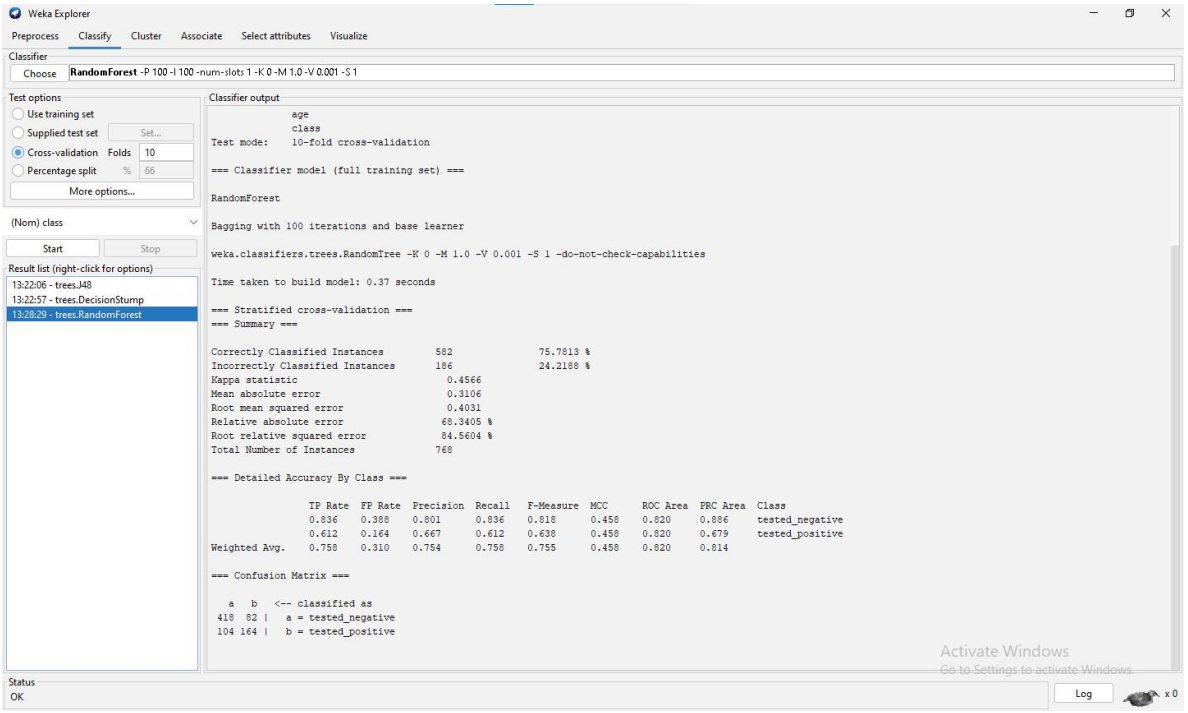






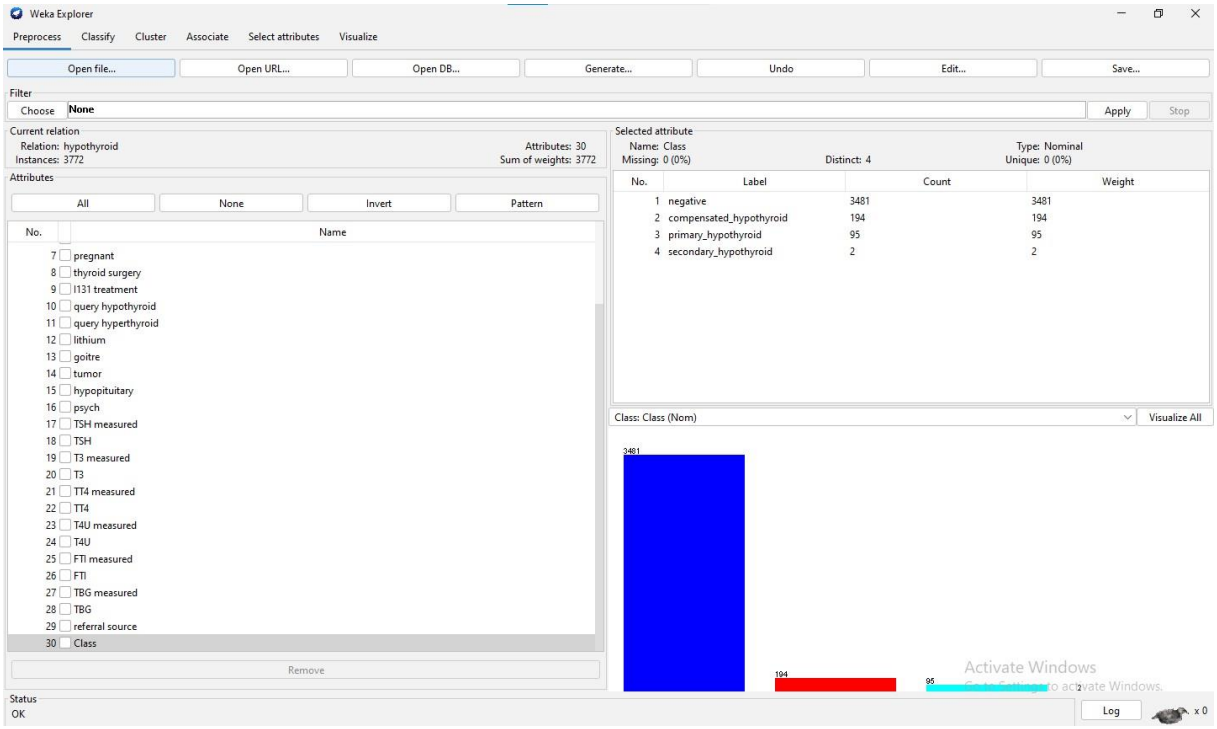
Step 5) tree.RandomForest Algorithm



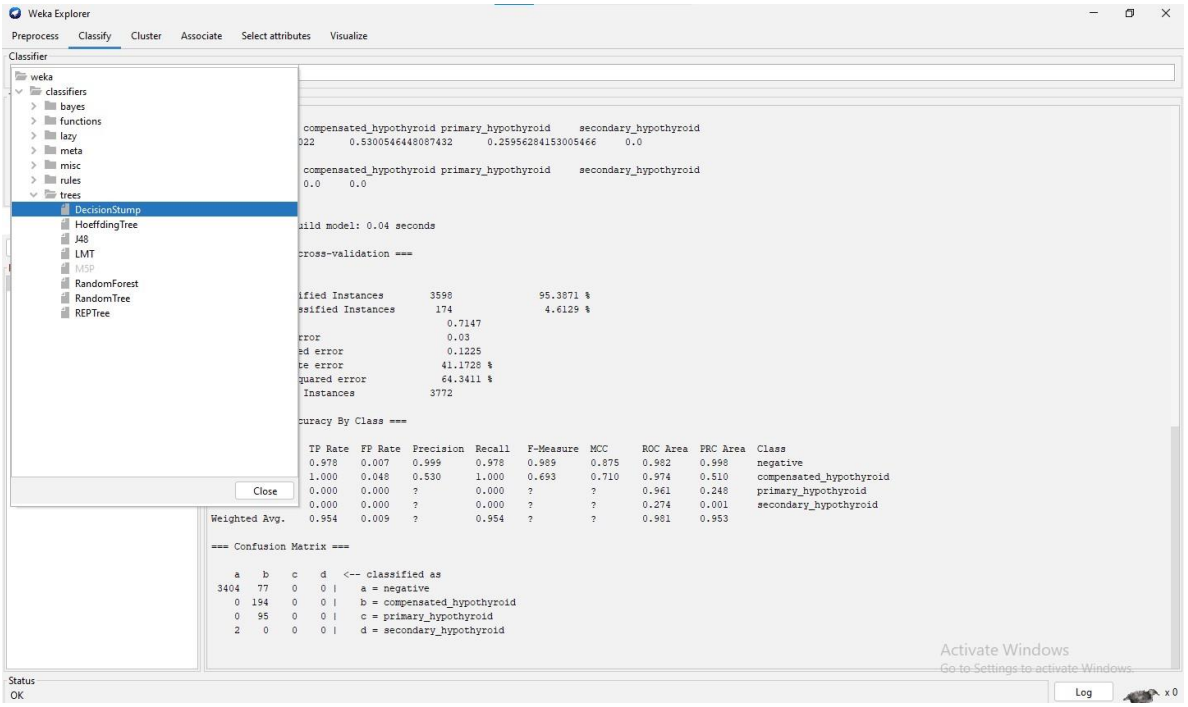

Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
RandomForest -P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1
Test options
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
66
More options...
(Nom) class
Start
Stop
Result list (right-click for options)
13:22:06 - trees.J48
13:22:57 - trees.DecisionStump
13:28:29 - trees.RandomForest
Classifier output
age
class
Test mode: 10-fold cross-validation
=== Classifier model (full training set) ===
RandomForest
Bagging with 100 iterations and base learner
weka.classifiers.trees.RandomTree -K 0 -M 1.0 -V 0.001 -S 1 -do-not-check-capabilities
Time taken to build model: 0.37 seconds
=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances 582 75.7813 %
Incorrectly Classified Instances 186 24.2188 %
Kappa statistic 0.4566
Mean absolute error 0.3106
Root mean squared error 0.4031
Relative absolute error 68.3405 %
Root relative squared error 84.5604 %
Total Number of Instances 768
=== Detailed Accuracy By Class ===
TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class
0.836 0.388 0.801 0.836 0.818 0.458 0.820 0.886 tested_negative
0.612 0.164 0.667 0.612 0.638 0.458 0.820 0.679 tested_positive
Weighted Avg. 0.758 0.310 0.754 0.758 0.755 0.458 0.820 0.814
=== Confusion Matrix ===
a b <-- classified as
418 82 | a = tested_negative
104 164 | b = tested_positive
Activate Windows
Go to Settings to activate Windows.
Status
OK
Log
x 0

➢ Dataset 2

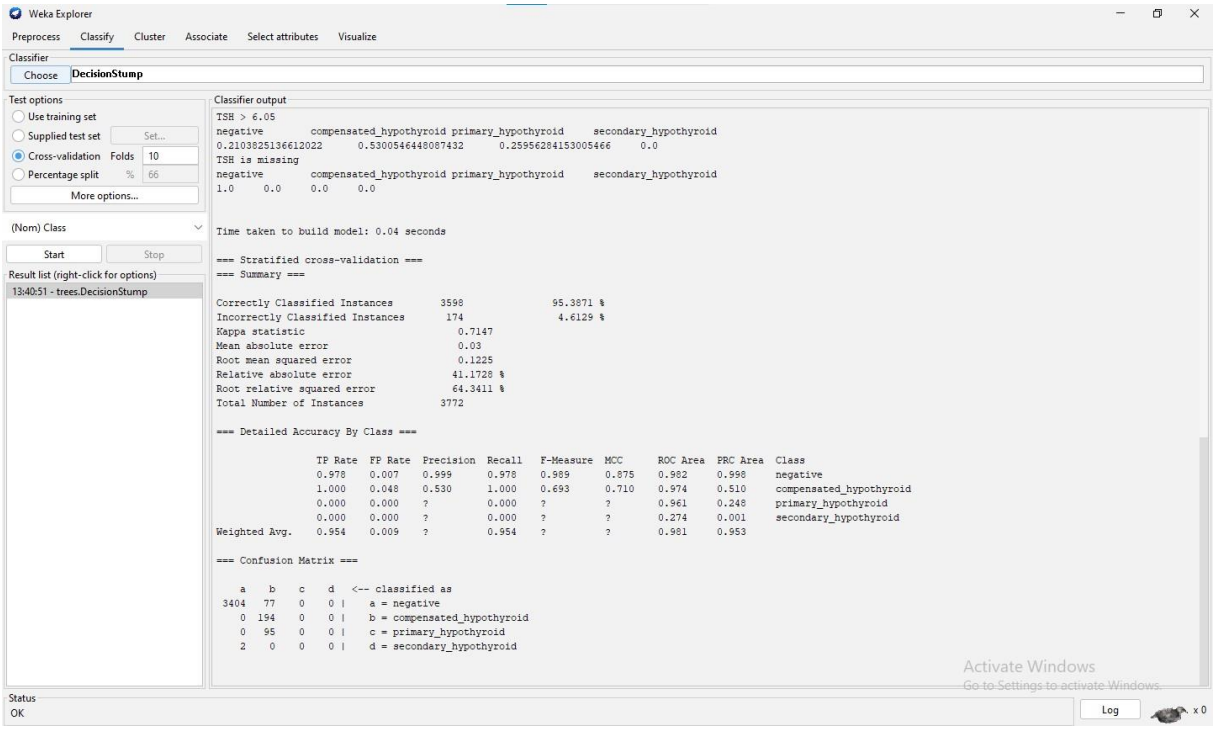
Step 1) Click on open file and load the data



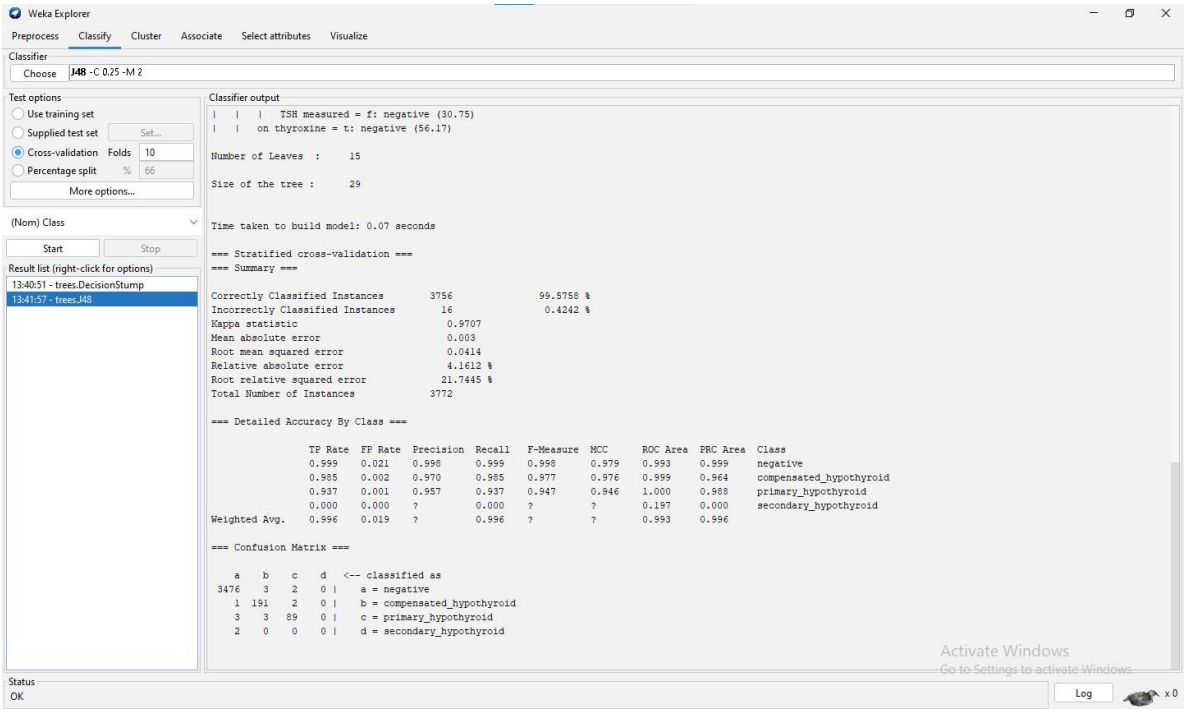
Step 2) Click on Classify and choose the algorithm of Decision tree.



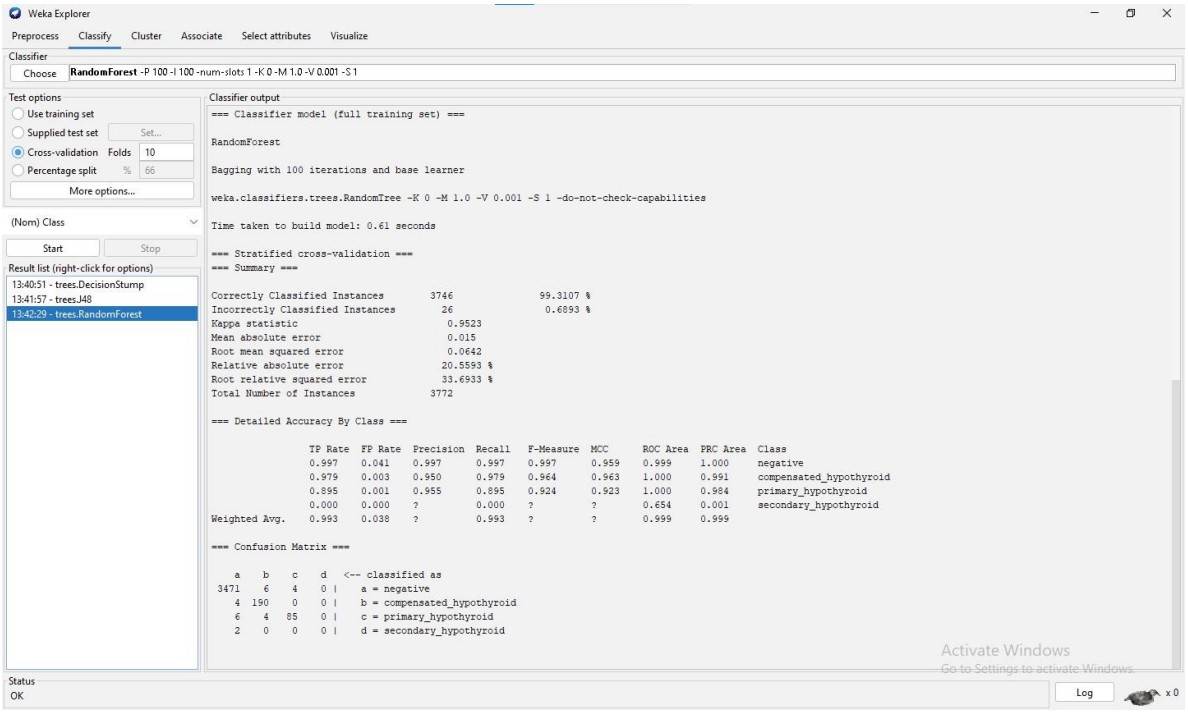
Step 3) DecisionStump Algorithm



Step 4) tree.J48 Algorithm



Step 5) tree.RandomForest Algorithm



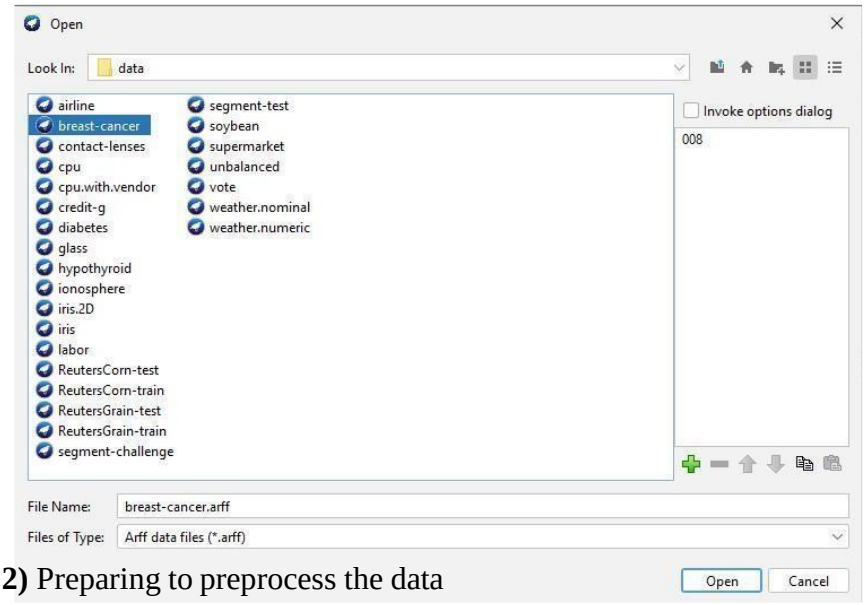
Practical: 7

Aim: Implement classification algorithm (Naïve Bayes) on any two datasets.

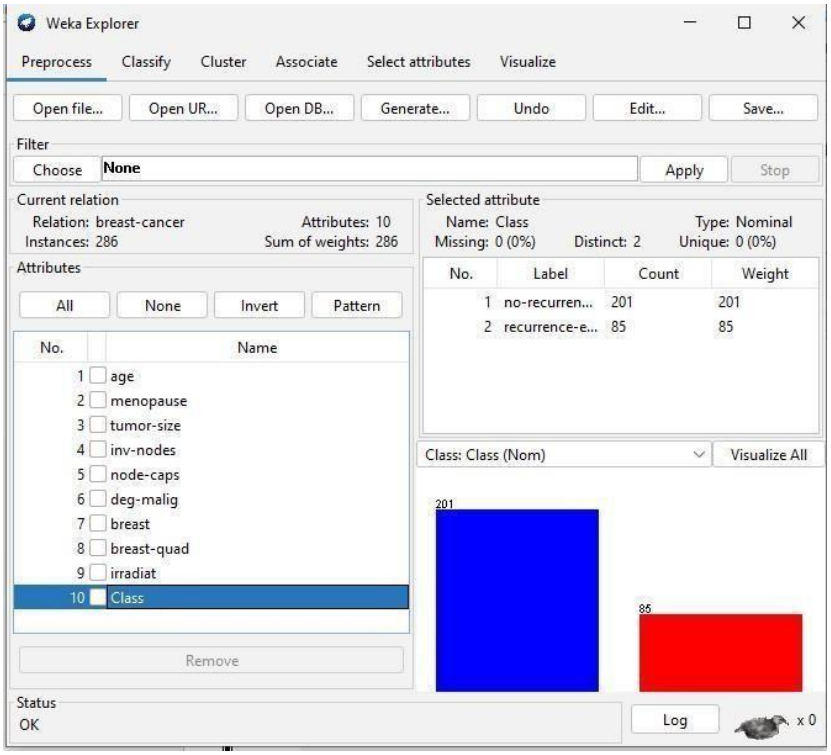
Code:

- Dataset: Breast Cancer

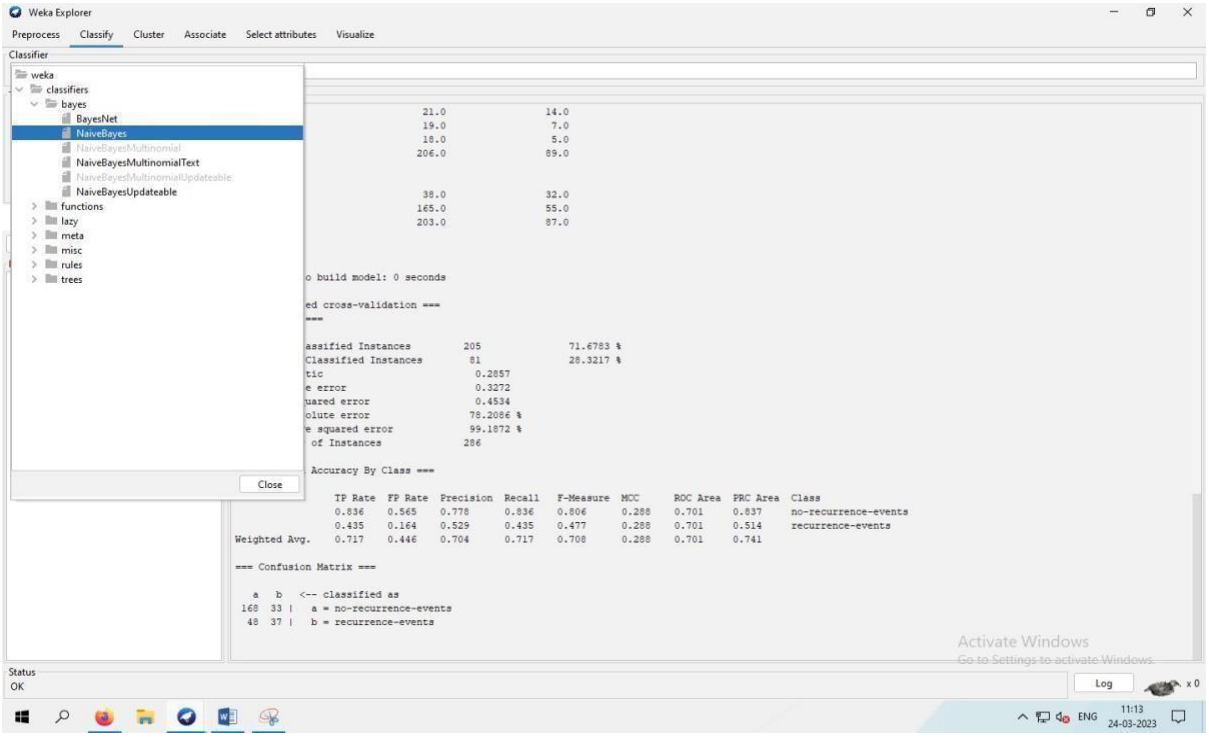
Step 1) Select the dataset



Step 2) Preparing to preprocess the data



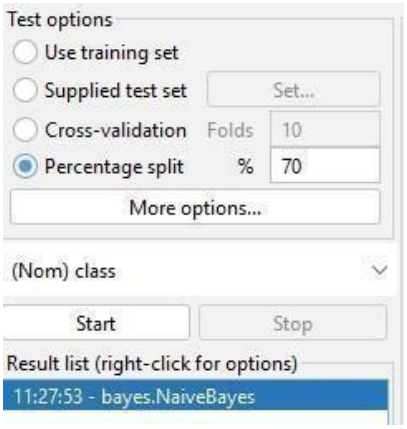
Step 3) Now go to the classify tab. Choose the Naïve Bayes classifier. Be sure that the Playattribute is selected as a class selector, and then press the Start button to build a model.

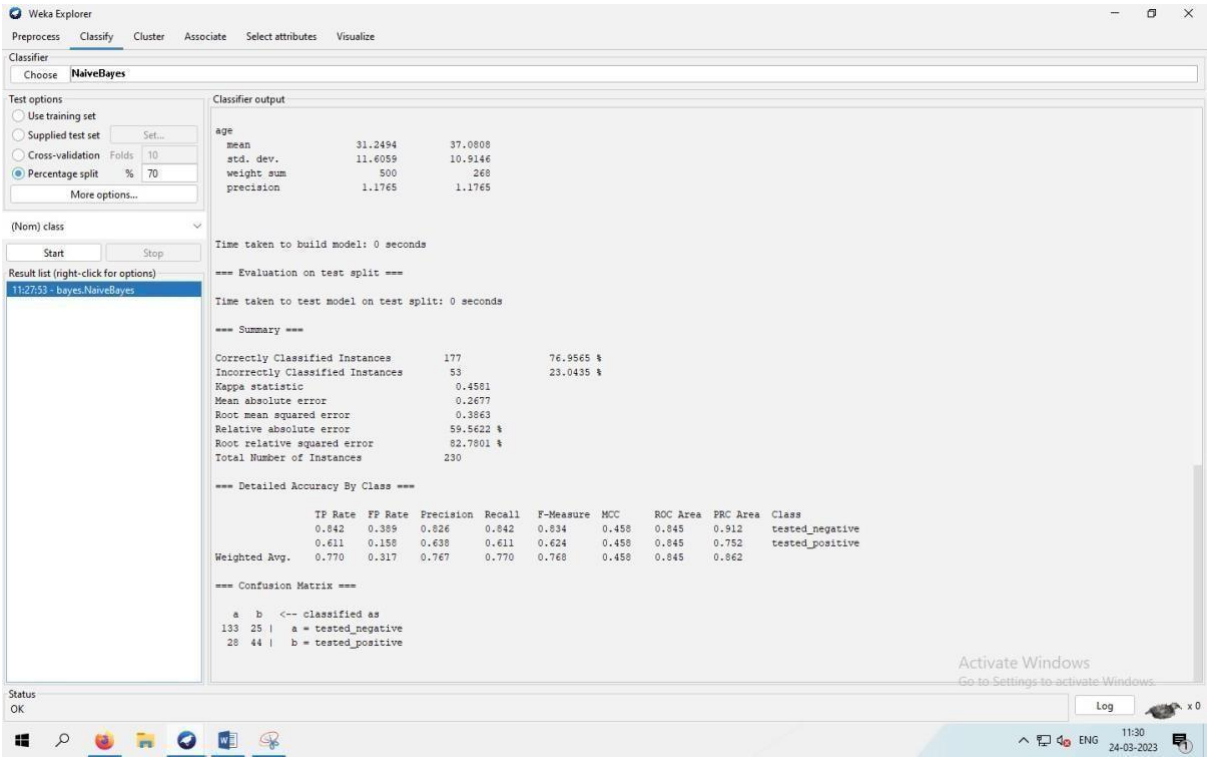


Step 4) Model outputs some information on how accurate it classifies and otherparameters.

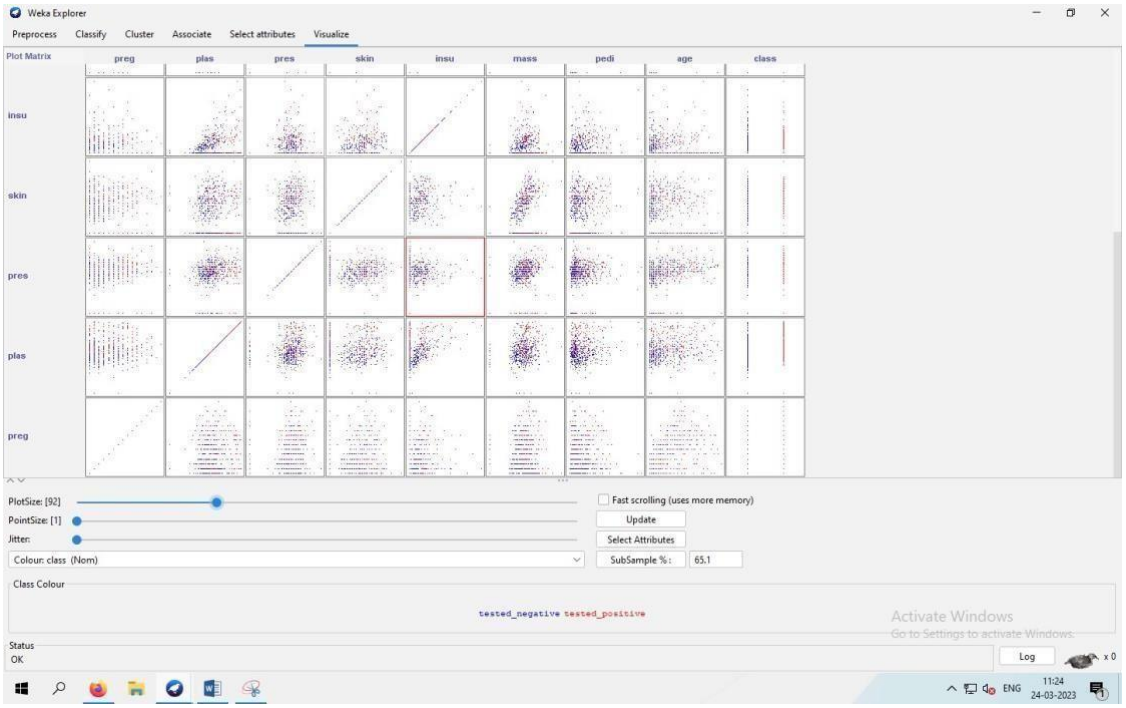
```
Correctly Classified Instances         205               71.6783 %
Incorrectly Classified Instances        81               28.3217 %
```

Step 5) Testing option to percentage split



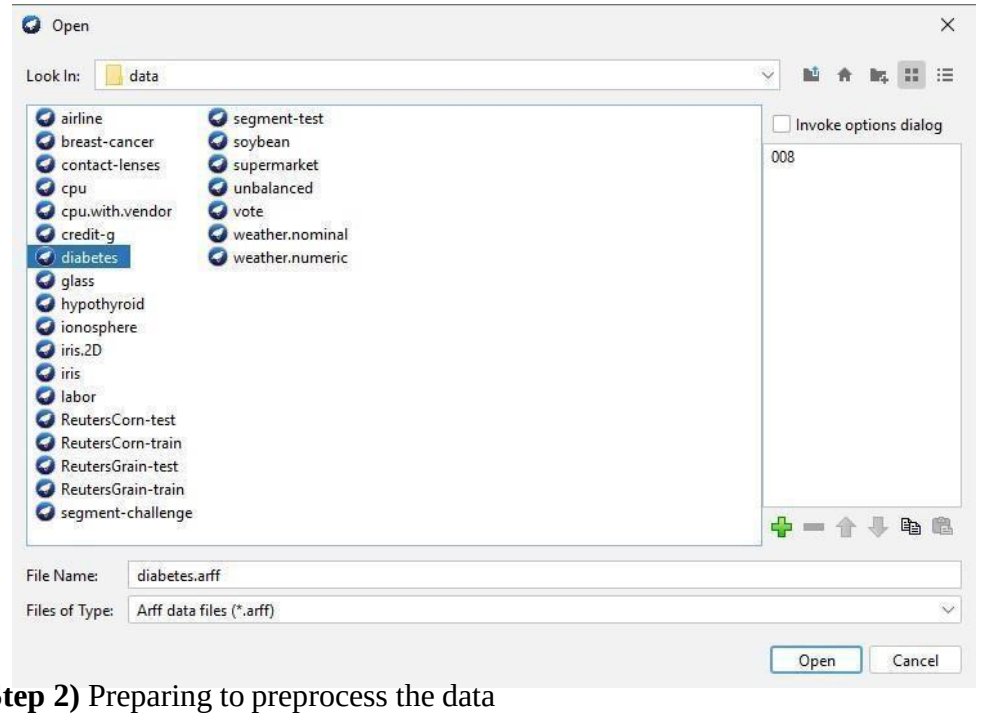


Step 6) Visualize

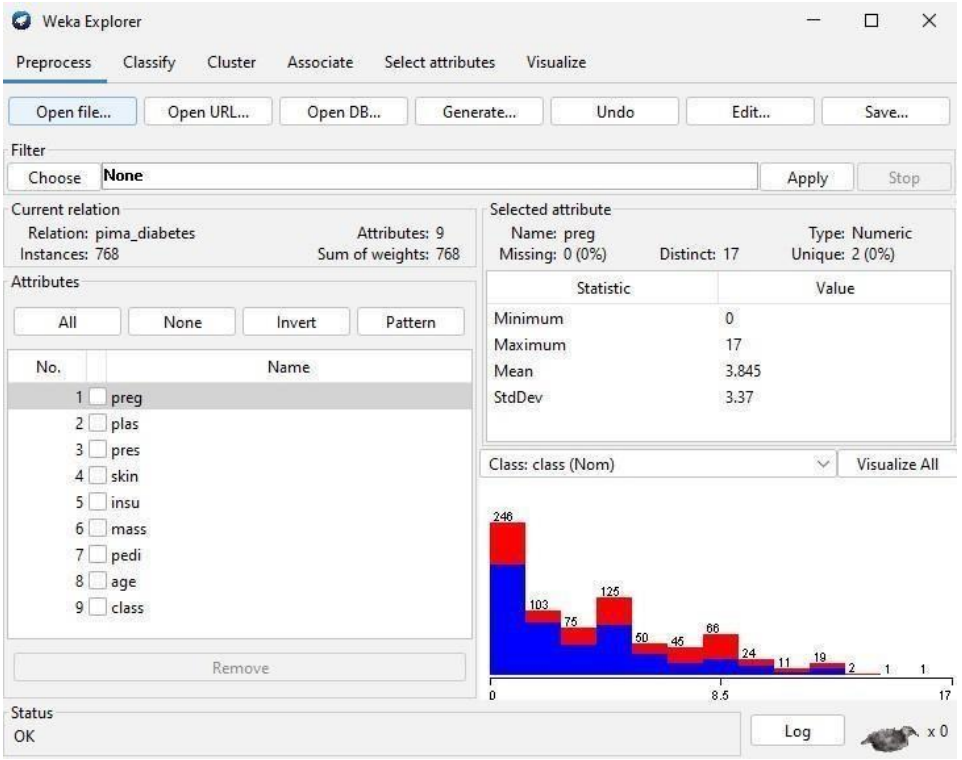


- Dataset: Diabetes

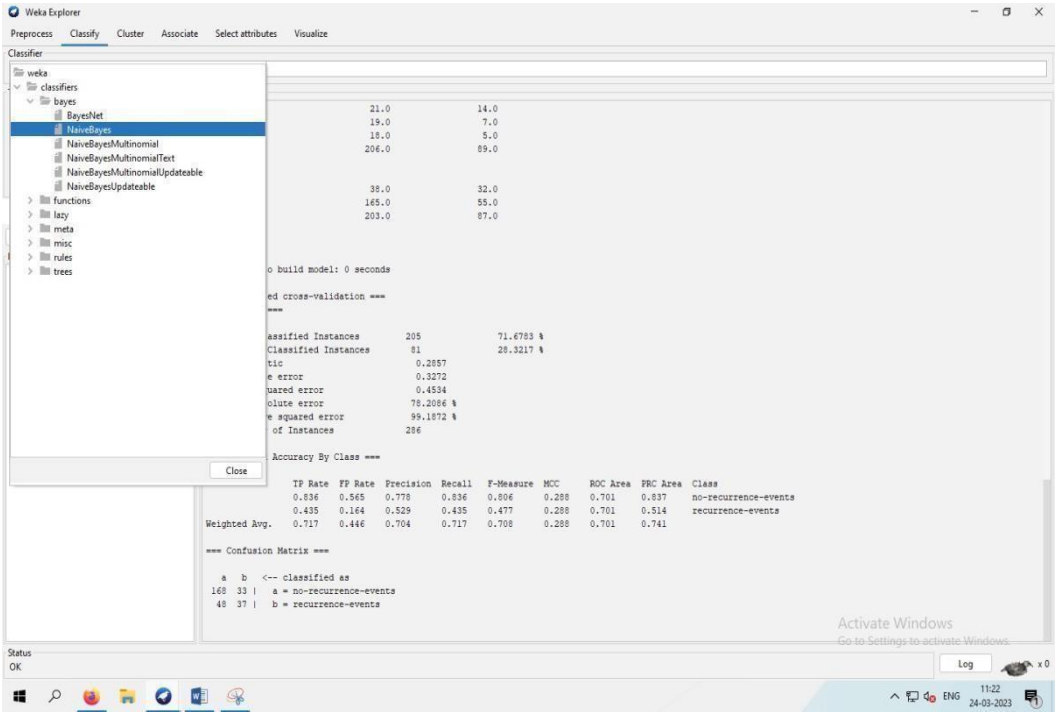
Step 1) Select the dataset



Step 2) Preparing to preprocess the data



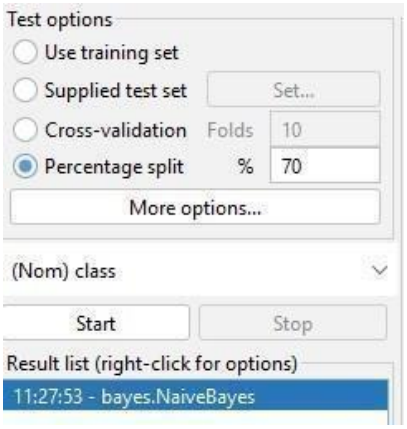
Step 3) Now go to the classify tab. Choose the Naïve Bayes classifier. Be sure that the Play attribute is selected as a class selector, and then press the Start button tobuild a model.

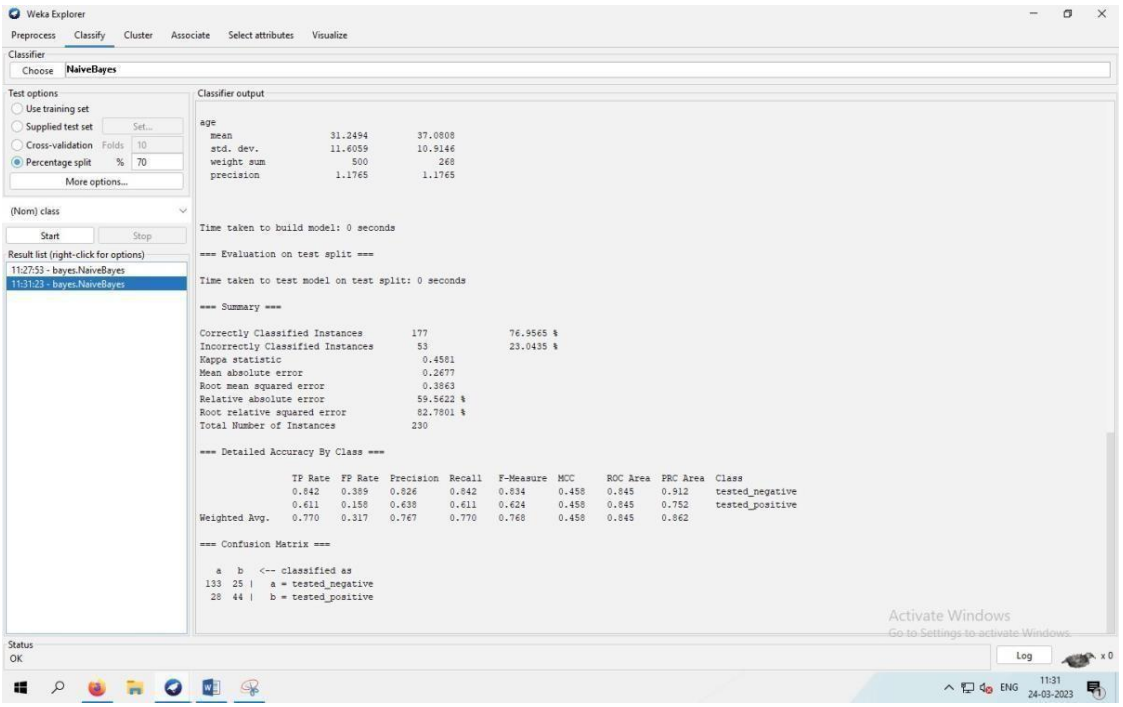


Step 4) Model outputs some information on how accurate it classifies and otherparameters.

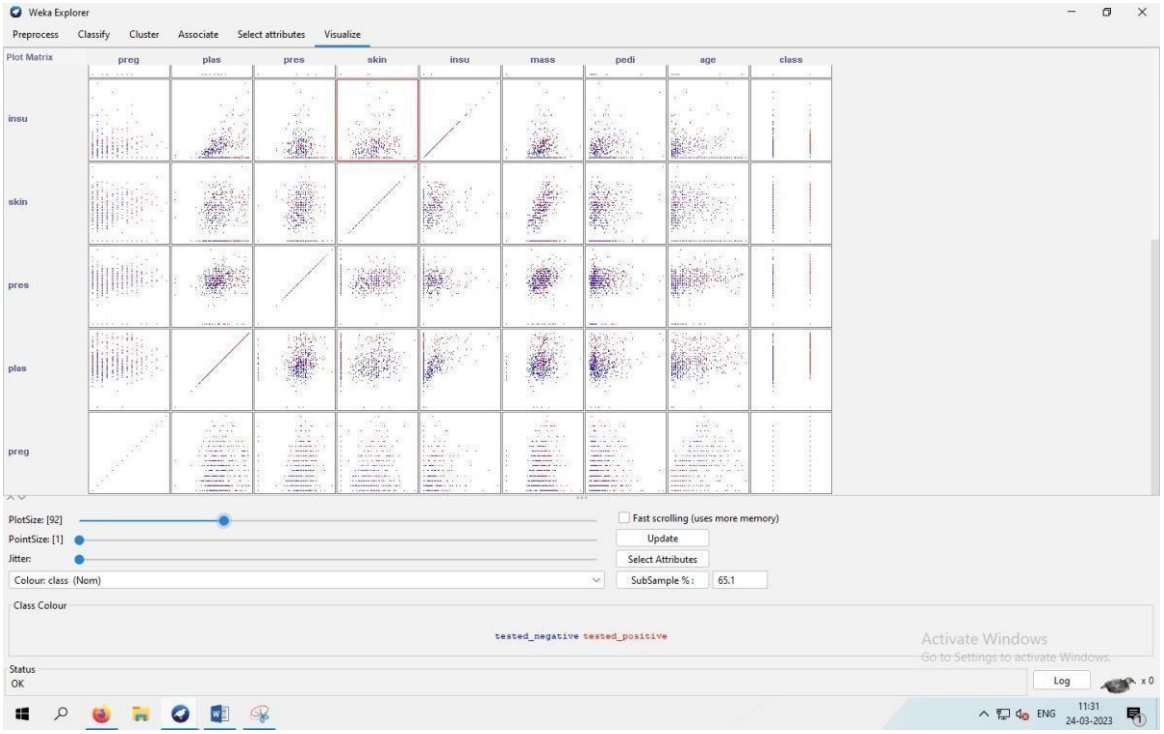
```
Correctly Classified Instances         205               71.6783 %
Incorrectly Classified Instances        81               28.3217 %
```

Step 5) Testing option to percentage split





Step 6) Visualize



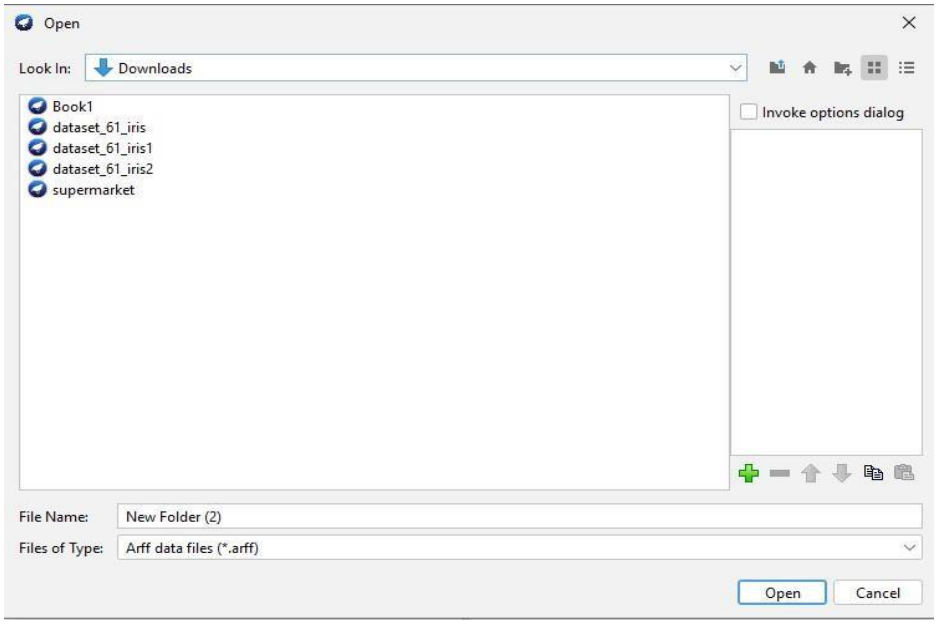
Practical: 8

Aim: Applying clustering algorithm using K-means.

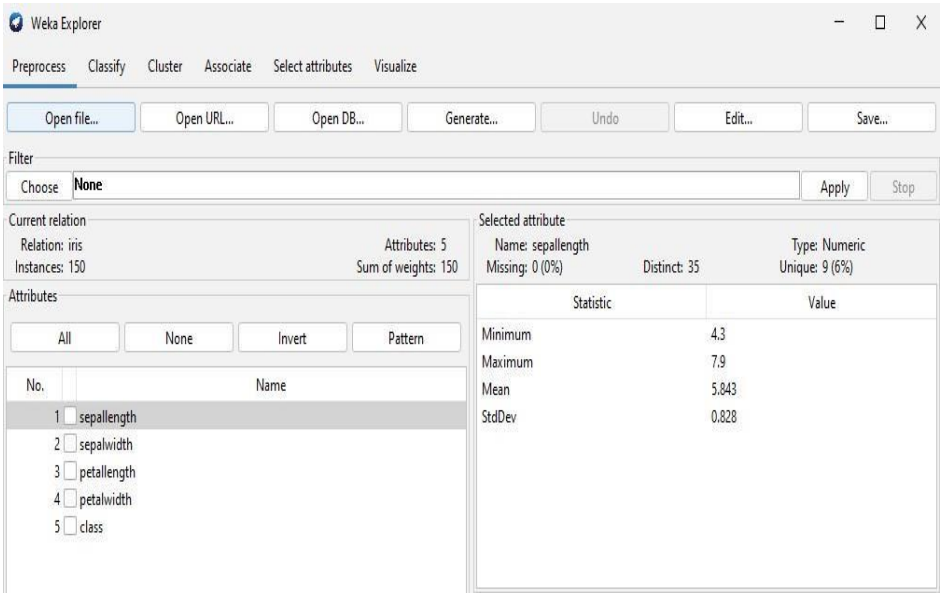
Code:

Method: 1

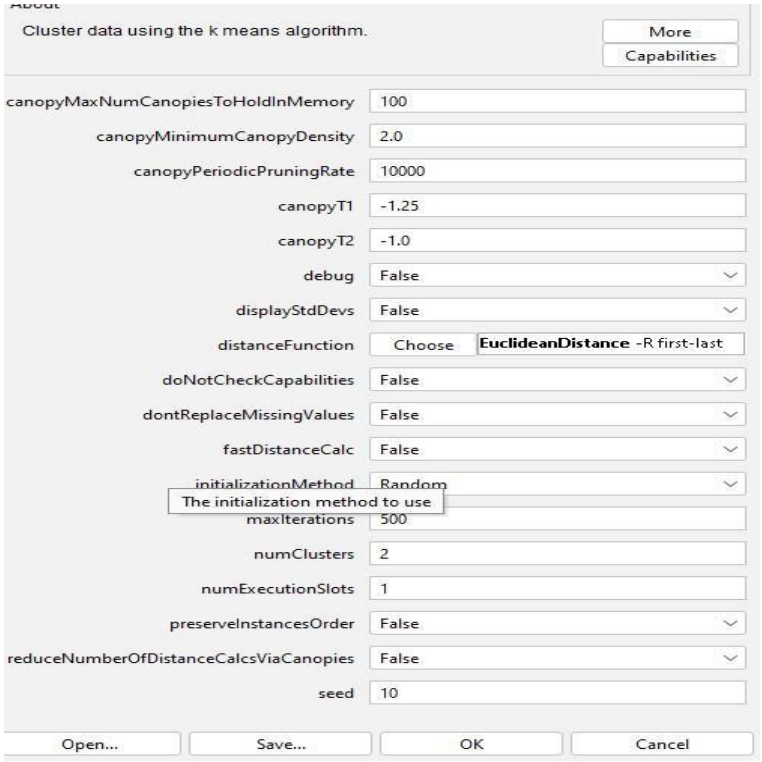
Step 1): In the pre-processing interface, open the Weka Explorer and load therequired dataset, and we are taking the iris.arff dataset



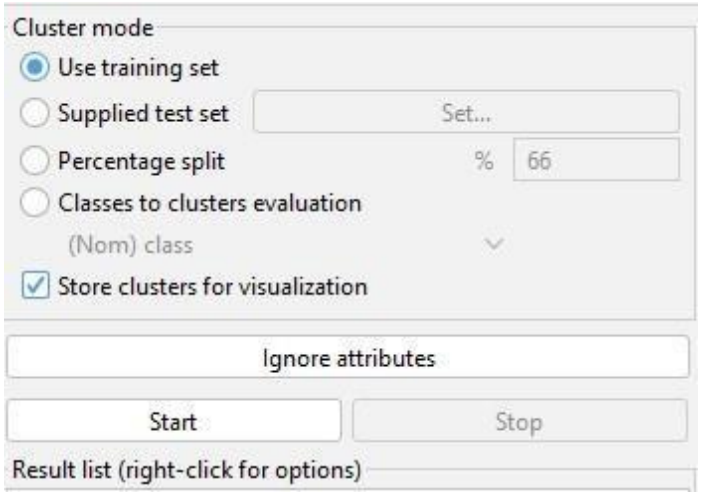
Step 2): Find the 'cluster' tab in the explorer and press the choose button to execute clustering.



Step 3): Then, to the right of the choose icon, press the text button to bring upthe popup window shown in the screenshots. We enter three for the numberof clusters in this window and leave the seed value alone.



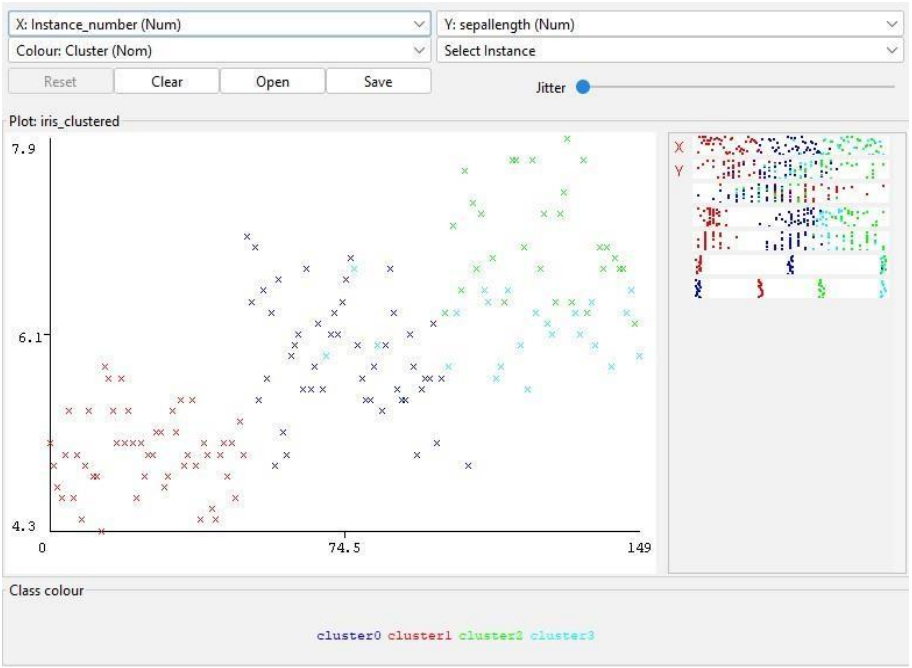
Step 4): One of the choices has been chosen. We must ensure that they are in the 'cluster mode' panel before running the clustering algorithm. The choice touse a training set is selected, and then the 'start' button is pressed.



Step 5): The centroid of each cluster is shown in the result window, along withstatistics on the number and percent of instances allocated to each cluster. Each cluster centroid is represented by a mean vector. This cluster can be usedto describe a cluster.

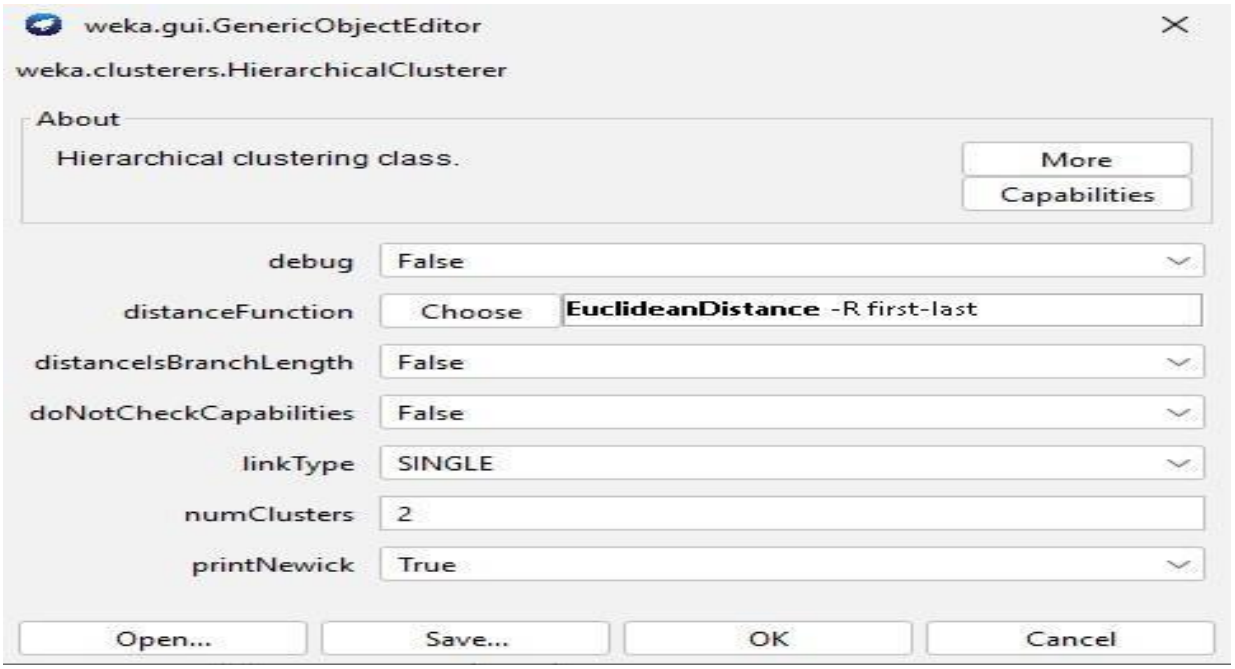
```
  mean                2.7519   3.418  3.1103  2.8088
  std. dev.           0.3103  0.3772  0.2952  0.2361

petallength
  mean                4.2267   1.464  5.8559  5.0993
  std. dev.            0.445  0.1718  0.4626  0.2462

petalwidth
  mean                1.3134   0.244  2.1495  1.8254
  std. dev.           0.1864  0.1061   0.232  0.2152

class
  Iris-setosa              1      51       1       1
  Iris-versicolor    48.1125       1  1.0182  3.8693
  Iris-virginica      2.0983       1 31.0375 19.8641
  [total]            51.2108      53 33.0557 24.7335


Time taken to build model (full training data) : 0.48 seconds

=== Model and evaluation on training set ===

Clustered Instances

0       48 ( 32%)
1       50 ( 33%)
2       29 ( 19%)
3       23 ( 15%)


Log likelihood: -2.03504
```

Step 6): Another way to grasp the characteristics of each cluster is to visualizethem. To do so, right-click the result set on the result. Selecting to visualize cluster assignments from the list column.

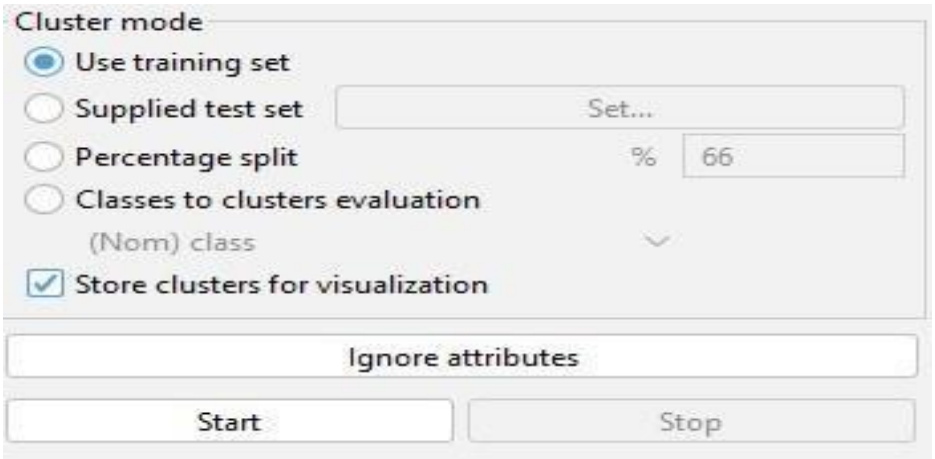


Method: 2(Hierarchical clustering)

Step 1): Then, to the right of the choose icon, press the text button to bring upthe popup window shown in the screenshots. We enter three for the numberof clusters in this window and leave the seed value alone.



Step 2): One of the choices has been chosen. We must ensure that they are in the ‘cluster mode’ panel before running the clustering algorithm. The choice touse a training set is selected, and then the ‘start’ button is pressed.

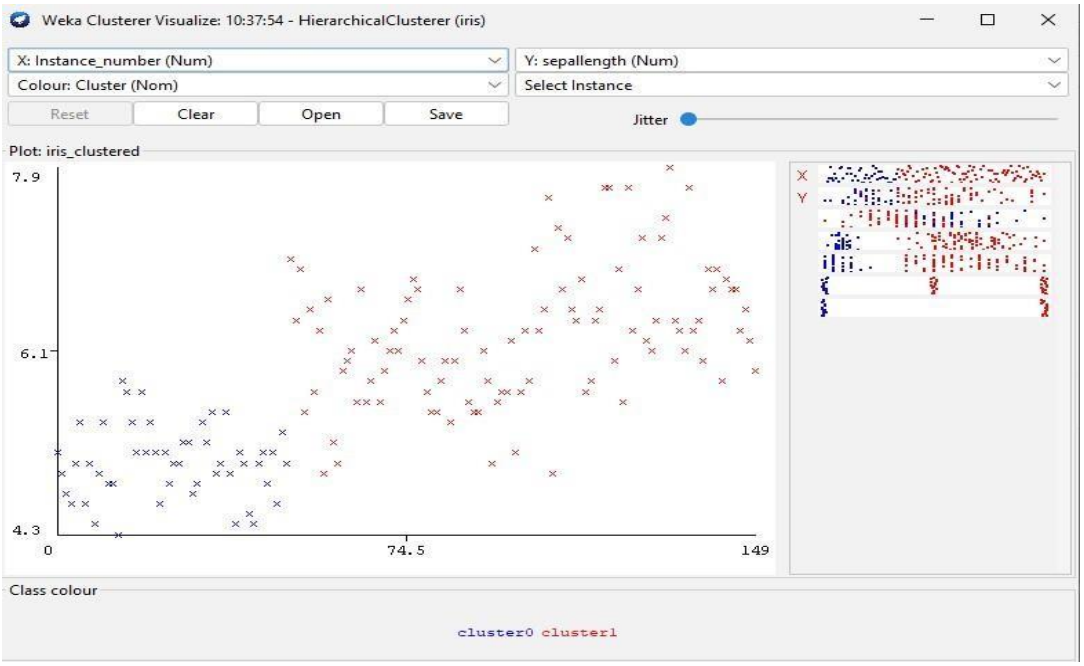


Step 3): The centroid of each cluster is shown in the result window, along withstatistics on the number and percent of instances allocated to each cluster.
Each cluster centroid is represented by a mean vector. This cluster can be usedto describe a cluster.

Clusterer output

```
Scheme:       weka.clusterers.HierarchicalClusterer -N 2 -L SINGLE -P -A "weka.core.Euclic
Relation:     iris
Instances:    150
Attributes:   5
              sepallength
              sepalwidth
              petallength
              petalwidth
              class
Test mode:    evaluate on training data

=== Clustering model (full training set) ===

Cluster 0
((((((((((((((((((((0.0:0.03254,0.0:0.03254):0.00913,(0.0:0.03254,0.0:0.03254):0.00913):0

Cluster 1
((((((((((((((((((1.0:0.07344,(((1.0:0.06508,1.0:0.06508):0.00066,(1.0:0.05008,1.0:0.0500

Time taken to build model (full training data) : 0.04 seconds

=== Model and evaluation on training set ===

Clustered Instances

0        50 ( 33%)
1       100 ( 67%)
```

Step 4): Another way to grasp the characteristics of each cluster is to visualize them. To do so, right-click the result set on the result. Selecting to visualize cluster assignments from the list column.



Practical: 9

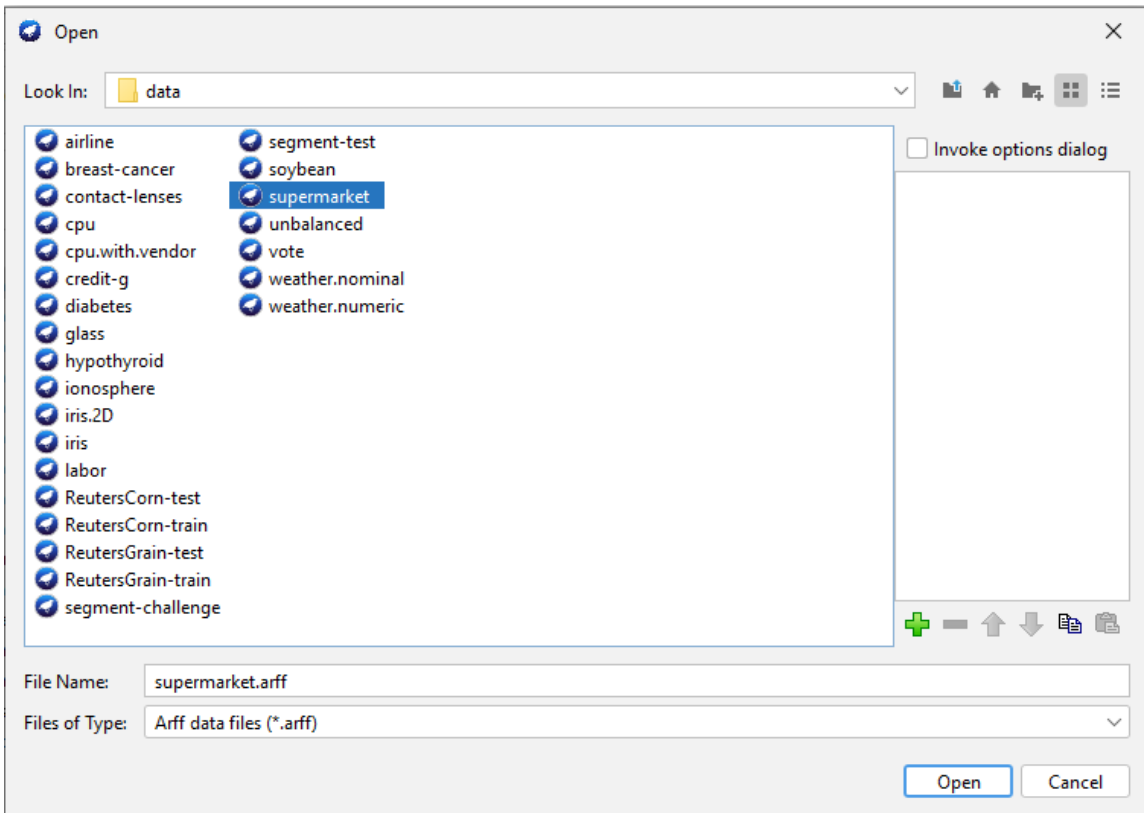
Aim: Generate Association rule, build classification model and clustering model for your dataset.

Code:

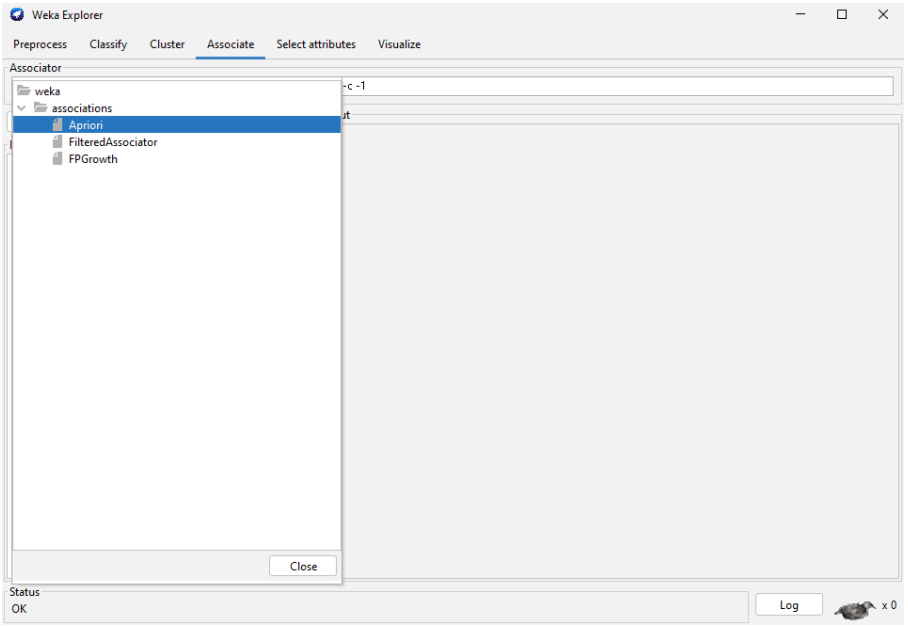
- **ASSOCIATION RULE**

Algorithm 1: Apriori Algorithm

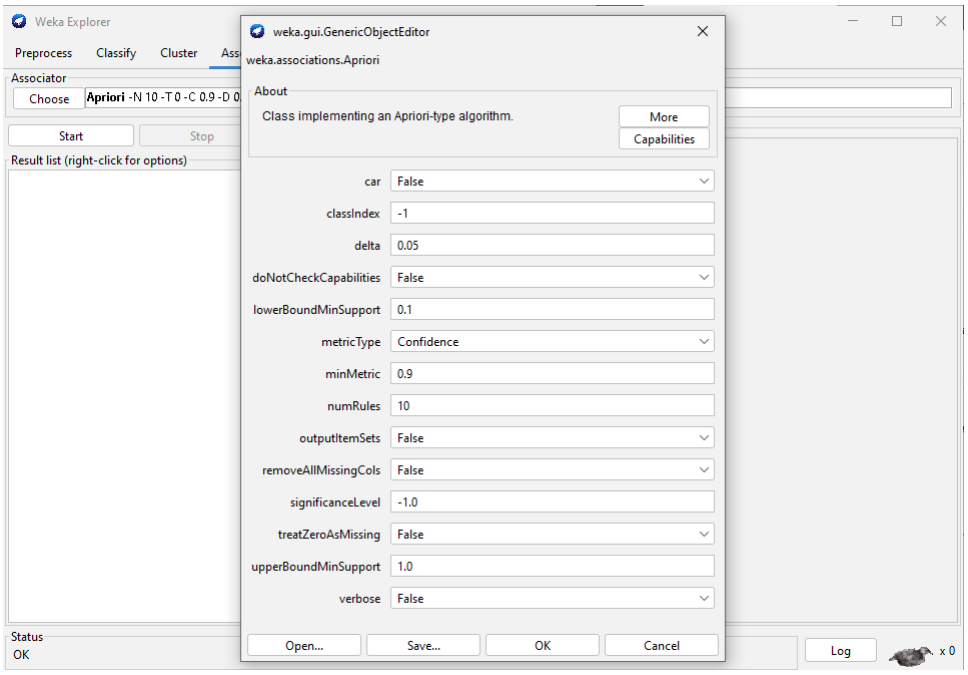
Step 1) In the WEKA explorer, open the **Preprocess** tab, click on the **Open file** ... button and select **supermarket.arff** database from the installation folder. After the data is loaded you will see the following screen –



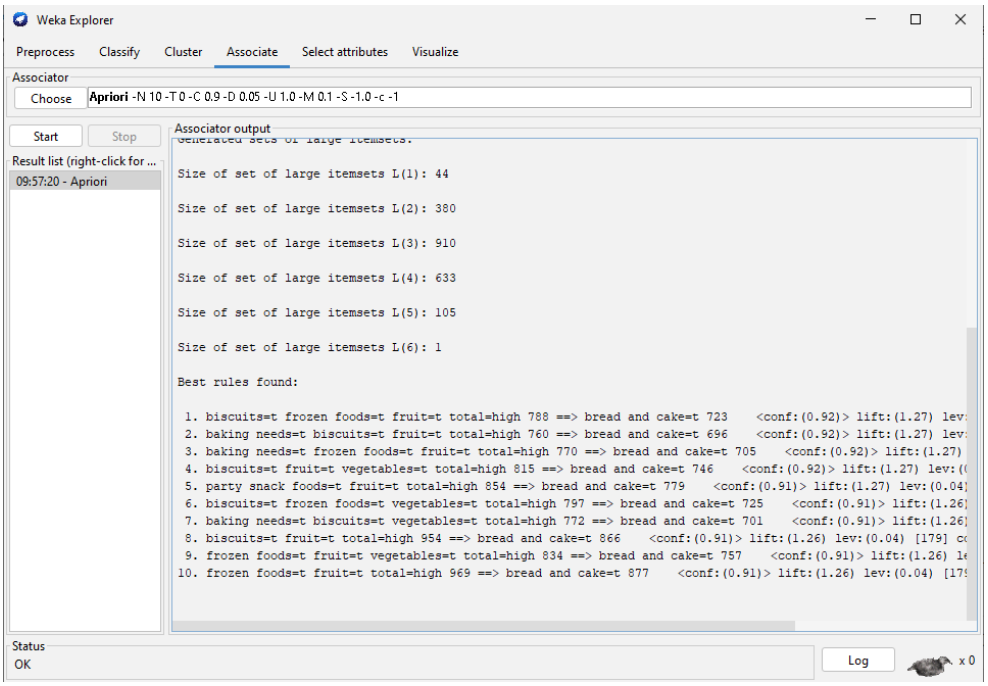
Step 2) Click on the **Associate** TAB and click on the **Choose** button. Select the **Apriori** association as shown in the screenshot –



Step 3) To set the parameters for the Apriori algorithm, click on its name, a window will pop up as shown below that allows you to set the parameters –



Step 4) After you set the parameters, click the **Start** button. After a while you will see the results as shown in the screenshot below –

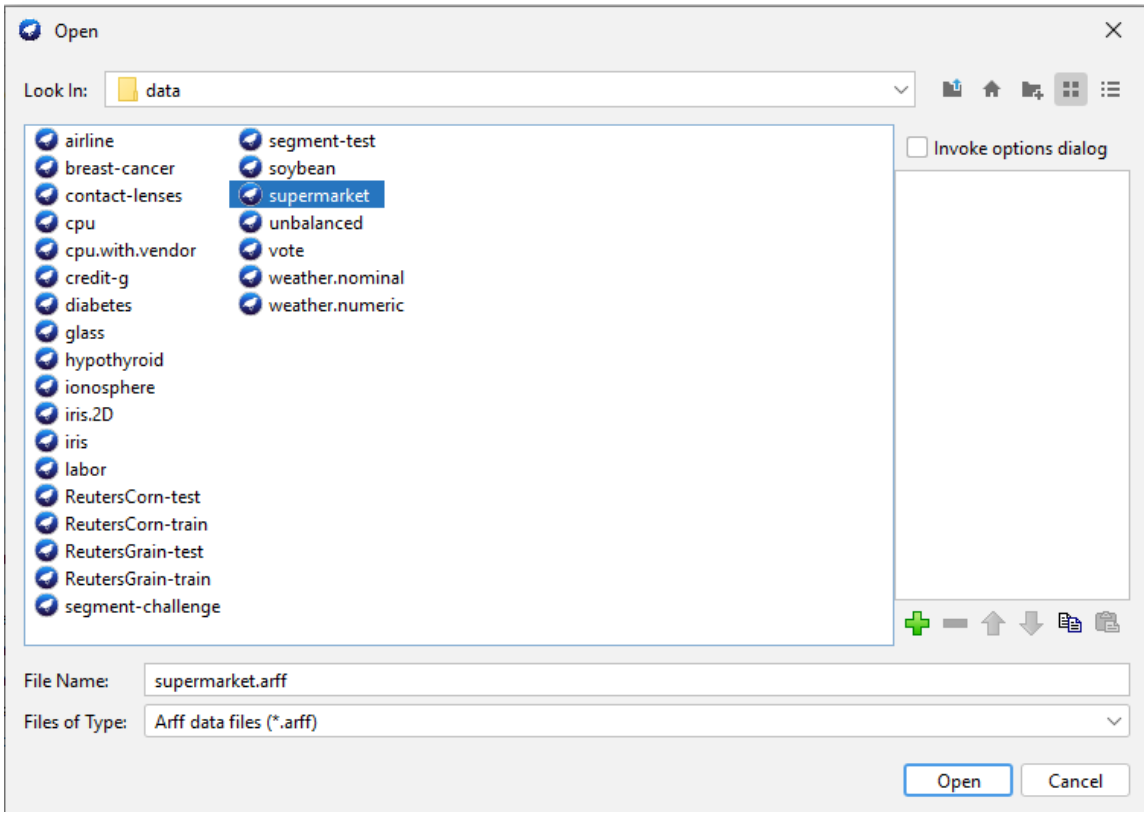


Step 5) At the bottom, you will find the detected best rules of associations. This will help the supermarket in stocking their products in appropriate shelves.

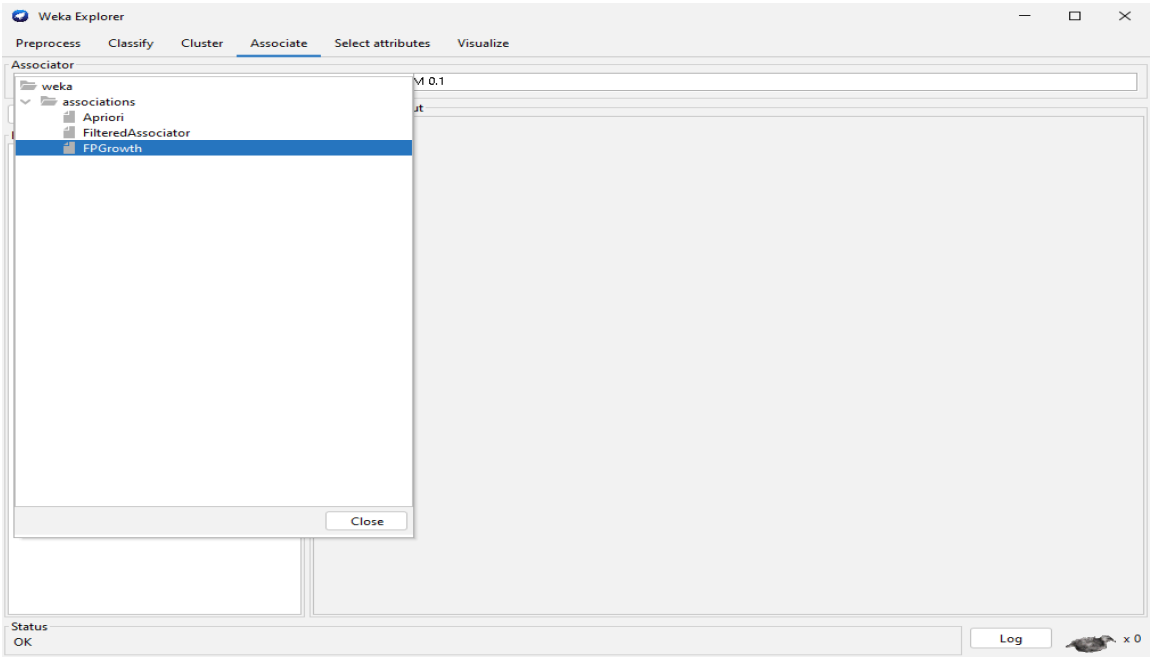
```
Best rules found:

 1. biscuits=t frozen foods=t fruit=t total=high 788 ==> bread and cake=t 723    <conf:(0.92)> lift:(1.27) lev:(0.03) [155] conv:(3.35)
 2. baking needs=t biscuits=t fruit=t total=high 760 ==> bread and cake=t 696    <conf:(0.92)> lift:(1.27) lev:(0.03) [149] conv:(3.28)
 3. baking needs=t frozen foods=t fruit=t total=high 770 ==> bread and cake=t 705    <conf:(0.92)> lift:(1.27) lev:(0.03) [150] conv:(3.27)
 4. biscuits=t fruit=t vegetables=t total=high 815 ==> bread and cake=t 746    <conf:(0.92)> lift:(1.27) lev:(0.03) [159] conv:(3.26)
 5. party snack foods=t fruit=t total=high 854 ==> bread and cake=t 779    <conf:(0.91)> lift:(1.27) lev:(0.04) [164] conv:(3.15)
 6. biscuits=t frozen foods=t vegetables=t total=high 797 ==> bread and cake=t 725    <conf:(0.91)> lift:(1.26) lev:(0.03) [151] conv:(3.06)
 7. baking needs=t biscuits=t vegetables=t total=high 772 ==> bread and cake=t 701    <conf:(0.91)> lift:(1.26) lev:(0.03) [145] conv:(3.01)
 8. biscuits=t fruit=t total=high 954 ==> bread and cake=t 866    <conf:(0.91)> lift:(1.26) lev:(0.04) [179] conv:(3)
 9. frozen foods=t fruit=t vegetables=t total=high 834 ==> bread and cake=t 757    <conf:(0.91)> lift:(1.26) lev:(0.03) [156] conv:(3)
10. frozen foods=t fruit=t total=high 969 ==> bread and cake=t 877    <conf:(0.91)> lift:(1.26) lev:(0.04) [179] conv:(2.92)
```

Algorithm 2: FPGrowth Algorithm

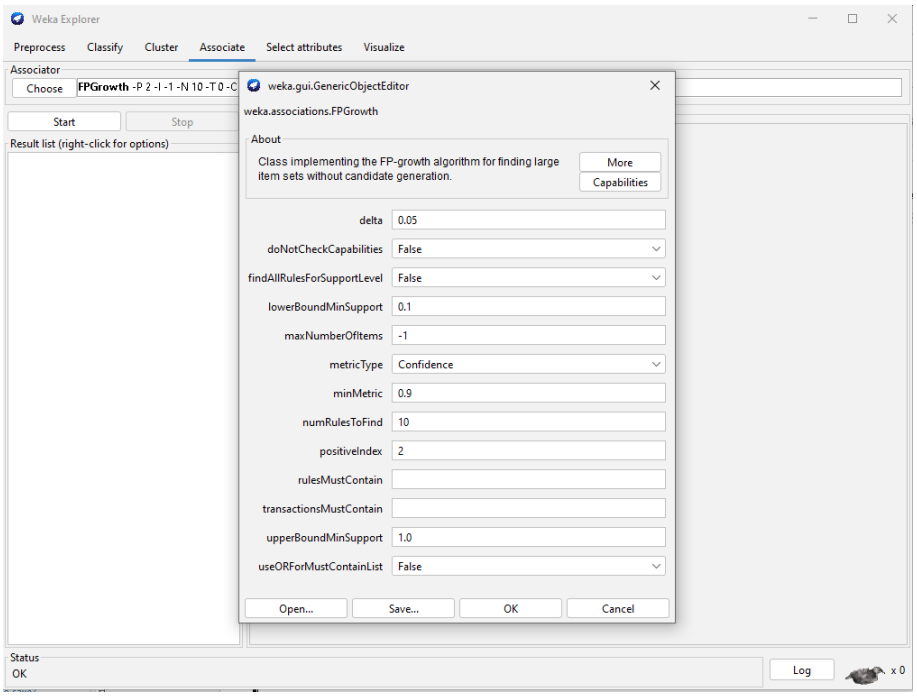
Step 1) In the WEKA explorer, open the **Preprocess** tab, click on the **Open file** ... button and select **supermarket.arff** database from the installation folder. After the data is loaded you will see the following screen –



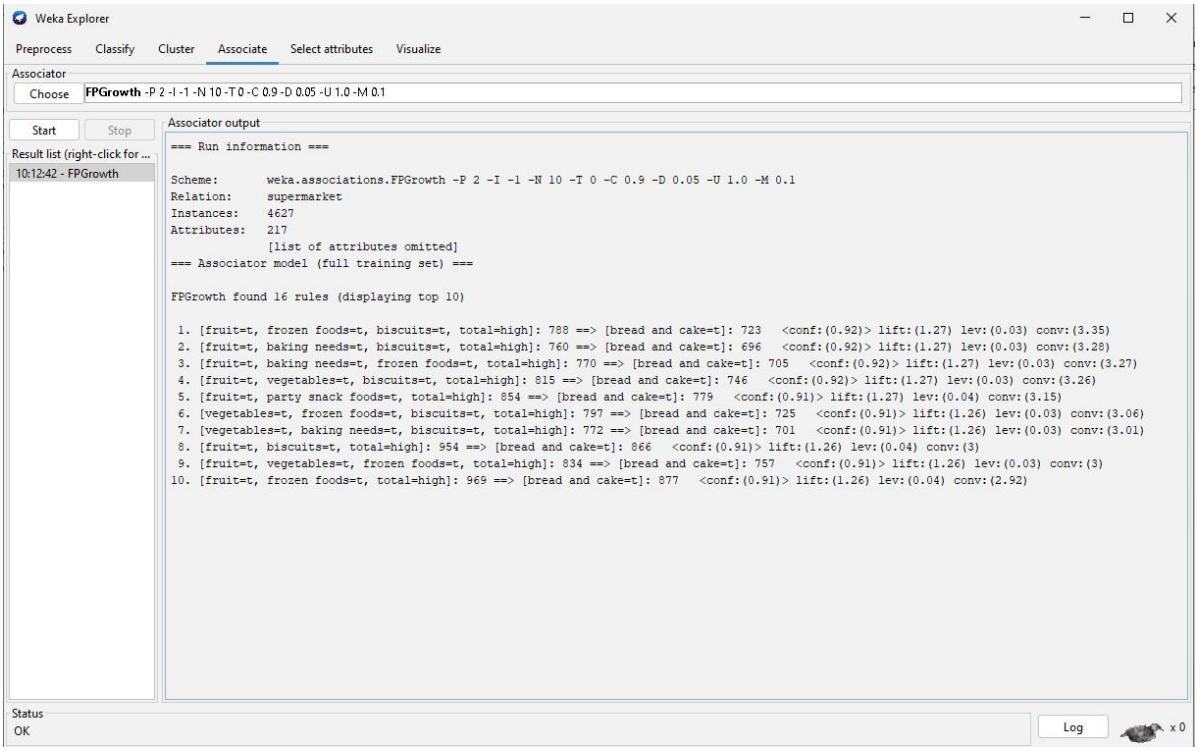
Step 2) Click on the **Associate** TAB and click on the **Choose** button. Select the **FPGrowth** association as shown in the screenshot –



Step 3) To set the parameters for the FPGrowth algorithm, click on its name, a window will pop up as shown below that allows you to set the parameters –



Step 4) After you set the parameters, click the **Start** button. After a while you will see the results as shown in the screenshot below –



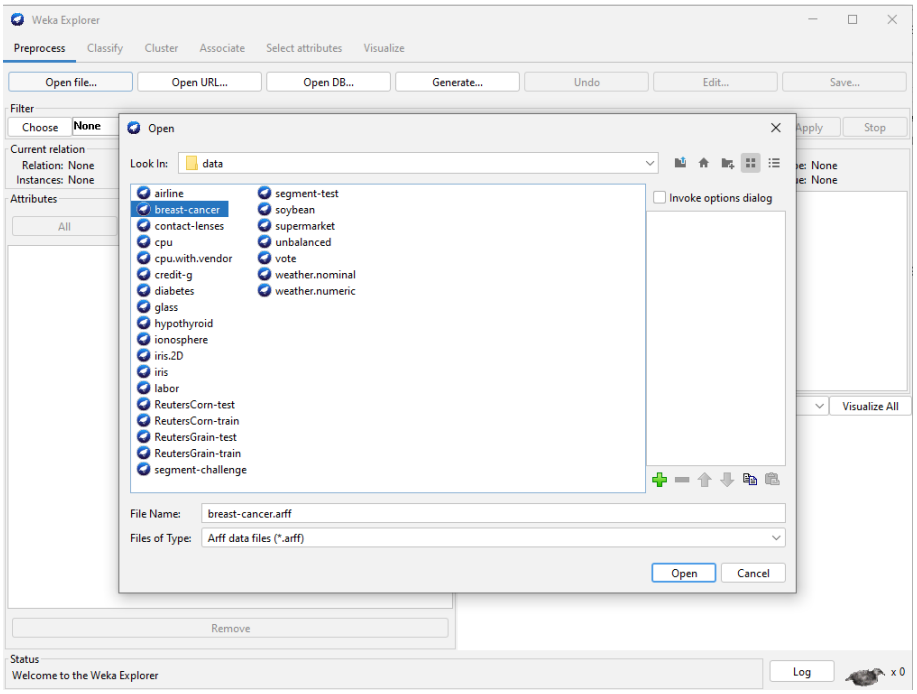
Step 5) At the bottom, you will find the detected best rules of associations. This will help the supermarket in stocking their products in appropriate shelves.

```
FPGrowth found 16 rules (displaying top 10)

 1. [fruit=t, frozen foods=t, biscuits=t, total=high]: 788 ==> [bread and cake=t]: 723   <conf:(0.92)> lift:(1.27) lev:(0.03) conv:(3.35)
 2. [fruit=t, baking needs=t, biscuits=t, total=high]: 760 ==> [bread and cake=t]: 696   <conf:(0.92)> lift:(1.27) lev:(0.03) conv:(3.28)
 3. [fruit=t, baking needs=t, frozen foods=t, total=high]: 770 ==> [bread and cake=t]: 705   <conf:(0.92)> lift:(1.27) lev:(0.03) conv:(3.27)
 4. [fruit=t, vegetables=t, biscuits=t, total=high]: 815 ==> [bread and cake=t]: 746   <conf:(0.92)> lift:(1.27) lev:(0.03) conv:(3.26)
 5. [fruit=t, party snack foods=t, total=high]: 854 ==> [bread and cake=t]: 779   <conf:(0.91)> lift:(1.27) lev:(0.04) conv:(3.15)
 6. [vegetables=t, frozen foods=t, biscuits=t, total=high]: 797 ==> [bread and cake=t]: 725   <conf:(0.91)> lift:(1.26) lev:(0.03) conv:(3.06)
 7. [vegetables=t, baking needs=t, biscuits=t, total=high]: 772 ==> [bread and cake=t]: 701   <conf:(0.91)> lift:(1.26) lev:(0.03) conv:(3.01)
 8. [fruit=t, biscuits=t, total=high]: 954 ==> [bread and cake=t]: 866   <conf:(0.91)> lift:(1.26) lev:(0.04) conv:(3)
 9. [fruit=t, vegetables=t, frozen foods=t, total=high]: 834 ==> [bread and cake=t]: 757   <conf:(0.91)> lift:(1.26) lev:(0.03) conv:(3)
10. [fruit=t, frozen foods=t, total=high]: 969 ==> [bread and cake=t]: 877   <conf:(0.91)> lift:(1.26) lev:(0.04) conv:(2.92)
```

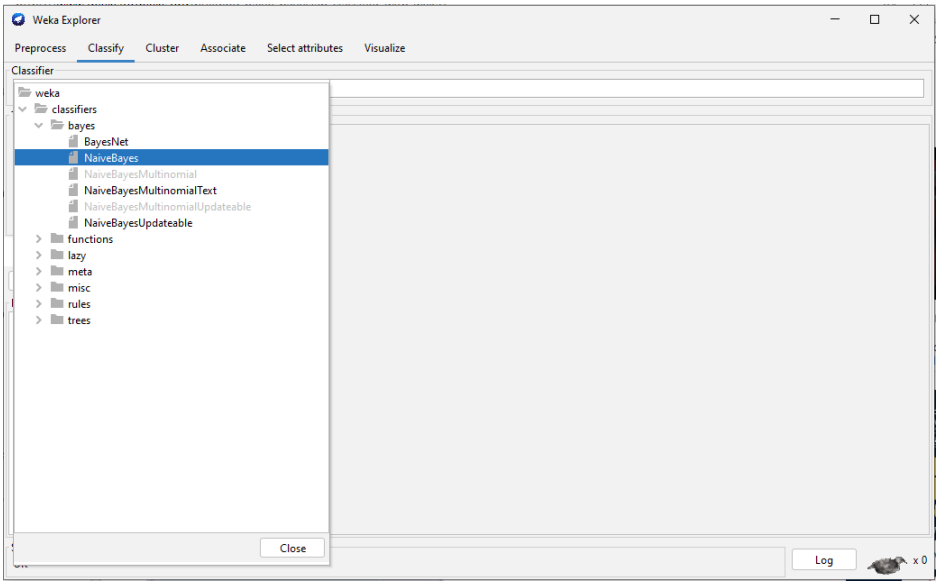
➢ CLASSIFICATION MODEL

Algorithm 1 : NaiveByes

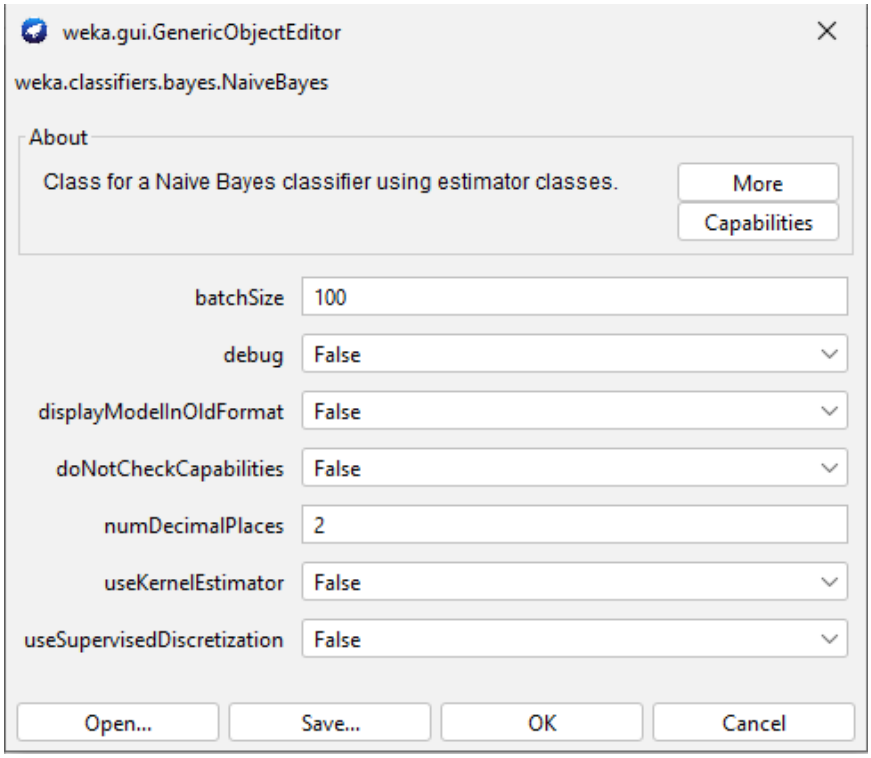
Step 1) Initially, we have to load the required dataset in the weka tool using choose file option. Here we are selecting the weather-nominal dataset to execute.



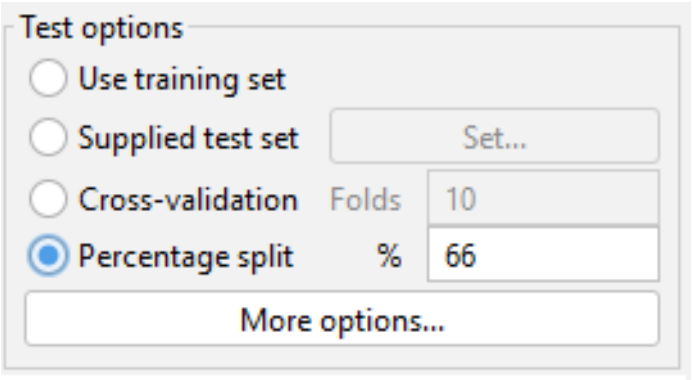
Step 2) Now we have to go to the classify tab on the top left side and click on the choose button and select the Naive Bayesian algorithm in it.



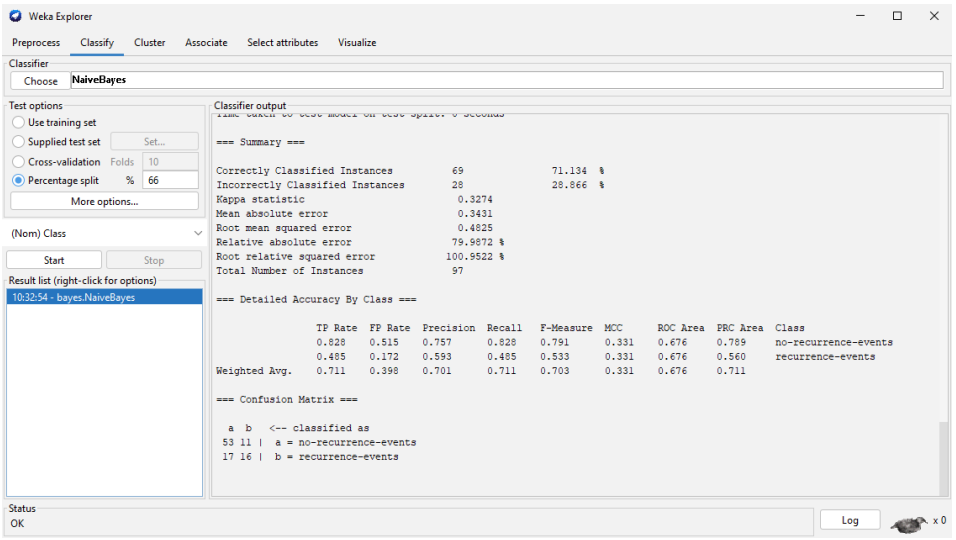
Step 3) Now to change the parameters click on the right side at the choose button, and we are accepting the default values in this example.



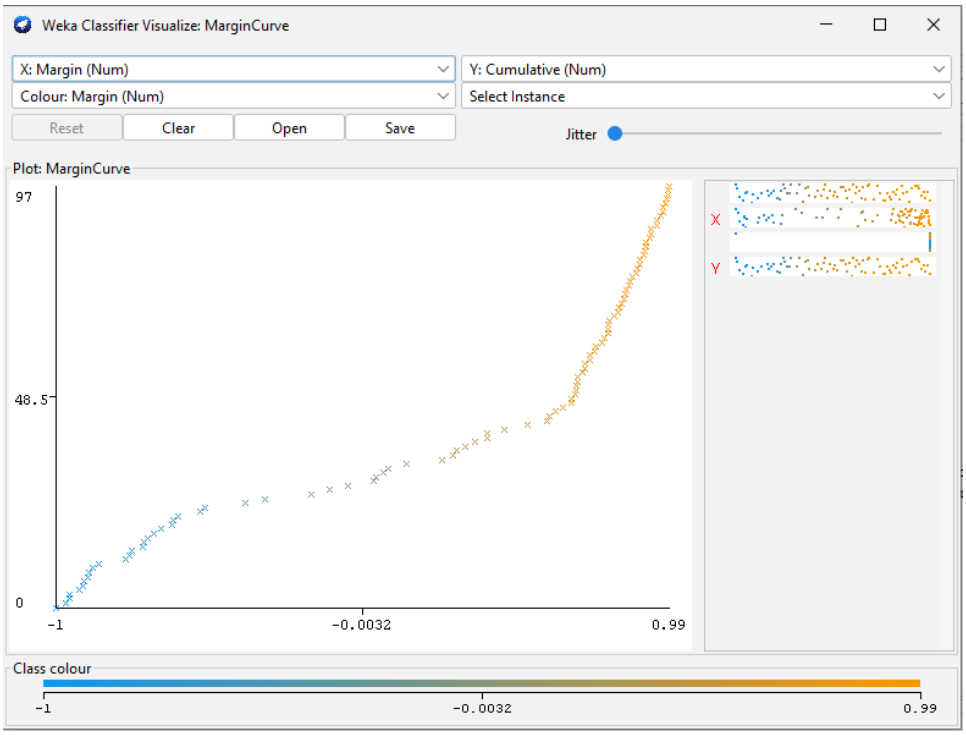
Step 4) We choose the Percentage split as our measurement method from the "Test" choices in the main panel. Since we don't have a separate test data collection, we'll use the percentage split of 66 percent to get a good idea of the model's accuracy. Our dataset contains 14 examples, with h9 being used for training and 5 being used for testing.



Step 5) To generate the model, we now click “start.” When the model is done, the evaluation statistic will appear in the right panel.

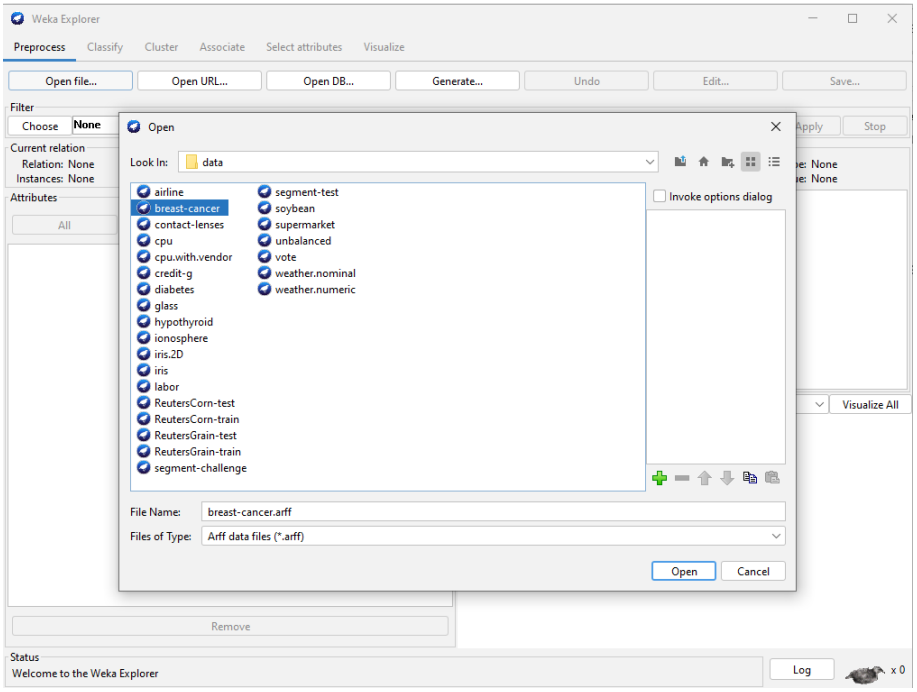


Step 6) Visualize the graph

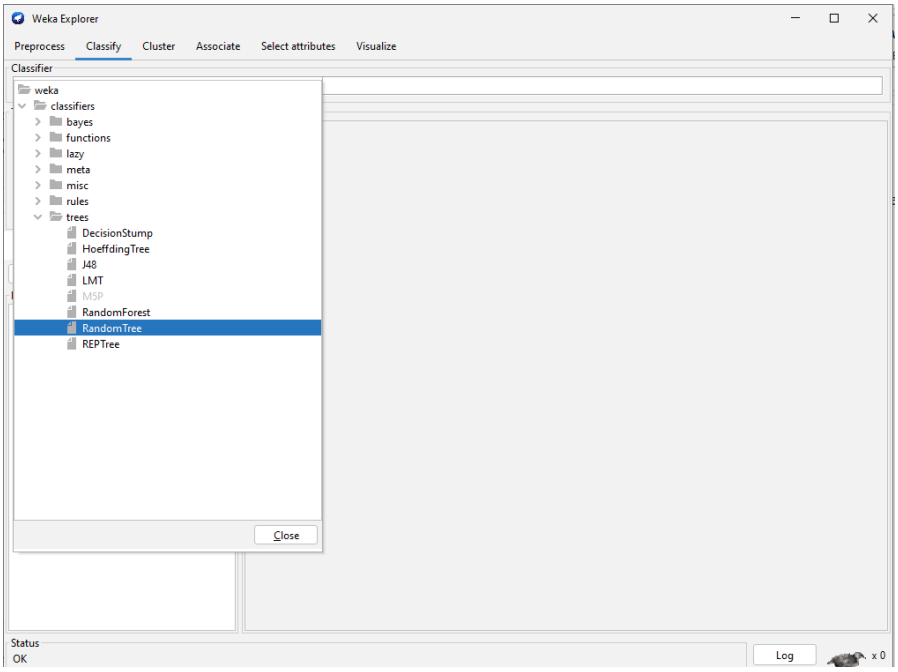


Algorithm 2: Random Tree

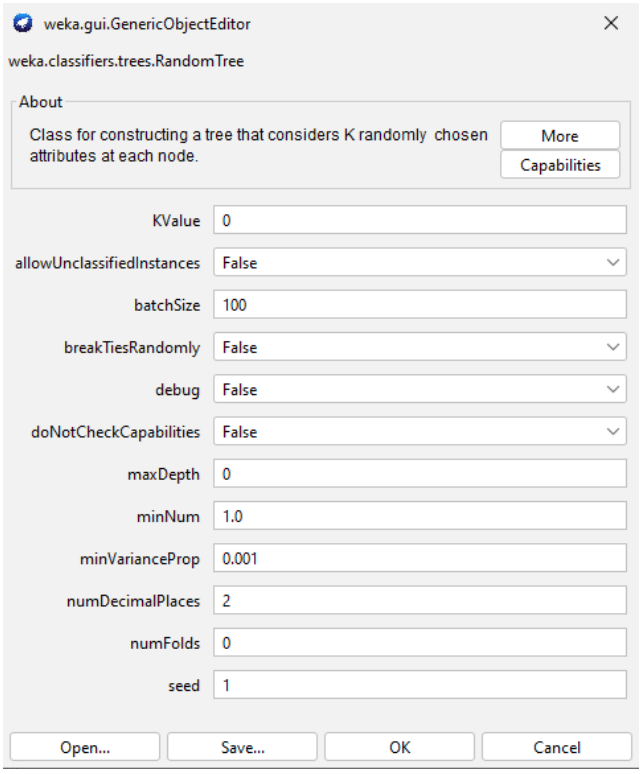
Step 1) Initially, we have to load the required dataset in the weka tool using choose file option. Here we are selecting the weather-nominal dataset to execute.



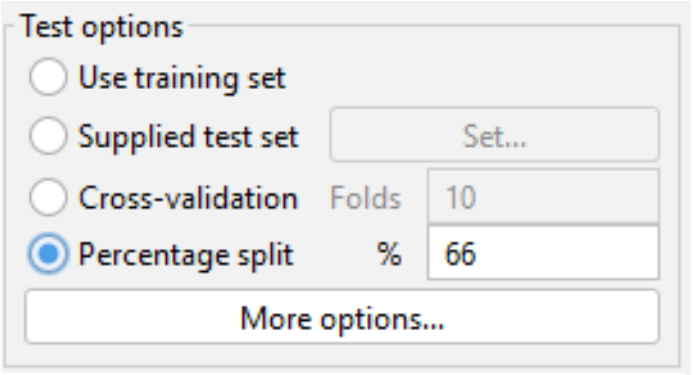
Step 2) Now we have to go to the classify tab on the top left side and click on the choose button and select the Random Tree algorithm in it.



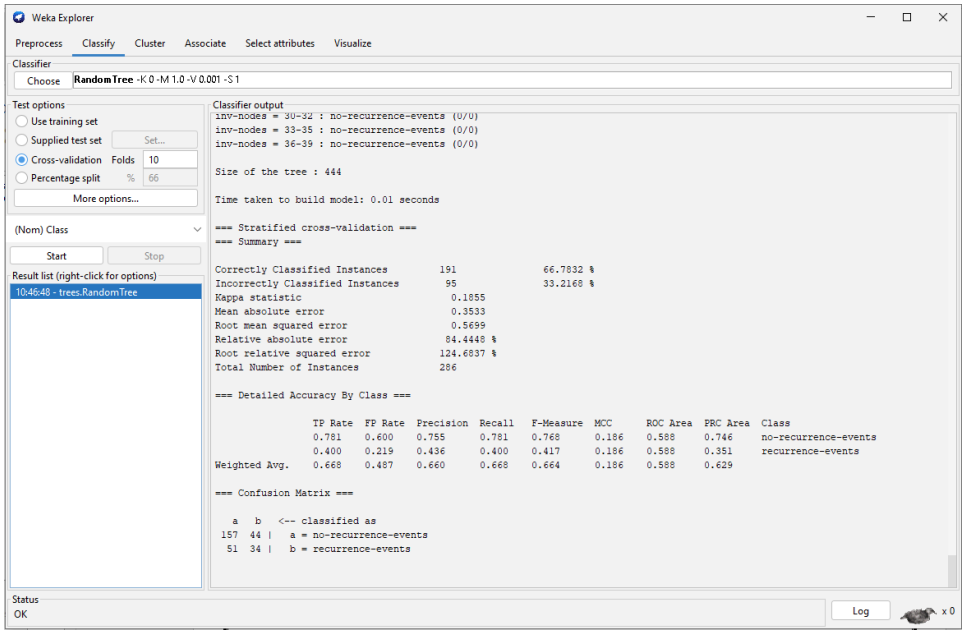
Step 3) Now to change the parameters click on the right side at the choose button, and we are accepting the default values in this example.



Step 4) We choose the Percentage split as our measurement method from the "Test" choices in the main panel. Since we don't have a separate test data collection, we'll use the percentage split of 66 percent to get a good idea of the model's accuracy. Our dataset contains 14 examples, with h9 being used for training and 5 being used for testing.

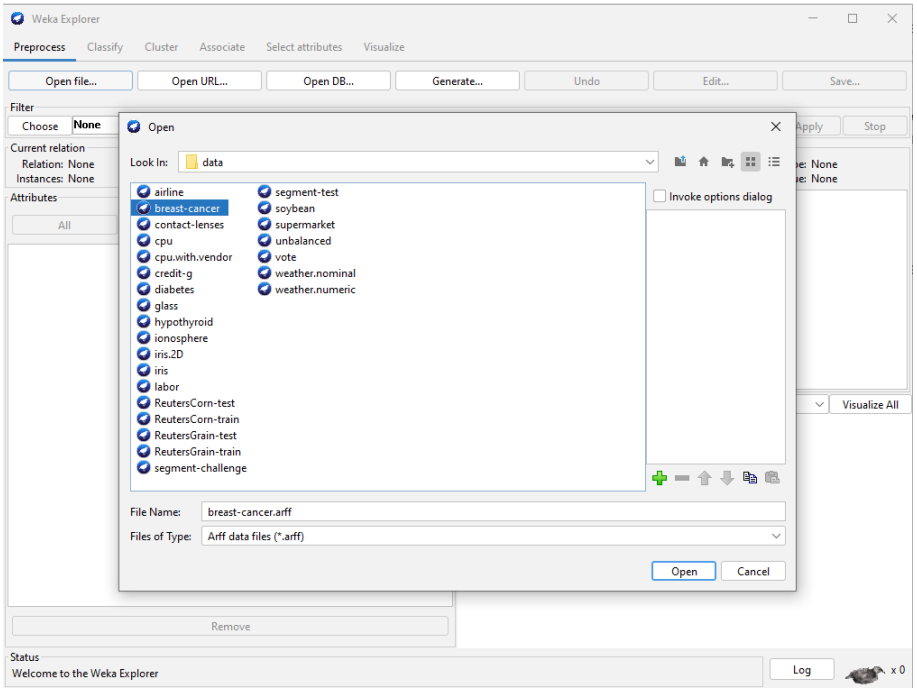


Step 5) To generate the model, we now click “start.” When the model is done, the evaluation statistic will appear in the right panel.

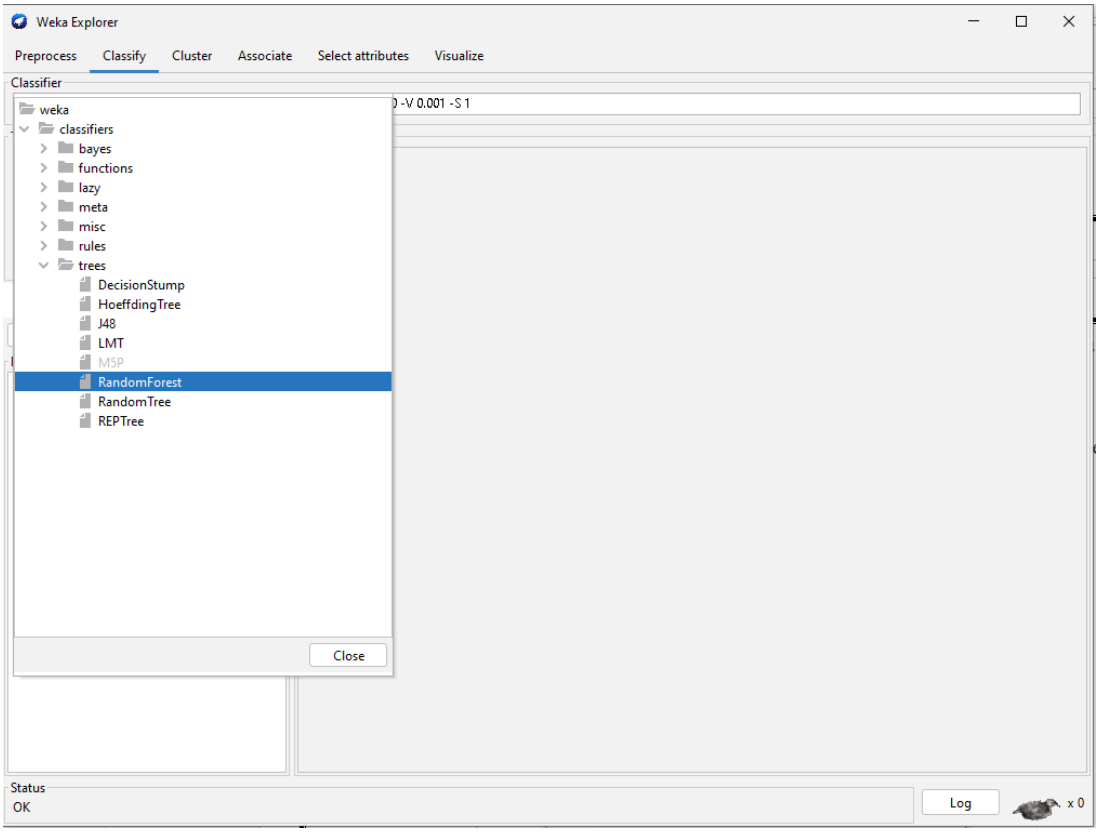


Algorithm 3: Random Forest

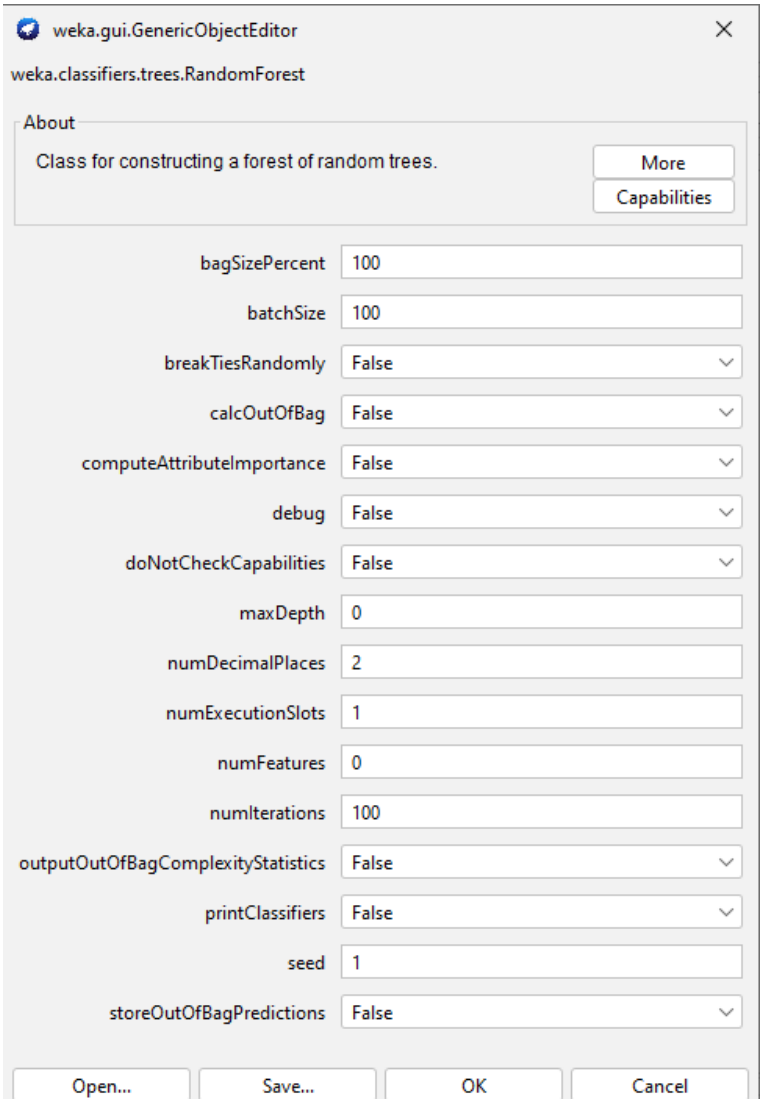
Step 1) Initially, we have to load the required dataset in the weka tool using choose file option. Here we are selecting the weather-nominal dataset to execute.



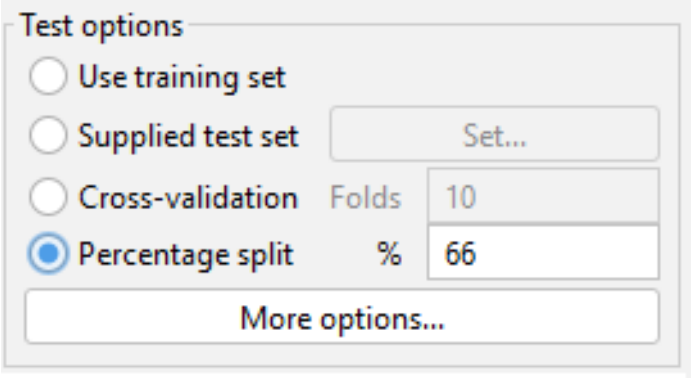
Step 2) Now we have to go to the classify tab on the top left side and click on the choose button and select the Random Tree algorithm in it.



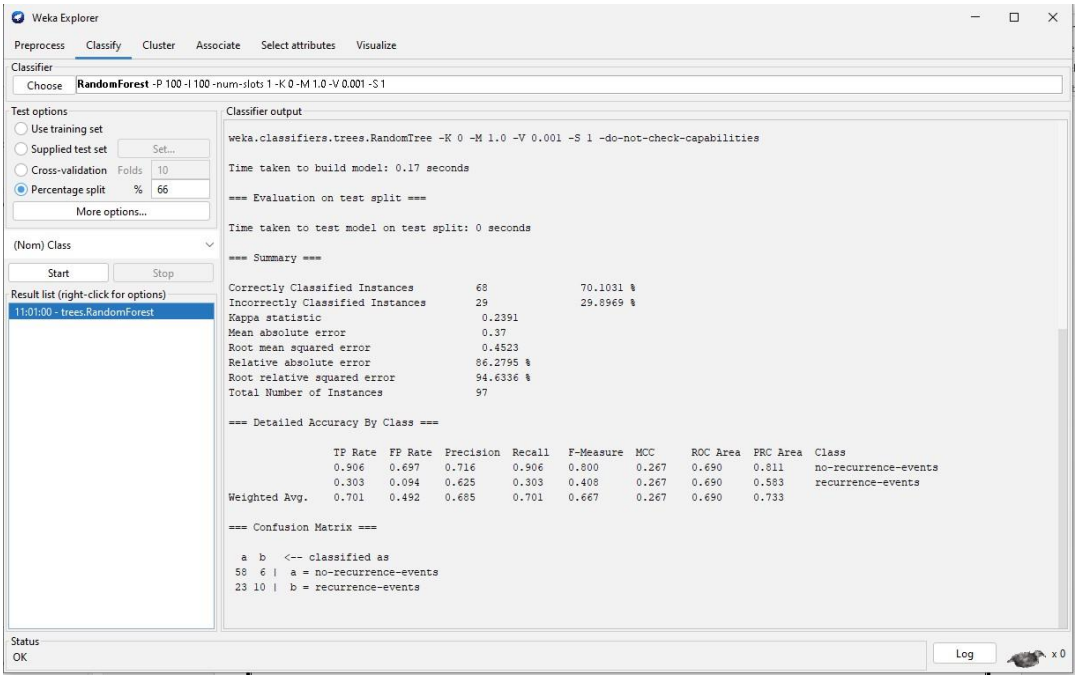
Step 3) Now to change the parameters click on the right side at the choose button, and we are accepting the default values in this example.



Step 4) We choose the Percentage split as our measurement method from the "Test" choices in the main panel. Since we don't have a separate test data collection, we'll use the percentage split of 66 percent to get a good idea of the model's accuracy. Our dataset contains 14 examples, with h9 being used for training and 5 being used for testing.



Step 5) To generate the model, we now click “start.” When the model is done, the evaluation statistic will appear in the right panel.



Practical: 10

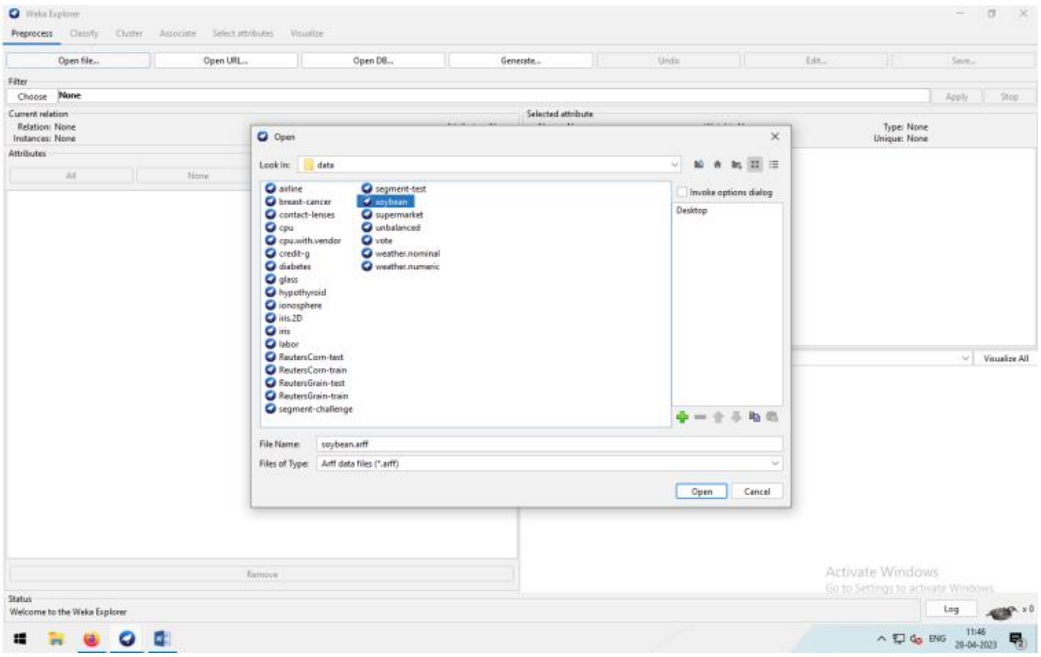
Aim: Calculating Information gains measures.

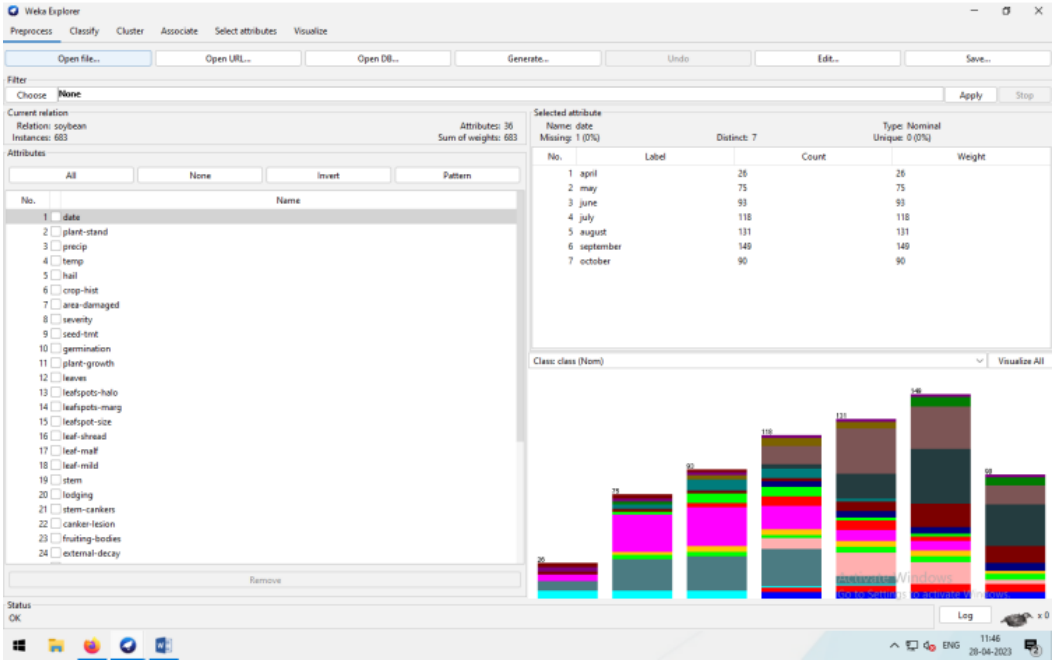
Code:

- Information gain (IG) measures how much "information" a feature gives us about the class. – Features that perfectly partition should give maximal information. – Unrelated features should give no information.
- It measures the reduction in entropy.
- CfsSubsetEval aims to identify a subset of attributes that are highly correlated with the target while not being strongly correlated with one another. It searches through the space of possible attribute subsets for the "best" one using theBestFirst search method by default, although other methods can be chosen.
- To use the wrapper method rather than a filter method, such as CfsSubsetEval, first select WrapperSubsetEval and then configure it by choosing a learning algorithm to apply and setting the number of crossvalidation folds to use when evaluating it on each attribute subset.
  Steps:

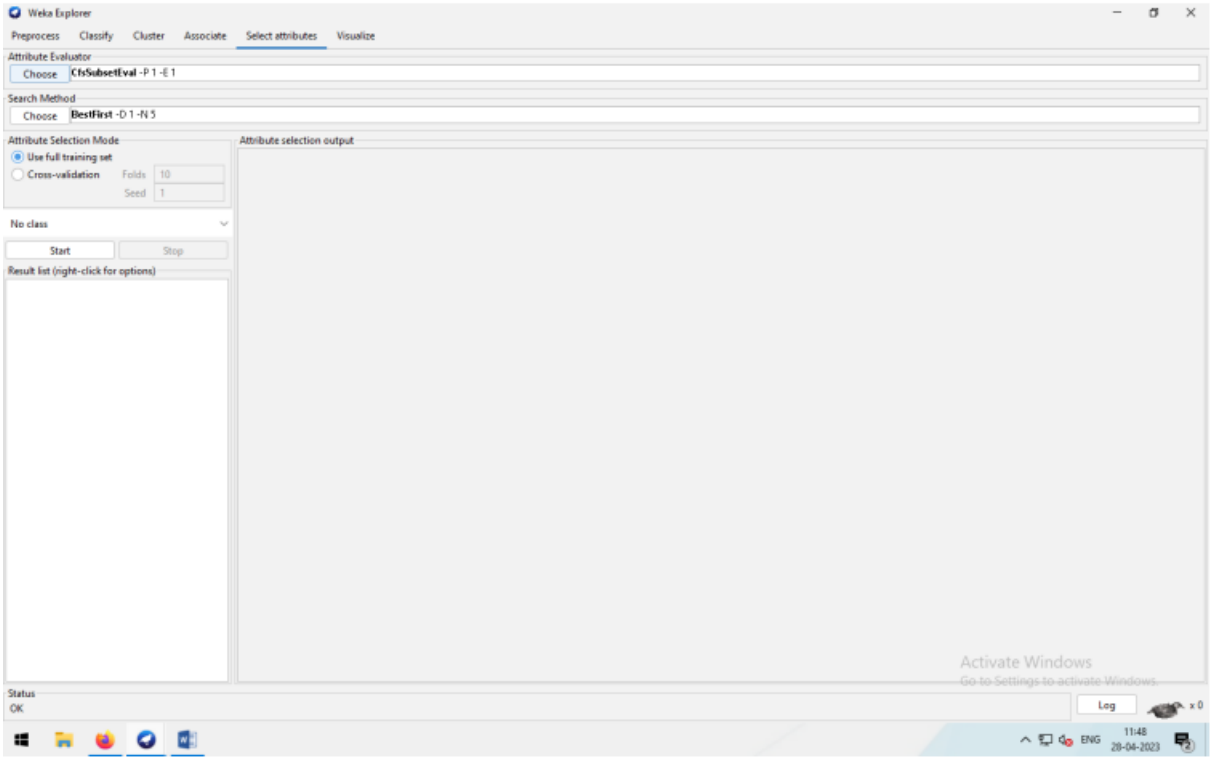
    - Open WEKA Tool.
    - Click on WEKA Explorer.
    - Click on Preprocessing tab button.
    - Click on open file button.
    - Select and Click on data option button.
    - Choose a data set and open file.
    - Click on select attribute tab and Choose attribute evaluator, search method algorithm

Step 1) In the WEKA explorer, open the **Preprocess** tab, click on the **Open file** ... button and select **supermarket.arff** database from the installation folder. After the data is loaded you will see the following screen –

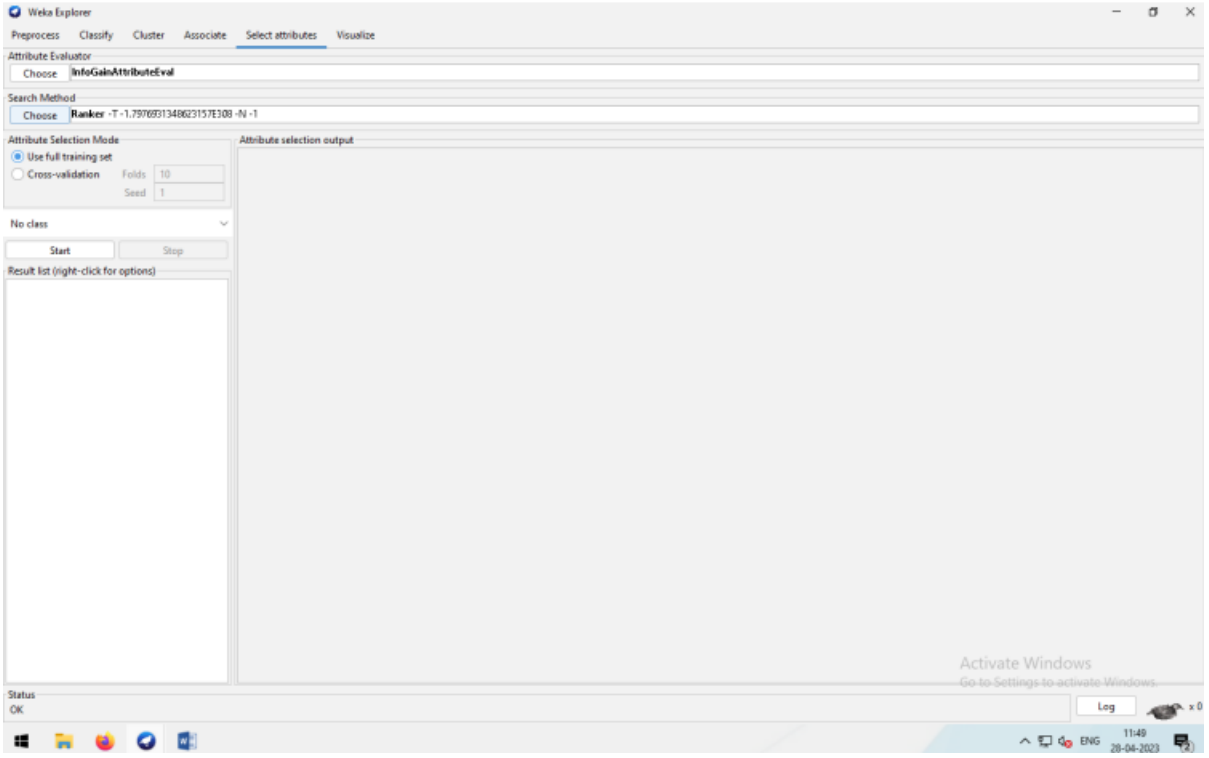




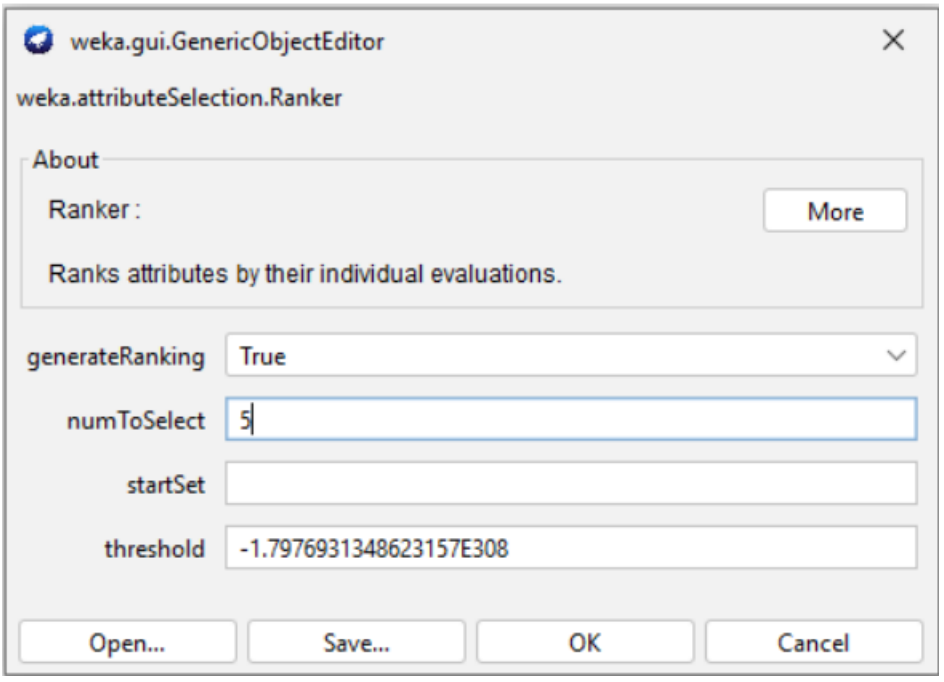
Step 2) Click on Select Attributes tab



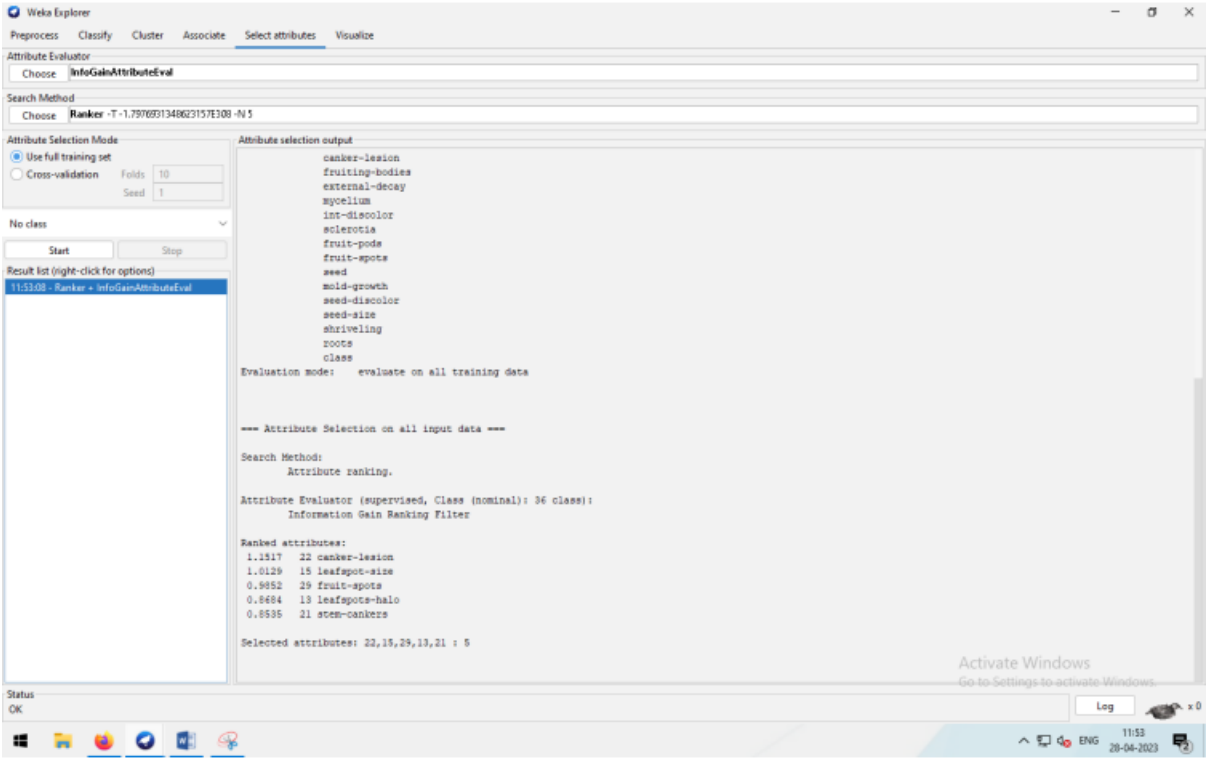
Step 3) Click on Choose [Attribute Evaluator - > InfoGainAttributeEval], [Search Method - > Ranker]



Step 4) Back to our task, where we need to find 5 and 10 best attributes, so first I will show how to get the best 5 attributes, then we just need to repeat step 5 to do for 10 best attributes. Click on Ranker, type 5 in numToSelect. Then click okay.



Step 5) - For Attribute Selection Mode, by default it will tick Use for training set, so don't change anything. Click Start. Then it will display the result at the right hand side.



From the result, we get 5 best attributes based on Information Gain and Ranker search method.

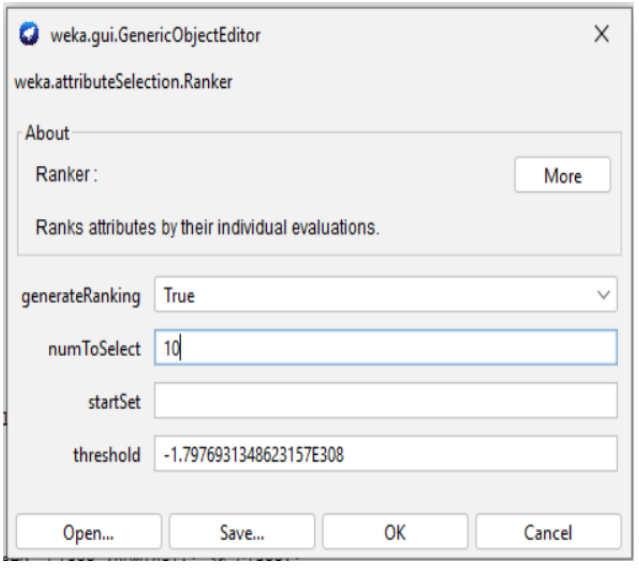
```
Ranked attributes:
 1.1517   22 canker-lesion
 1.0129   15 leafspot-size
 0.9852   29 fruit-spots
 0.8684   13 leafspots-halo
 0.8535   21 stem-cankers
```

Output:

Attribute selection output

```
                canker-lesion
                fruiting-bodies
                external-decay
                mycelium
                int-discolor
                sclerotia
                fruit-pods
                fruit-spots
                seed
                mold-growth
                seed-discolor
                seed-size
                shriveling
                roots
                class
Evaluation mode:    evaluate on all training data


=== Attribute Selection on all input data ===

Search Method:
        Attribute ranking.

Attribute Evaluator (supervised, Class (nominal): 36 class):
        Information Gain Ranking Filter

Ranked attributes:
 1.1517   22 canker-lesion
 1.0129   15 leafspot-size
 0.9852   29 fruit-spots
 0.8684   13 leafspots-halo
 0.8535   21 stem-cankers

Selected attributes: 22,15,29,13,21 : 5
```

Step 6)



Step 7)

```
Ranked attributes:
 1.1517   22 canker-lesion
 1.0129   15 leafspot-size
 0.9852   29 fruit-spots
 0.8684   13 leafspots-halo
 0.8535   21 stem-cankers
 0.8504   14 leafspots-marg
 0.8437   28 fruit-pods
 0.6918   19 stem
 0.6715    1 date
 0.6265   11 plant-growth
```

Step 8)

Attribute selection output

```
              sclerotia
              fruit-pods
              fruit-spots
              seed
              mold-growth
              seed-discolor
              seed-size
              shriveling
              roots
              class
Evaluation mode:    evaluate on all training data



=== Attribute Selection on all input data ===

Search Method:
        Attribute ranking.

Attribute Evaluator (supervised, Class (nominal): 36 class):
        Information Gain Ranking Filter

Ranked attributes:
 1.1517   22 canker-lesion
 1.0129   15 leafspot-size
 0.9852   29 fruit-spots
 0.8684   13 leafspots-halo
 0.8535   21 stem-cankers
 0.8504   14 leafspots-marg
 0.8437   28 fruit-pods
 0.6918   19 stem
 0.6715    1 date
 0.6265   11 plant-growth

Selected attributes: 22,15,29,13,21,14,28,19,1,11 : 10
```