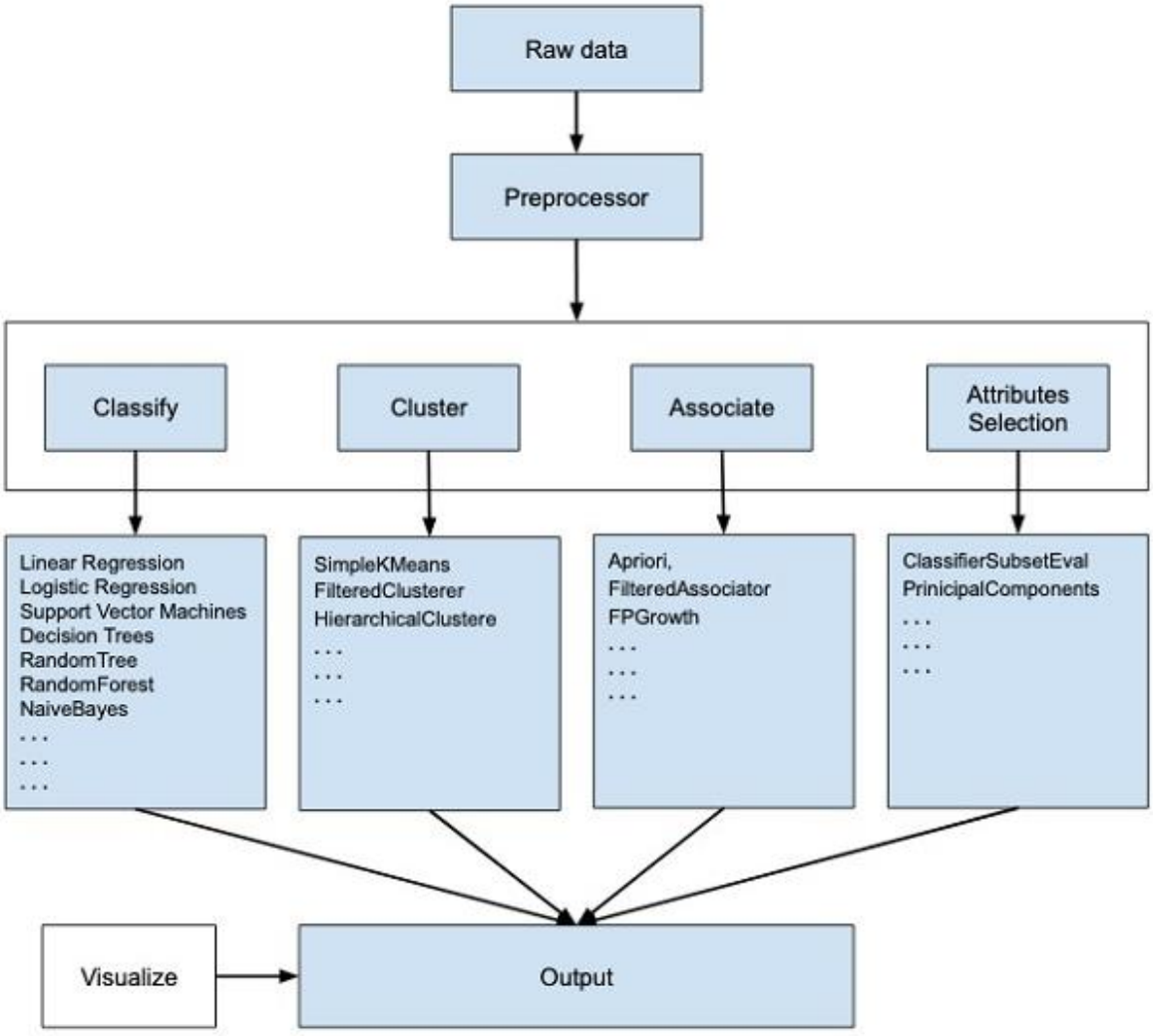
# Weka

## Introduction

WEKA - an open source software provides tools for data preprocessing, implementation of several Machine Learning algorithms, and visualization tools so that you can develop machine learning techniques and apply them to real-world data mining problems.
Below is the flowchart of what WEKA offers



First, you will start with the raw data collected from the field. This data may contain several null values and irrelevant fields. You use the data preprocessing tools provided in WEKA to cleanse the data.

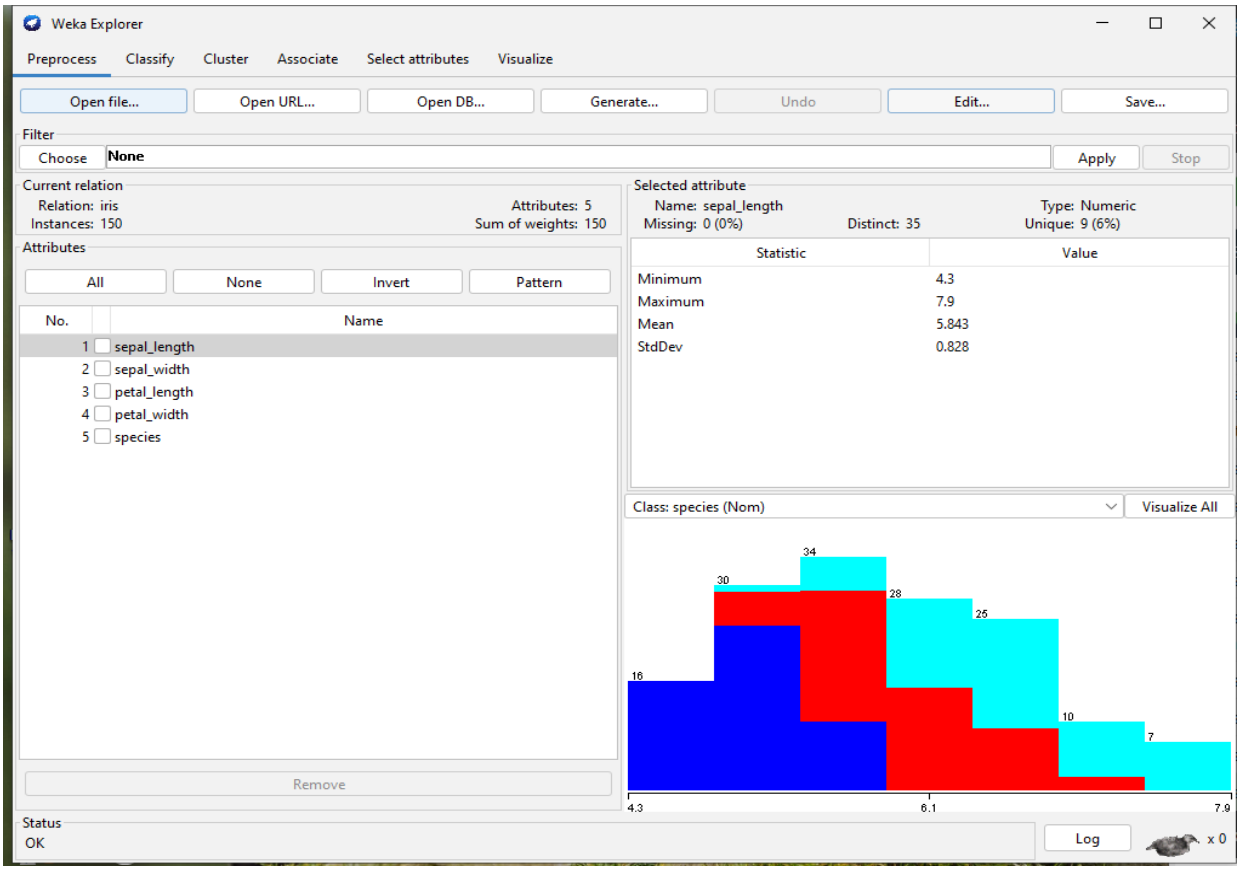
Then, you would save the preprocessed data in your local storage for applying ML algorithms.

Next, depending on the kind of ML model that you are trying to develop you would select one of the options such as **Classify, Cluster,** or **Associate**. The Attributes Selection allows the automatic selection of features to create a reduced dataset.

Then, WEKA would give you the statistical output of the model processing. It provides you a visualization tool to inspect the data.

The various models can be applied on the same dataset. You can then compare the outputs of different models and select the best that meets your purpose.

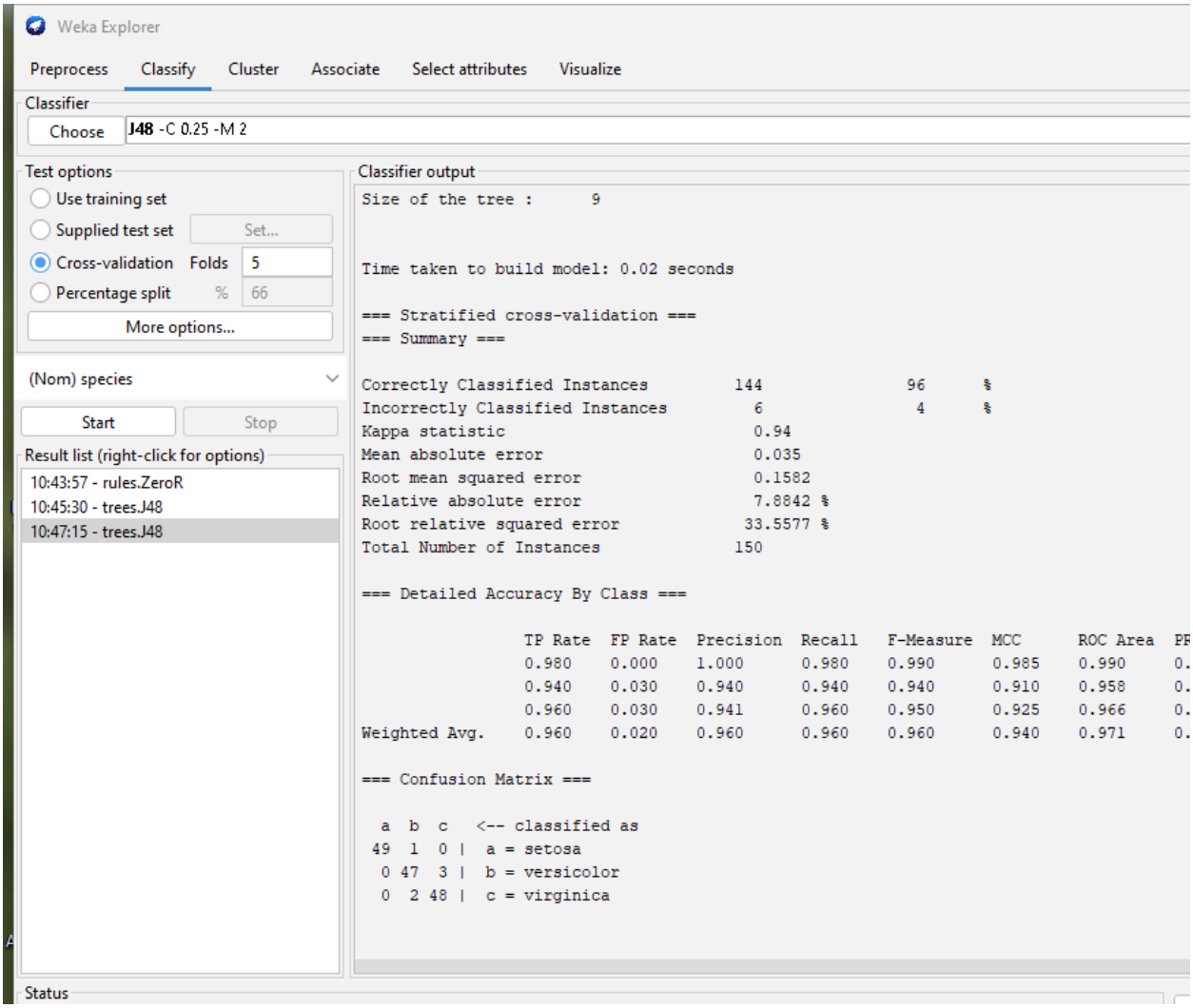
Below we are performing the Classification using J48 algorithms and clustering using EM algorithms which are inbuilt algorithms



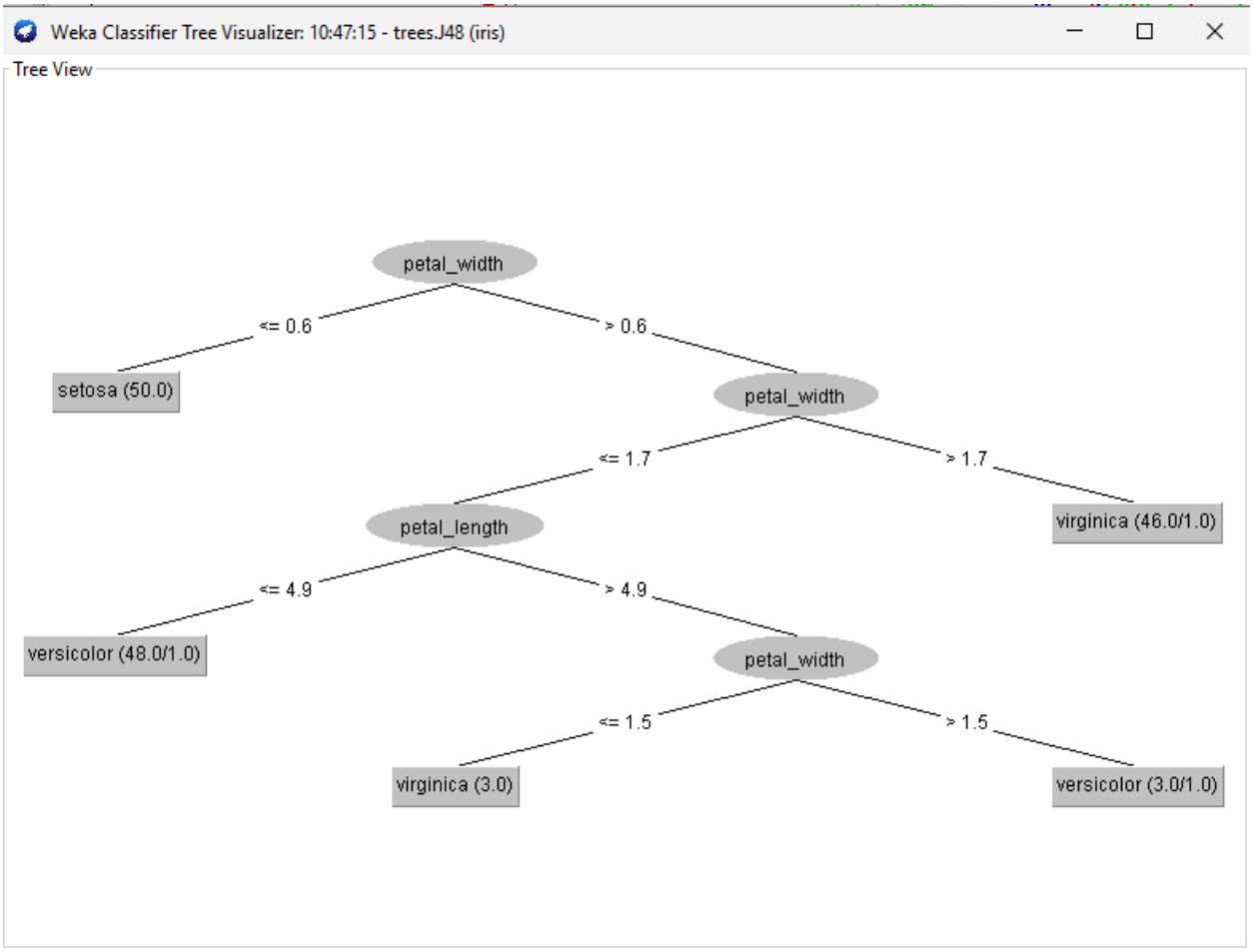
There are 150 instances and 5 attributes. The names of attributes are listed as sepallength, sepalwidth, petallength, petalwidth and class. The first four attributes are of numeric type while the class is a nominal type with 3 distinct values.

We will not do any preprocessing on this data and straight-away proceed to model building.

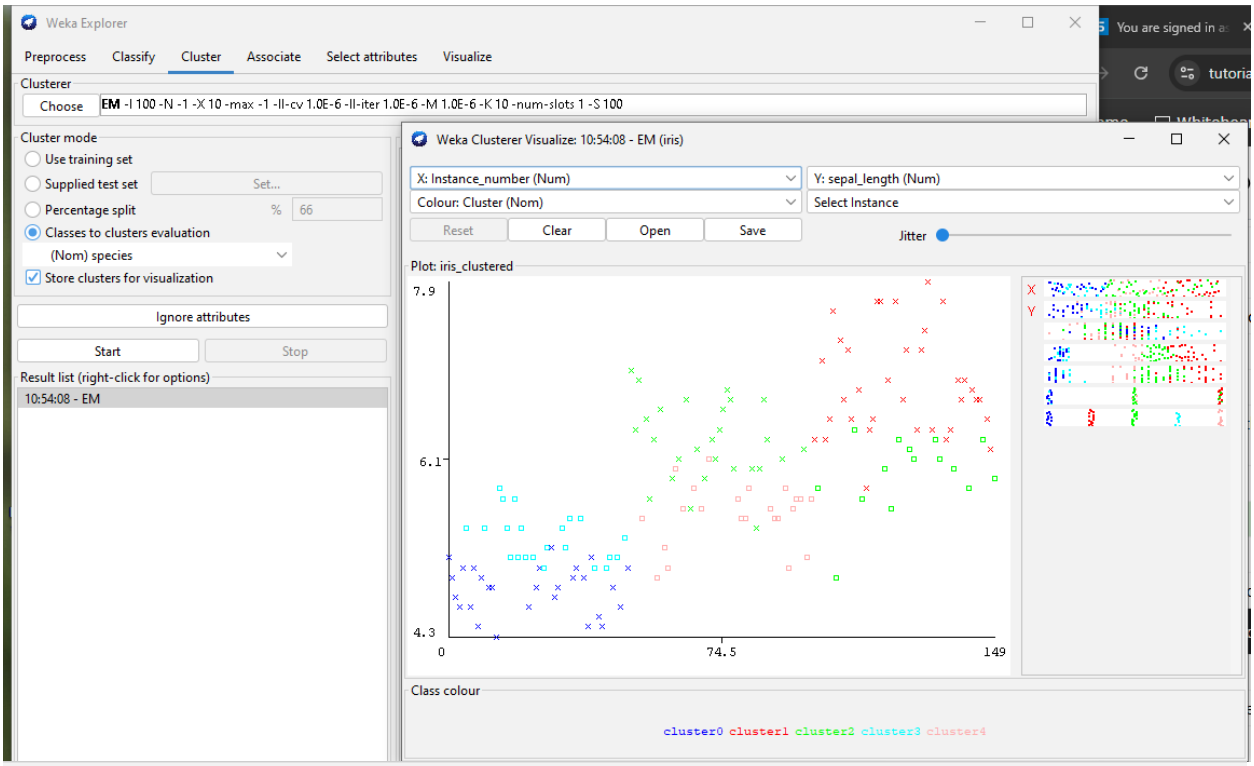
## Working on IRIS dataset using J48 Classifier



Tree Visualizer:



# Working on IRIS dataset using EM Cluster




Clusterer output

```
=== Run information ===

Scheme:       weka.clusterers.EM -I 100 -N -1 -X 10 -max -1 -ll-cv 1.0E-6 -ll
Relation:     iris
Instances:    150
Attributes:   5
              sepal_length
              sepal_width
              petal_length
              petal_width
Ignored:
              species
Test mode:    Classes to clusters evaluation on training data

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


EM
==

Number of clusters selected by cross validation: 5
Number of iterations performed: 16


                Cluster
Attribute            0       1       2       3       4
                  (0.18)  (0.23)  (0.28)  (0.15)  (0.15)
============================================================
sepal_length
  mean           4.7748  6.8585  6.1613  5.2823  5.5432
```

Clusterer output

```
==

Number of clusters selected by cross validation: 5
Number of iterations performed: 16


               Cluster
Attribute            0       1       2       3       4
                (0.18)  (0.23)  (0.28)  (0.15)  (0.15)
=========================================================
sepal_length
  mean           4.7748  6.8585  6.1613  5.2823  5.5432
  std. dev.      0.2405  0.5228  0.4138  0.2407  0.3159

sepal_width
  mean           3.1789  3.0862  2.8547  3.7037  2.5786
  std. dev.      0.2599  0.2891  0.2687  0.2857  0.2512

petal_length
  mean           1.4194  5.7859  4.7484  1.5173   3.863
  std. dev.      0.1692  0.4745  0.3193  0.1592  0.3516

petal_width
  mean           0.1948  2.1327  1.5757  0.3028  1.1696
  std. dev.      0.0557  0.2359  0.2196  0.1212  0.1351



Time taken to build model (full training data) : 0.22 seconds

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

Clustered Instances
```

Clusterer output

```
Time taken to build model (full training data) : 0.22 seconds

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

Clustered Instances

0       28 ( 19%)
1       35 ( 23%)
2       42 ( 28%)
3       22 ( 15%)
4       23 ( 15%)


Log likelihood: -1.60803


Class attribute: species
Classes to Clusters:

  0  1  2  3  4  <-- assigned to cluster
 28  0  0 22  0 | setosa
  0  0 27  0 23 | versicolor
```