Data Preprocessing Applying Filters Example

WEKA – Feature selection

When a database contains a large number of attributes, there will be several attributes which do not become significant in the analysis that you are currently seeking.

Thus, removing the unwanted attributes from the dataset becomes an important task in developing a good machine learning model.

WEKA — Feature selection

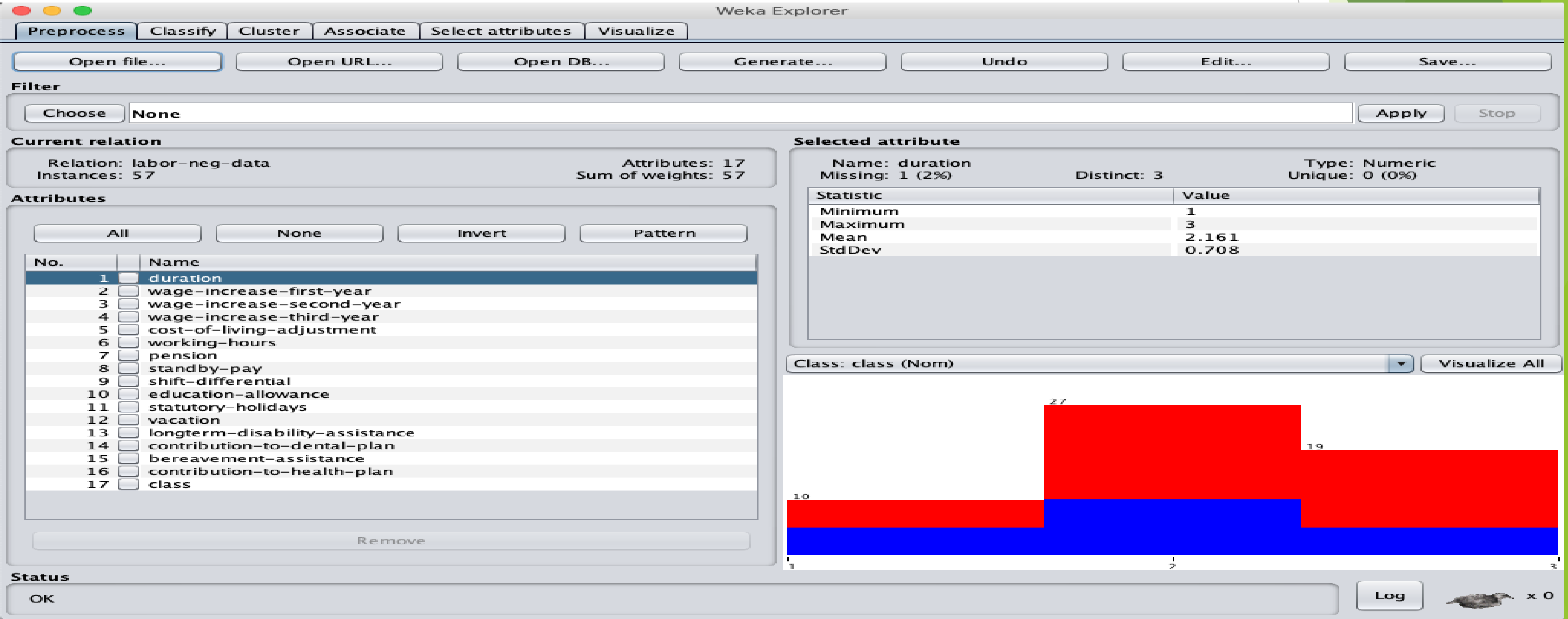
You may examine the entire dataset visually and decide on the irrelevant attributes. This could be a huge task for databases containing a large number of attributes like the supermarket case that you saw in an earlier discussion.

WEKA provides an automated tool for feature selection. To demonstrate this feature on a database containing a large number of attributes.

In the Preprocess tag of the WEKA explorer, select the labor.arff file for loading into the system.

WEKA – Feature selection

When you load the data, you will see the following screen:Notice that there are 17 attributes. Our task is to create a reduced dataset by eliminating some of the attributes which are irrelevant to our analysis.



Filter: None

Current relation

Relation: labor-neg-data
Instances: 57
Attributes: 17
Sum of weights: 57

Selected attribute

Name: duration
Missing: 1 (2%)
Distinct: 3
Type: Numeric
Unique: 0 (0%)

Statistic	Value
Minimum	1
Maximum	3
Mean	2.161
StdDev	0.708

Attributes

No.		Name
1	□	duration
2	□	wage-increase-first-year
3	□	wage-increase-second-year
4	□	wage-increase-third-year
5	□	cost-of-living-adjustment
6	□	working-hours
7	□	pension
8	□	standby-pay
9	□	shift-differential
10	□	education-allowance
11	□	statutory-holidays
12	□	vacation
13	□	longterm-disability-assistance
14	□	contribution-to-dental-plan
15	□	bereavement-assistance
16	□	contribution-to-health-plan
17	□	class

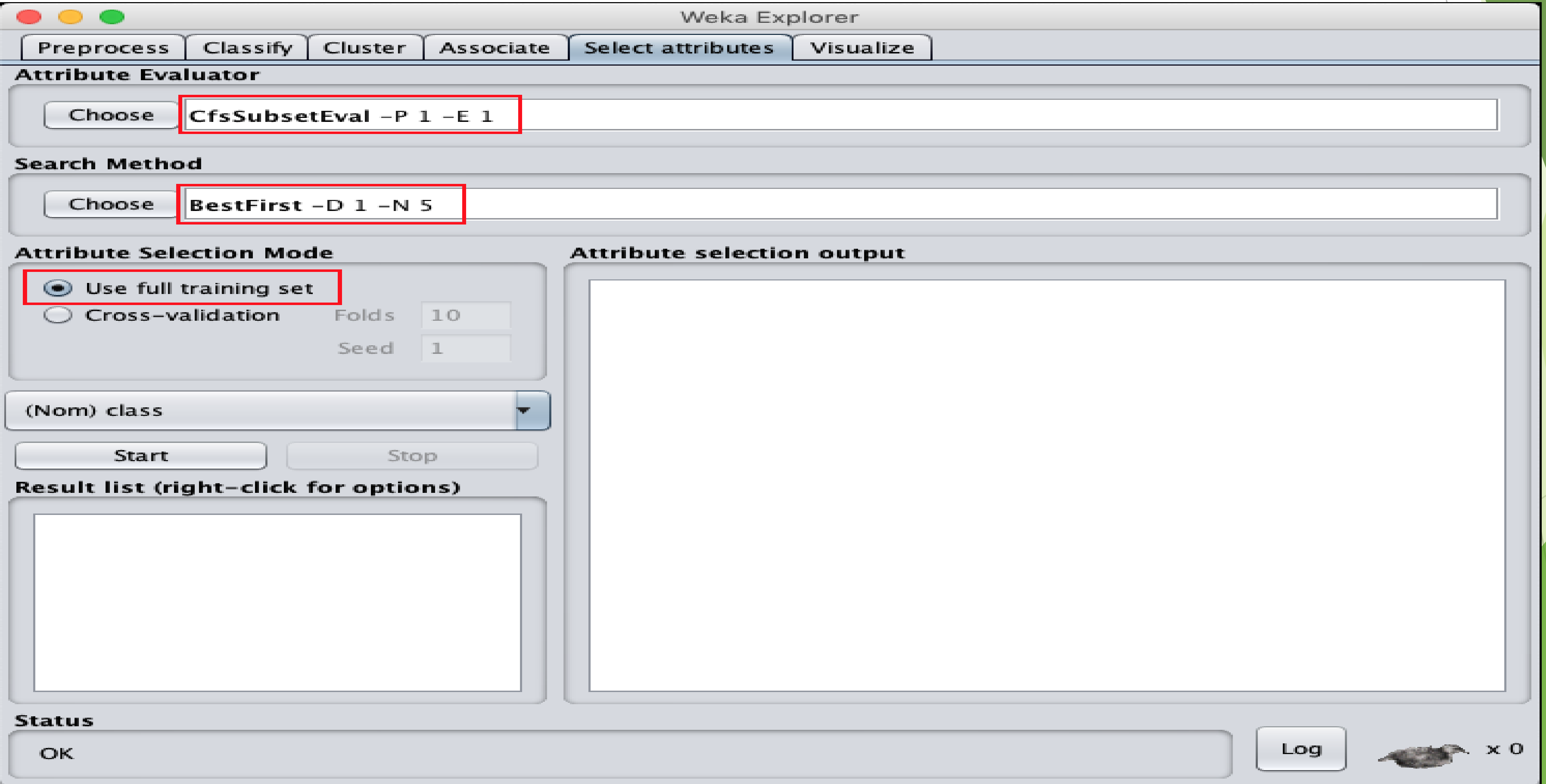
Class: class (Nom)

Status

OK

WEKA – Feature selection

Click on the **Select attributes** TAB. You will see the following screen:



Attribute Evaluator

CfsSubsetEval -P 1 -E 1

Search Method

BestFirst -D 1 -N 5

Attribute Selection Mode

● Use full training set

○ Cross-validation Folds 10

Seed 1

(Nom) class

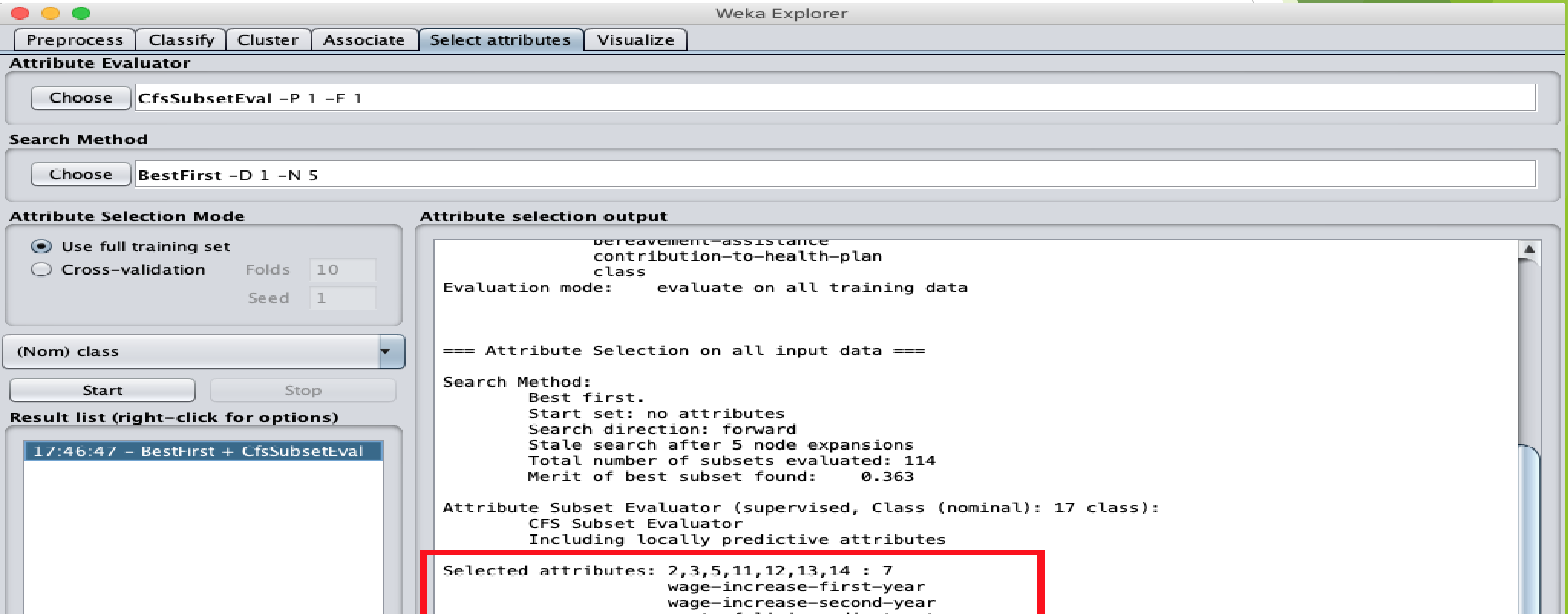
Start
Stop
Result list (right-click for options)
Attribute selection output

Status

OK

WEKA – Feature selection

Under the **Attribute Evaluator** and **Search Method**, you will find several options. We will just use the defaults here. In the **Attribute Selection Mode**, use full training set option. Click on the **Start** button to process the dataset. You will see the following output:



Attribute Evaluator

CfsSubsetEval -P 1 -E 1

Search Method

BestFirst -D 1 -N 5

Attribute Selection Mode

● Use full training set

○ Cross-validation Folds 10

Seed 1

(Nom) class

Result list (right-click for options)

17:46:47 - BestFirst + CfsSubsetEval

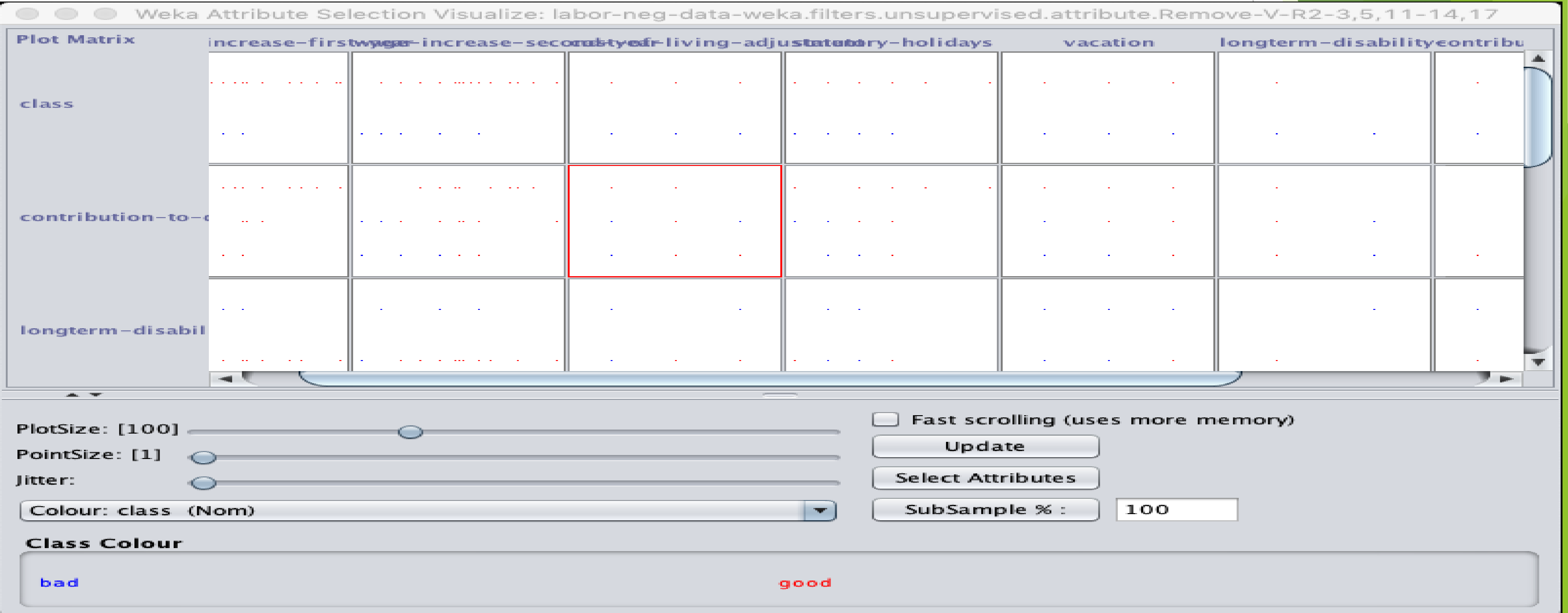
Attribute selection output

```
                    bereavement-assistance
                    contribution-to-health-plan
                    class
Evaluation mode:    evaluate on all training data



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

Search Method:
        Best first.
        Start set: no attributes
        Search direction: forward
        Stale search after 5 node expansions
        Total number of subsets evaluated: 114
        Merit of best subset found:    0.363

Attribute Subset Evaluator (supervised, Class (nominal): 17 class):
        CFS Subset Evaluator
        Including locally predictive attributes

Selected attributes: 2,3,5,11,12,13,14 : 7
                     wage-increase-first-year
                     wage-increase-second-year
```

WEKA – Feature selection

At the bottom of the result window, you will get the list of **Selected** attributes. To get the visual representation, right click on the result in the **Result** list.

The output is shown in the following screenshot: This is similar to the ones we have seen in the earlier chapters.



Weka Attribute Selection Visualize: labor-neg-data-weka.filters.unsupervised.attribute.Remove-V-R2-3,5,11-14,17

PlotSize: [100]

PointSize: [1]

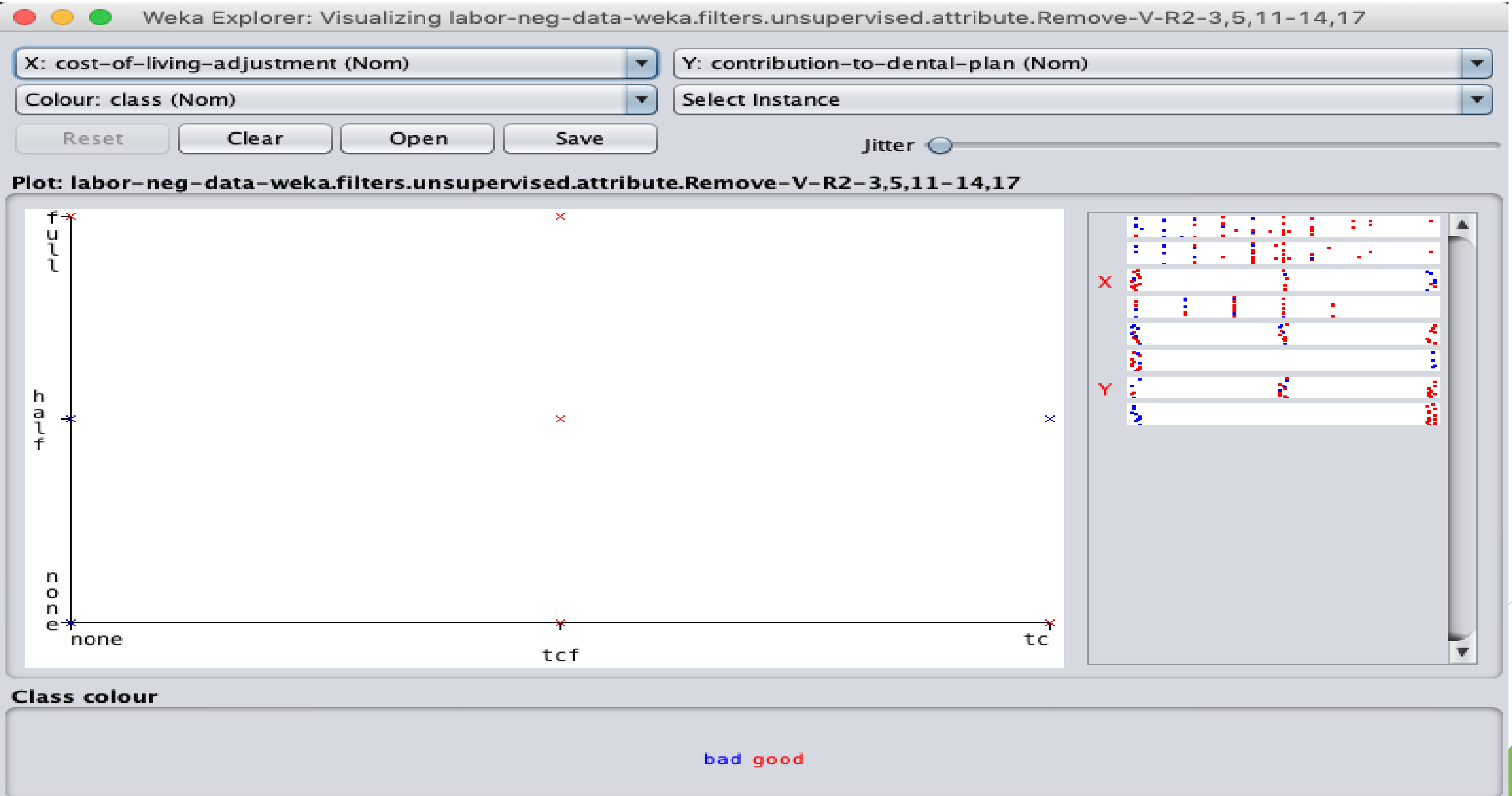
Jitter:

Colour: class (Nom)

SubSample % : 100

WEKA – Feature selection

Clicking on any of the squares will give you the data plot for your further analysis. A typical data plot is shown below: This is similar to the ones we have seen in the earlier slides in previous lab sesssions. Play around with the different options available to analyze the results.



Weka Explorer: Visualizing labor-neg-data-weka.filters.unsupervised.attribute.Remove-V-R2-3,5,11-14,17

X: cost-of-living-adjustment (Nom)

Y: contribution-to-dental-plan (Nom)

Colour: class (Nom)

Plot: labor-neg-data-weka.filters.unsupervised.attribute.Remove-V-R2-3,5,11-14,17

Class colour

WEKA — Clustering

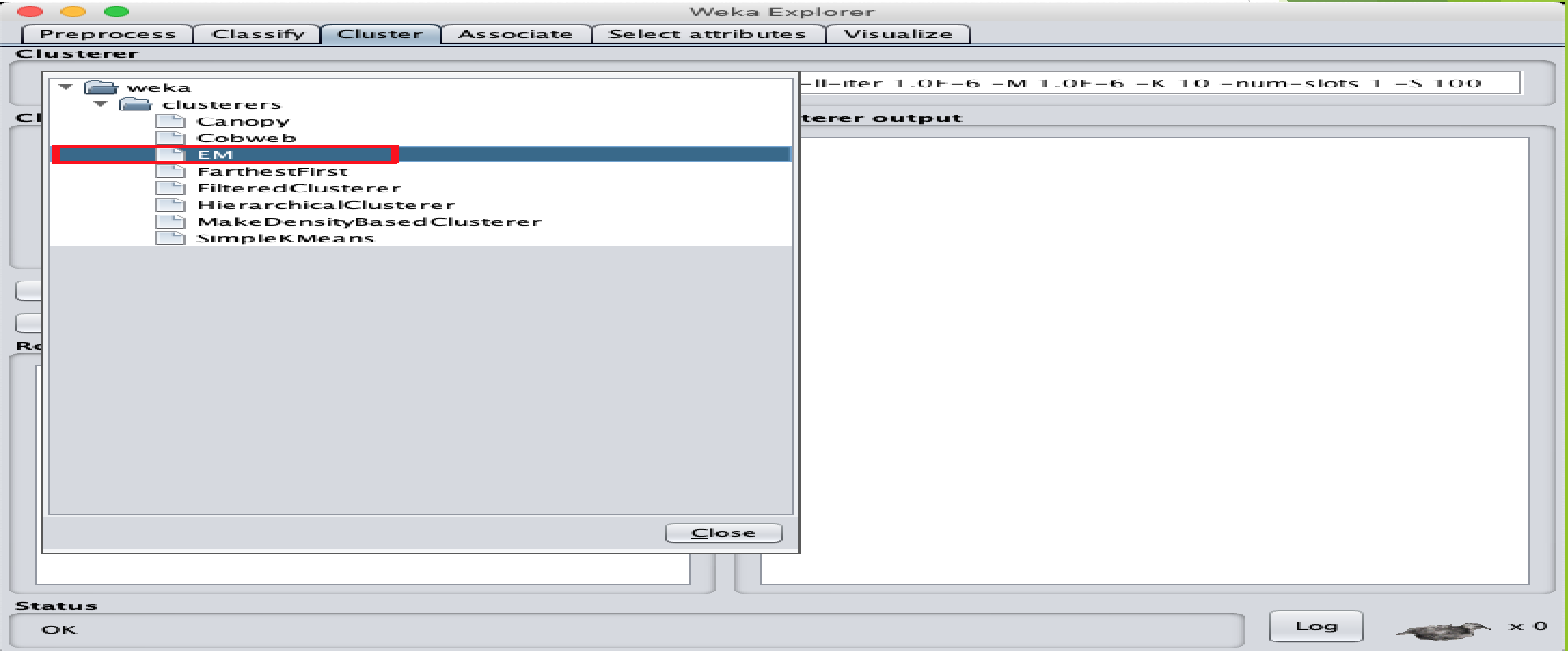
You can observe that there are 150 instances and 5 attributes. The names of attributes are listed as **sepallength**, **sepalwidth**, **petallength**, **petalwidth** and **class**.

The first four attributes are of numeric type while the **class** is a nominal type with 3 distinct values.

Examine each attribute to understand the features of the database. We will not do any preprocessing on this data and straight-away proceed to model building.

WEKA – Clustering

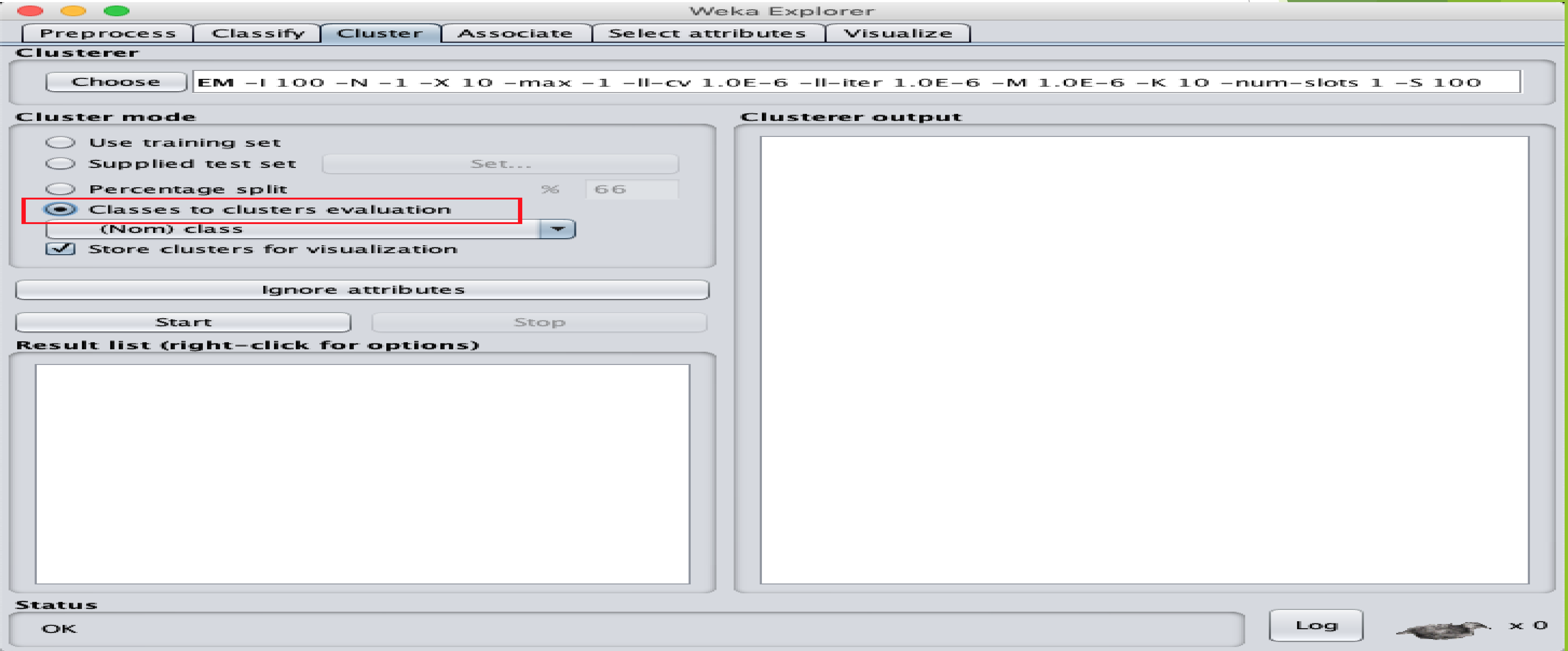
Clustering - Click on the **Cluster** TAB to apply the clustering algorithms to our loaded data. Click on the **Choose** button. You will see the following screen:



-ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100

WEKA – Clustering

Now, select **EM** as the clustering algorithm. In the **Cluster mode** sub window, select the **Classes to clusters evaluation** option as shown in the screenshot below:



Clusterer

EM -I 100 -N -1 -X 10 -max -1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100

Cluster mode

○ Use training set
○ Supplied test set
○ Percentage split % 66
● Classes to clusters evaluation
(Nom) class
☑ Store clusters for visualization

Clusterer output

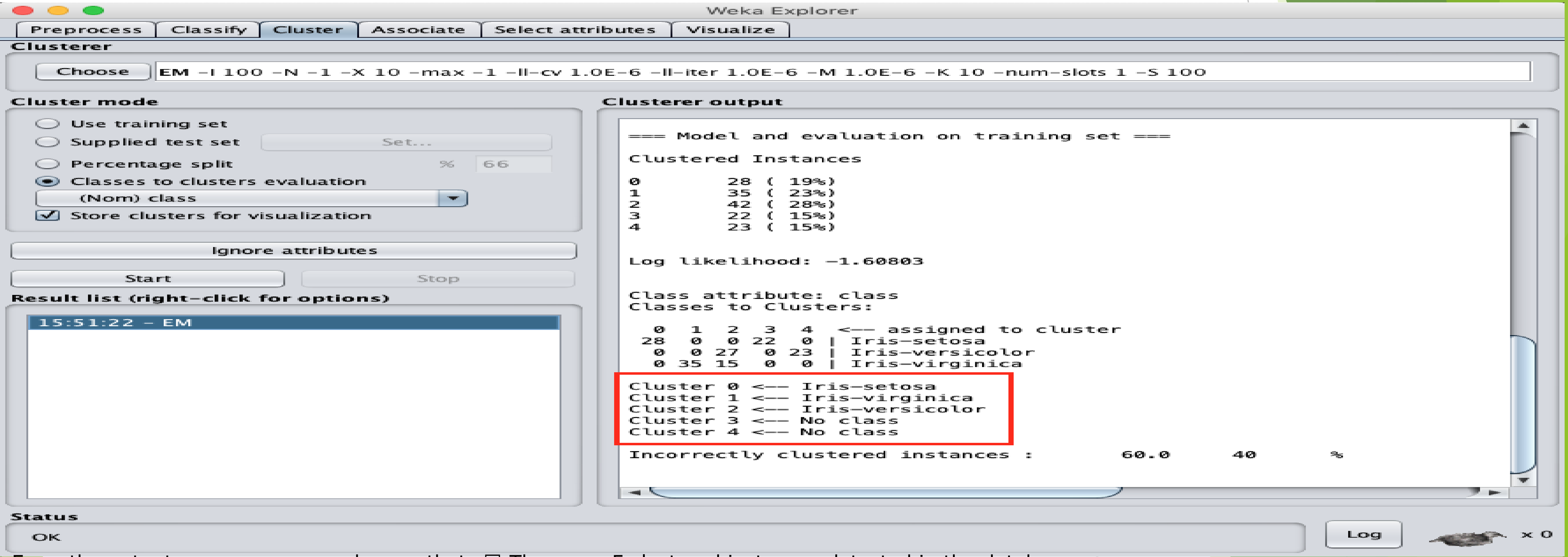
Result list (right-click for options)

Status

OK

WEKA – Clustering

Click on the **Start** button to process the data. After a while, the results will be presented on the screen. Next, let us study the results. The output of the data processing is shown in the screen below:



Clusterer

EM -I 100 -N -1 -X 10 -max -1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100

Cluster mode

○ Use training set

○ Supplied test set

○ Percentage split % 66

● Classes to clusters evaluation

(Nom) class

☑ Store clusters for visualization

Result list (right-click for options)

15:51:22 - EM

Clusterer output

```
=== 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: class
Classes to Clusters:

  0  1  2  3  4  <-- assigned to cluster
 28  0  0 22  0 | Iris-setosa
  0  0 27  0 23 | Iris-versicolor
  0 35 15  0  0 | Iris-virginica

Cluster 0 <-- Iris-setosa
Cluster 1 <-- Iris-virginica
Cluster 2 <-- Iris-versicolor
Cluster 3 <-- No class
Cluster 4 <-- No class

Incorrectly clustered instances :       60.0      40      %
```

Status

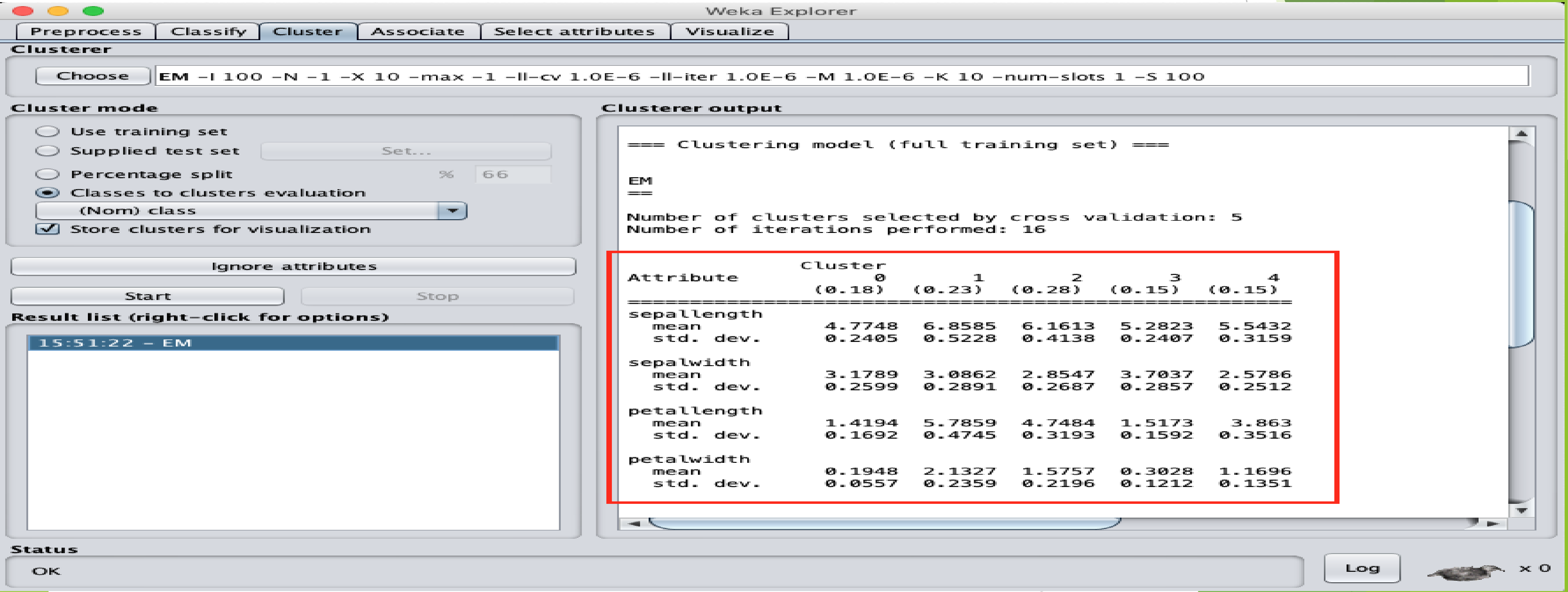
OK

From the output screen, you can observe that: □ There are 5 clustered instances detected in the database.

□ The **Cluster 0** represents setosa, **Cluster 1** represents virginica, **Cluster 2** represents versicolor, while the last two clusters do not have any class associated with them.

WEKA – Clustering

If you scroll up the output window, you will also see some statistics that gives the mean and standard deviation for each of the attributes in the various detected clusters. This is shown in the screenshot given below:



```
EM -I 100 -N -1 -X 10 -max -1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100
```

○ Use training set
○ Supplied test set
○ Percentage split % 66
● Classes to clusters evaluation
(Nom) class
☑ Store clusters for visualization

15:51:22 - EM

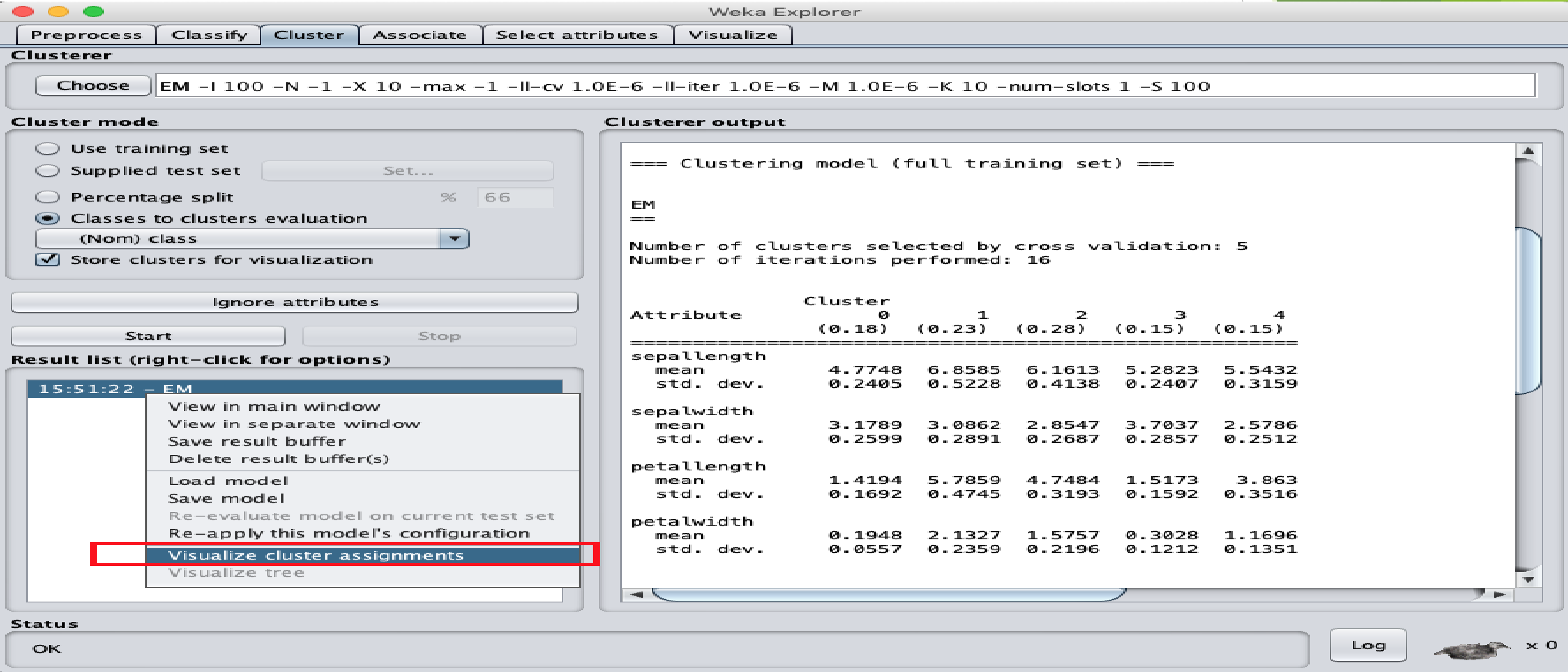
```
=== Clustering model (full training set) ===


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)
=============================================================
sepallength
  mean            4.7748   6.8585   6.1613   5.2823   5.5432
  std. dev.       0.2405   0.5228   0.4138   0.2407   0.3159

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

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

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

OK

Next, we will look at the visual representation of the clusters.

WEKA – Clustering

To visualize the clusters, right click on the **EM** result in the **Result list**. You will see the following options:



```
EM -I 100 -N -1 -X 10 -max -1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100
```

○ Use training set

○ Supplied test set

○ Percentage split % 66

● Classes to clusters evaluation

(Nom) class

☑ Store clusters for visualization

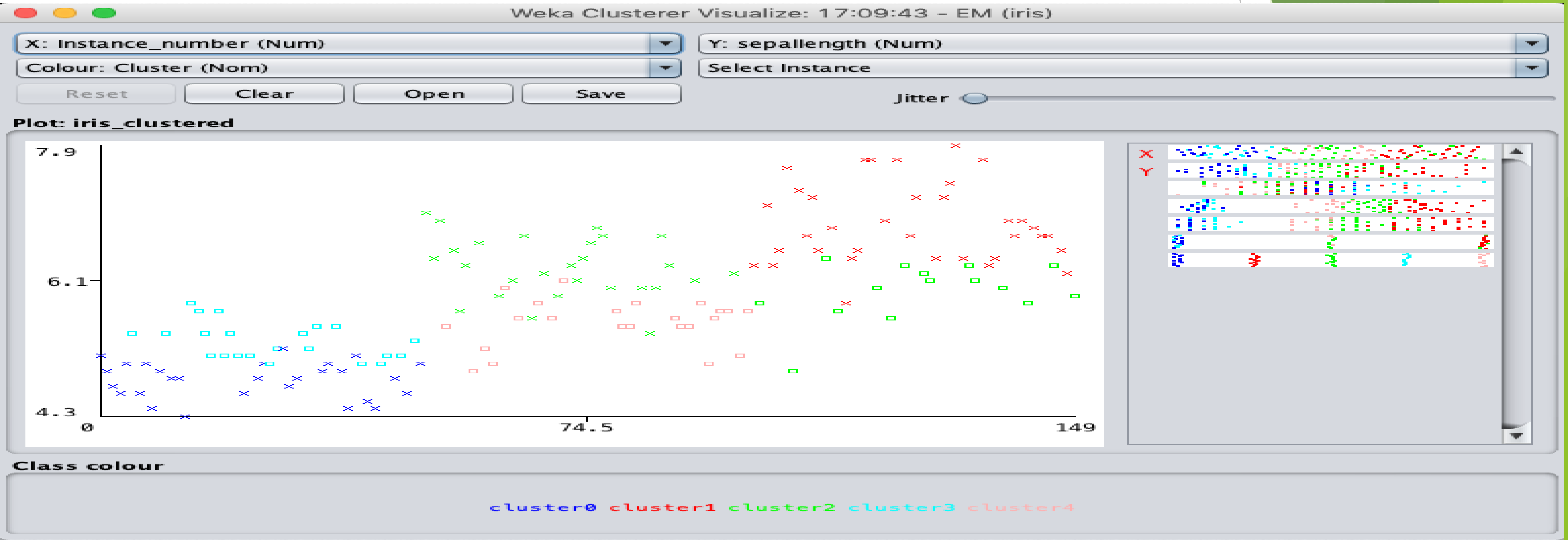
15:51:22 - EM

```
=== 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)
========================================================
sepallength
  mean            4.7748  6.8585  6.1613  5.2823  5.5432
  std. dev.       0.2405  0.5228  0.4138  0.2407  0.3159

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

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

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

OK

WEKA – Clustering

Select **Visualize cluster assignments**. You will see the following output:



Weka Clusterer Visualize: 17:09:43 - EM (iris)

X: Instance_number (Num)

Y: sepallength (Num)

Colour: Cluster (Nom)

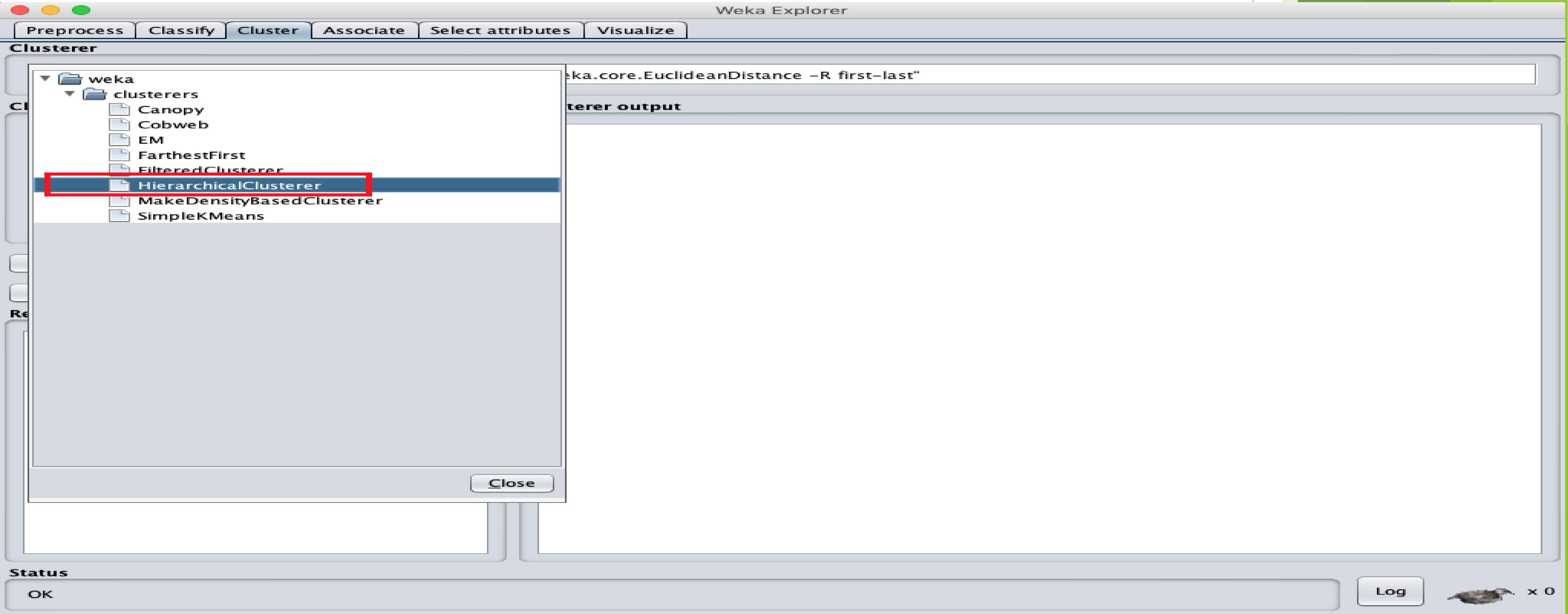
Plot: iris_clustered

As in the case of classification, you will notice the distinction between the correctly and incorrectly identified instances. You can play around by changing the X and Y axes to analyze the results. You may use jittering as in the case of classification to find out the concentration of correctly identified instances. The operations in visualization plot are similar to the one you studied in the case of classification.

WEKA – Clustering

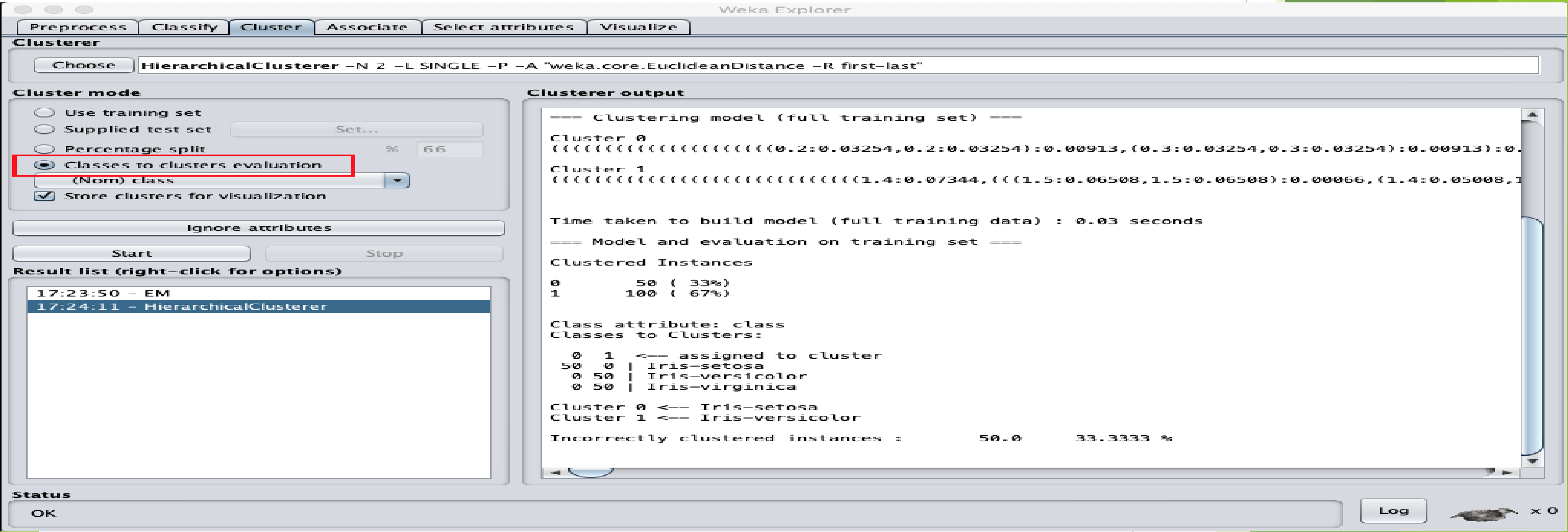
Applying Hierarchical Clusterer

To demonstrate the power of WEKA, let us now look into an application of another clustering algorithm. In the WEKA explorer, select the **HierarchicalClusterer** as your ML algorithm as shown in the screenshot shown below:



WEKA – Clustering

Choose the **Cluster mode** selection to **Classes to cluster evaluation**, and click on the **Start** button. You will see the following output:



HierarchicalClusterer -N 2 -L SINGLE -P -A "weka.core.EuclideanDistance -R first-last"

○ Use training set
○ Supplied test set
○ Percentage split % 66
● Classes to clusters evaluation
(Nom) class
☑ Store clusters for visualization

Result list (right-click for options)

17:23:50 - EM
17:24:11 - HierarchicalClusterer

Clusterer output

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

Cluster 0
((((((((((((((((((((0.2:0.03254,0.2:0.03254):0.00913,(0.3:0.03254,0.3:0.03254):0.00913):0.

Cluster 1
((((((((((((((((((((((((((((1.4:0.07344,(((1.5:0.06508,1.5:0.06508):0.00066,(1.4:0.05008,1


Time taken to build model (full training data) : 0.03 seconds

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

Clustered Instances

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


Class attribute: class
Classes to Clusters:

  0  1  <-- assigned to cluster
 50  0 | Iris-setosa
  0 50 | Iris-versicolor
  0 50 | Iris-virginica

Cluster 0 <-- Iris-setosa
Cluster 1 <-- Iris-versicolor

Incorrectly clustered instances :       50.0      33.3333 %
```

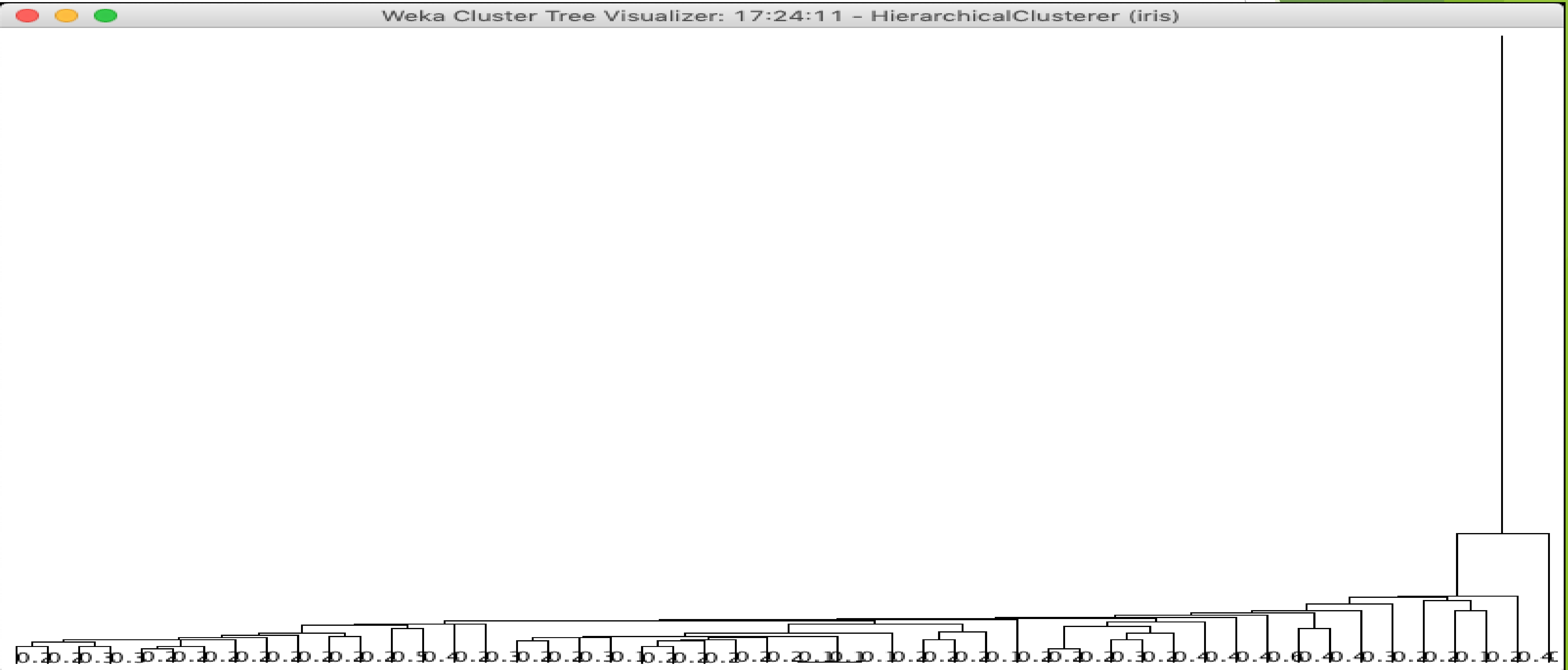
Status

OK

Notice that in the **Result list,** there are two results listed: the first one is the EM result and the second one is the current Hierarchical. Likewise, you can apply multiple ML algorithms to the same dataset and quickly compare their results.

WEKA – Clustering

If you examine the tree produced by this algorithm, you will see the following output:


Weka Cluster Tree Visualizer: 17:24:11 – HierarchicalClusterer (iris)