PREPARED BY: RAJARAJESWARI VAIDYANATHAN **CWID**: A20362220 **COURSE**: CS422 – DATA MINING

TABLE OF CONTENTS

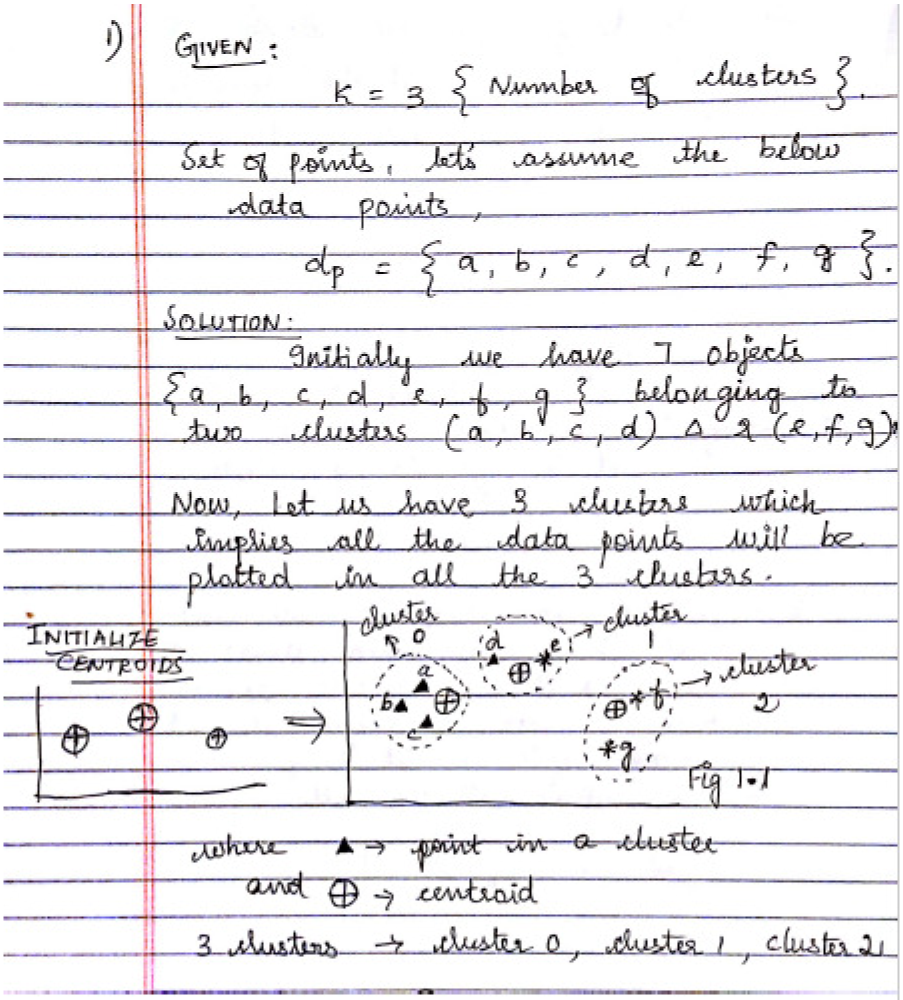
EXPLORING WEKA SOFTWARE TOOL AND ANALYZING THE RESULTS PRODUCED BY DIFFERENT DATASETS WITH RESPECT TO CLUSTERING.

TABLE OF CONTENTS

CLUSTERING PROBLEMS:

1. **Find an example of a small set of points and three initial centroids so that kMeans with k=3 converges to a clustering with an empty cluster. Note that the initial centroids do not have to be members of the set of points.**

Solution:



1) GIVEN :

K = 3 { Number of clusters }.

Set of points, let's assume the below data points,

dp = { a, b, c, d, e, f, g }.

SOLUTION:

Initially we have 7 objects { a, b, c, d, e, f, g } belonging to two clusters (a, b, c, d) & (e, f, g).

Now, Let us have 3 clusters which implies all the data points will be plotted in all the 3 clusters.

INITIALIZE CENTROIDS

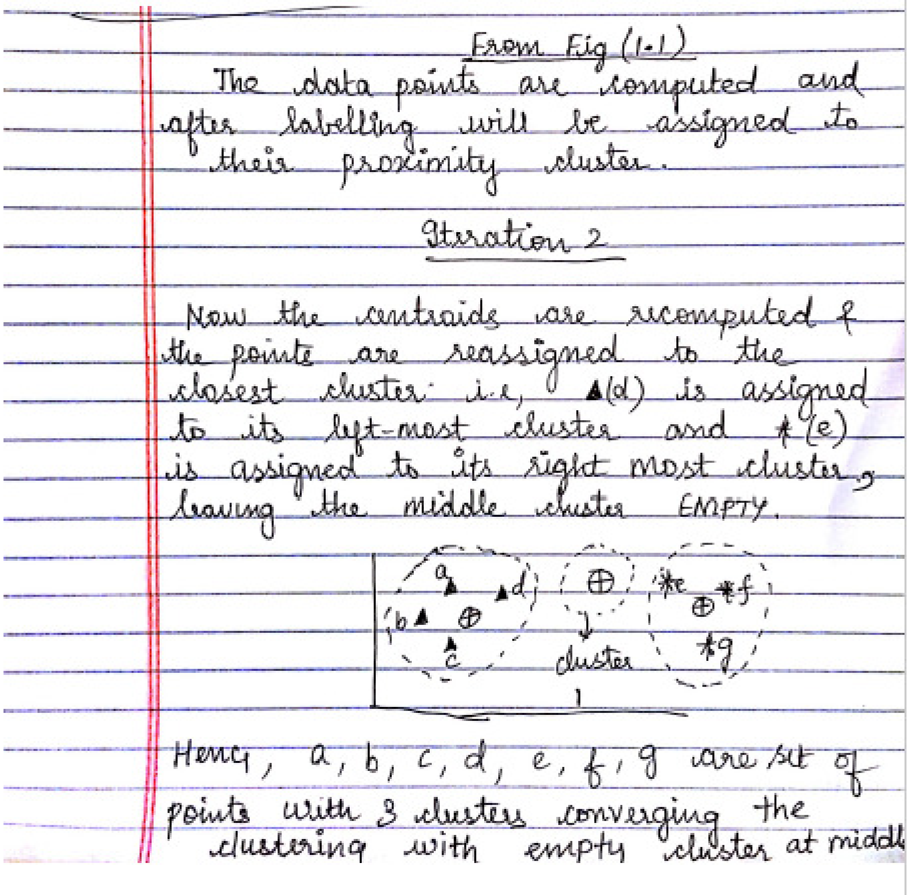
⊕
⊕
⊕
⇒
cluster 0
a
b
c
d
e
cluster 1
f
g
cluster 2

Fig 1.1

where ▲ → point in a cluster

and ⊕ → centroid

3 clusters → cluster 0, cluster 1, cluster 2.



From Fig (1.1)

The data points are computed and after labelling will be assigned to their proximity cluster.

Iteration 2

Now the centroids are recomputed & the points are reassigned to the closest cluster. i.e, ▲(d) is assigned to its left-most cluster and *(e) is assigned to its right most cluster, leaving the middle cluster EMPTY.

a
d
b
c
⊕
cluster
e
f
g

Hence, a, b, c, d, e, f, g are set of points with 3 clusters converging the clustering with empty cluster at middl

Iterations performed:

Mean taken of all the instances to compute the new Centroid and the distance is calculated from point (d) to its own cluster and to the distance between point (d) to the other cluster (leftmost).

Since the distance from point (d) to the leftmost cluster is less, (SSE less) point (d) will come into left most cluster thus leaving its own cluster.

Similarly the same case happens with point (e) in the 2nd iteration when the rightmost cluster is computing the centroid for its own cluster.

Hence from the assumed set of points and three initial centroids (kMeans) with k=3 converges to a clustering with an empty cluster.

2. **We use SSE (RSS) as the measure of cluster quality and kMeans minimizes it. If there is an empty cluster, can that clustering be the global minimum solution based on RSS?**

Solution:

RSS(Residual sum of squared distance between each point in the cluster to its centroid) shows how well the centroids represent the members of their clusters. For a good cluster the RSS value has to be low. It is defined as the measure of squared distance of each vector from its centroid that is added with all vectors.

For an **empty cluster the RSS=0 ,for a global minimum solution the RSS should be very low**.

But while considering the **RSS value for each cluster which includes the RSS value of the Non empty clusters ,hence the the total RSS value is no more zero**.

Hence **the empty cluster doesn't correspond to global minimum solution.**

Also, in kMeans we can get different solutions by trying different random seeds and we choose the best one. This factor leads to random initialization.

Global minimum solutions is a **NP hard problem**.

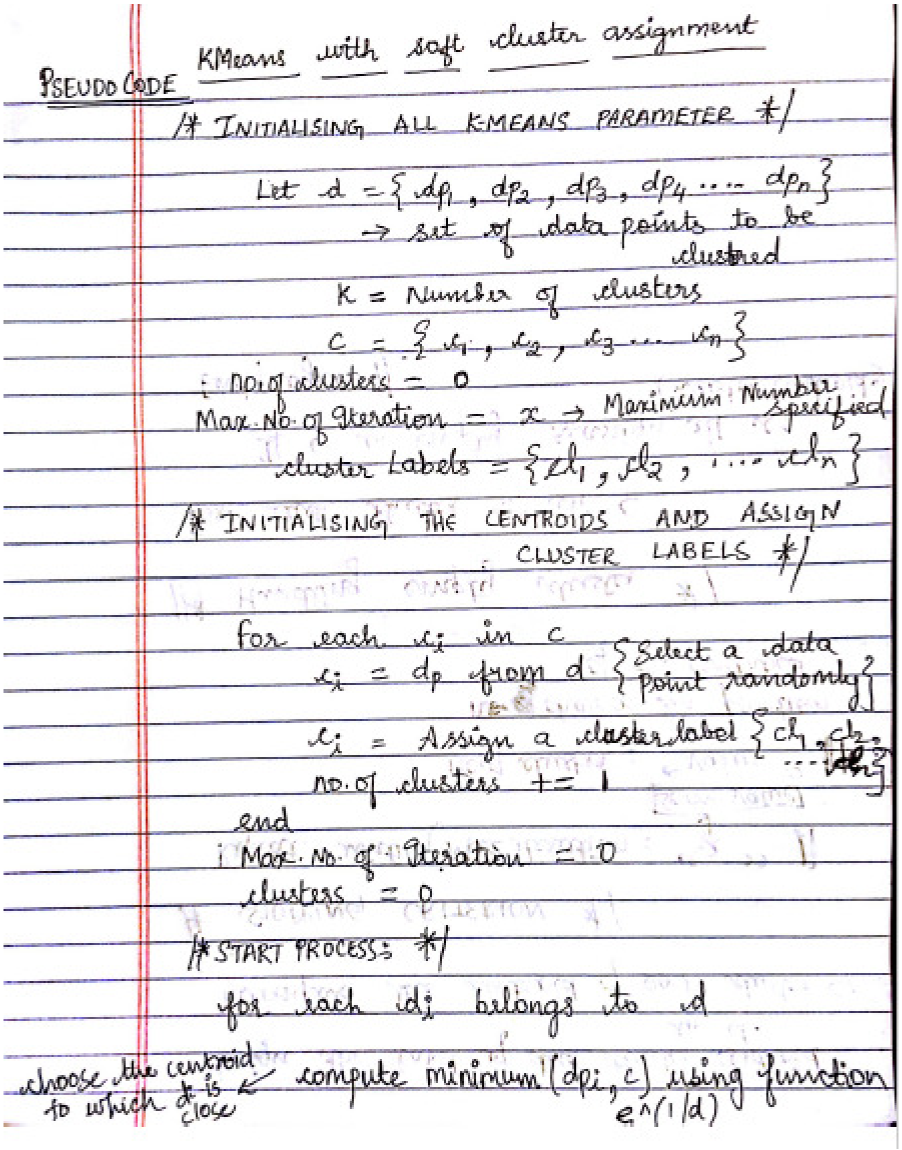
If the result of clustering is only the empty cluster then it leads to global minimum solution ,but an empty cluster makes no meaning and hence always ignored.

There may be many empty clusters or many single points clusters which are also outliers, but they are given less importance because they don't have any significant change in the mean that is being calculated to compute the centroid.

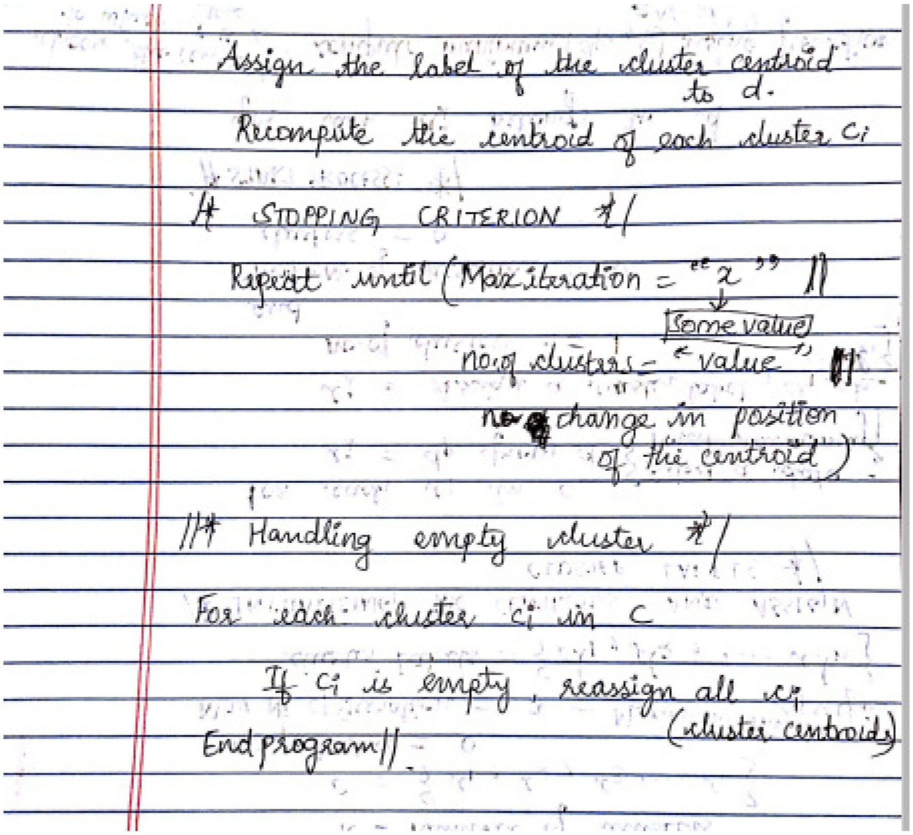
3. **kMeans with soft cluster assignment**
   Computes the fractional membership of a document in a cluster as a function of the distance D from its centroid. That function is monotonically decreasing, e.g., as e^(1/d)

Solution:

Pseudocode:



```
PSEUDO CODE   KMeans with soft cluster assignment

/* INITIALISING ALL KMEANS PARAMETER */

        Let d = { dp_1, dp_2, dp_3, dp_4 .... dp_n }
                → set of data points to be clustered
            K = Number of clusters
            c = { c_1, c_2, c_3 ... c_n }
    no.of clusters = 0
    Max.No.of Iteration = x → Maximum Number specified
        cluster Labels = { cl_1, cl_2, .... cl_n }

/* INITIALISING THE CENTROIDS AND ASSIGN
                        CLUSTER LABELS */

    for each c_i in c
        c_i = d_p from d  { Select a data point randomly }
        c_i = Assign a cluster label { cl_1, cl_2, ... cl_n }
        no.of clusters += 1
    end
    Max. No. of Iteration = 0
    clusters = 0
/* START PROCESS: */
    for each d_i belongs to d
choose the centroid  ←  compute minimum (dp_i, c) using function
to which d is close                                 e^(1/d)
```



```
    Assign the label of the cluster centroid
                                       to d.
    Recompute the centroid of each cluster ci
/* STOPPING CRITERION */
    Repeat until (Max iteration = "x" ||
                                   Some value
                  no.of clusters = "value" ||
                  no change in position
                        of the centroid)
//* Handling empty cluster */
For each cluster ci in C
    If ci is empty, reassign all ci
                          (cluster centroid)
End program//
```

For Fractional computation of a document membership.

```
/* Cluster with threshold value */
If softness > threshold value
        Split cluster
        Compute centroid for another cluster
// Number of documents to be found in each cluster:
If documents in cluster < threshold
        Stop
Else
        If cluster is empty
                Change the centroid position
End//
```

Exercise 7.4.1:

Consider two clusters that are a circle and a surrounding ring, as in the running example of this section. Suppose:
i. The radius of the circle is c.
ii. The inner and outer circles forming the ring have radii i and o, respectively.
iii. All representative points for the two clusters are on the boundaries of the clusters.
iv. Representative points are moved 20% of the distance from their initial position toward the centroid of their cluster.
v. Clusters are merged if, after repositioning, there are representative points from the two clusters at distance d or less.

In terms of d, c, i, and o, under what circumstances will the ring and circle be merged into a single cluster?

Solution:

Given:

C = Circle

I = radius of the inner circle

O = radius of the outer circle

Lets assume a, b, c be the distances after they started moving from their initial distance. Once they started moving 20% of their distance from their initial position towards the centroid of their cluster, then

We can conclude that 80% of the reduction is corresponding to the original distances c, I, o

Based on some distance we can merge the clusters and we reduce all the 3 radii to 20% as they have moved from their initial distance.

D = (b – a) or (c – a) depending on the ring and the circle (inner radii and outer radii) if it is to be merged in one cluster.

Value of d now should be greater than these 2 combinations if we have to merge them into a single cluster.

- ➔ B – c <= d
- ➔ 0.8(b) – 0.8(a) <= d -----------→ equation 1

- ➔ Similarly, c – a <= d
- ➔ 0.8(c) – 0.8(a) <= d ----------------→ equation 2

Once you combine these 2, add them

0.8(b-a-a-c) <= 2d

0.8(b - c – 2a) <= 2d

0.4(b-c-2a) <= d

CLUSTERING:

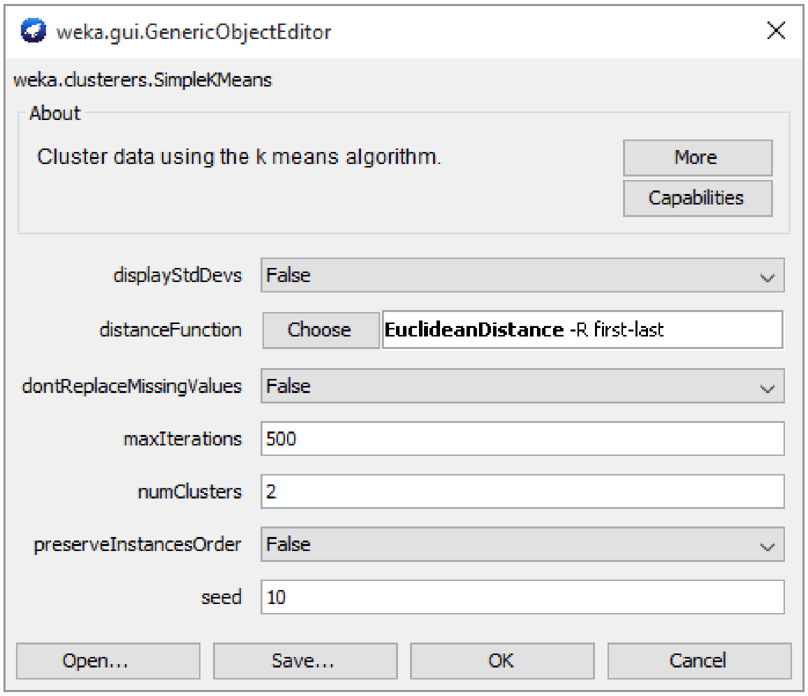
- Clusters are formed when No Class is defined.
- They divide the instances into natural groups or clusters.
- It also helps in recovering the deleted class attributes and grouping them in one or more clusters.
- Disjoint sets and Hierarchical sets are the two types of Clusters.

KMeans Algorithm:

- Specify k, number of clusters.
- Choose k points at random as cluster centers.
- Assign all instances to their closest cluster center.
- Calculate the centroid (average) of the instances in each cluster.
- These centroids are the new cluster centers.
- Now, repeat the above until the cluster centers don't change however you iterate it.

This helps in minimizing the total squared distance from instances to their cluster centers where different results gets generated with different random seeds.

Inferences in Weka with K-means Clustering – Default values.



weka.gui.GenericObjectEditor

weka.clusterers.SimpleKMeans

About

Cluster data using the k means algorithm.

displayStdDevs	False
distanceFunction	EuclideanDistance -R first-last
dontReplaceMissingValues	False
maxIterations	500
numClusters	2
preserveInstancesOrder	False
seed	10

INPUT PARAMETERS FOR SIMPLE K-MEANS ALGORITHM:

displayStdDevs – It is used to display the standard deviation of the attributes in the dataset. Displays standard deviations of numeric attributes and counts of nominal attributes.

distanceFunction – This is the objective function that calculates the proximity of the points in the dataset or the proximity of the points to the centroid of the cluster.

dontReplaceMissingValues– The dataset may contain some missing values, those values may be replaced with global mean/mode.

maxIterations – Used to set the number of iterations to be done . It can also be the convergence/stopping criterion

numClusters – Used to set the number of clusters to be formed. It can also be the convergence/stopping criterion

preserveInstancesOrder –It takes the same order of instances in the dataset.

seed – Seed is used to calculate the initial centroid.

CLUSTER MODES:

There are different types of modes in Clusters to test the data. These effectively uses **training set, Percentage split, Classes to Cluster evaluation, Supplied test set.**

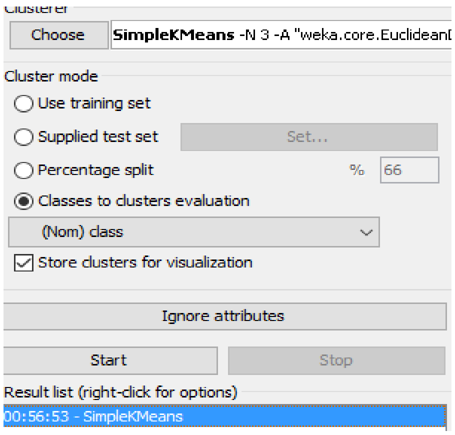
CLASSES TO CLUSTERS EVALUATION:

- This is a default value which is set under Cluster modes during Clustering. In this mode, the class attribute is first ignored in the dataset and the clustering is executed.

- During the test phase again, the class attribute is included in the Dataset and Clustering again takes place.

- The misclassification error is then computed and it is represented in confusion matrix.

- Incorrect instances are specified under Clustering.



Clusterer

Choose **SimpleKMeans** -N 3 -A "weka.core.Euclidean[

Cluster mode

○ Use training set

○ Supplied test set Set...

○ Percentage split % 66

● Classes to clusters evaluation

(Nom) class

☑ Store clusters for visualization

Ignore attributes

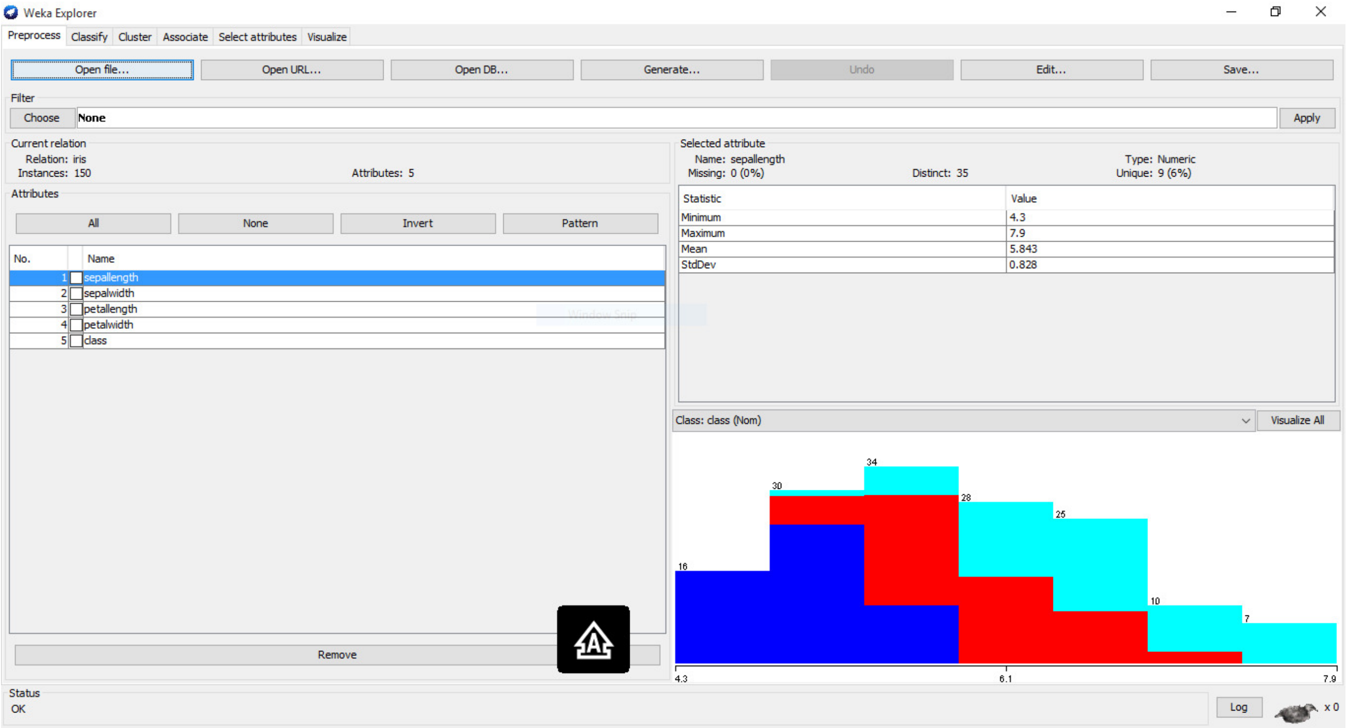
Start Stop

Result list (right-click for options)

00:56:53 - SimpleKMeans

PREPARED BY: RAJARAJESWARI VAIDYANATHAN CWID: A20362220 COURSE: CS422 – DATA MINING

IRIS DATASET:



Filter

None

Current relation

Relation: iris
Instances: 150 Attributes: 5

Attributes

No.		Name
1	□	sepallength
2	□	sepalwidth
3	□	petallength
4	□	petalwidth
5	□	class

Selected attribute

Name: sepallength Type: Numeric
Missing: 0 (0%) Distinct: 35 Unique: 9 (6%)

Statistic	Value
Minimum	4.3
Maximum	7.9
Mean	5.843
StdDev	0.828

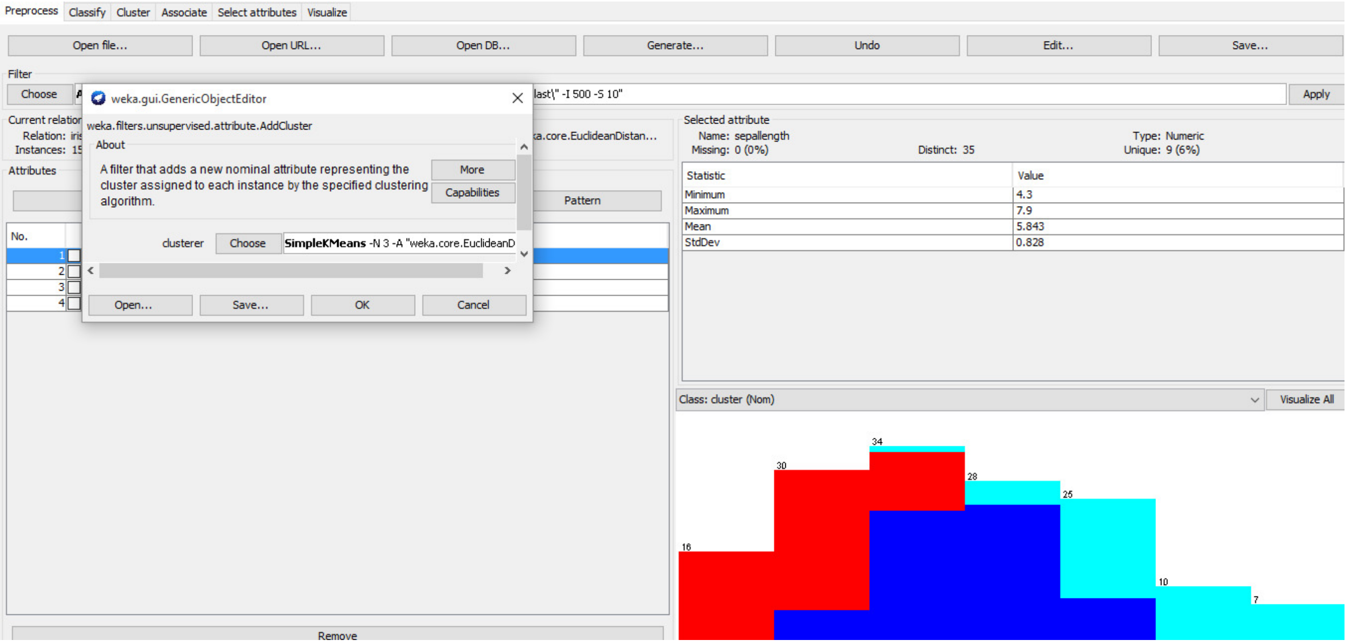
Class: class (Nom)

Status
OK

1. Inference from IRIS Dataset:

Dataset Name	Number of attributes present	Name of the attributes	Class attribute – Name of the instances	Number of attribute instances
IRIS dataset	5	Sepal Length, Sepal Width, Petal Length, Petal width and Class.	Setosa, Virginica and Veriscolor	150 (Setosa, Virginica, Veriscolor – each 50 instances)

Test case 1:- IRIS Dataset is tested under pre-processing to explore and visualize the attributes.



last\" -I 500 -S 10"

Relation: iris

Instances: 15

a.core.EuclideanDistan...

weka.gui.GenericObjectEditor

weka.filters.unsupervised.attribute.AddCluster

About

A filter that adds a new nominal attribute representing the cluster assigned to each instance by the specified clustering algorithm.

clusterer: **SimpleKMeans** -N 3 -A "weka.core.EuclideanD

Selected attribute

Name: sepallength Type: Numeric

Missing: 0 (0%) Distinct: 35 Unique: 9 (6%)

Statistic	Value
Minimum	4.3
Maximum	7.9
Mean	5.843
StdDev	0.828

Preprocessing visualizing:

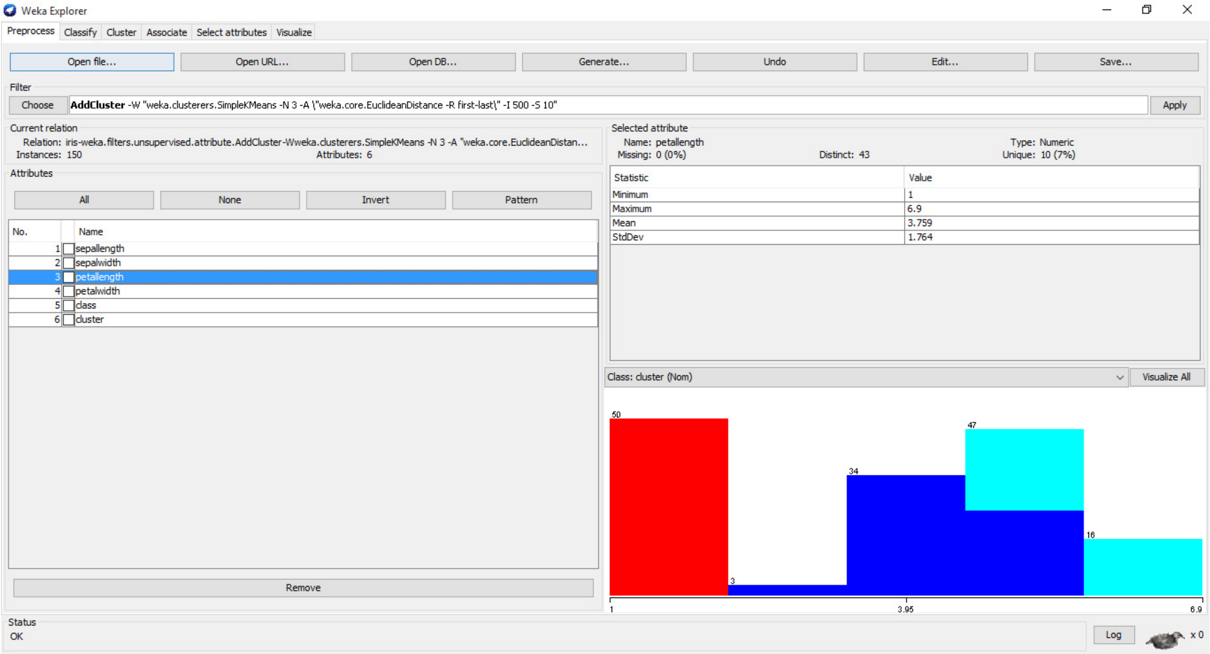
Petal Length: The points in the cluster are dense for Iris-Setosa and hence this would result in a good cluster. SSE is minimum and the number of clusters are also less which will result in a best clustering/grouping data according to the attributes.

Petal Width: The points in here, are also resembling the Iris-setosa and the points are dense enough to group them in one cluster.

Sepal Length and Sepal width: Have Hierarchial clusterings as the points in them are scattered which in few iterations will be distorted. In this case, SSE is more and hence resulting in a poor cluster.

Inferences:

- The number of clusters is increased and various possibility of SSE (Sum of Squared Error) and Number of Incorrectly observed instances are observed in 2 attributes.
- As the number of clusters increases the SSE decreases, this is because as the number of clusters increases, the points in the dataset are placed with more appropriate clusters i.e. data point moves to the closest cluster centroid which in turn causes the SSE of each cluster to reduce.
- As the number of cluster increases, the Incorrectly clustered instances initially decreases , when the number of cluster further increases it leads to outliers and missing data values fall into those clusters and there by the Incorrectly clustered instances increases.



Filter: AddCluster -W "weka.clusterers.SimpleKMeans -N 3 -A \"weka.core.EuclideanDistance -R first-last\" -I 500 -S 10"

Current relation
Relation: iris-weka.filters.unsupervised.attribute.AddCluster-Wweka.clusterers.SimpleKMeans -N 3 -A "weka.core.EuclideanDistan...
Instances: 150 Attributes: 6

Attributes

No.	Name
1	sepallength
2	sepalwidth
3	petallength
4	petalwidth
5	class
6	cluster

Selected attribute
Name: petallength Type: Numeric
Missing: 0 (0%) Distinct: 43 Unique: 10 (7%)

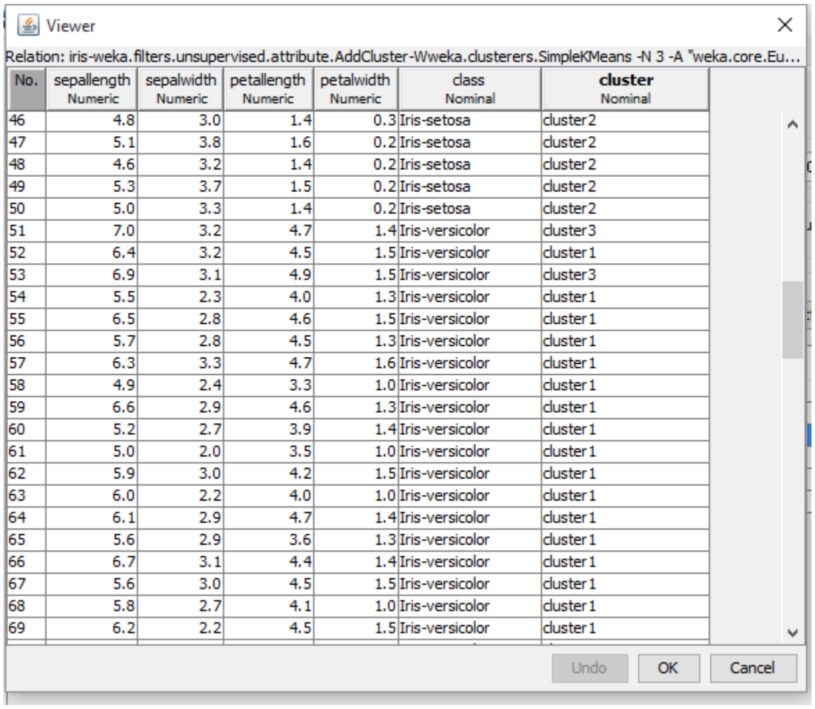
Statistic	Value
Minimum	1
Maximum	6.9
Mean	3.759
StdDev	1.764

Class: cluster (Nom)

Status
OK

After adding the Cluster as an attribute in the filter in Preprocessing tab, the data groups themselves in one cluster by knowing their nearest points.

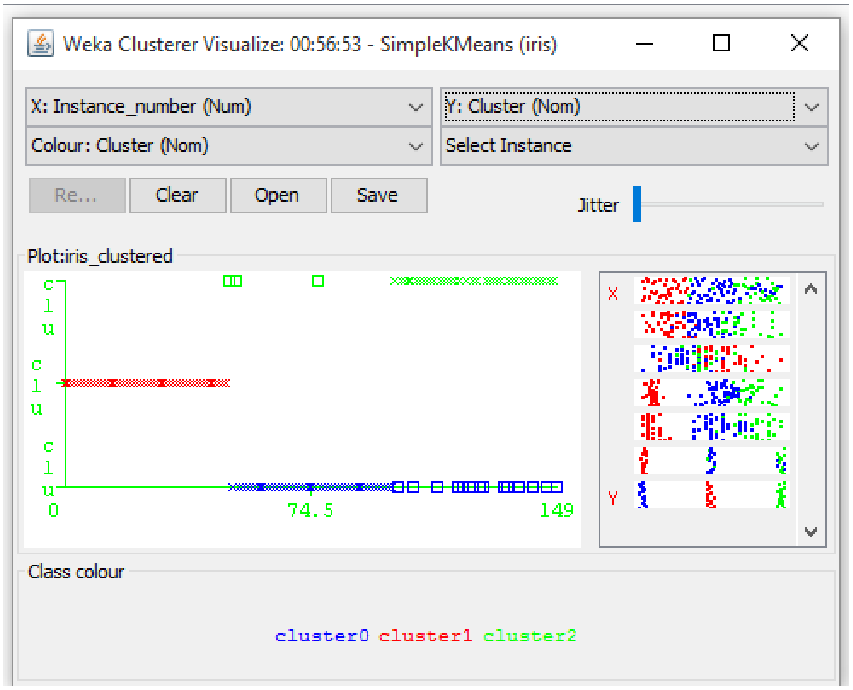
Input dataset- IRIS after adding clusters.



Relation: iris-weka.filters.unsupervised.attribute.AddCluster-Wweka.clusterers.SimpleKMeans -N 3 -A "weka.core.Eu...

No.	sepallength Numeric	sepalwidth Numeric	petallength Numeric	petalwidth Numeric	class Nominal	cluster Nominal
46	4.8	3.0	1.4	0.3	Iris-setosa	cluster2
47	5.1	3.8	1.6	0.2	Iris-setosa	cluster2
48	4.6	3.2	1.4	0.2	Iris-setosa	cluster2
49	5.3	3.7	1.5	0.2	Iris-setosa	cluster2
50	5.0	3.3	1.4	0.2	Iris-setosa	cluster2
51	7.0	3.2	4.7	1.4	Iris-versicolor	cluster3
52	6.4	3.2	4.5	1.5	Iris-versicolor	cluster1
53	6.9	3.1	4.9	1.5	Iris-versicolor	cluster3
54	5.5	2.3	4.0	1.3	Iris-versicolor	cluster1
55	6.5	2.8	4.6	1.5	Iris-versicolor	cluster1
56	5.7	2.8	4.5	1.3	Iris-versicolor	cluster1
57	6.3	3.3	4.7	1.6	Iris-versicolor	cluster1
58	4.9	2.4	3.3	1.0	Iris-versicolor	cluster1
59	6.6	2.9	4.6	1.3	Iris-versicolor	cluster1
60	5.2	2.7	3.9	1.4	Iris-versicolor	cluster1
61	5.0	2.0	3.5	1.0	Iris-versicolor	cluster1
62	5.9	3.0	4.2	1.5	Iris-versicolor	cluster1
63	6.0	2.2	4.0	1.0	Iris-versicolor	cluster1
64	6.1	2.9	4.7	1.4	Iris-versicolor	cluster1
65	5.6	2.9	3.6	1.3	Iris-versicolor	cluster1
66	6.7	3.1	4.4	1.4	Iris-versicolor	cluster1
67	5.6	3.0	4.5	1.5	Iris-versicolor	cluster1
68	5.8	2.7	4.1	1.0	Iris-versicolor	cluster1
69	6.2	2.2	4.5	1.5	Iris-versicolor	cluster1

Visualizing the attributes individually:

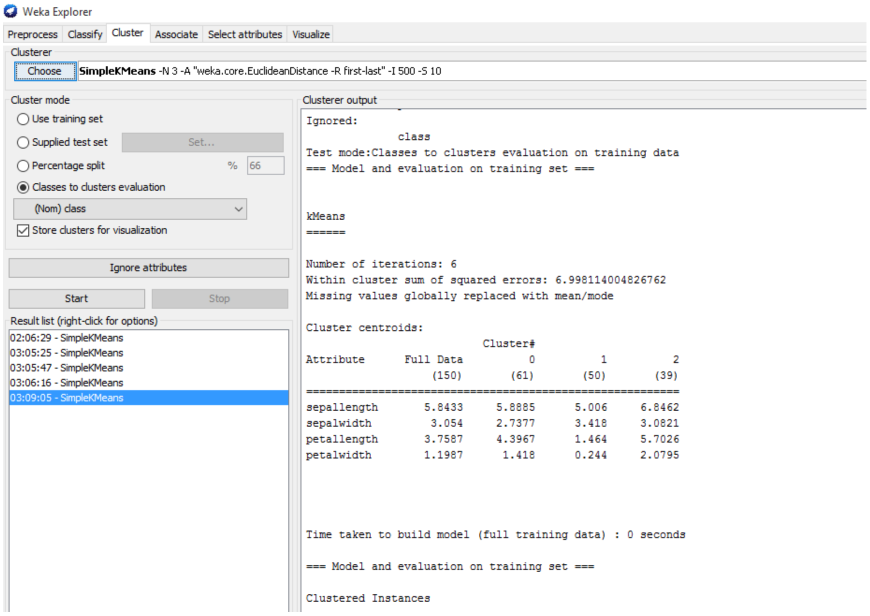


The blue points that fall in one cluster has more dense region than the ones that are separate.

When blue cluster is formed with respect to the centroid taken, noise is also taken as a factor but cannot be inclusive in the cluster as their SSE would be very high.

Test case 2:- Output KMeans algorithm for IRIS Dataset.

Output KMeans algorithm:

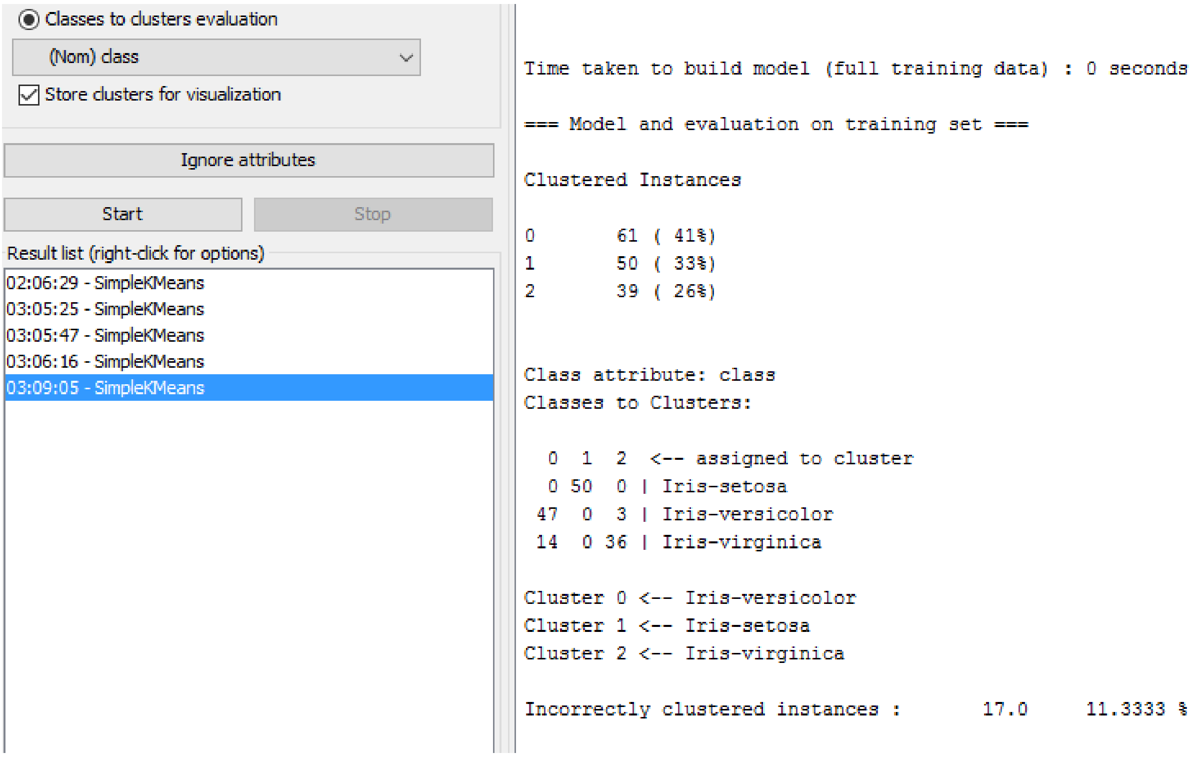


Clusterer: **SimpleKMeans** -N 3 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 10

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

```
Ignored:
              class
Test mode:Classes to clusters evaluation on training data
=== Model and evaluation on training set ===


kMeans
======

Number of iterations: 6
Within cluster sum of squared errors: 6.998114004826762
Missing values globally replaced with mean/mode

Cluster centroids:
                           Cluster#
Attribute      Full Data          0          1          2
                   (150)       (61)       (50)       (39)
=========================================================
sepallength       5.8433     5.8885      5.006     6.8462
sepalwidth         3.054     2.7377      3.418     3.0821
petallength       3.7587     4.3967      1.464     5.7026
petalwidth        1.1987      1.418      0.244     2.0795




Time taken to build model (full training data) : 0 seconds

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

Clustered Instances
```

- Shows the number of iterations taken for the clustering of the instances into each Cluster.
- If the maximum iterations have reached but the centroid remains the same, we stop iterating it further by introducing stopping/convergence criterion.
- It also shows the efficiency of the clustering as SSE has been taken as a global function for clustering.
- Missing values if any present in the dataset are globally replaced with mean/mode.
- kMeans algorithm is sensitive to noise parameter and hence the iterations and the initial centroids should be more appropriate for a good cluster to be achieved.



● Classes to clusters evaluation

(Nom) class

☑ Store clusters for visualization

```
Time taken to build model (full training data) : 0 seconds

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

Clustered Instances

0       61 ( 41%)
1       50 ( 33%)
2       39 ( 26%)


Class attribute: class
Classes to Clusters:

  0  1  2  <-- assigned to cluster
  0 50  0 | Iris-setosa
 47  0  3 | Iris-versicolor
 14  0 36 | Iris-virginica

Cluster 0 <-- Iris-versicolor
Cluster 1 <-- Iris-setosa
Cluster 2 <-- Iris-virginica

Incorrectly clustered instances :      17.0     11.3333 %
```

- The above fig, shows that there are 17 misclassified instances and 3 clusters are formed after performing 6 iterations.

- Within the cluster: sum of squared errors = 6.998

- Each record in the Cluster Centroid, gives the centroid co-ordinate for the specific dimension or attribute in the dataset with respect to Clustered instances.

- Each cluster belongs to a cluster where Cluster 0 = Iris versicolor, Cluster 1 = Iris Setosa, Cluster 2 = Iris Virginica.

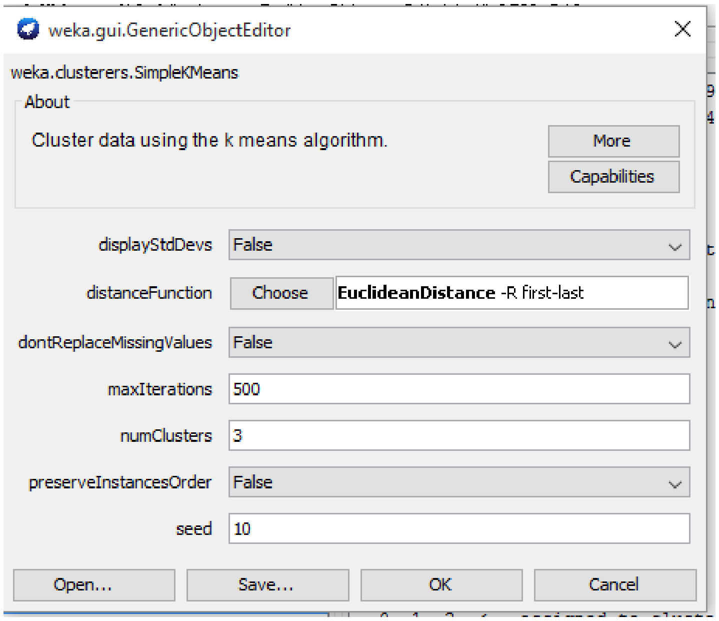
- The actual clusters of the instances is now compared with the instances in each cluster and whenever a mismatch is found, we have an incorrectly instance marked.

Test case 3:- IRIS Dataset under KMeans algorithm without changing parameters.

Number of Iterations keeps varying at each run and SimpleK Means can be achieved by finding how Number of iterations can affect the TOTAL SSE and the number of misclassified instances.

Default parameters for Simple KMeans for Iris dataset:

The Seed parameter is set to default "10". Rest other default parameters are set as shown below.



weka.clusterers.SimpleKMeans

About

Cluster data using the k means algorithm.

displayStdDevs: False

distanceFunction: Choose **EuclideanDistance** -R first-last

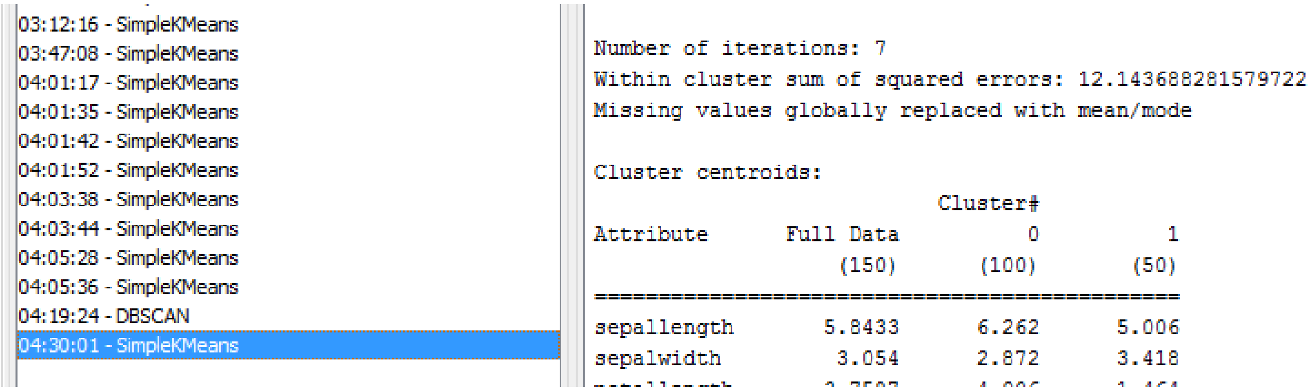
dontReplaceMissingValues: False

maxIterations: 500

numClusters: 3

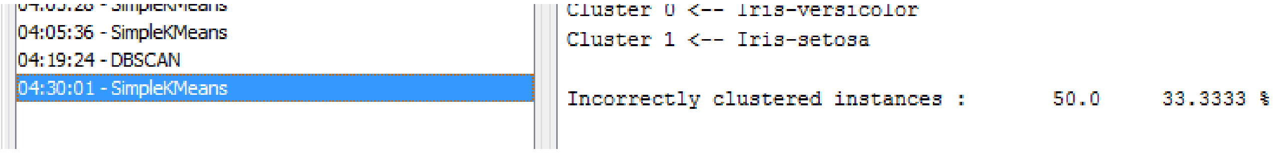
preserveInstancesOrder: False

seed: 10



```
03:12:16 - SimpleKMeans
03:47:08 - SimpleKMeans
04:01:17 - SimpleKMeans
04:01:35 - SimpleKMeans
04:01:42 - SimpleKMeans
04:01:52 - SimpleKMeans
04:03:38 - SimpleKMeans
04:03:44 - SimpleKMeans
04:05:28 - SimpleKMeans
04:05:36 - SimpleKMeans
04:19:24 - DBSCAN
04:30:01 - SimpleKMeans
```

```
Number of iterations: 7
Within cluster sum of squared errors: 12.143688281579722
Missing values globally replaced with mean/mode

Cluster centroids:
                          Cluster#
Attribute      Full Data         0          1
                   (150)     (100)       (50)
==============================================
sepallength       5.8433     6.262      5.006
sepalwidth         3.054     2.872      3.418
petallength       3.7587     4.906      1.464
```



```
04:05:28 - SimpleKMeans
04:05:36 - SimpleKMeans
04:19:24 - DBSCAN
04:30:01 - SimpleKMeans
```

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

Incorrectly clustered instances :       50.0     33.3333 %
```

Inference:

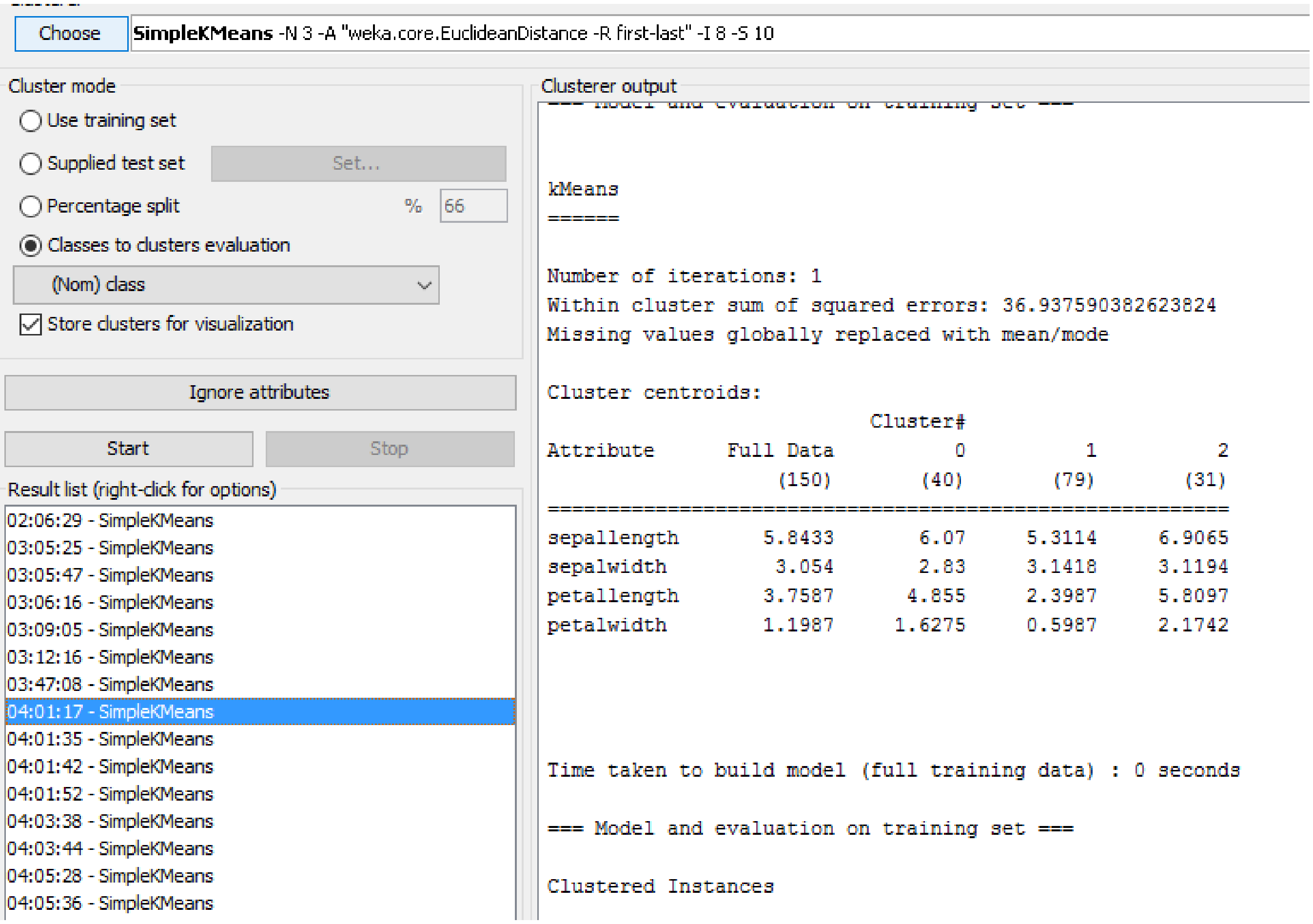
- Default parameters, it is inferred that there are many incorrectly clustered instances and the number of iterations is more.

Test case 4:- IRIS Dataset under KMeans algorithm after changing Parameters = Number of iterations.

After changing the Number of iterations parameter until maximum number of iterations reached,

Change the number of iterations from 1 to 6 and keep number of clusters = Constant

When number of Iteration = 1



Choose **SimpleKMeans** -N 3 -A "weka.core.EuclideanDistance -R first-last" -I 8 -S 10

Cluster mode

○ Use training set

○ Supplied test set Set...

○ Percentage split % 66

● Classes to clusters evaluation

(Nom) class

☑ Store clusters for visualization

Ignore attributes

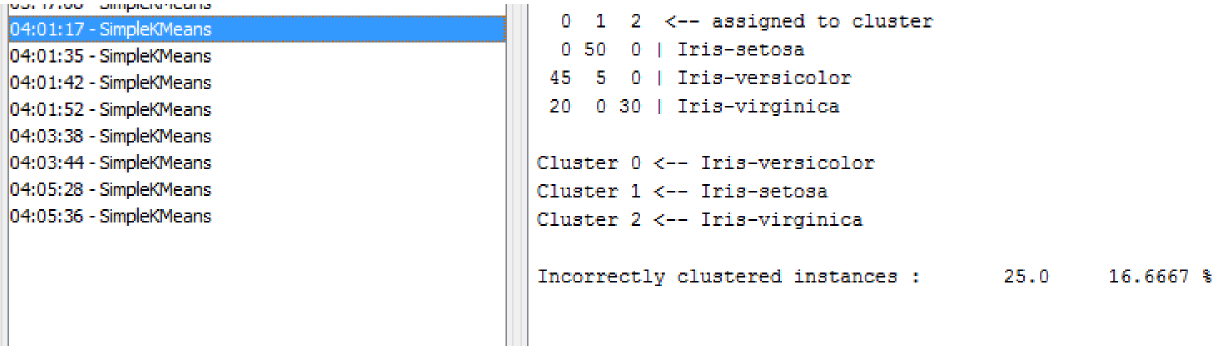
Start Stop

Result list (right-click for options)

02:06:29 - SimpleKMeans
03:05:25 - SimpleKMeans
03:05:47 - SimpleKMeans
03:06:16 - SimpleKMeans
03:09:05 - SimpleKMeans
03:12:16 - SimpleKMeans
03:47:08 - SimpleKMeans
04:01:17 - SimpleKMeans
04:01:35 - SimpleKMeans
04:01:42 - SimpleKMeans
04:01:52 - SimpleKMeans
04:03:38 - SimpleKMeans
04:03:44 - SimpleKMeans
04:05:28 - SimpleKMeans
04:05:36 - SimpleKMeans

Clusterer output

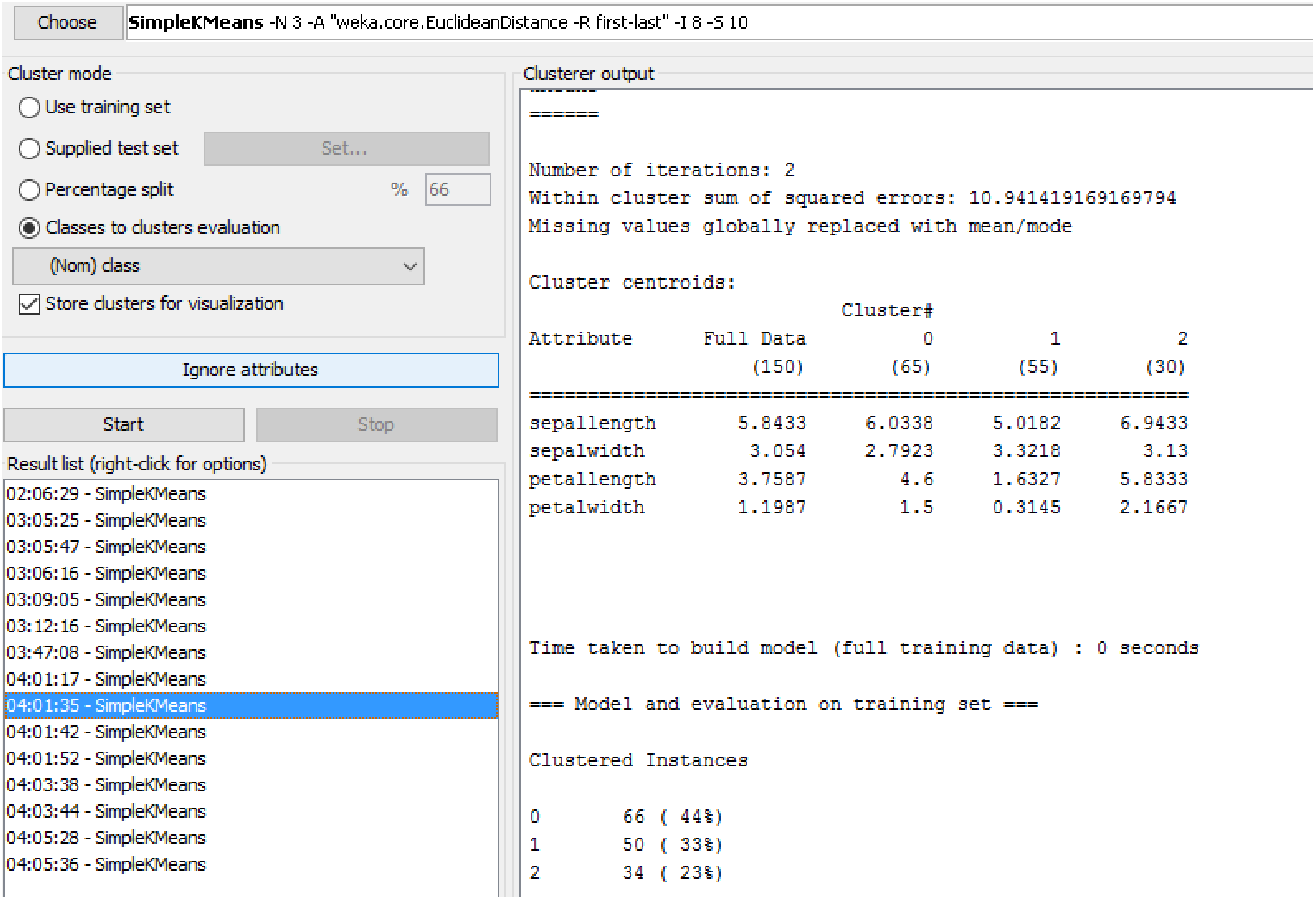
```
=== Model and evaluation on training set ===


kMeans
======

Number of iterations: 1
Within cluster sum of squared errors: 36.937590382623824
Missing values globally replaced with mean/mode

Cluster centroids:
                          Cluster#
Attribute      Full Data          0          1          2
                   (150)       (40)       (79)       (31)
==========================================================
sepallength       5.8433       6.07     5.3114     6.9065
sepalwidth         3.054       2.83     3.1418     3.1194
petallength       3.7587      4.855     2.3987     5.8097
petalwidth        1.1987     1.6275     0.5987     2.1742




Time taken to build model (full training data) : 0 seconds

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

Clustered Instances
```



04:01:17 - SimpleKMeans
04:01:35 - SimpleKMeans
04:01:42 - SimpleKMeans
04:01:52 - SimpleKMeans
04:03:38 - SimpleKMeans
04:03:44 - SimpleKMeans
04:05:28 - SimpleKMeans
04:05:36 - SimpleKMeans

```
  0  1  2  <-- assigned to cluster
  0 50  0 | Iris-setosa
 45  5  0 | Iris-versicolor
 20  0 30 | Iris-virginica

Cluster 0 <-- Iris-versicolor
Cluster 1 <-- Iris-setosa
Cluster 2 <-- Iris-virginica

Incorrectly clustered instances :       25.0     16.6667 %
```

When Iterations = 2



Choose **SimpleKMeans** -N 3 -A "weka.core.EuclideanDistance -R first-last" -I 8 -S 10

Cluster mode

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

Ignore attributes

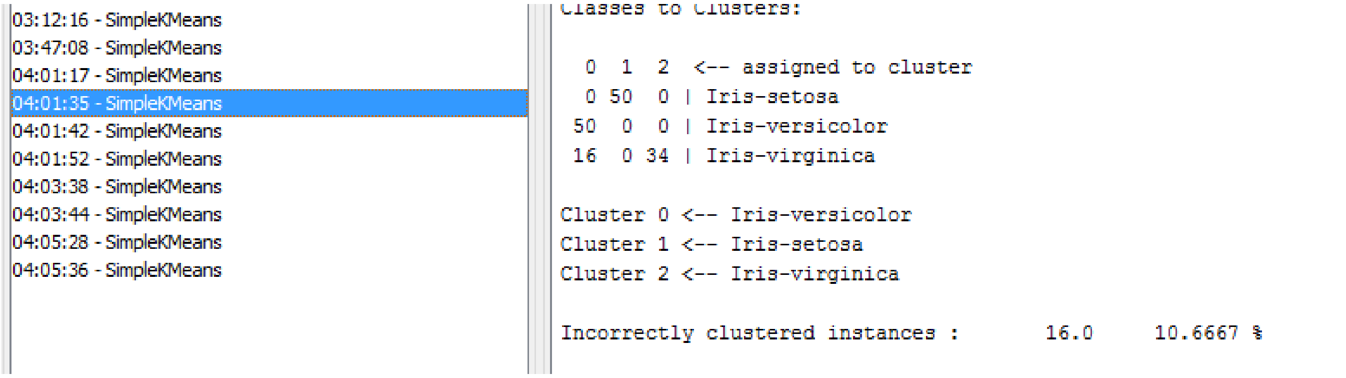
Start Stop

Result list (right-click for options)

02:06:29 - SimpleKMeans
03:05:25 - SimpleKMeans
03:05:47 - SimpleKMeans
03:06:16 - SimpleKMeans
03:09:05 - SimpleKMeans
03:12:16 - SimpleKMeans
03:47:08 - SimpleKMeans
04:01:17 - SimpleKMeans
04:01:35 - SimpleKMeans
04:01:42 - SimpleKMeans
04:01:52 - SimpleKMeans
04:03:38 - SimpleKMeans
04:03:44 - SimpleKMeans
04:05:28 - SimpleKMeans
04:05:36 - SimpleKMeans

Clusterer output

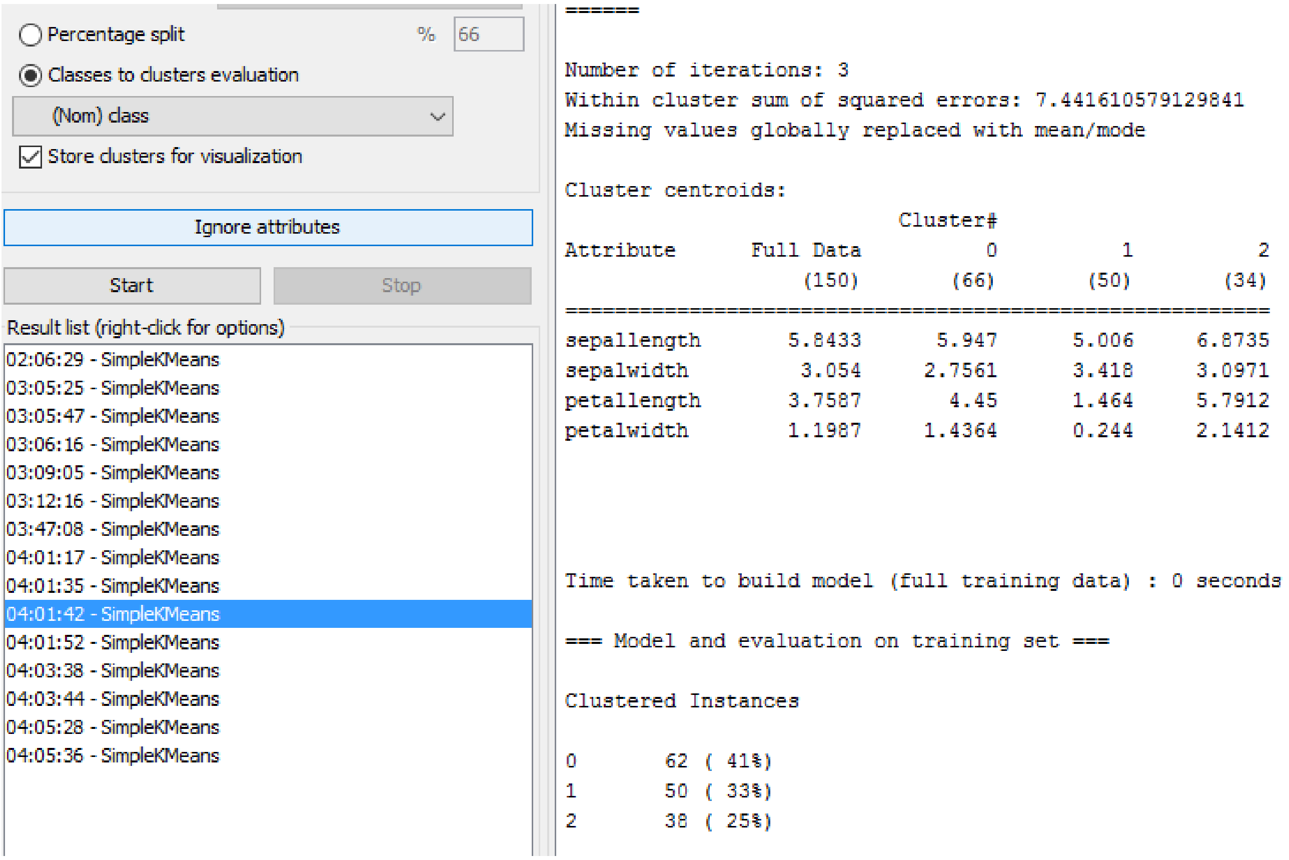
```
======

Number of iterations: 2
Within cluster sum of squared errors: 10.941419169169794
Missing values globally replaced with mean/mode

Cluster centroids:
                           Cluster#
Attribute      Full Data          0          1          2
                   (150)       (65)       (55)       (30)
=========================================================
sepallength       5.8433     6.0338     5.0182     6.9433
sepalwidth         3.054     2.7923     3.3218       3.13
petallength       3.7587        4.6     1.6327     5.8333
petalwidth        1.1987        1.5     0.3145     2.1667




Time taken to build model (full training data) : 0 seconds

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

Clustered Instances

0       66 ( 44%)
1       50 ( 33%)
2       34 ( 23%)
```



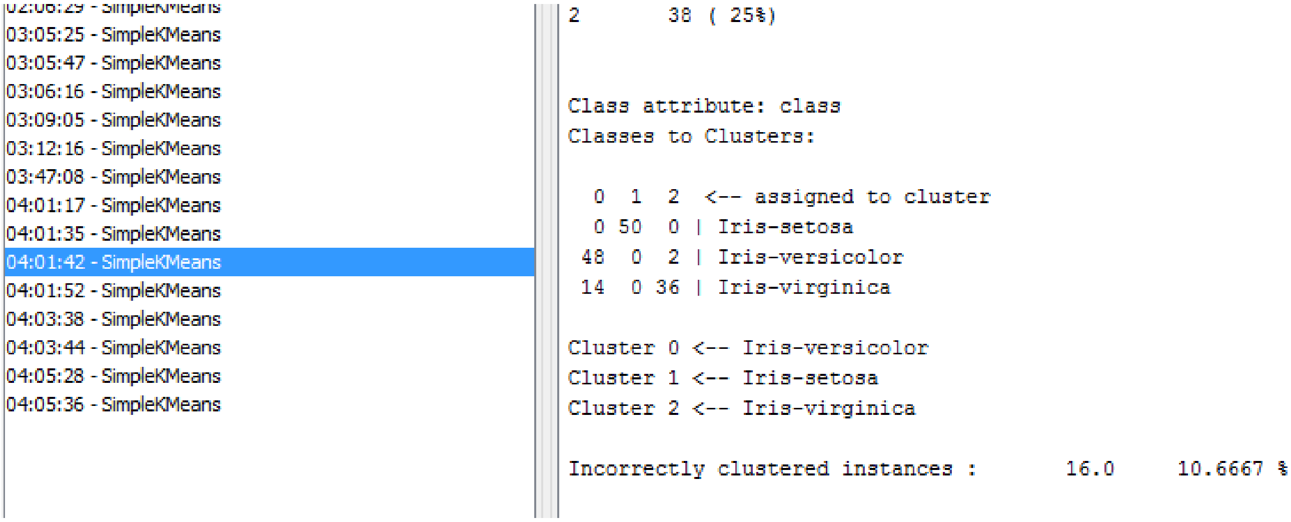
03:12:16 - SimpleKMeans
03:47:08 - SimpleKMeans
04:01:17 - SimpleKMeans
04:01:35 - SimpleKMeans
04:01:42 - SimpleKMeans
04:01:52 - SimpleKMeans
04:03:38 - SimpleKMeans
04:03:44 - SimpleKMeans
04:05:28 - SimpleKMeans
04:05:36 - SimpleKMeans

```
Classes to Clusters:

  0  1  2  <-- assigned to cluster
  0 50  0 | Iris-setosa
 50  0  0 | Iris-versicolor
 16  0 34 | Iris-virginica

Cluster 0 <-- Iris-versicolor
Cluster 1 <-- Iris-setosa
Cluster 2 <-- Iris-virginica

Incorrectly clustered instances :       16.0     10.6667 %
```

When Iteration = 3

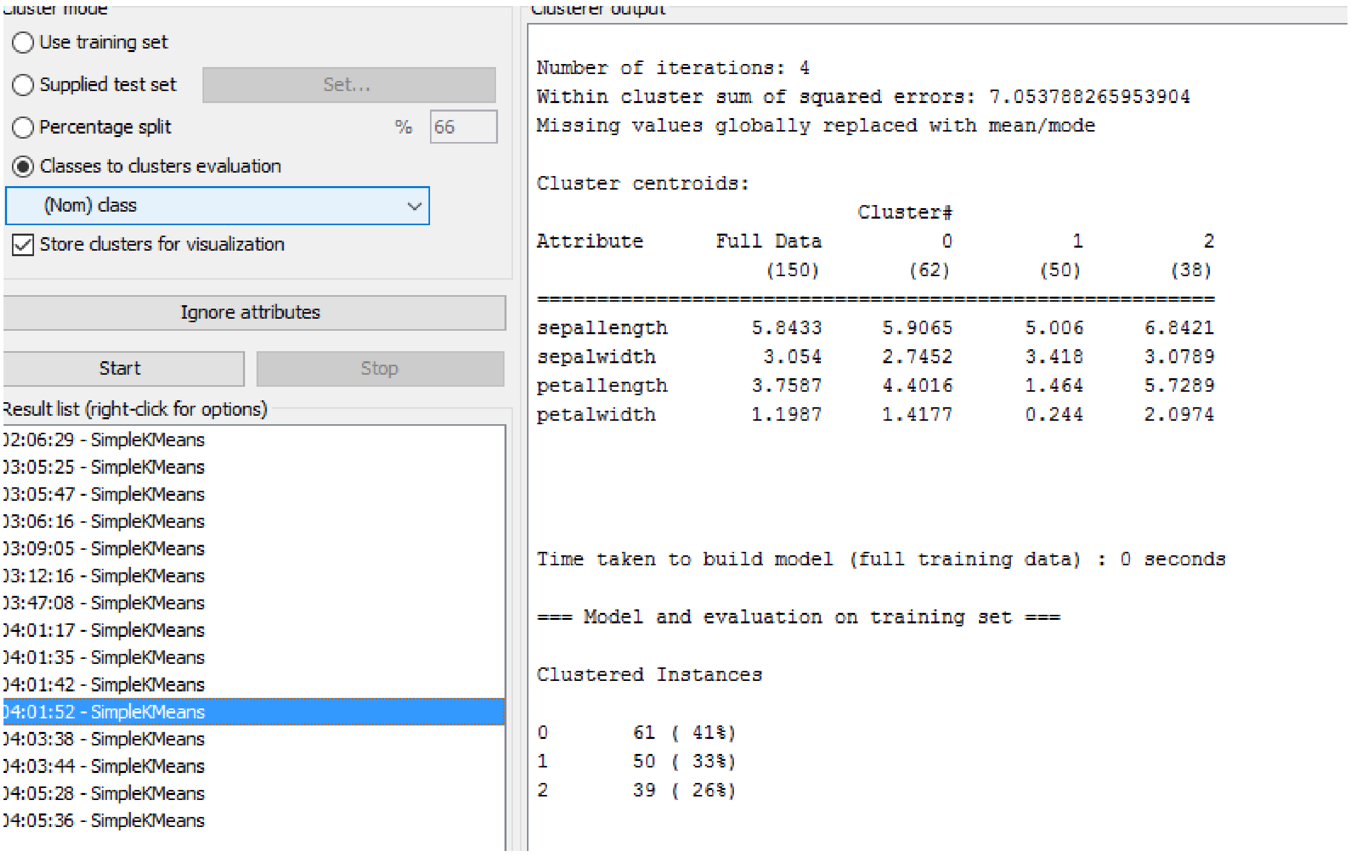


```
======

Number of iterations: 3
Within cluster sum of squared errors: 7.441610579129841
Missing values globally replaced with mean/mode

Cluster centroids:
                           Cluster#
Attribute      Full Data          0          1          2
                   (150)       (66)       (50)       (34)
==========================================================
sepallength       5.8433      5.947      5.006     6.8735
sepalwidth         3.054     2.7561      3.418     3.0971
petallength       3.7587       4.45      1.464     5.7912
petalwidth        1.1987     1.4364      0.244     2.1412




Time taken to build model (full training data) : 0 seconds

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

Clustered Instances

0       62 ( 41%)
1       50 ( 33%)
2       38 ( 25%)
```



```
2       38 ( 25%)


Class attribute: class
Classes to Clusters:

  0  1  2  <-- assigned to cluster
  0 50  0 | Iris-setosa
 48  0  2 | Iris-versicolor
 14  0 36 | Iris-virginica

Cluster 0 <-- Iris-versicolor
Cluster 1 <-- Iris-setosa
Cluster 2 <-- Iris-virginica

Incorrectly clustered instances :        16.0     10.6667 %
```

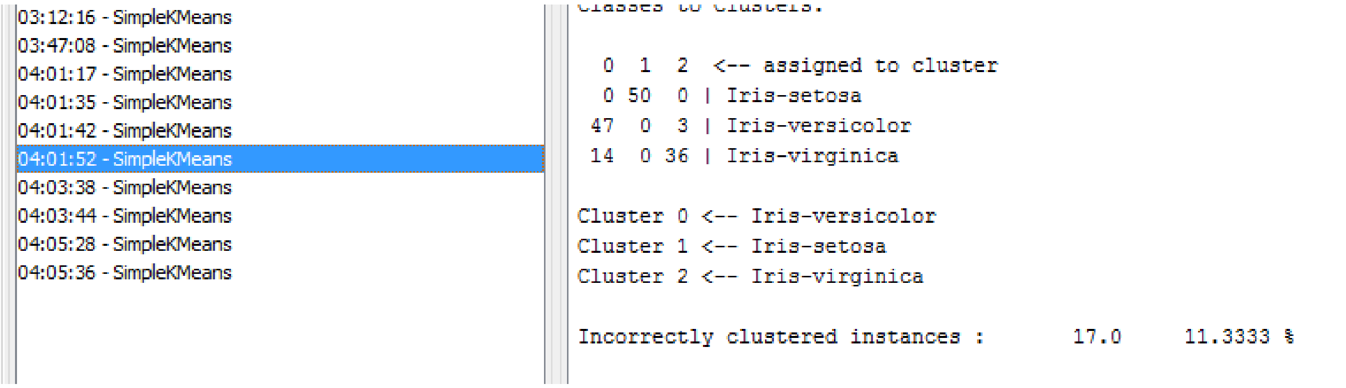
When Iterations = 4



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

```
02:06:29 - SimpleKMeans
03:05:25 - SimpleKMeans
03:05:47 - SimpleKMeans
03:06:16 - SimpleKMeans
03:09:05 - SimpleKMeans
03:12:16 - SimpleKMeans
03:47:08 - SimpleKMeans
04:01:17 - SimpleKMeans
04:01:35 - SimpleKMeans
04:01:42 - SimpleKMeans
04:01:52 - SimpleKMeans
04:03:38 - SimpleKMeans
04:03:44 - SimpleKMeans
04:05:28 - SimpleKMeans
04:05:36 - SimpleKMeans
```

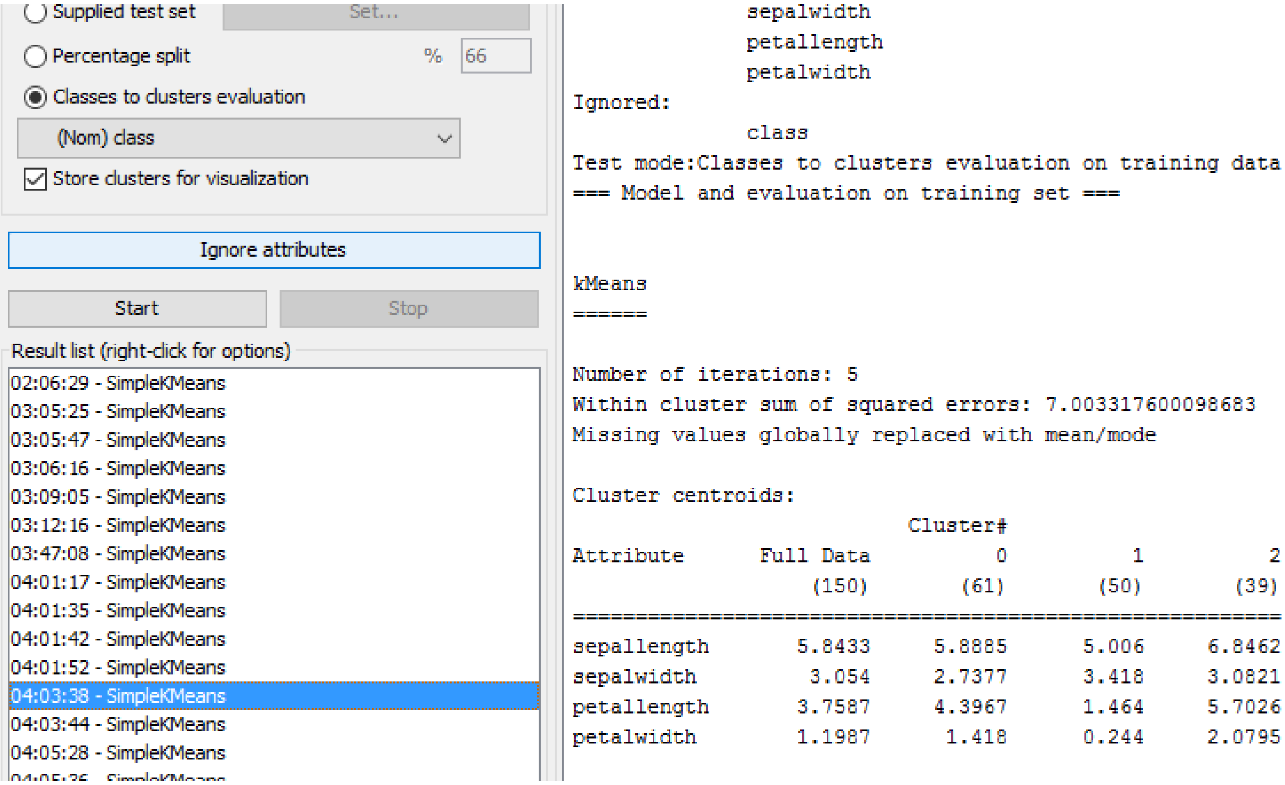
```
Number of iterations: 4
Within cluster sum of squared errors: 7.053788265953904
Missing values globally replaced with mean/mode

Cluster centroids:
                           Cluster#
Attribute      Full Data          0          1          2
                   (150)       (62)       (50)       (38)
==========================================================
sepallength       5.8433     5.9065      5.006     6.8421
sepalwidth         3.054     2.7452      3.418     3.0789
petallength       3.7587     4.4016      1.464     5.7289
petalwidth        1.1987     1.4177      0.244     2.0974




Time taken to build model (full training data) : 0 seconds

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

Clustered Instances

0       61 ( 41%)
1       50 ( 33%)
2       39 ( 26%)
```



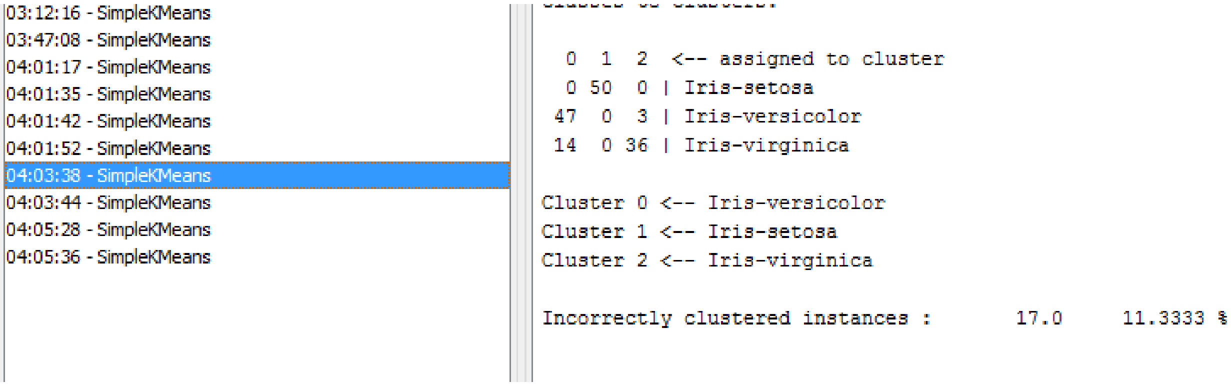
```
03:12:16 - SimpleKMeans
03:47:08 - SimpleKMeans
04:01:17 - SimpleKMeans
04:01:35 - SimpleKMeans
04:01:42 - SimpleKMeans
04:01:52 - SimpleKMeans
04:03:38 - SimpleKMeans
04:03:44 - SimpleKMeans
04:05:28 - SimpleKMeans
04:05:36 - SimpleKMeans
```

```
Classes to Clusters:

  0  1  2  <-- assigned to cluster
  0 50  0 | Iris-setosa
 47  0  3 | Iris-versicolor
 14  0 36 | Iris-virginica

Cluster 0 <-- Iris-versicolor
Cluster 1 <-- Iris-setosa
Cluster 2 <-- Iris-virginica

Incorrectly clustered instances :      17.0      11.3333 %
```

When Iterations = 5

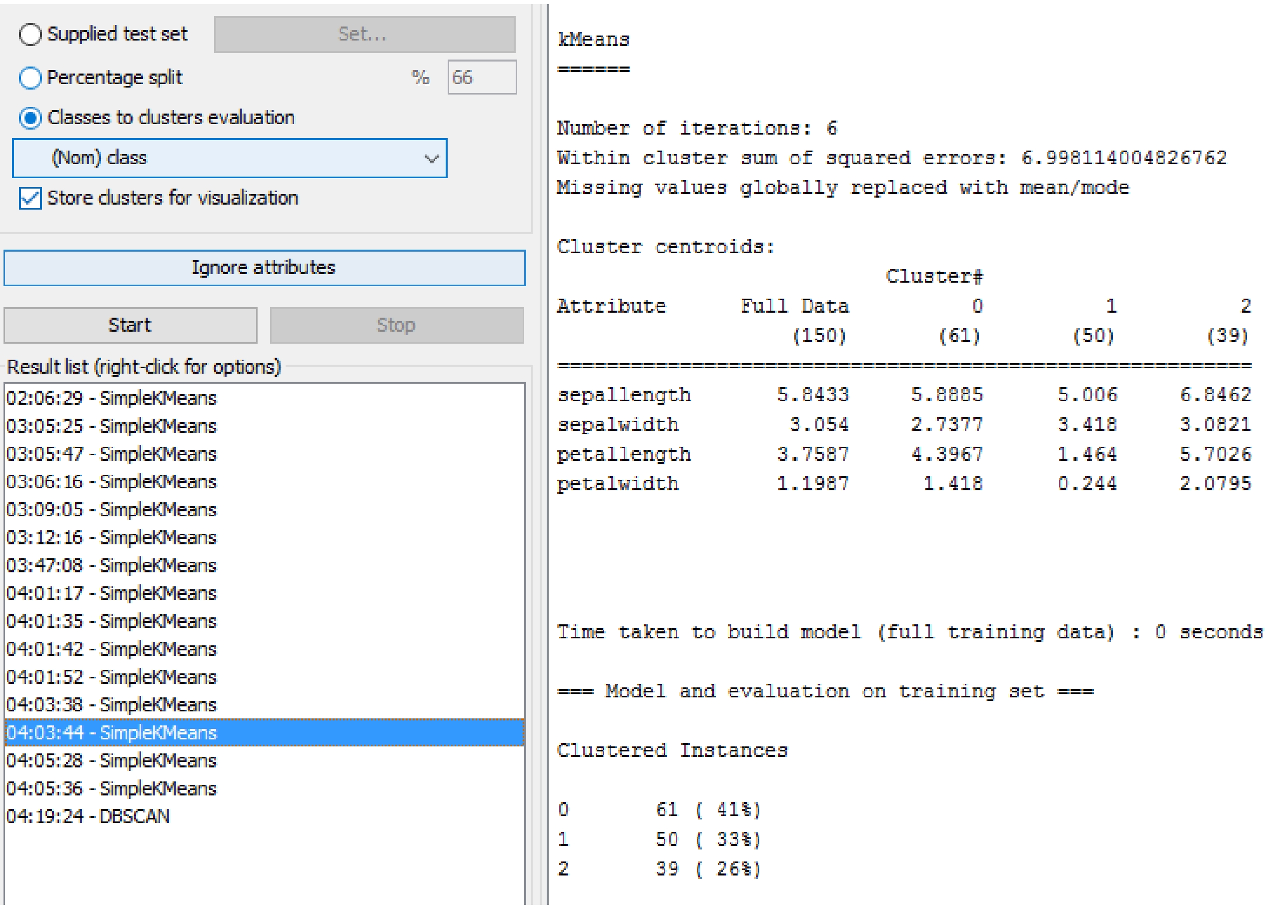


```
              sepalwidth
              petallength
              petalwidth
Ignored:
              class
Test mode:Classes to clusters evaluation on training data
=== Model and evaluation on training set ===


kMeans
======

Number of iterations: 5
Within cluster sum of squared errors: 7.003317600098683
Missing values globally replaced with mean/mode

Cluster centroids:
                          Cluster#
Attribute      Full Data         0          1          2
                   (150)      (61)       (50)       (39)
=========================================================
sepallength       5.8433    5.8885      5.006     6.8462
sepalwidth         3.054    2.7377      3.418     3.0821
petallength       3.7587    4.3967      1.464     5.7026
petalwidth        1.1987     1.418      0.244     2.0795
```



```
Classes to Clusters:

  0  1  2  <-- assigned to cluster
  0 50  0 | Iris-setosa
 47  0  3 | Iris-versicolor
 14  0 36 | Iris-virginica

Cluster 0 <-- Iris-versicolor
Cluster 1 <-- Iris-setosa
Cluster 2 <-- Iris-virginica

Incorrectly clustered instances :       17.0     11.3333 %
```

When Iteration = 6:



○ Supplied test set

○ Percentage split % 66

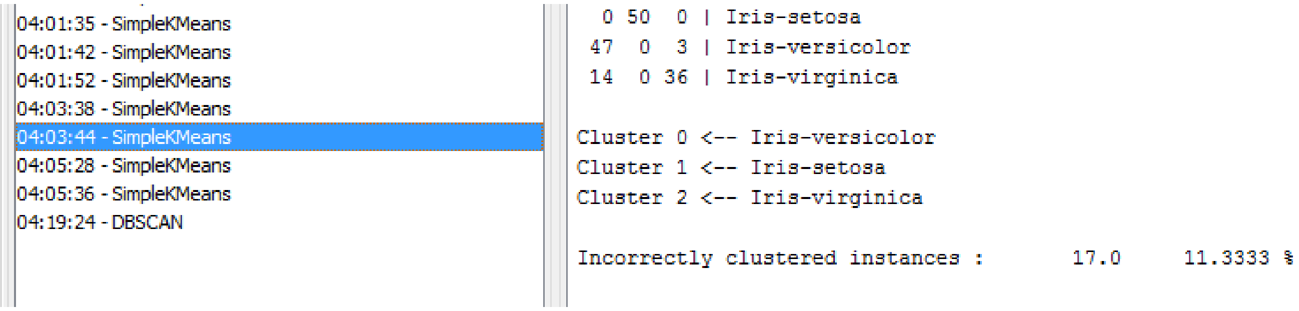
● Classes to clusters evaluation

(Nom) class

☑ Store clusters for visualization

```
02:06:29 - SimpleKMeans
03:05:25 - SimpleKMeans
03:05:47 - SimpleKMeans
03:06:16 - SimpleKMeans
03:09:05 - SimpleKMeans
03:12:16 - SimpleKMeans
03:47:08 - SimpleKMeans
04:01:17 - SimpleKMeans
04:01:35 - SimpleKMeans
04:01:42 - SimpleKMeans
04:01:52 - SimpleKMeans
04:03:38 - SimpleKMeans
04:03:44 - SimpleKMeans
04:05:28 - SimpleKMeans
04:05:36 - SimpleKMeans
04:19:24 - DBSCAN
```

```
kMeans
======

Number of iterations: 6
Within cluster sum of squared errors: 6.998114004826762
Missing values globally replaced with mean/mode

Cluster centroids:
                           Cluster#
Attribute      Full Data          0          1          2
                   (150)       (61)       (50)       (39)
=========================================================
sepallength       5.8433     5.8885      5.006     6.8462
sepalwidth         3.054     2.7377      3.418     3.0821
petallength       3.7587     4.3967      1.464     5.7026
petalwidth        1.1987      1.418      0.244     2.0795




Time taken to build model (full training data) : 0 seconds

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

Clustered Instances

0       61 ( 41%)
1       50 ( 33%)
2       39 ( 26%)
```



```
04:01:35 - SimpleKMeans
04:01:42 - SimpleKMeans
04:01:52 - SimpleKMeans
04:03:38 - SimpleKMeans
04:03:44 - SimpleKMeans
04:05:28 - SimpleKMeans
04:05:36 - SimpleKMeans
04:19:24 - DBSCAN
```

```
  0 50  0 | Iris-setosa
 47  0  3 | Iris-versicolor
 14  0 36 | Iris-virginica

Cluster 0 <-- Iris-versicolor
Cluster 1 <-- Iris-setosa
Cluster 2 <-- Iris-virginica

Incorrectly clustered instances :       17.0     11.3333 %
```

The above is represented in the below tabular column.

No. Of Iterations	Number of Clusters	SSE	Incorrect instances
1	3	36.9375	25
2	3	10.9414	16
3	3	7.4416	16
4	3	7.0537	17
5	3	7.00331	17
6	3	6.9981	17

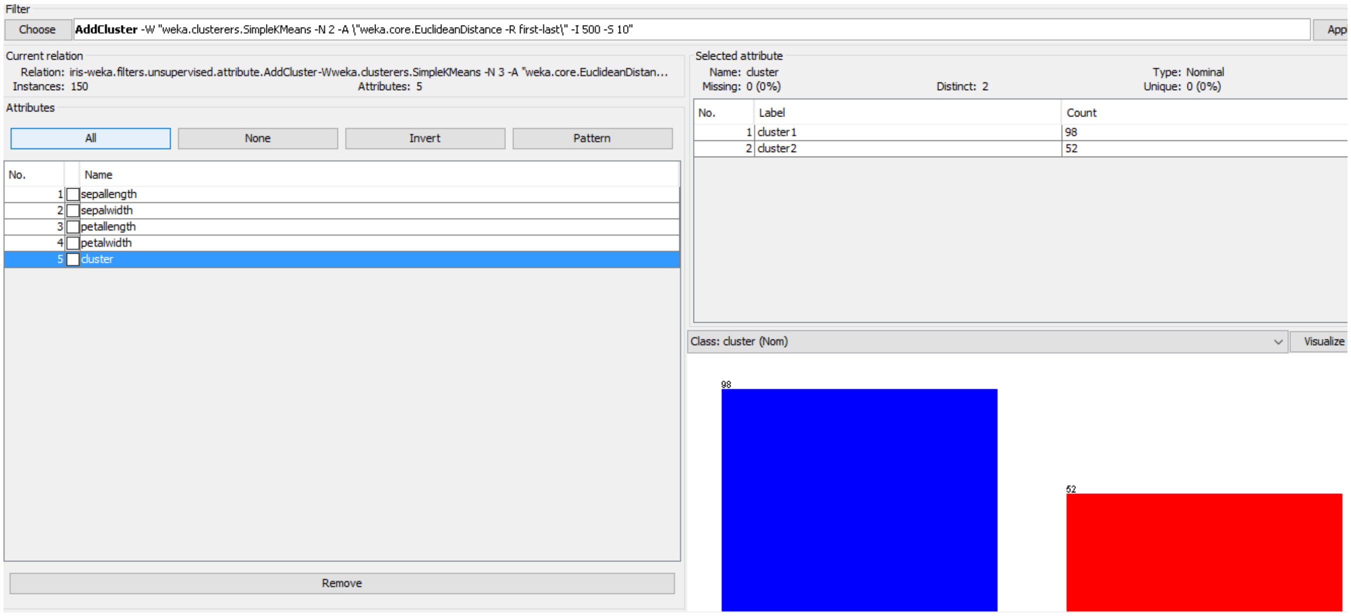
Inference:

- For IRIS dataset, optimum number of clusters is 3 as highlighted above as the SSE is also less and number of incorrect instances is also less when compared to the previous runs.

- Also, though we try increasing the maximum number of iterations, the cycle stops at 6 and does not iterate further. This implies that the centroid has been achieved and cannot be iterated further.

- When Number of clusters = 2, (default value) **without depending on the Class variable** the clusters formed are very poor as it has got high SSE as shown below.

When Number of clusters = 2(default), without depending on Class variable.



AddCluster -W "weka.clusterers.SimpleKMeans -N 2 -A \"weka.core.EuclideanDistance -R first-last\" -I 500 -S 10"

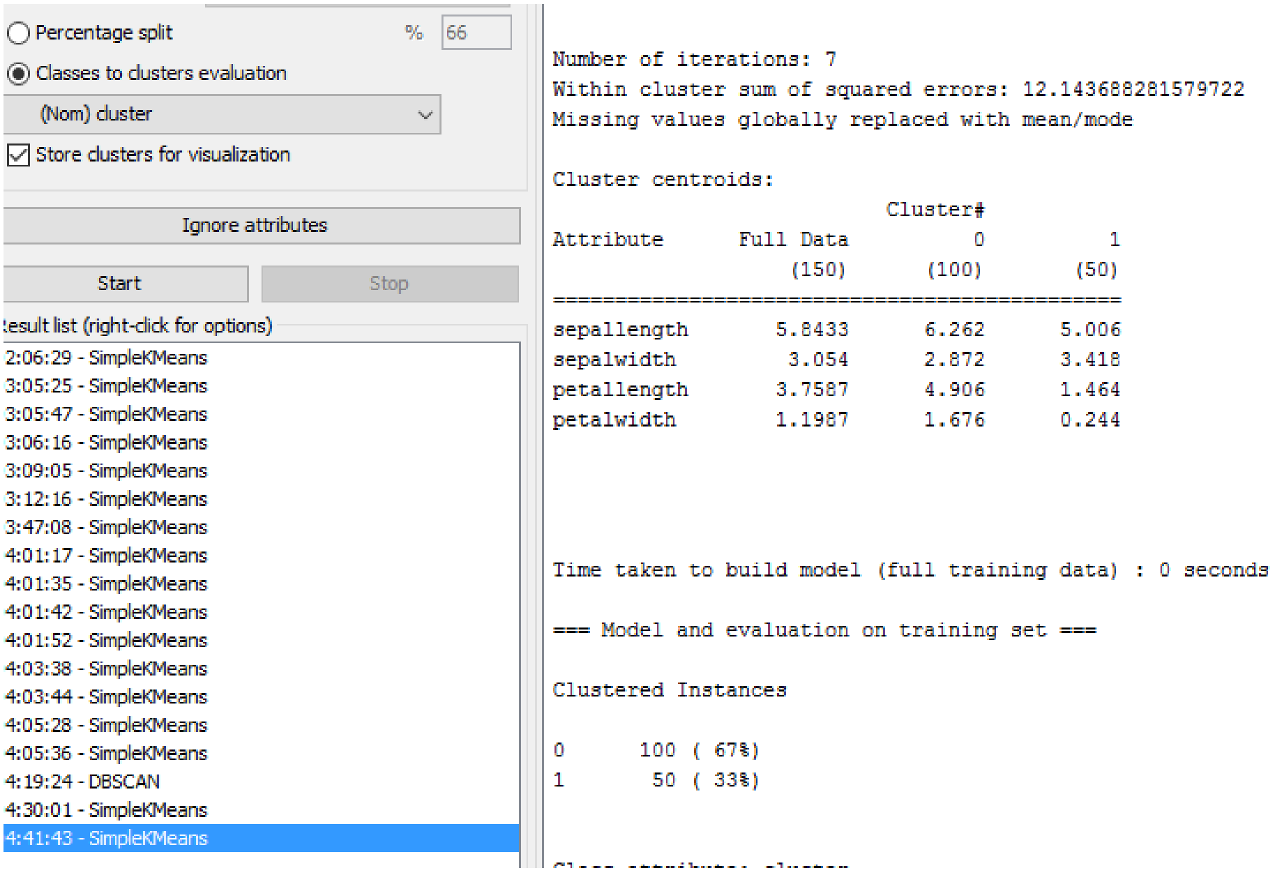
Current relation
Relation: iris-weka.filters.unsupervised.attribute.AddCluster-Wweka.clusterers.SimpleKMeans -N 3 -A "weka.core.EuclideanDistan...
Instances: 150 Attributes: 5

No.		Name
1	□	sepallength
2	□	sepalwidth
3	□	petallength
4	□	petalwidth
5	□	cluster

Selected attribute
Name: cluster Type: Nominal
Missing: 0 (0%) Distinct: 2 Unique: 0 (0%)

No.	Label	Count
1	cluster1	98
2	cluster2	52

Class: cluster (Nom)



○ Percentage split % 66

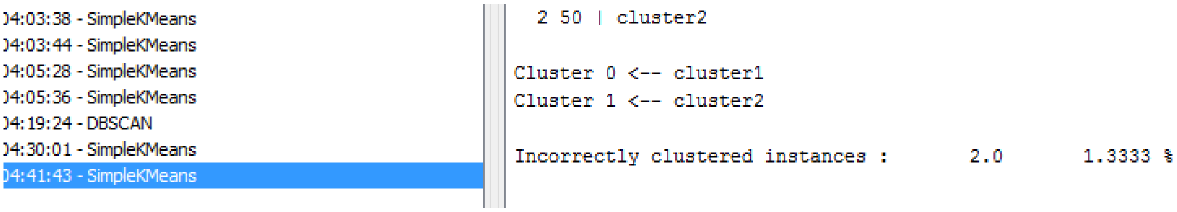
● Classes to clusters evaluation

(Nom) cluster

☑ Store clusters for visualization

```
2:06:29 - SimpleKMeans
3:05:25 - SimpleKMeans
3:05:47 - SimpleKMeans
3:06:16 - SimpleKMeans
3:09:05 - SimpleKMeans
3:12:16 - SimpleKMeans
3:47:08 - SimpleKMeans
4:01:17 - SimpleKMeans
4:01:35 - SimpleKMeans
4:01:42 - SimpleKMeans
4:01:52 - SimpleKMeans
4:03:38 - SimpleKMeans
4:03:44 - SimpleKMeans
4:05:28 - SimpleKMeans
4:05:36 - SimpleKMeans
4:19:24 - DBSCAN
4:30:01 - SimpleKMeans
4:41:43 - SimpleKMeans
```

```
Number of iterations: 7
Within cluster sum of squared errors: 12.143688281579722
Missing values globally replaced with mean/mode

Cluster centroids:
                            Cluster#
Attribute       Full Data          0           1
                    (150)      (100)        (50)
==============================================
sepallength        5.8433      6.262       5.006
sepalwidth          3.054      2.872       3.418
petallength        3.7587      4.906       1.464
petalwidth         1.1987      1.676       0.244




Time taken to build model (full training data) : 0 seconds

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

Clustered Instances

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


Class attribute: cluster
```



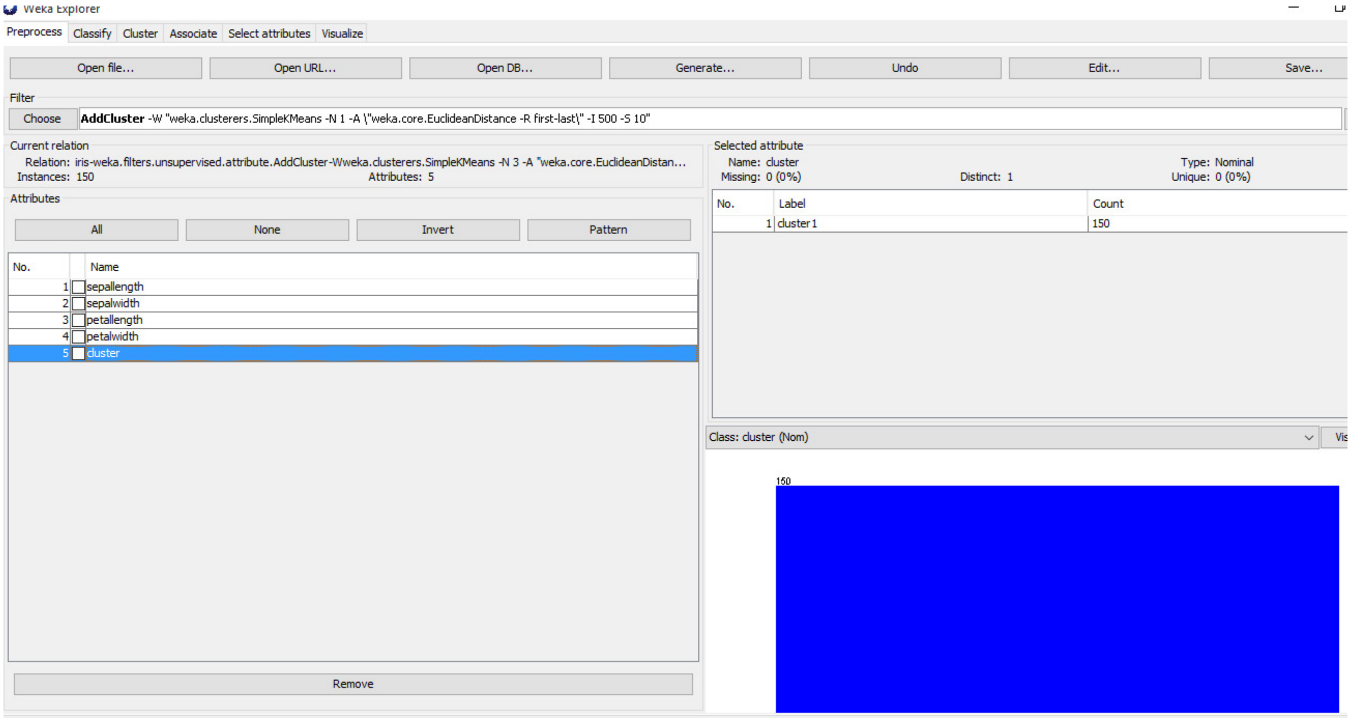
```
04:03:38 - SimpleKMeans
04:03:44 - SimpleKMeans
04:05:28 - SimpleKMeans
04:05:36 - SimpleKMeans
04:19:24 - DBSCAN
04:30:01 - SimpleKMeans
04:41:43 - SimpleKMeans
```

```
  2 50 | cluster2

Cluster 0 <-- cluster1
Cluster 1 <-- cluster2

Incorrectly clustered instances :       2.0       1.3333 %
```

Test case 5:- IRIS Dataset under KMeans algorithm after changing Parameters = Number of clusters.

When Number of clusters parameters are being changed.

When Number of clusters = 1



Filter

AddCluster -W "weka.clusterers.SimpleKMeans -N 1 -A \"weka.core.EuclideanDistance -R first-last\" -I 500 -S 10"

Current relation

Relation: iris-weka.filters.unsupervised.attribute.AddCluster-Wweka.clusterers.SimpleKMeans -N 3 -A "weka.core.EuclideanDistan...

Instances: 150 Attributes: 5

Attributes

No.	Name
1	□ sepallength
2	□ sepalwidth
3	□ petallength
4	□ petalwidth
5	□ cluster

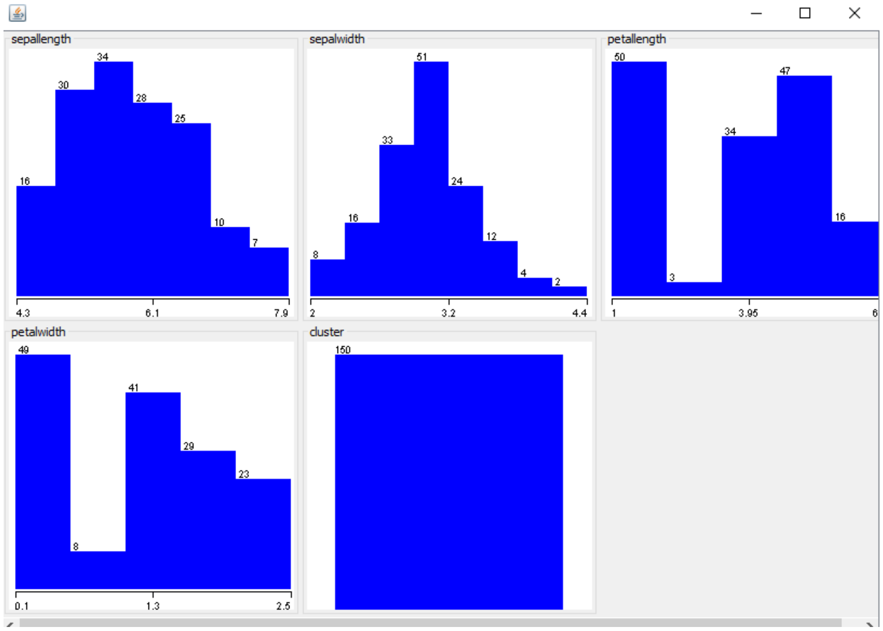
Selected attribute

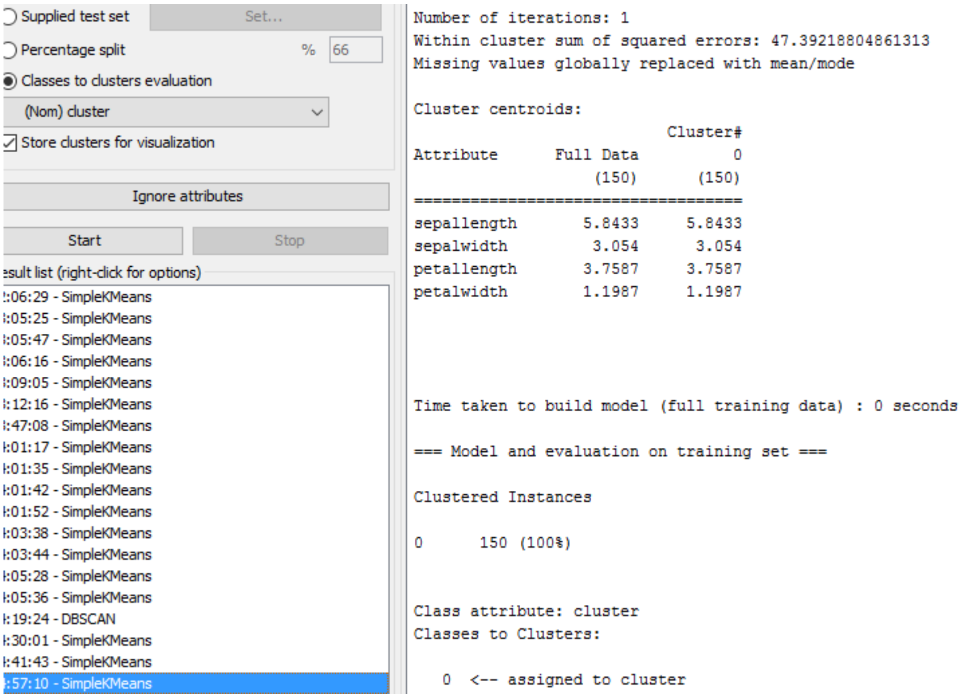
Name: cluster Type: Nominal

Missing: 0 (0%) Distinct: 1 Unique: 0 (0%)

No.	Label	Count
1	cluster1	150

Class: cluster (Nom)





○ Supplied test set

○ Percentage split % 66

● Classes to clusters evaluation

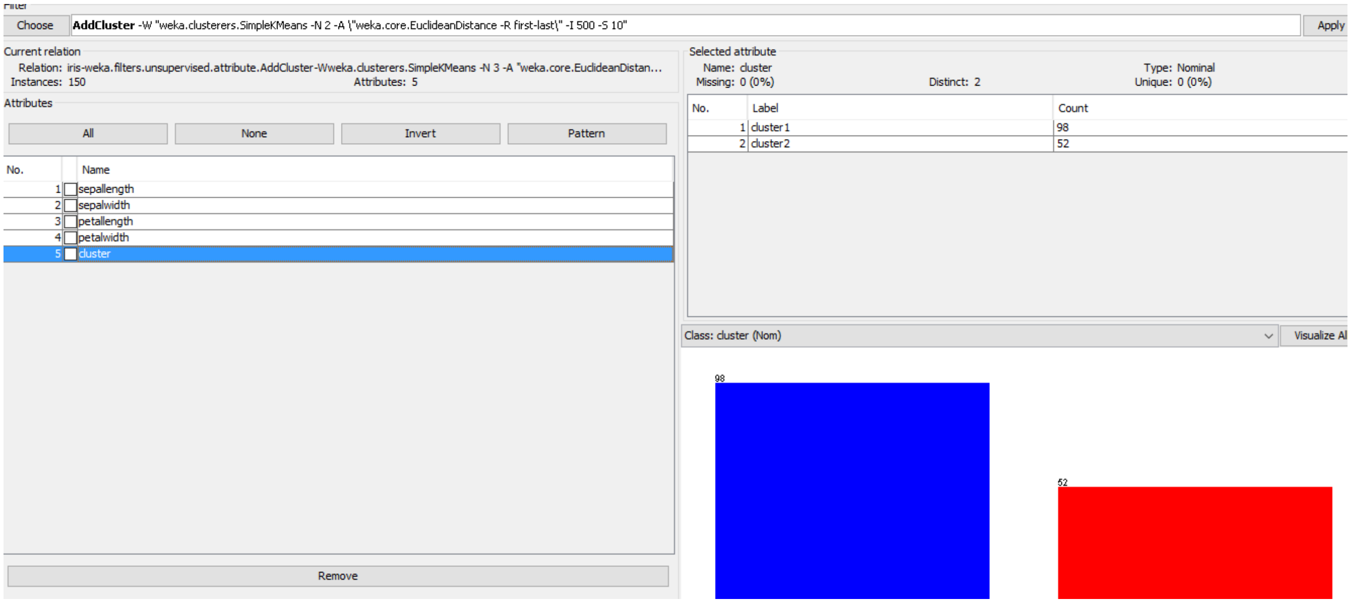
(Nom) cluster

☑ Store clusters for visualization

```
Number of iterations: 1
Within cluster sum of squared errors: 47.39218804861313
Missing values globally replaced with mean/mode

Cluster centroids:
                            Cluster#
Attribute      Full Data           0
                   (150)       (150)
====================================
sepallength       5.8433      5.8433
sepalwidth         3.054       3.054
petallength       3.7587      3.7587
petalwidth        1.1987      1.1987




Time taken to build model (full training data) : 0 seconds

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

Clustered Instances

0      150 (100%)


Class attribute: cluster
Classes to Clusters:

   0  <-- assigned to cluster
```

Number of incorrectly classified instances = 0 as all the instances have been put under one main cluster.

When Number of clusters = 2:



Filter

AddCluster -W "weka.clusterers.SimpleKMeans -N 2 -A \"weka.core.EuclideanDistance -R first-last\" -I 500 -S 10"

Current relation

Relation: iris-weka.filters.unsupervised.attribute.AddCluster-Wweka.clusterers.SimpleKMeans -N 3 -A "weka.core.EuclideanDistan...

Instances: 150 Attributes: 5

Attributes

No.		Name
1	□	sepallength
2	□	sepalwidth
3	□	petallength
4	□	petalwidth
5	□	cluster

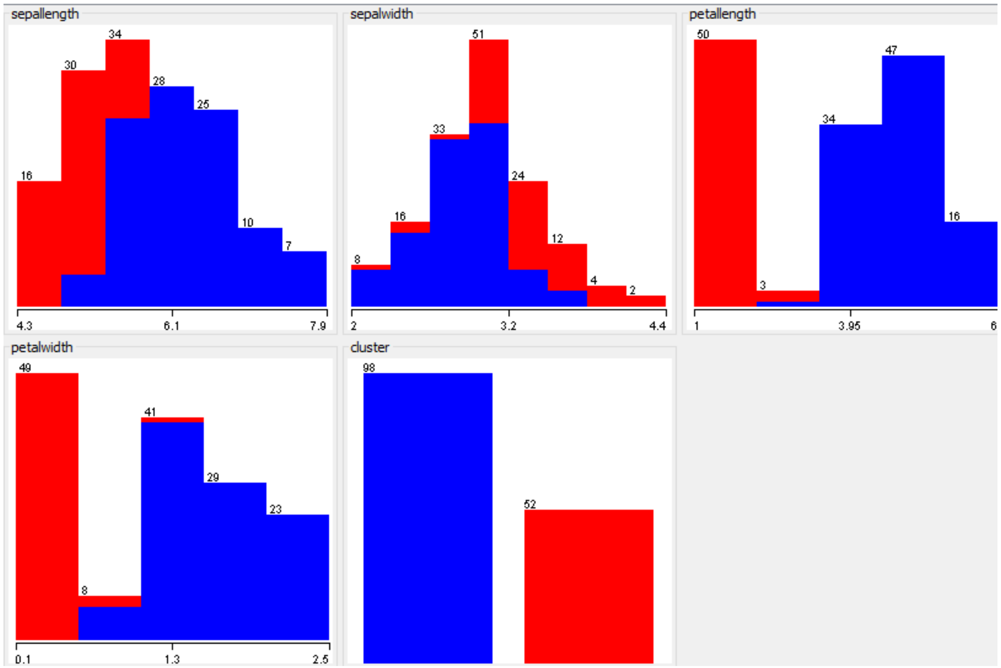
Selected attribute

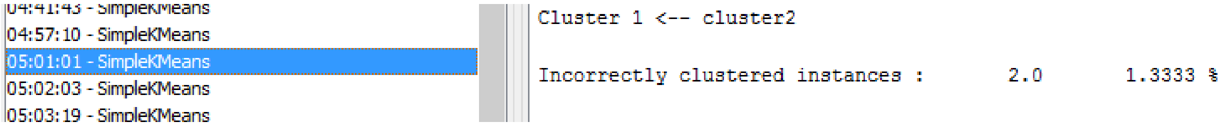
Name: cluster Type: Nominal

Missing: 0 (0%) Distinct: 2 Unique: 0 (0%)

No.	Label	Count
1	cluster1	98
2	cluster2	52

Class: cluster (Nom)

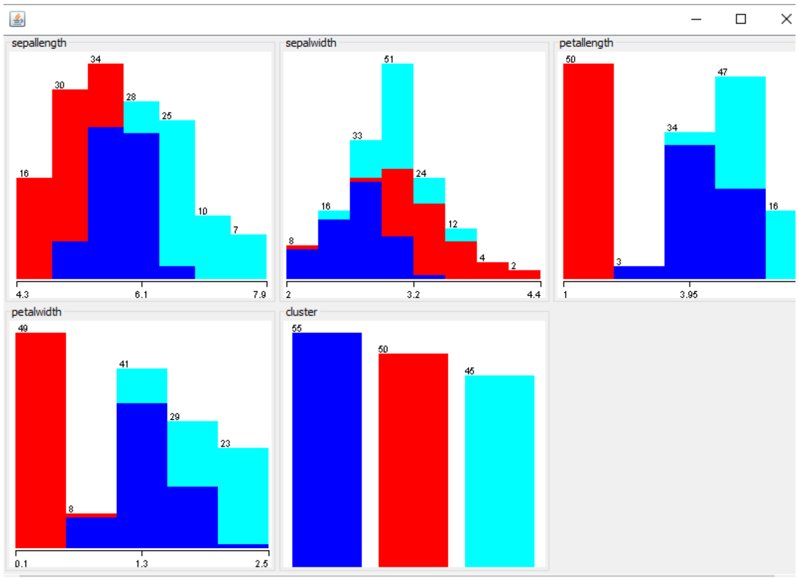




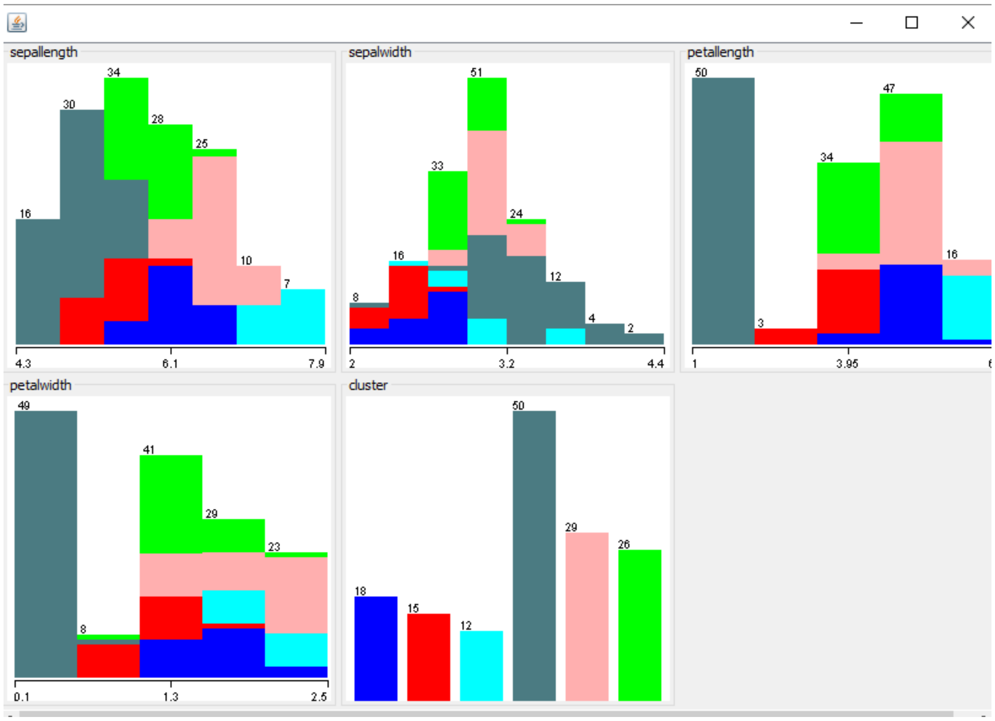
```
04:41:43 - SimpleKMeans
04:57:10 - SimpleKMeans
05:01:01 - SimpleKMeans
05:02:03 - SimpleKMeans
05:03:19 - SimpleKMeans
```

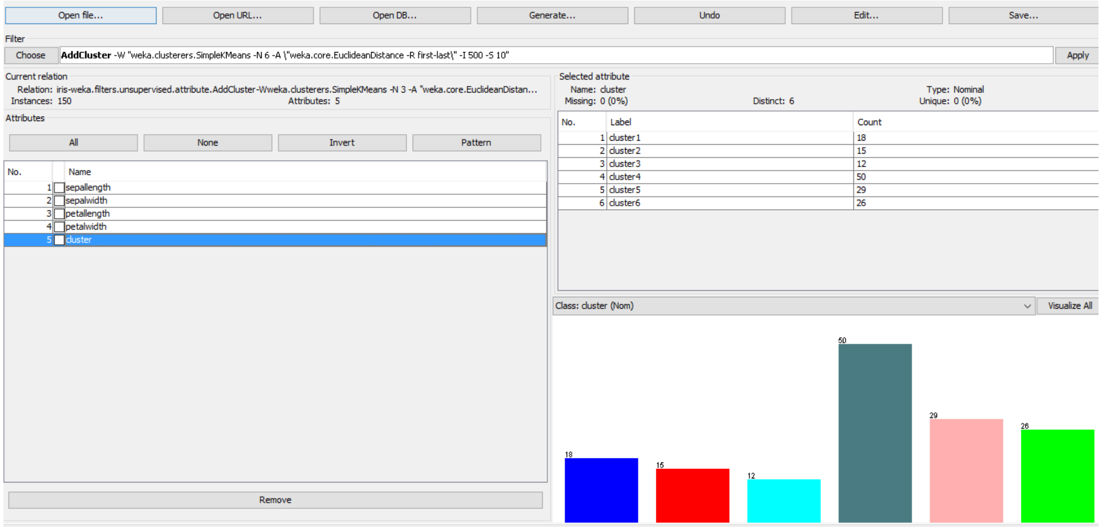
```
Cluster 1 <-- cluster2

Incorrectly clustered instances :       2.0       1.3333 %
```

When Number of clusters = 3



Similarly for the 6 clusters:





Filter

AddCluster -W "weka.clusterers.SimpleKMeans -N 6 -A \"weka.core.EuclideanDistance -R first-last\" -I 500 -S 10"

Current relation
Relation: iris-weka.filters.unsupervised.attribute.AddCluster-Wweka.clusterers.SimpleKMeans -N 3 -A "weka.core.EuclideanDistan...
Instances: 150 Attributes: 5

Attributes

No.		Name
1	□	sepallength
2	□	sepalwidth
3	□	petallength
4	□	petalwidth
5	□	cluster

Selected attribute
Name: cluster Type: Nominal
Missing: 0 (0%) Distinct: 6 Unique: 0 (0%)

No.	Label	Count
1	cluster1	18
2	cluster2	15
3	cluster3	12
4	cluster4	50
5	cluster5	29
6	cluster6	26

Class: cluster (Nom)

Inference:

- The number of clusters is increased and more possible chances of SSE and incorrect instances being reported.
- When the number of cluster increases, the misplaced clustered instances decreases.
- When the number of clusters further increases and eventually leads to outliers and missing data thereby increasing the incorrect instances.

The below tabular column is derived from the above screenshots.

Max No. Of Iterations	Actual No. of iterations	Number of Clusters	SSE	Incorrect Instances
500	1	1	47.3921	0
500	7	2 (default)	12.1436	2
500	6	3	6.998	8
500	4	4	5.5328	17
500	9	5	5.1307	32
500	7	6	4.6870	28

Here the number of clusters increases and better SSE is achieved.

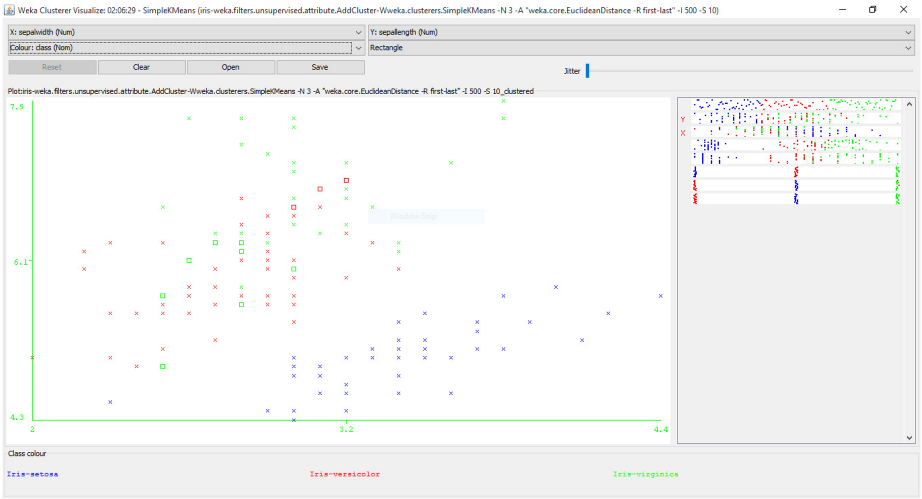
Once threshold level is reached, SSE decreases but Number of incorrectly clustered instances increases.

Test case 6:- IRIS Dataset under KMeans algorithm after opting some of the attributes.

Output of KMeans:

Cluster using attributes Sepal Width, Sepal Length:

The clusters are formed using the attributes Sepal Length and Sepal Width.



Weka Clusterer Visualize: 02:06:29 - SimpleKMeans (iris-weka.filters.unsupervised.attribute.AddCluster-Wweka.clusterers.SimpleKMeans -N 3 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 10)

X: sepalwidth (Num)

Y: sepallength (Num)

Colour: class (Nom)

Rectangle

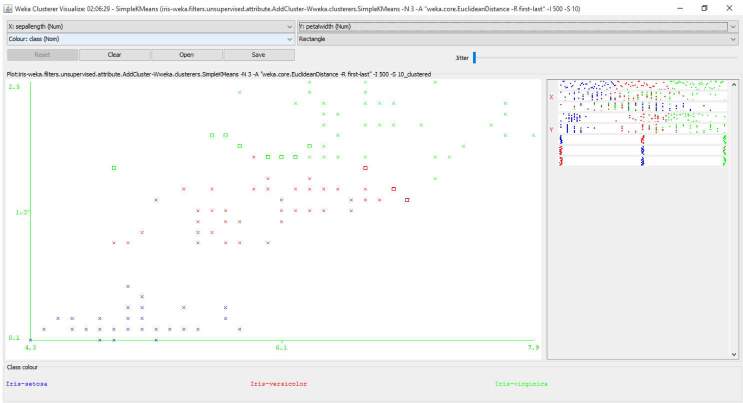
Plot:iris-weka.filters.unsupervised.attribute.AddCluster-Wweka.clusterers.SimpleKMeans -N 3 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 10_clustered

We will be able to find the inter cluster distance within each cluster. The Inter-cluster distance is very high in this case.

Inference: The **points inside the clusters are not uniform and are distorted**, leading to a total SSE which is larger.

Cluster using attributes Sepal Length, Petal width:

The clusters are formed using the attributes Sepal Length and Petal Width.



Weka Clusterer Visualize: 02:06:29 - SimpleKMeans (iris-weka.filters.unsupervised.attribute.AddCluster-Wweka.clusterers.SimpleKMeans -N 3 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 10)

X: sepallength (Num)

Y: petalwidth (Num)

Colour: class (Nom)

Rectangle

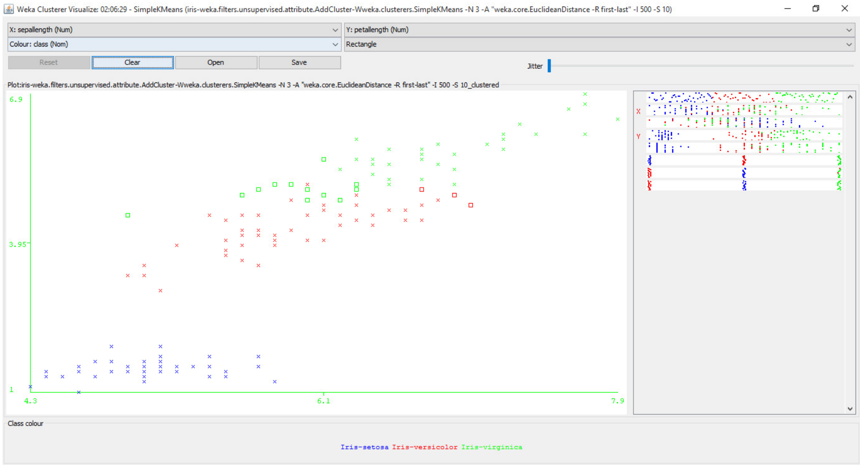
Plot:iris-weka.filters.unsupervised.attribute.AddCluster-Wweka.clusterers.SimpleKMeans -N 3 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 10_clustered

The distribution of the clusters in petallength and sepal width attributes shows that, **Iris setosa and Iris versicolor are dense while the other cluster is less dense**.

Hence the SSE which is calculated is slightly high when compared to the previous run.

Cluster using attributes Sepal Length and Petal Length:

The clusters are formed using the attributes Sepal Length and Petal Length.



Weka Clusterer Visualize: 02:06:29 - SimpleKMeans (iris-weka.filters.unsupervised.attribute.AddCluster-Wweka.clusterers.SimpleKMeans -N 3 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 10)

X: sepallength (Num)

Y: petallength (Num)

Colour: class (Nom)

Rectangle

Plot:iris-weka.filters.unsupervised.attribute.AddCluster-Wweka.clusterers.SimpleKMeans -N 3 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 10_clustered

In the above figure, Intra cluster distance is more.

In this figure **the intra cluster distance is more in Iris-Virginica but the clusters Iris-versicolor and Iris virginica have very small inter cluster distance, hence the total SSE is very small** when compared to the other cluster formed during the previous runs.

Also, Iris-Setosa has less intra cluster distance and hence SSE is small in this case leading to a pure Cluster.

The above figure shows that, there are some Misclassified instances being reported due to which they don't fall under any Class or cluster.

Inference:

- Selecting specific attributes results in good clustering with min SSE and min Number of incorrect instances.
- If the attribute which are selected represents the data well then they are to be called good clusters.

DBSCAN on IRIS dataset:

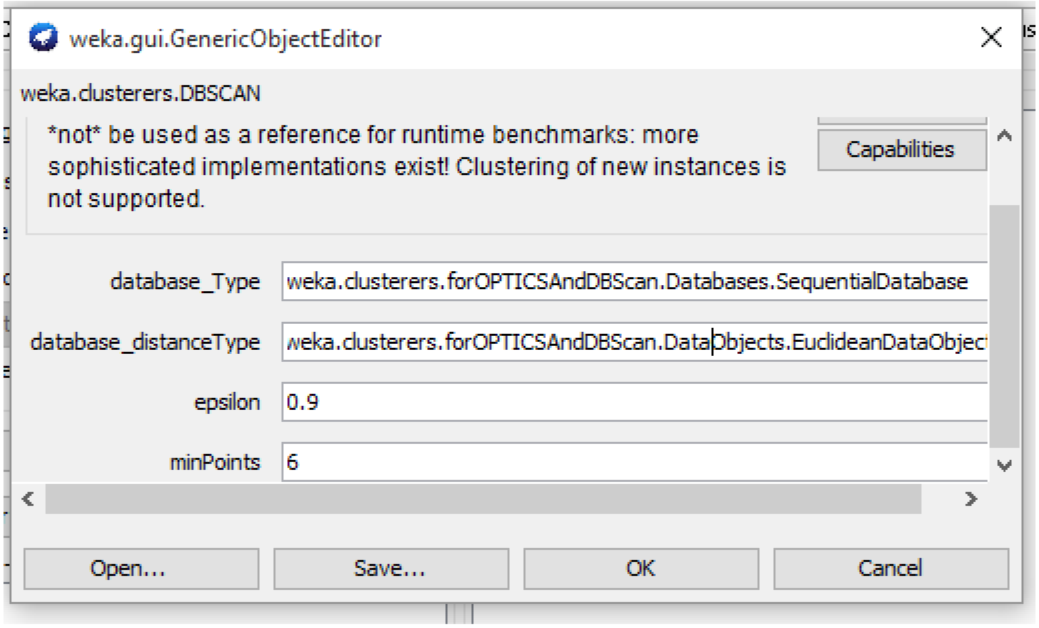
Input to DB Scan Algorithm:

database_Type – Describes what type of dataset is used.

database_distanceType – Describes the distance type of instances in the dataset

epsilon -- radius of the core point within which the data points should fall.

minPoints – specifies the number of data points that falls under the radius epsilon.



weka.gui.GenericObjectEditor

weka.clusterers.DBSCAN

not be used as a reference for runtime benchmarks: more sophisticated implementations exist! Clustering of new instances is not supported.

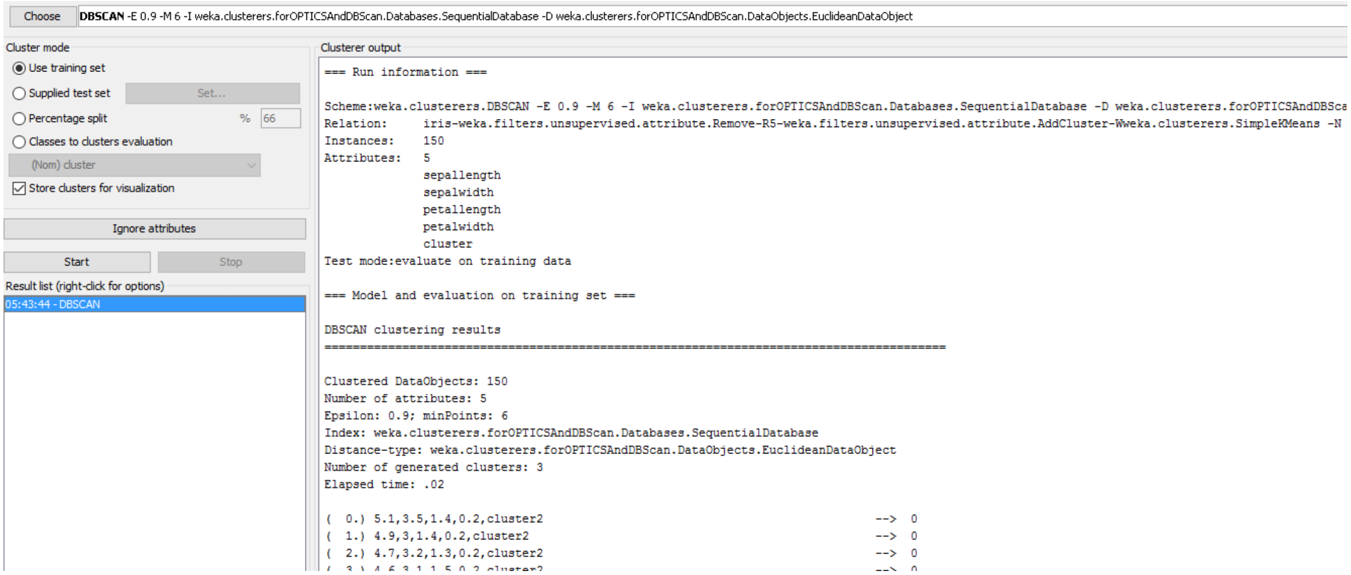
database_Type weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase

database_distanceType weka.clusterers.forOPTICSAndDBScan.DataObjects.EuclideanDataObject

epsilon 0.9

minPoints 6

Output of DBSCAN for IRIS dataset:



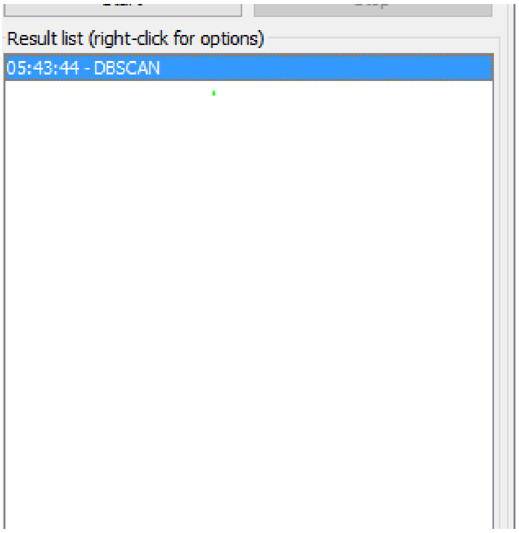
DBSCAN -E 0.9 -M 6 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAndDBScan.DataObjects.EuclideanDataObject

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

05:43:44 - DBSCAN

Clusterer output

```
=== Run information ===

Scheme:weka.clusterers.DBSCAN -E 0.9 -M 6 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAndDBSca
Relation:     iris-weka.filters.unsupervised.attribute.Remove-R5-weka.filters.unsupervised.attribute.AddCluster-Wweka.clusterers.SimpleKMeans -N
Instances:    150
Attributes:   5
              sepallength
              sepalwidth
              petallength
              petalwidth
              cluster
Test mode:evaluate on training data

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

DBSCAN clustering results
========================================================================================

Clustered DataObjects: 150
Number of attributes: 5
Epsilon: 0.9; minPoints: 6
Index: weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase
Distance-type: weka.clusterers.forOPTICSAndDBScan.DataObjects.EuclideanDataObject
Number of generated clusters: 3
Elapsed time: .02

(  0.) 5.1,3.5,1.4,0.2,cluster2                                          -->  0
(  1.) 4.9,3,1.4,0.2,cluster2                                            -->  0
(  2.) 4.7,3.2,1.3,0.2,cluster2                                          -->  0
(  3.) 4.6,3.1,1.5,0.2,cluster2                                          -->  0
```



Result list (right-click for options)

05:43:44 - DBSCAN

```
(144.) 6.7,3.3,5.7,2.5,cluster3
(145.) 6.7,3,5.2,2.3,cluster3
(146.) 6.3,2.5,5,1.9,cluster1
(147.) 6.5,3,5.2,2,cluster3
(148.) 6.2,3.4,5.4,2.3,cluster3
(149.) 5.9,3,5.1,1.8,cluster1



Time taken to build model (full training data) : 0.02 seconds

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

Clustered Instances

0       50 ( 33%)
1       45 ( 30%)
2       55 ( 37%)
```

This shows the clustering of the instances in the given dataset (IRIS).

A single instance clustered to a label or labelled as noise, if it does not fall in any range between the epsilon range of any core point.

Test case 7:- Impact of changing epsilon and minpts parameter in DBSCAN for IRIS Dataset.

For a good cluster, number of incorrectly specified instances must be less.

Clustering value happens when epsilon value is low and minpts is high.

After changing minpts from default value to 15, 20, 25, 30, 35, 15, 30, 35, 15, 35,10

Epsilon value keeps changing from 0.2 → 0.3 → 0.4 → 0.5

Number of clusters generated = 3

Epsilon	Minpts	Number of Incorrectly clustered Instances	Number of unclustered Instances
0.2	15	23	16
0.2	20	16	35
0.2	25	8	63
0.2	30	4	105
0.2	35	0	109
0.3	15	17	2
0.3	30	16	6
0.3	35	15	9
0.4	15	17	0
0.4	30	17	0
0.4	35	17	0
0.5	15	17	0

- Initially the epsilon value is kept constant and the minpts is increased ,we could infer that the Number of Incorrectly clustered instances decreases and Number of Unclustered Instances increases.
- Here since the unclustered instances leads to noise we can ignore them or give no/less importance while forming the clusters.
- When the Epsilon value is increased to 0.3, and when minPts is also increased, number of incorrectly clustered instances decreases with the number of unclustered instance.
- Similarly when Epsilon = 5 or more than 5, there are high chances of all instances to be reported incorrectly clustered instances.

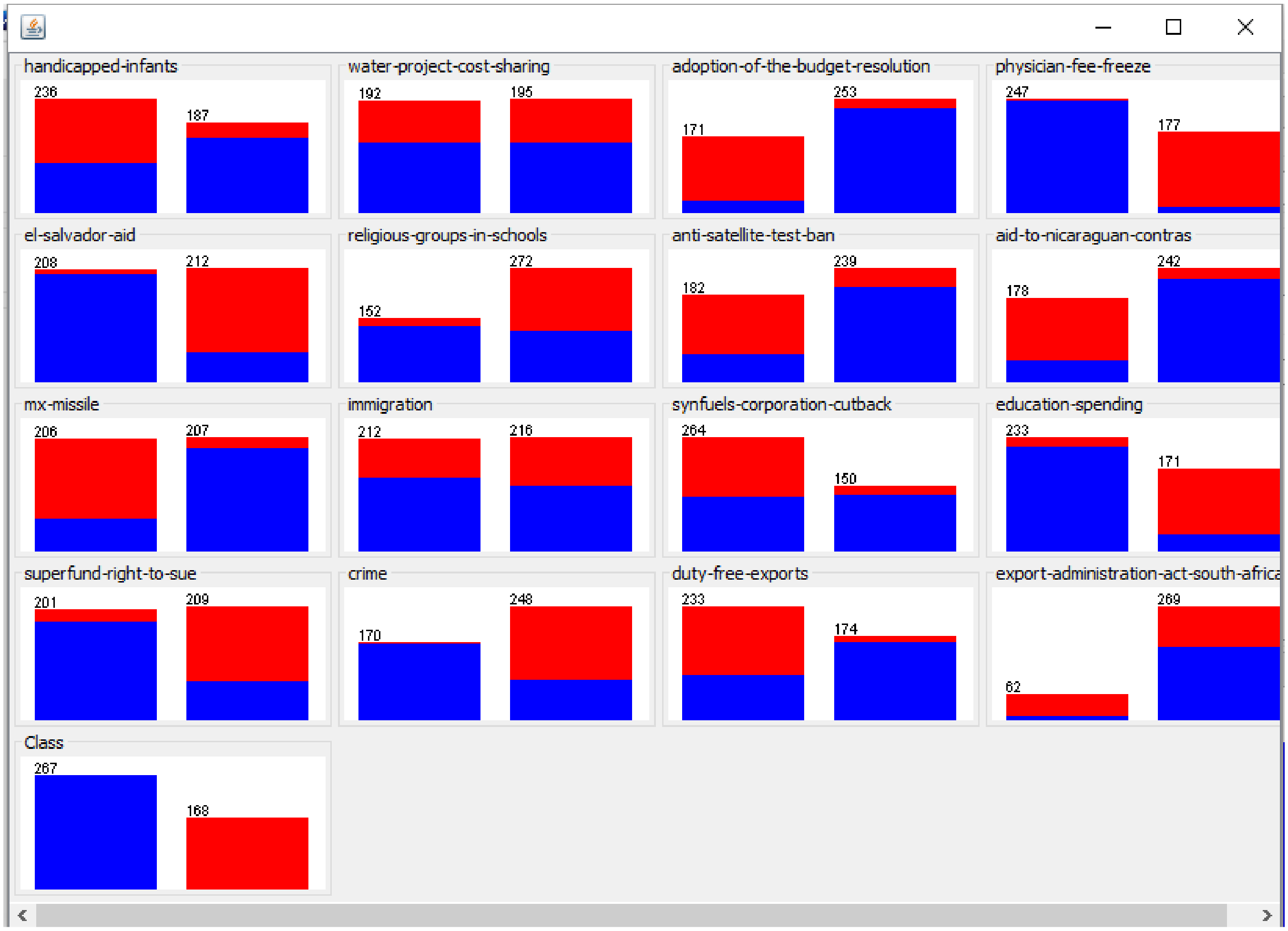
Test case 8:- Impact of changing epsilon and minpts parameter in DBSCAN for IRIS Dataset.

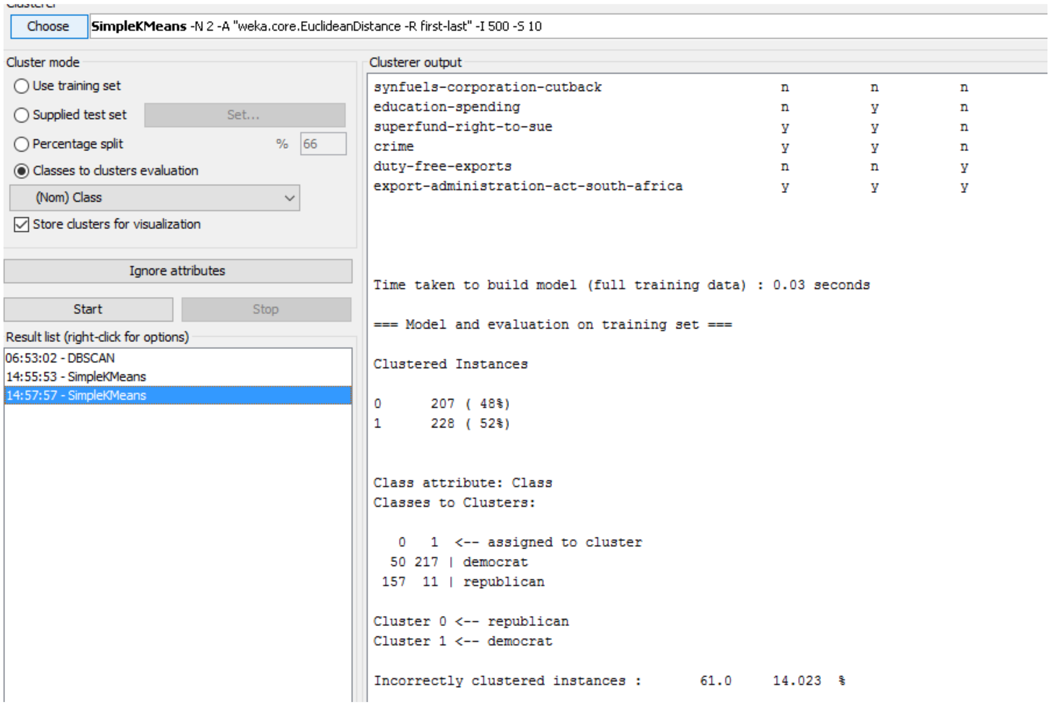
Attributes selected for Clustering	Number of UnClustered Instances	Number of Incorrectly Clustered Instances
sepal width ,petal width	9	42
petal length, petal width	5	45
sepal length ,sepal width	27	36

The attributes above were given in the same order and then inferences are made accordingly on the Number of unclustered instances and number of incorrectly clustered instances.

Running kMeans algorithm for VOTE DATASET

The below fig, shows the visualize part of VOTE DATASET with all the attributes.





SimpleKMeans -N 2 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 10

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

Result list:

- 06:53:02 - DBSCAN
- 14:55:53 - SimpleKMeans
- 14:57:57 - SimpleKMeans

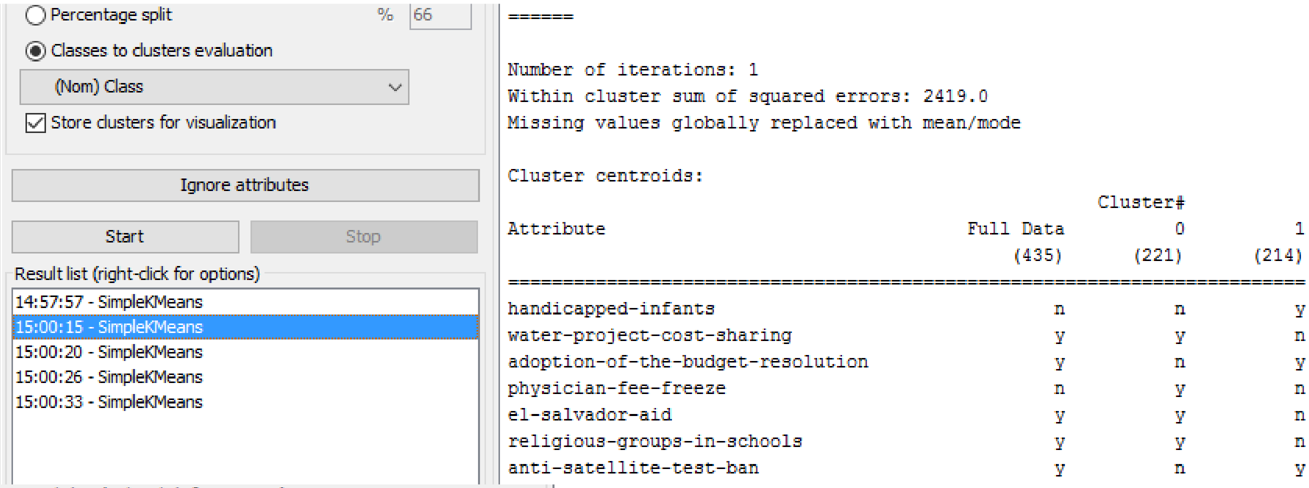
```
synfuels-corporation-cutback               n          n          n
education-spending                         n          y          n
superfund-right-to-sue                     y          y          n
crime                                      y          y          n
duty-free-exports                          n          n          y
export-administration-act-south-africa     y          y          y




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

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

Clustered Instances

0       207 ( 48%)
1       228 ( 52%)


Class attribute: Class
Classes to Clusters:

   0   1  <-- assigned to cluster
  50 217 | democrat
 157  11 | republican

Cluster 0 <-- republican
Cluster 1 <-- democrat

Incorrectly clustered instances :       61.0     14.023  %
```

The above fig, shows the simple kMeans algorithm when run against VOTE Datset specifies 61 incorrectly clustered instances and the SSE 1449.0 with iterations 3.

Impact of Changing the Number of Iterations parameter in K-means algorithm over Vote Dataset:

NO. of Iterations during clustering	Number of Clusters (Constant)	SSE	Number of Incorrectly Clustered Instances
1	2	2419	61
2	2	1449	61
3	2	1449	61
4	2	1449	61

When Iterations = 1



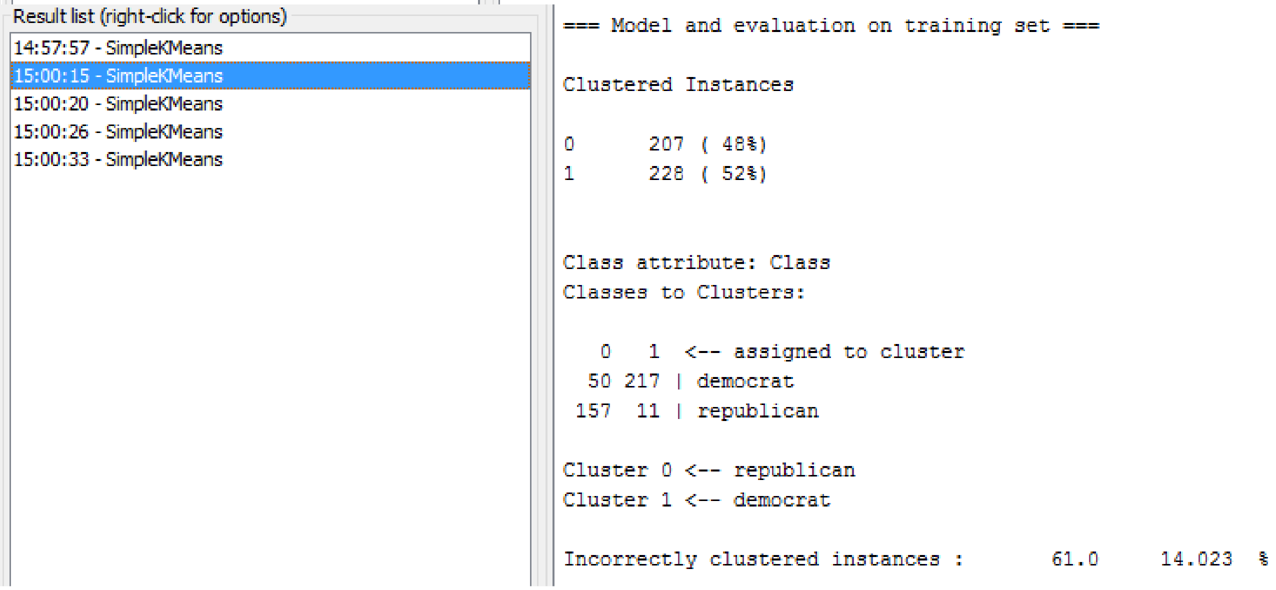
○ Percentage split % 66

● Classes to clusters evaluation

(Nom) Class

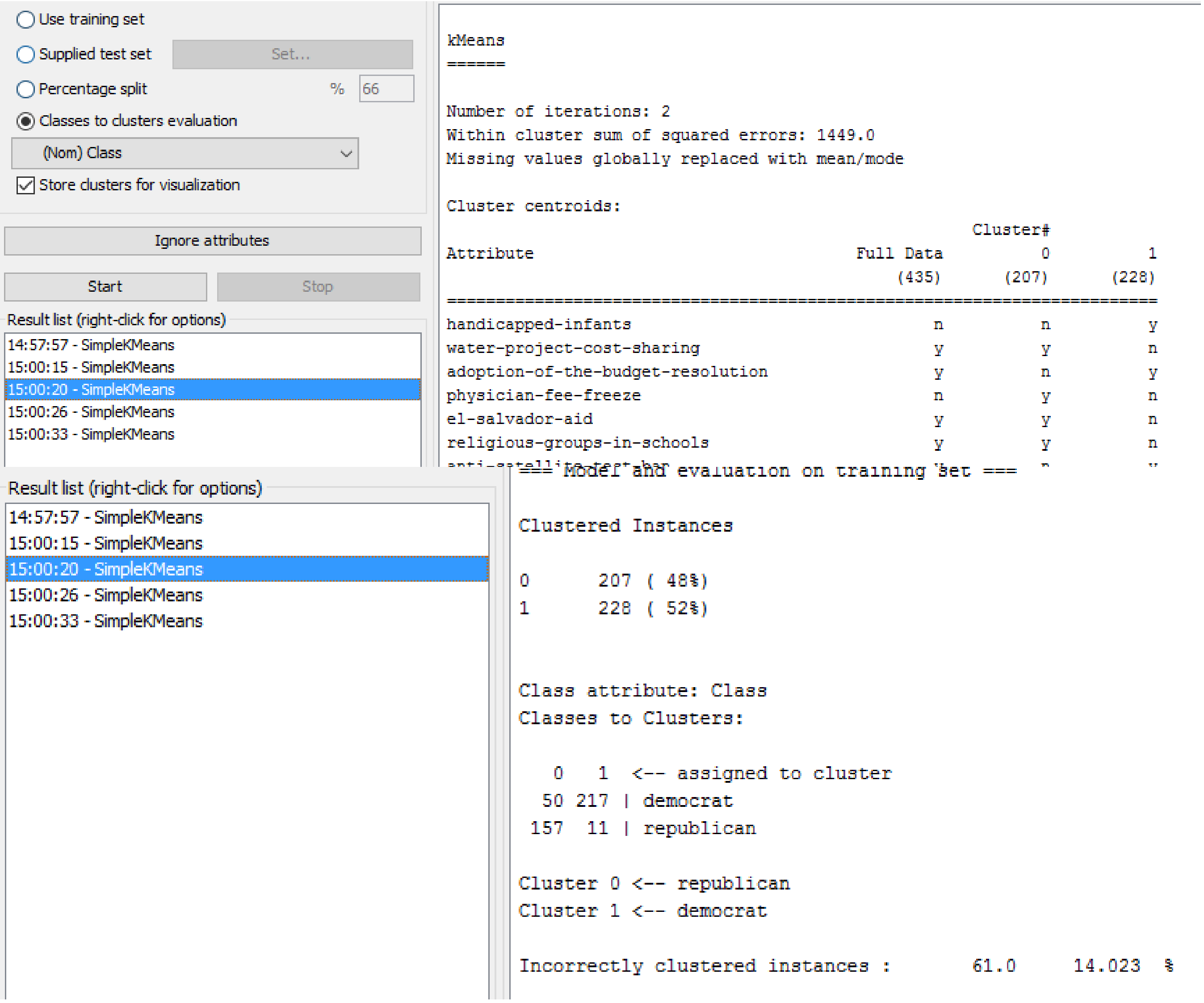
☑ Store clusters for visualization

```
======

Number of iterations: 1
Within cluster sum of squared errors: 2419.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                                  Cluster#
Attribute                              Full Data         0         1
                                           (435)     (221)     (214)
=====================================================================
handicapped-infants                            n         n         y
water-project-cost-sharing                     y         y         n
adoption-of-the-budget-resolution              y         n         y
physician-fee-freeze                           n         y         n
el-salvador-aid                                y         y         n
religious-groups-in-schools                    y         y         n
anti-satellite-test-ban                        y         n         y
```



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

Clustered Instances

0      207 ( 48%)
1      228 ( 52%)


Class attribute: Class
Classes to Clusters:

   0   1  <-- assigned to cluster
  50 217 | democrat
 157  11 | republican

Cluster 0 <-- republican
Cluster 1 <-- democrat

Incorrectly clustered instances :       61.0      14.023  %
```

When Iterations = 2



○ Use training set
○ Supplied test set [Set...]
○ Percentage split % 66
● Classes to clusters evaluation
(Nom) Class
☑ Store clusters for visualization

[Ignore attributes]

[Start] [Stop]

Result list (right-click for options)

14:57:57 - SimpleKMeans
15:00:15 - SimpleKMeans
15:00:20 - SimpleKMeans
15:00:26 - SimpleKMeans
15:00:33 - SimpleKMeans

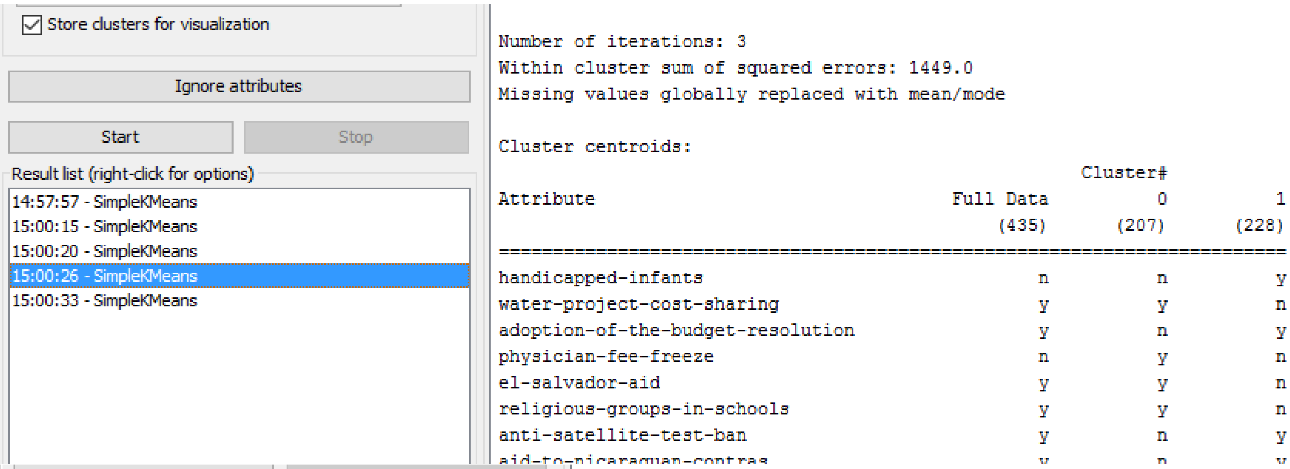
```
kMeans
======

Number of iterations: 2
Within cluster sum of squared errors: 1449.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                              Cluster#
Attribute                          Full Data         0         1
                                       (435)     (207)     (228)
=================================================================
handicapped-infants                        n         n         y
water-project-cost-sharing                 y         y         n
adoption-of-the-budget-resolution          y         n         y
physician-fee-freeze                       n         y         n
el-salvador-aid                            y         y         n
religious-groups-in-schools                y         y         n
anti-satellite-test-ban
```

Result list (right-click for options)

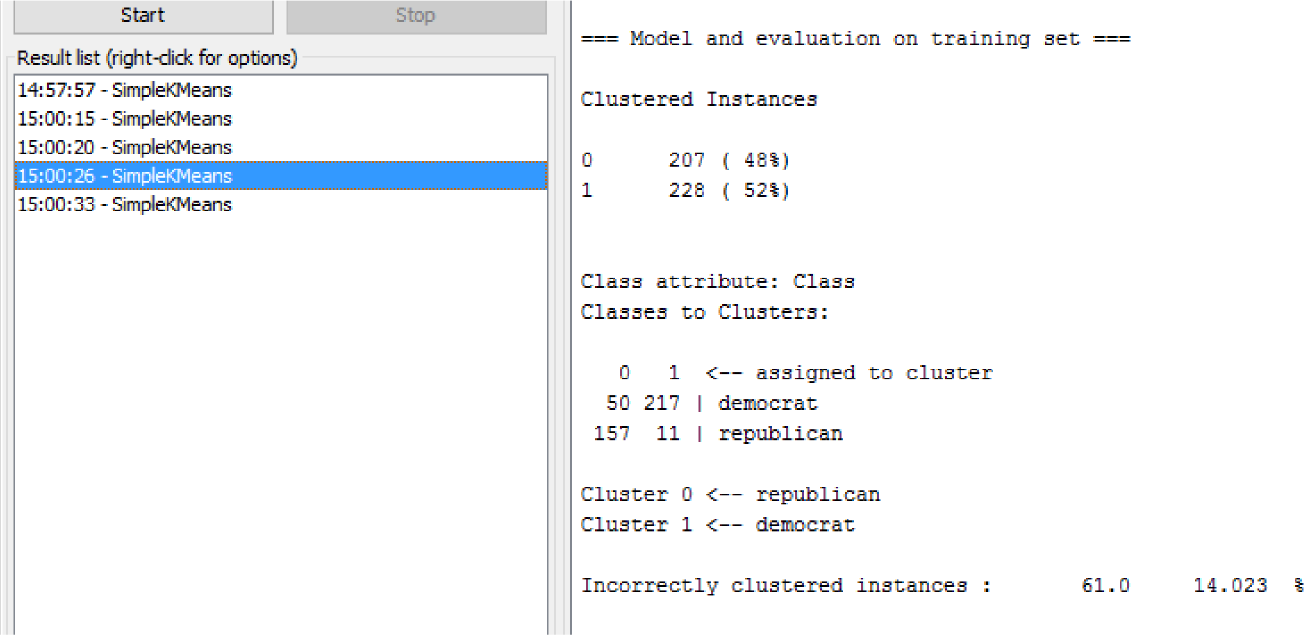
14:57:57 - SimpleKMeans
15:00:15 - SimpleKMeans
15:00:20 - SimpleKMeans
15:00:26 - SimpleKMeans
15:00:33 - SimpleKMeans

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

Clustered Instances

0       207 ( 48%)
1       228 ( 52%)


Class attribute: Class
Classes to Clusters:

   0   1  <-- assigned to cluster
  50 217 | democrat
 157  11 | republican

Cluster 0 <-- republican
Cluster 1 <-- democrat

Incorrectly clustered instances :       61.0     14.023  %
```

When iterations = 3:

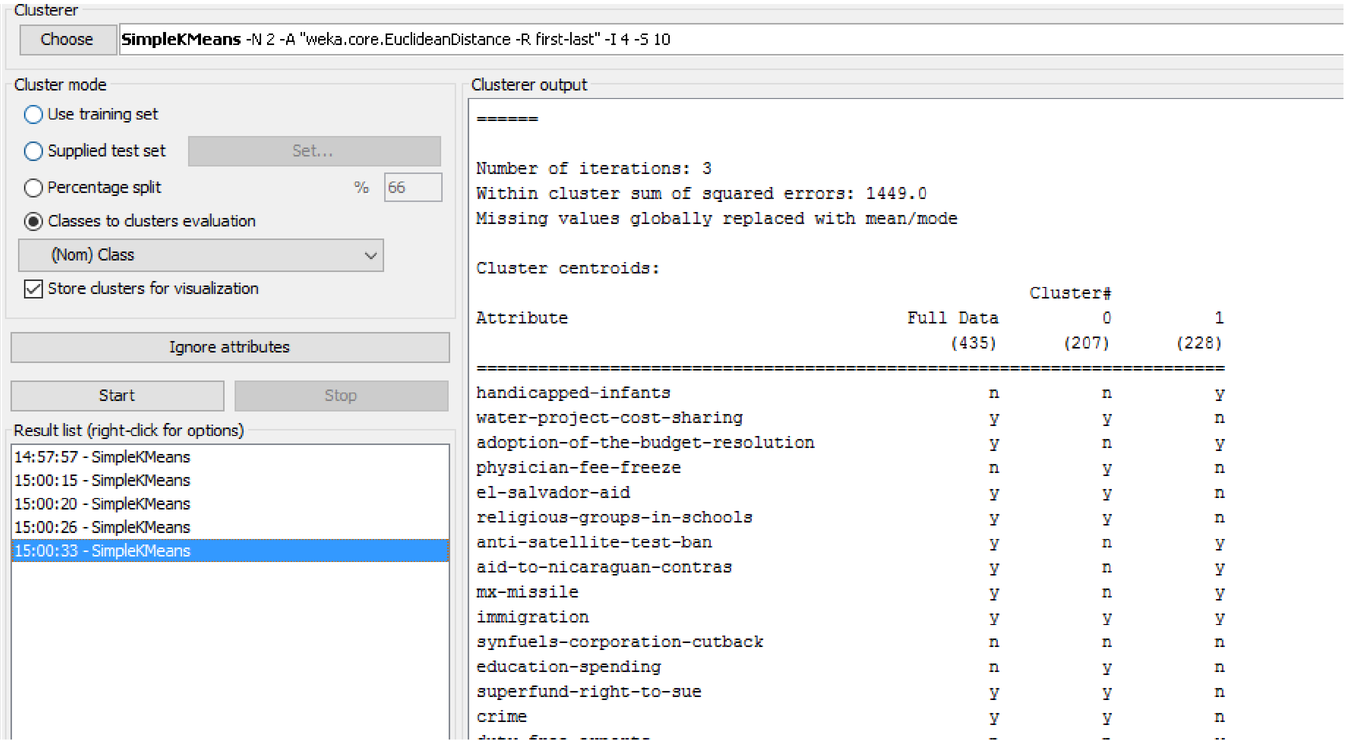


```
Number of iterations: 3
Within cluster sum of squared errors: 1449.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                                Cluster#
Attribute                           Full Data          0          1
                                        (435)      (207)      (228)
===================================================================
handicapped-infants                         n          n          y
water-project-cost-sharing                  y          y          n
adoption-of-the-budget-resolution           y          n          y
physician-fee-freeze                        n          y          n
el-salvador-aid                             y          y          n
religious-groups-in-schools                 y          y          n
anti-satellite-test-ban                     y          n          y
aid-to-nicaraguan-contras                   y          n          y
```



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

Clustered Instances

0       207 ( 48%)
1       228 ( 52%)


Class attribute: Class
Classes to Clusters:

   0   1  <-- assigned to cluster
  50 217 | democrat
 157  11 | republican

Cluster 0 <-- republican
Cluster 1 <-- democrat

Incorrectly clustered instances :       61.0     14.023  %
```

When Iteration = 4:



Clusterer

Choose **SimpleKMeans** -N 2 -A "weka.core.EuclideanDistance -R first-last" -I 4 -S 10

Cluster mode

○ Use training set

○ Supplied test set Set...

○ Percentage split % 66

● Classes to clusters evaluation

(Nom) Class

☑ Store clusters for visualization

Ignore attributes

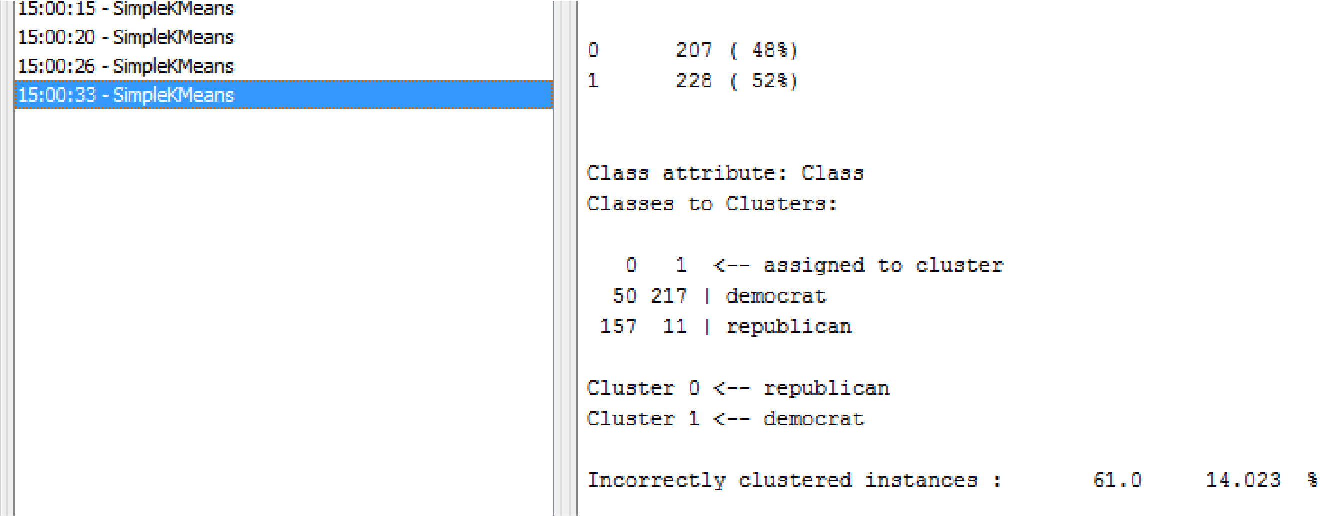
Start Stop

Result list (right-click for options)

14:57:57 - SimpleKMeans
15:00:15 - SimpleKMeans
15:00:20 - SimpleKMeans
15:00:26 - SimpleKMeans
15:00:33 - SimpleKMeans

Clusterer output

```
======

Number of iterations: 3
Within cluster sum of squared errors: 1449.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                              Cluster#
Attribute                          Full Data         0          1
                                       (435)     (207)      (228)
==================================================================
handicapped-infants                        n         n          y
water-project-cost-sharing                 y         y          n
adoption-of-the-budget-resolution          y         n          y
physician-fee-freeze                       n         y          n
el-salvador-aid                            y         y          n
religious-groups-in-schools                y         y          n
anti-satellite-test-ban                    y         n          y
aid-to-nicaraguan-contras                  y         n          y
mx-missile                                 y         n          y
immigration                                y         y          y
synfuels-corporation-cutback               n         n          n
education-spending                         n         y          n
superfund-right-to-sue                     y         y          n
crime                                      y         y          n
```



15:00:15 - SimpleKMeans
15:00:20 - SimpleKMeans
15:00:26 - SimpleKMeans
15:00:33 - SimpleKMeans

```
0       207 ( 48%)
1       228 ( 52%)


Class attribute: Class
Classes to Clusters:

   0   1  <-- assigned to cluster
  50 217 | democrat
 157  11 | republican

Cluster 0 <-- republican
Cluster 1 <-- democrat

Incorrectly clustered instances :       61.0     14.023  %
```

Inference:

- When Number of iterations increases, better SSE is achieved.
- Once the centroid is fixed, in our case when number of iterations is 2, the centroid is fixed. This implies that even if more iterations will be performed, the SSE and number of incorrectly clustered instances remains constant.

- It can also be inferred that as the **Number of Iterations during clustering decreases the SSE and after certain number of iterations the SSE and number of Incorrectly clustered Instances becomes stable.**

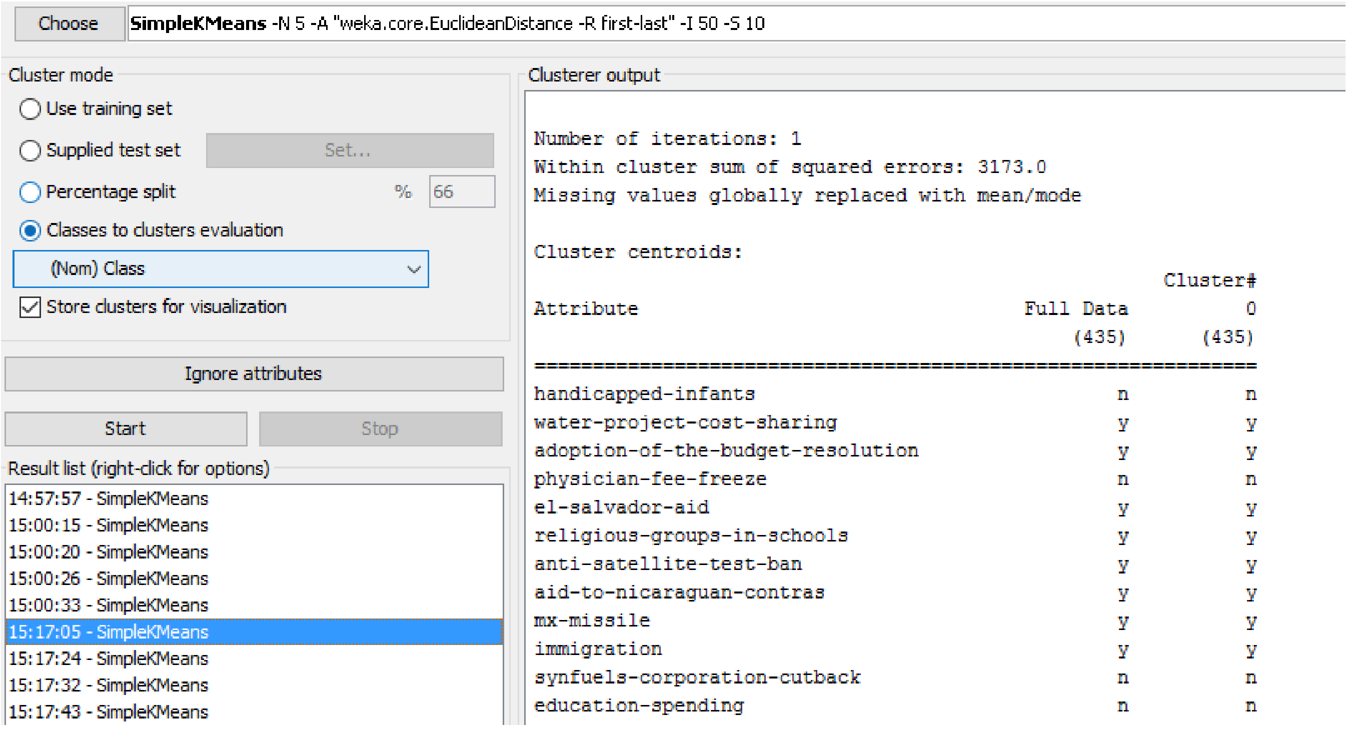
- Hence changing the number of iterations will not result in good clusters because each time it iterates, point is reassigned to the nearest cluster and centroid needs to be re-calculated. Hence computation process is high as the number of iterations increases.

- In our case, 2nd iteration gives the optimal SSE and this run is considered to be the optimal clustering. The clustering done after the 2nd iteration has the same SSE and same number of Incorrectly clustered instances.

- When Iterations = 4 is given, the number of iterations performed is still 3 because it has reached the maximum number of iterations for this particular DS.

Impact of Changing the Number of Clusters parameter in K-means algorithm over Vote Dataset:

When numOf clusters = 1:



SimpleKMeans -N 5 -A "weka.core.EuclideanDistance -R first-last" -I 50 -S 10

Cluster mode

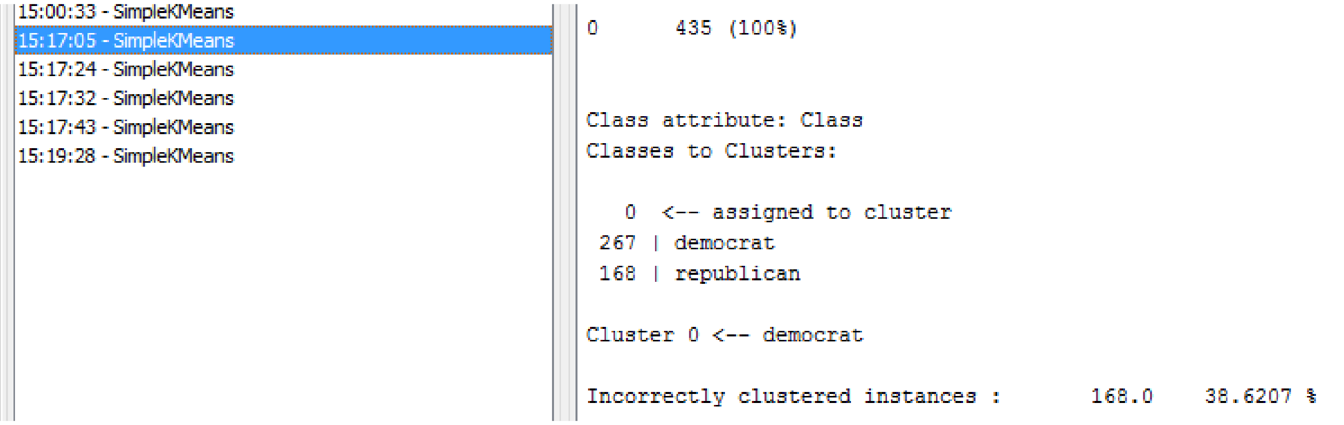
○ Use training set
○ Supplied test set
○ Percentage split % 66
● Classes to clusters evaluation
(Nom) Class
☑ Store clusters for visualization

Result list (right-click for options)

14:57:57 - SimpleKMeans
15:00:15 - SimpleKMeans
15:00:20 - SimpleKMeans
15:00:26 - SimpleKMeans
15:00:33 - SimpleKMeans
15:17:05 - SimpleKMeans
15:17:24 - SimpleKMeans
15:17:32 - SimpleKMeans
15:17:43 - SimpleKMeans

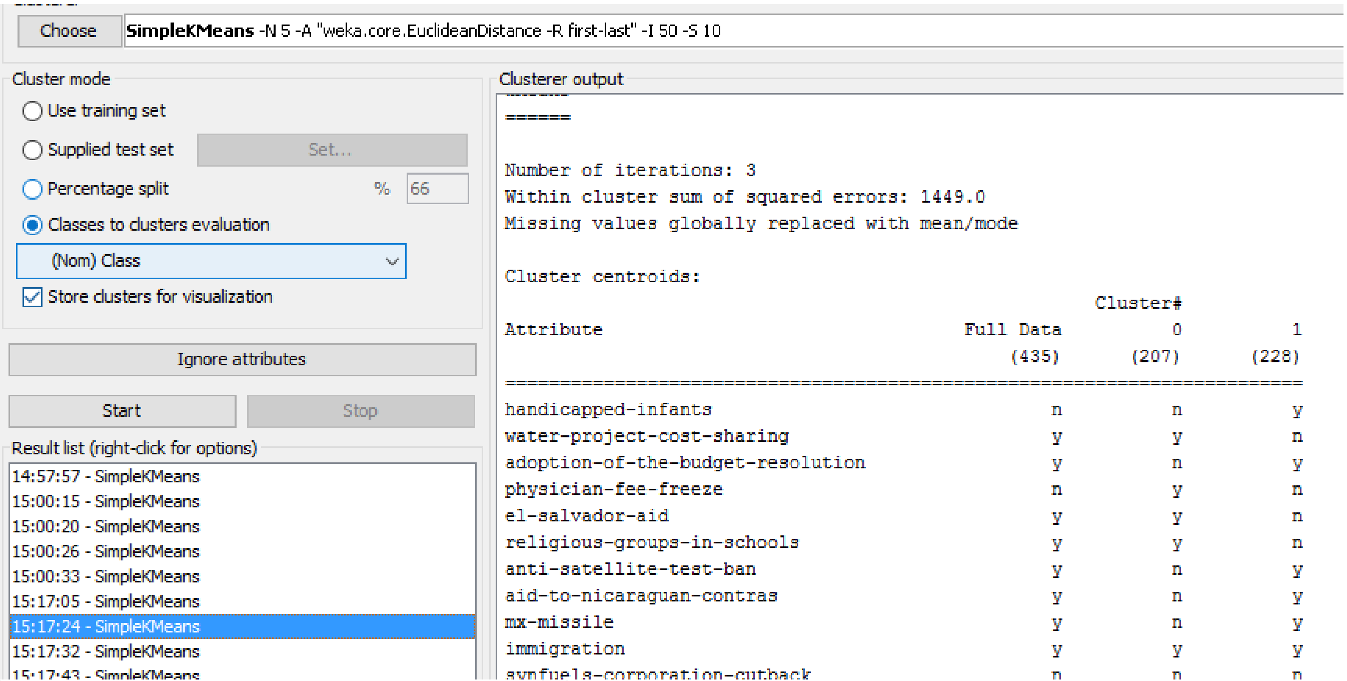
Clusterer output

```
Number of iterations: 1
Within cluster sum of squared errors: 3173.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                                    Cluster#
Attribute                                Full Data         0
                                             (435)     (435)
==============================================================
handicapped-infants                              n         n
water-project-cost-sharing                       y         y
adoption-of-the-budget-resolution                y         y
physician-fee-freeze                             n         n
el-salvador-aid                                  y         y
religious-groups-in-schools                      y         y
anti-satellite-test-ban                          y         y
aid-to-nicaraguan-contras                        y         y
mx-missile                                       y         y
immigration                                      y         y
synfuels-corporation-cutback                     n         n
education-spending                               n         n
```



```
0       435 (100%)


Class attribute: Class
Classes to Clusters:

   0  <-- assigned to cluster
 267 | democrat
 168 | republican

Cluster 0 <-- democrat

Incorrectly clustered instances :       168.0    38.6207 %
```

When number of clusters = 2



SimpleKMeans -N 5 -A "weka.core.EuclideanDistance -R first-last" -I 50 -S 10

Cluster mode

○ Use training set

○ Supplied test set

○ Percentage split % 66

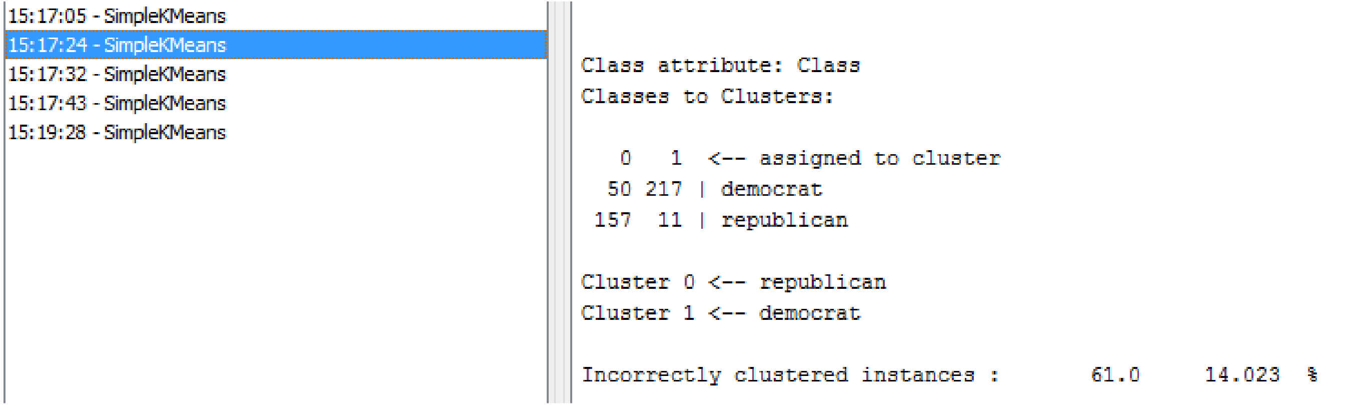
● Classes to clusters evaluation

(Nom) Class

☑ Store clusters for visualization

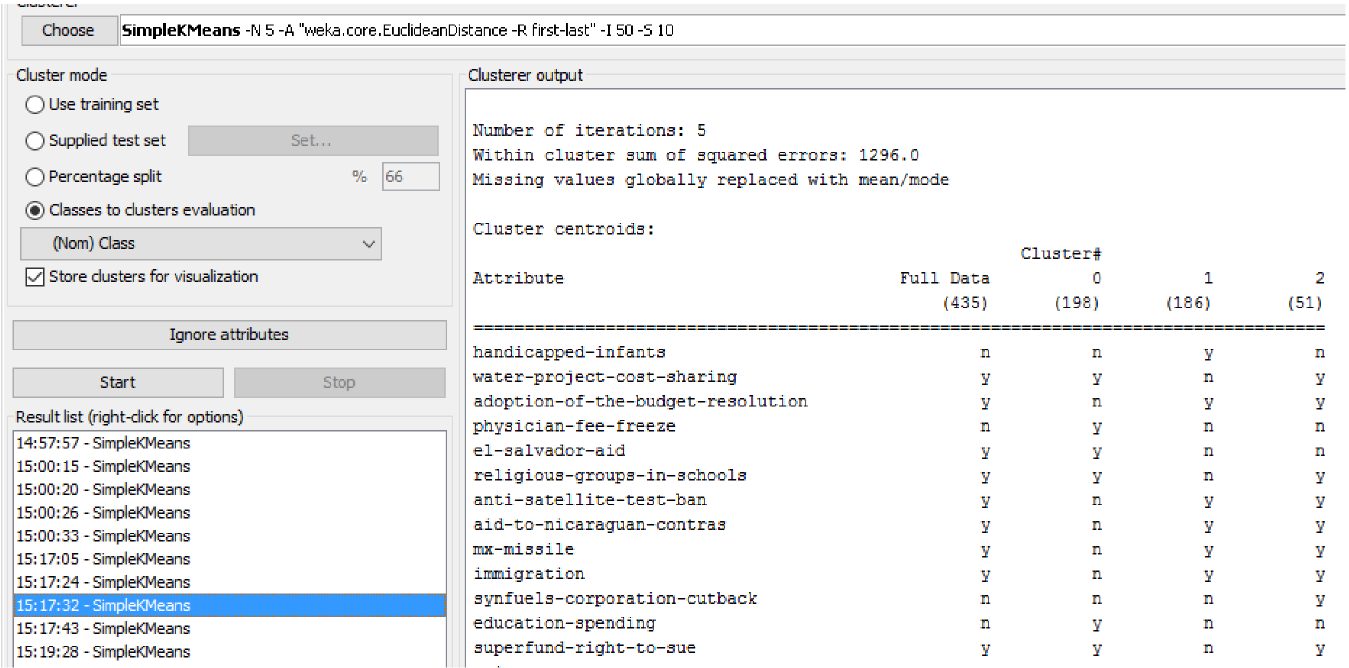
Clusterer output

```
======

Number of iterations: 3
Within cluster sum of squared errors: 1449.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                                   Cluster#
Attribute                               Full Data         0          1
                                            (435)     (207)      (228)
========================================================================
handicapped-infants                             n         n          y
water-project-cost-sharing                      y         y          n
adoption-of-the-budget-resolution               y         n          y
physician-fee-freeze                            n         y          n
el-salvador-aid                                 y         y          n
religious-groups-in-schools                     y         y          n
anti-satellite-test-ban                         y         n          y
aid-to-nicaraguan-contras                       y         n          y
mx-missile                                      y         n          y
immigration                                     y         y          y
synfuels-corporation-cutback                    n         n          n
```



```
Class attribute: Class
Classes to Clusters:

   0   1  <-- assigned to cluster
  50 217 | democrat
 157  11 | republican

Cluster 0 <-- republican
Cluster 1 <-- democrat

Incorrectly clustered instances :       61.0      14.023  %
```

When Number of clusters = 3



SimpleKMeans -N 5 -A "weka.core.EuclideanDistance -R first-last" -I 50 -S 10

Cluster mode

○ Use training set

○ Supplied test set

○ Percentage split % 66

● Classes to clusters evaluation

(Nom) Class

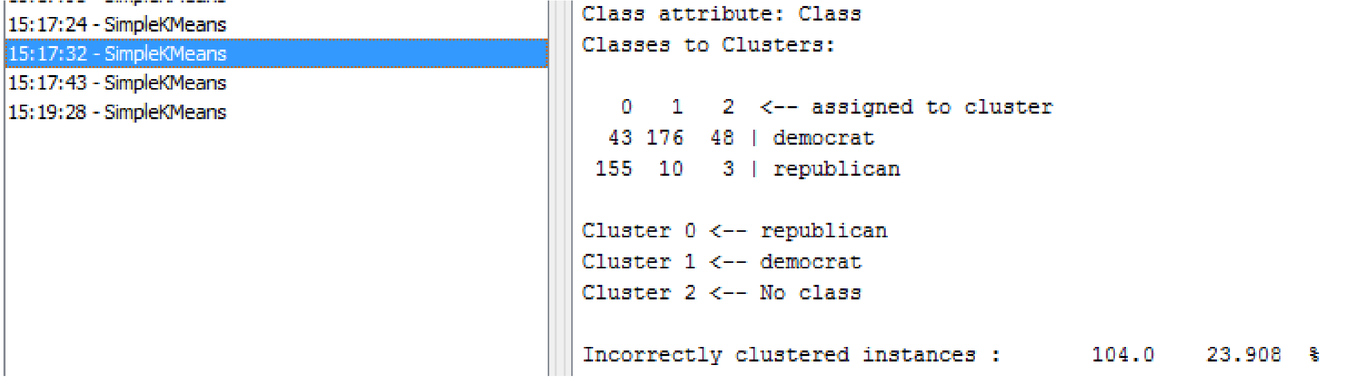
☑ Store clusters for visualization

Result list (right-click for options)

```
14:57:57 - SimpleKMeans
15:00:15 - SimpleKMeans
15:00:20 - SimpleKMeans
15:00:26 - SimpleKMeans
15:00:33 - SimpleKMeans
15:17:05 - SimpleKMeans
15:17:24 - SimpleKMeans
15:17:32 - SimpleKMeans
15:17:43 - SimpleKMeans
15:19:28 - SimpleKMeans
```

Clusterer output

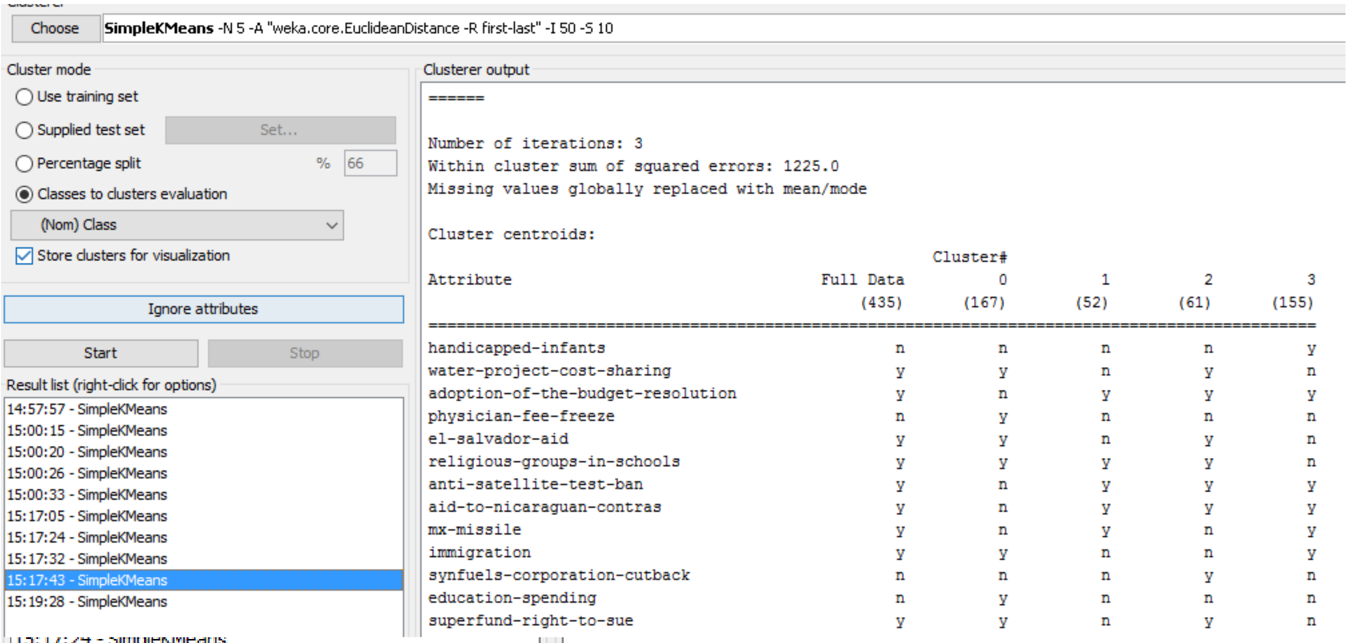
```
Number of iterations: 5
Within cluster sum of squared errors: 1296.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                                  Cluster#
Attribute                            Full Data          0          1          2
                                         (435)      (198)      (186)       (51)
================================================================================
handicapped-infants                          n          n          y          n
water-project-cost-sharing                   y          y          n          y
adoption-of-the-budget-resolution            y          n          y          y
physician-fee-freeze                         n          y          n          n
el-salvador-aid                              y          y          n          n
religious-groups-in-schools                  y          y          n          y
anti-satellite-test-ban                      y          n          y          y
aid-to-nicaraguan-contras                    y          n          y          y
mx-missile                                   y          n          y          y
immigration                                  y          n          y          y
synfuels-corporation-cutback                 n          n          n          y
education-spending                           n          y          n          n
superfund-right-to-sue                       y          y          n          y
```



```
15:17:24 - SimpleKMeans
15:17:32 - SimpleKMeans
15:17:43 - SimpleKMeans
15:19:28 - SimpleKMeans
```

```
Class attribute: Class
Classes to Clusters:

   0   1   2  <-- assigned to cluster
  43 176  48 | democrat
 155  10   3 | republican

Cluster 0 <-- republican
Cluster 1 <-- democrat
Cluster 2 <-- No class

Incorrectly clustered instances :       104.0     23.908  %
```

When number of clusters = 4



Choose | SimpleKMeans -N 5 -A "weka.core.EuclideanDistance -R first-last" -I 50 -S 10

Cluster mode

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

Ignore attributes

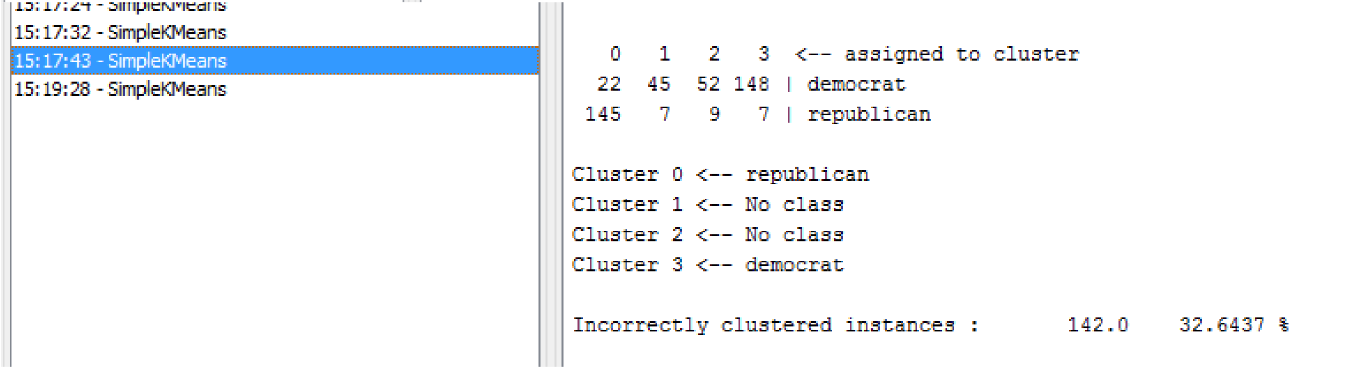
Start Stop

Result list (right-click for options)

14:57:57 - SimpleKMeans
15:00:15 - SimpleKMeans
15:00:20 - SimpleKMeans
15:00:26 - SimpleKMeans
15:00:33 - SimpleKMeans
15:17:05 - SimpleKMeans
15:17:24 - SimpleKMeans
15:17:32 - SimpleKMeans
15:17:43 - SimpleKMeans
15:19:28 - SimpleKMeans

Clusterer output

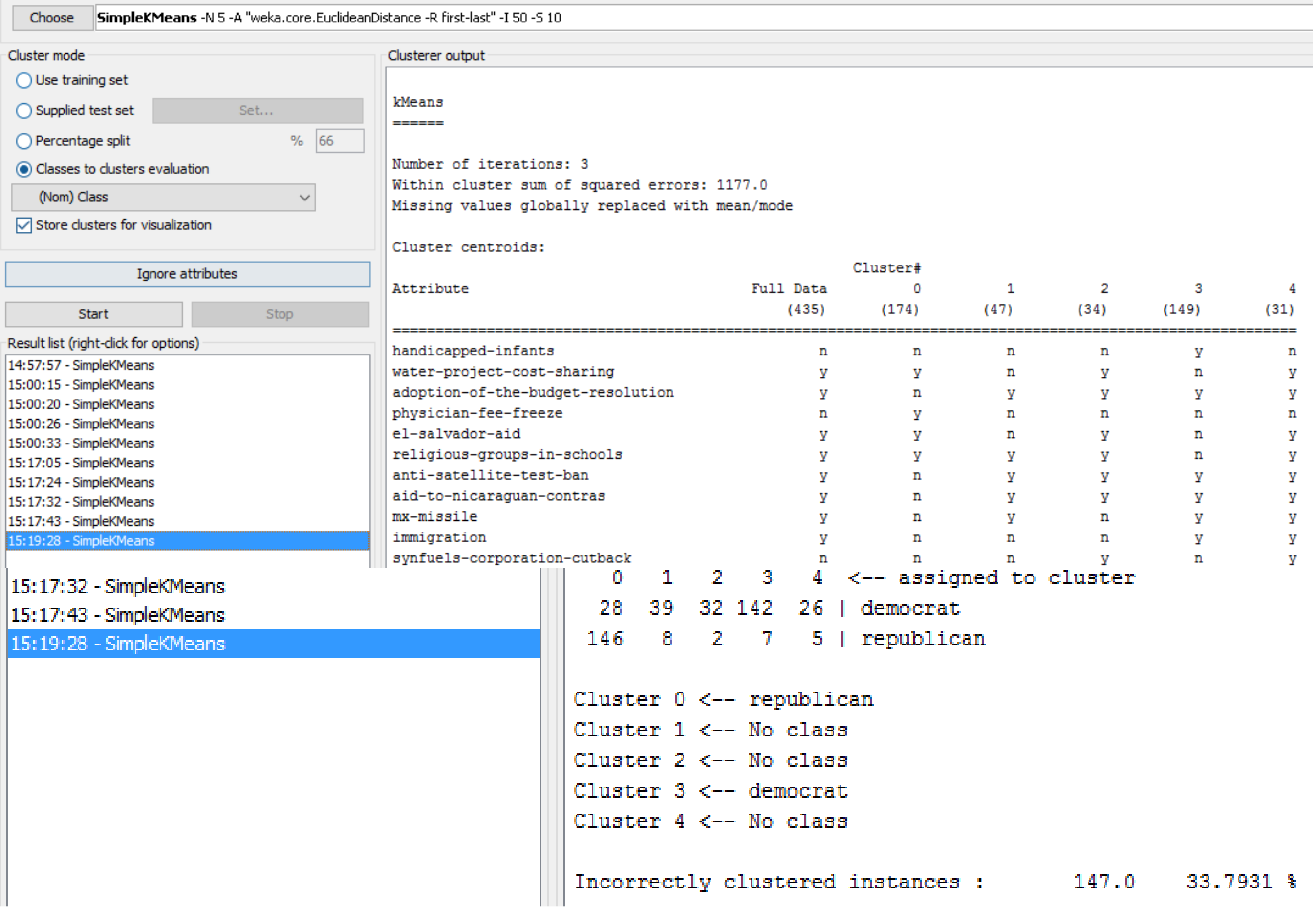
```
======

Number of iterations: 3
Within cluster sum of squared errors: 1225.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                                  Cluster#
Attribute                             Full Data          0          1          2          3
                                          (435)      (167)       (52)       (61)      (155)
===========================================================================================
handicapped-infants                           n          n          n          n          y
water-project-cost-sharing                    y          y          n          y          n
adoption-of-the-budget-resolution             y          n          y          y          y
physician-fee-freeze                          n          y          n          n          n
el-salvador-aid                               y          y          n          y          n
religious-groups-in-schools                   y          y          y          y          n
anti-satellite-test-ban                       y          n          y          y          y
aid-to-nicaraguan-contras                     y          n          y          y          y
mx-missile                                    y          n          y          n          y
immigration                                   y          y          n          n          y
synfuels-corporation-cutback                  n          n          n          y          n
education-spending                            n          y          n          n          n
superfund-right-to-sue                        y          y          n          y          n
```



15:17:24 - SimpleKMeans
15:17:32 - SimpleKMeans
15:17:43 - SimpleKMeans
15:19:28 - SimpleKMeans

```
  0   1   2   3  <-- assigned to cluster
 22  45  52 148 | democrat
145   7   9   7 | republican

Cluster 0 <-- republican
Cluster 1 <-- No class
Cluster 2 <-- No class
Cluster 3 <-- democrat

Incorrectly clustered instances :       142.0    32.6437 %
```

When Number of iterations = 5:



SimpleKMeans -N 5 -A "weka.core.EuclideanDistance -R first-last" -I 50 -S 10

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)

14:57:57 - SimpleKMeans
15:00:15 - SimpleKMeans
15:00:20 - SimpleKMeans
15:00:26 - SimpleKMeans
15:00:33 - SimpleKMeans
15:17:05 - SimpleKMeans
15:17:24 - SimpleKMeans
15:17:32 - SimpleKMeans
15:17:43 - SimpleKMeans
15:19:28 - SimpleKMeans

Clusterer output

```
kMeans
======

Number of iterations: 3
Within cluster sum of squared errors: 1177.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                              Cluster#
Attribute                          Full Data         0          1          2          3          4
                                       (435)     (174)       (47)       (34)      (149)       (31)
==================================================================================================
handicapped-infants                        n         n          n          n          y          n
water-project-cost-sharing                 y         y          n          y          n          y
adoption-of-the-budget-resolution          y         n          y          y          y          y
physician-fee-freeze                       n         y          n          n          n          n
el-salvador-aid                            y         y          n          y          n          y
religious-groups-in-schools                y         y          y          y          n          y
anti-satellite-test-ban                    y         n          y          y          y          y
aid-to-nicaraguan-contras                  y         n          y          y          y          y
mx-missile                                 y         n          y          n          y          y
immigration                                y         n          n          n          y          y
synfuels-corporation-cutback               n         n          n          y          n          y
```

15:17:32 - SimpleKMeans
15:17:43 - SimpleKMeans
15:19:28 - SimpleKMeans

```
  0   1   2   3   4  <-- assigned to cluster
 28  39  32 142  26 | democrat
146   8   2   7   5 | republican

Cluster 0 <-- republican
Cluster 1 <-- No class
Cluster 2 <-- No class
Cluster 3 <-- democrat
Cluster 4 <-- No class

Incorrectly clustered instances :	147.0	 33.7931 %
```

The below tabular column is derived from the above screenshots.

Maximum NO. of Iterations in clustering (default)	Actual No. of Iterations during clustering	Number of Clusters (Constant)	SSE	Number of Incorrectly Clustered Instances
50	1	1	3173	168
50	3	2	1449	61
50	5	3	1296	104
50	3	4	1225	142
50	3	5	1177	147

Inferences:

- It can inferred that when number of clusters increases, it results in better SSE.

- But at the same time, optimum number of clusters should be taken into account along with number of incorrectly clustered instances.

- In the above run, the maximum number of Iterations in clustering is set to a default value =50, but the k-means algorithm would end up with even small number of iterations .It is due to the size of the dataset.

- Number of clusters increases and SSE decreases. This is because the points in the dataset are placed to more closest clusters. Implies point moves to their proximity cluster centroid and which in turn causes SSE of each point to decrease.

- As the number of cluster increases, the Incorrectly clustered instances initially decreases , when the number of cluster further increases it leads to outliers and missing data , these values fall into those clusters and there by the Incorrectly clustered instances increases.

It can be inferred that optimistic clustering is achieved in the 2nd run with 2 clusters as they have less SSE when compared to the other runs.

Impact on including only the specific attributes for Clustering:

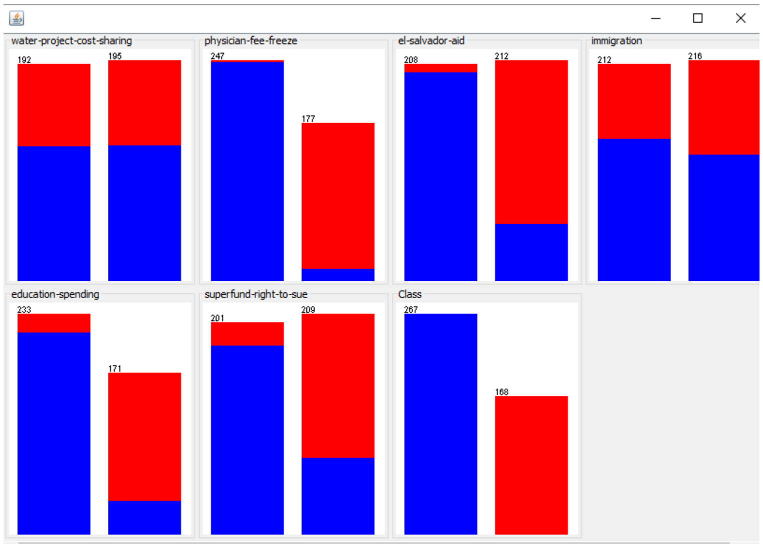
Attributes selected for Clustering	SSE	Number of Incorrectly Clustered Instances
Water-project cost sharing, immigration	191	206
education, spending superfund right to use	111	112
Physician fee freeze ,el salvor aid	66	71

The above table shows the execution of the k-means clustering using only some of the attributes / dimensions in the dataset. The global optimum function SSE (Sum of Squared Error) and Number of Incorrectly clustered instances is used to determine/predict the optimistic clustering.

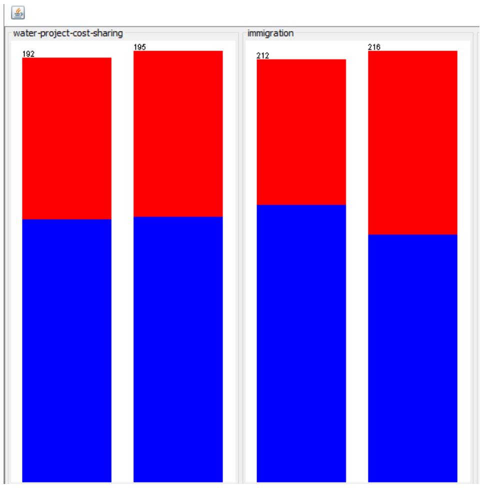
Inferences:

- Always selecting some specific attributes will result in a good clustering with minimum SSE and minimum number of Incorrectly clustered instances.

- The attributes decides a good cluster.



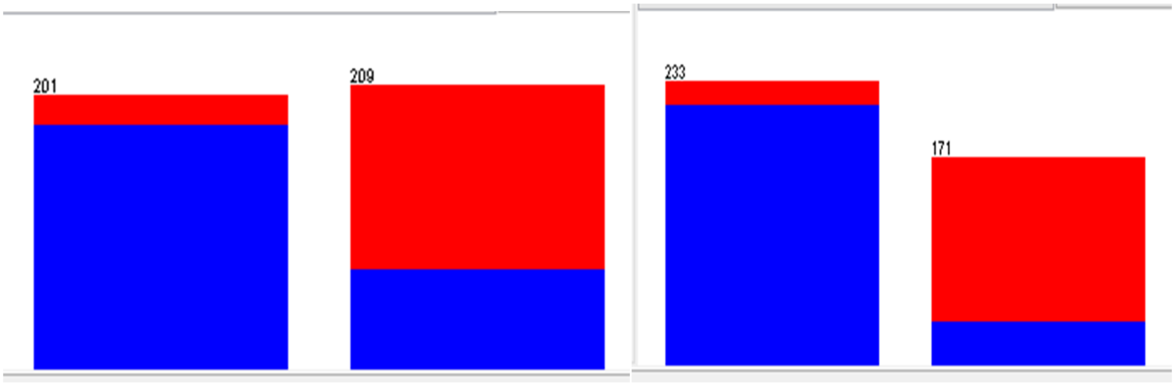
Cluster using attributes Water-project cost sharing, immigration:



The above figure shows the clusters formed using the attributes Water-project cost sharing and immigration. It doesn't represent the both the classes and the distribution is not uniform hence the SSE is very high. The data points in the cluster are not uniform and distorted. Hence the total SSE is larger.

Cluster using attributes education, spending superfund right to use:

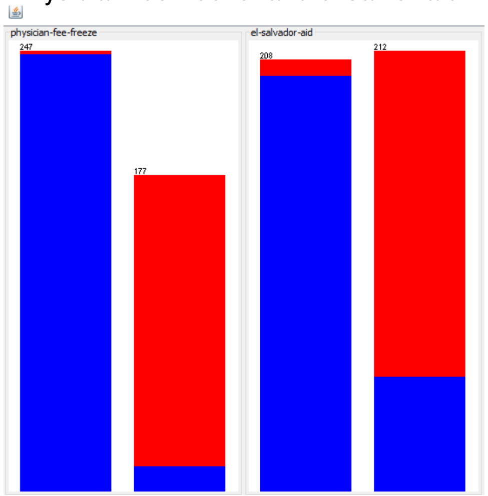
Superfund right to use Education - Spending



This figure shows the distribution of the clusters in education, spending superfund right to use attributes , here the distribution is somewhat close to uniform, hence the clustering has less SSE when compared to previous run.

Cluster using attributes Physician fee freeze , el salvor aid:

Physician fee freeze and el salvor aid

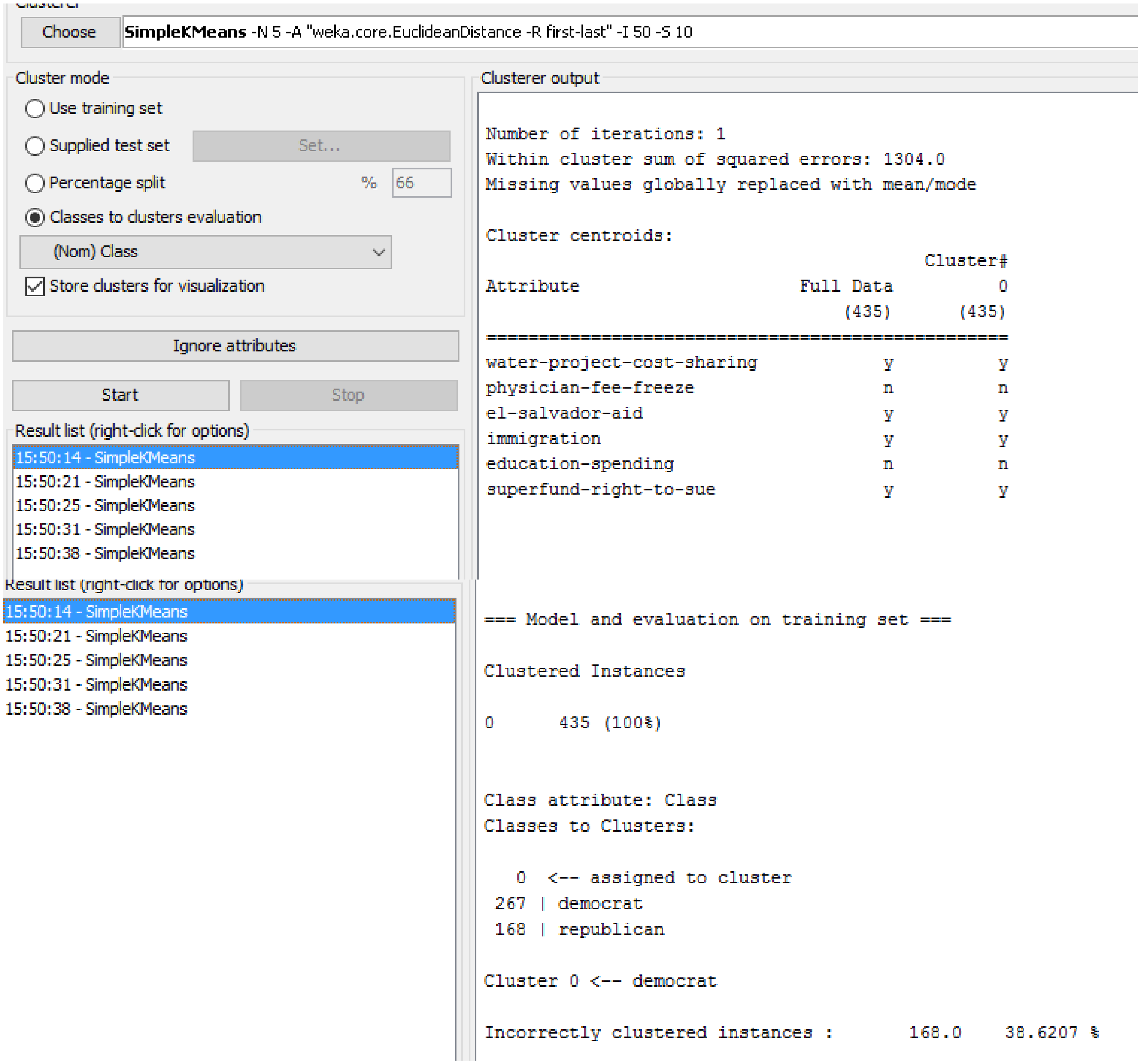


Inference:

In this figure the distribution of the instances is uniform. Hence the attributes Physician fee freeze ,el salvor aid yields to the optimistic SSE and incorrectly clustered instances.

After changing the parameters – Number of clusters:

Number of clusters = 1



Choose **SimpleKMeans** -N 5 -A "weka.core.EuclideanDistance -R first-last" -I 50 -S 10

Cluster mode

○ Use training set

○ Supplied test set Set...

○ Percentage split % 66

● Classes to clusters evaluation

(Nom) Class

☑ Store clusters for visualization

Result list (right-click for options)

15:50:14 - SimpleKMeans
15:50:21 - SimpleKMeans
15:50:25 - SimpleKMeans
15:50:31 - SimpleKMeans
15:50:38 - SimpleKMeans

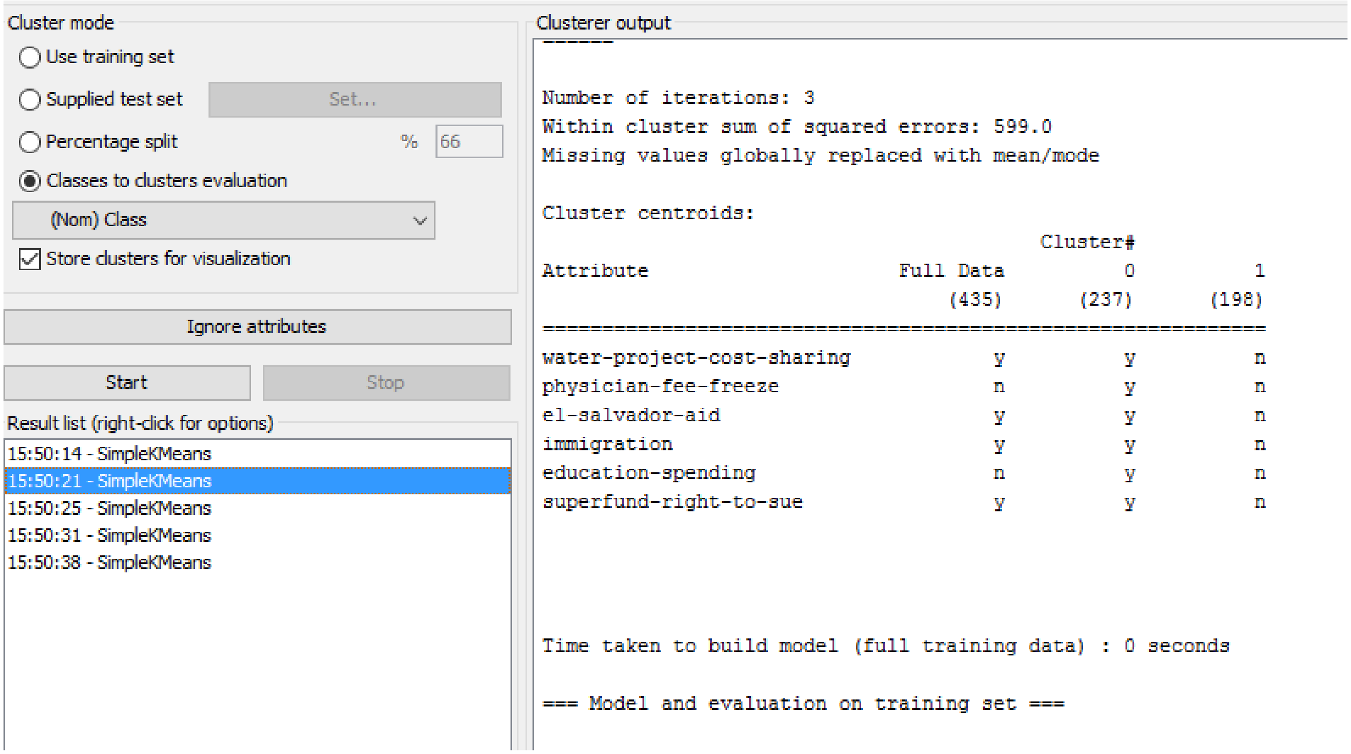
Result list (right-click for options)

15:50:14 - SimpleKMeans
15:50:21 - SimpleKMeans
15:50:25 - SimpleKMeans
15:50:31 - SimpleKMeans
15:50:38 - SimpleKMeans

Clusterer output

```
Number of iterations: 1
Within cluster sum of squared errors: 1304.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                        Cluster#
Attribute                 Full Data            0
                              (435)        (435)
==================================================
water-project-cost-sharing        y            y
physician-fee-freeze              n            n
el-salvador-aid                   y            y
immigration                       y            y
education-spending                n            n
superfund-right-to-sue            y            y




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

Clustered Instances

0       435 (100%)


Class attribute: Class
Classes to Clusters:

   0  <-- assigned to cluster
 267 | democrat
 168 | republican

Cluster 0 <-- democrat

Incorrectly clustered instances :       168.0    38.6207 %
```

When Number of clusters = 2



Cluster mode

○ Use training set

○ Supplied test set

○ Percentage split % 66

● Classes to clusters evaluation

(Nom) Class

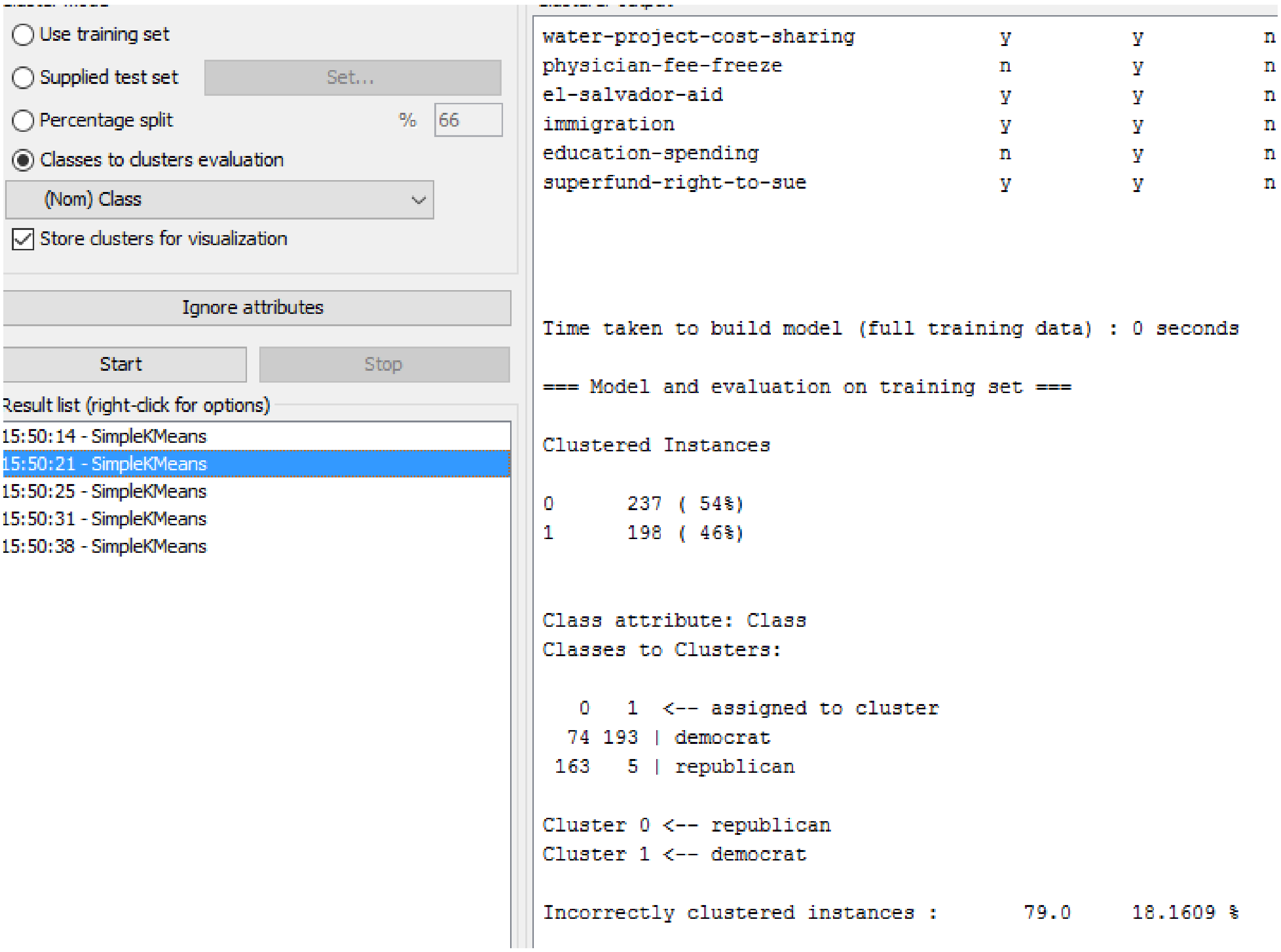
☑ Store clusters for visualization

Result list (right-click for options)

15:50:14 - SimpleKMeans
15:50:21 - SimpleKMeans
15:50:25 - SimpleKMeans
15:50:31 - SimpleKMeans
15:50:38 - SimpleKMeans

Clusterer output

```
======

Number of iterations: 3
Within cluster sum of squared errors: 599.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                         Cluster#
Attribute                    Full Data          0          1
                                 (435)      (237)      (198)
=============================================================
water-project-cost-sharing           y          y          n
physician-fee-freeze                 n          y          n
el-salvador-aid                      y          y          n
immigration                          y          y          n
education-spending                   n          y          n
superfund-right-to-sue               y          y          n




Time taken to build model (full training data) : 0 seconds

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

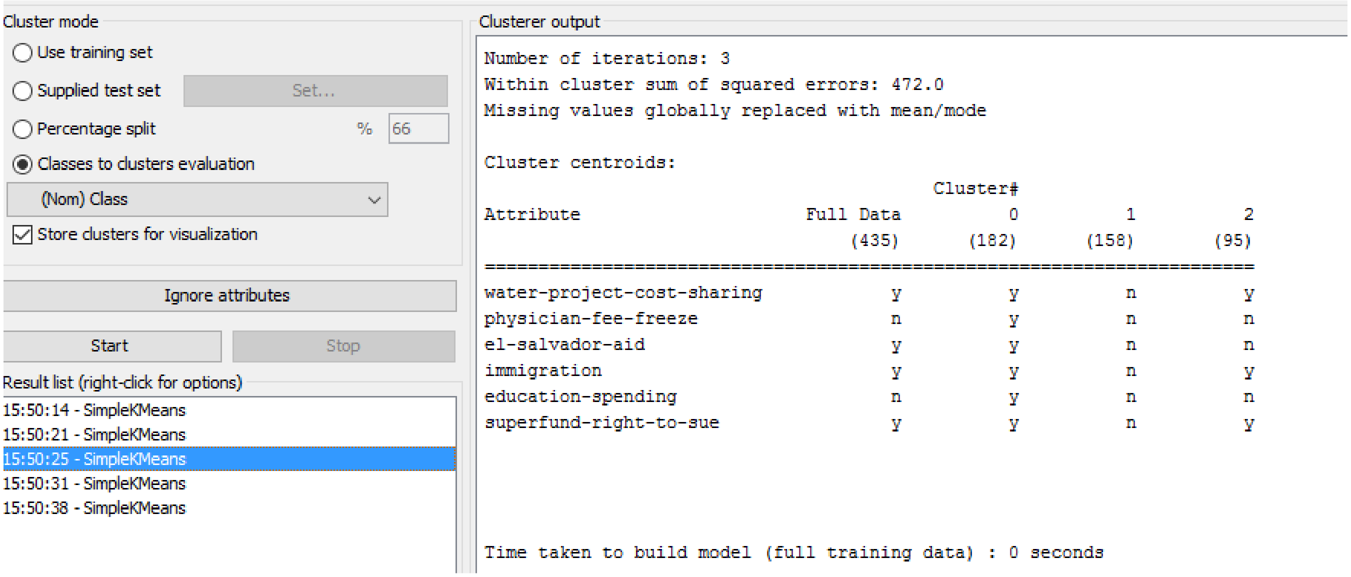


○ Use training set
○ Supplied test set
○ Percentage split % 66
● Classes to clusters evaluation
(Nom) Class
☑ Store clusters for visualization

Result list (right-click for options)
15:50:14 - SimpleKMeans
15:50:21 - SimpleKMeans
15:50:25 - SimpleKMeans
15:50:31 - SimpleKMeans
15:50:38 - SimpleKMeans

```
water-project-cost-sharing            y          y          n
physician-fee-freeze                  n          y          n
el-salvador-aid                       y          y          n
immigration                           y          y          n
education-spending                    n          y          n
superfund-right-to-sue                y          y          n




Time taken to build model (full training data) : 0 seconds

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

Clustered Instances

0      237 ( 54%)
1      198 ( 46%)


Class attribute: Class
Classes to Clusters:

   0   1  <-- assigned to cluster
  74 193 | democrat
 163   5 | republican

Cluster 0 <-- republican
Cluster 1 <-- democrat

Incorrectly clustered instances :      79.0     18.1609 %
```

When Number of clusters = 3

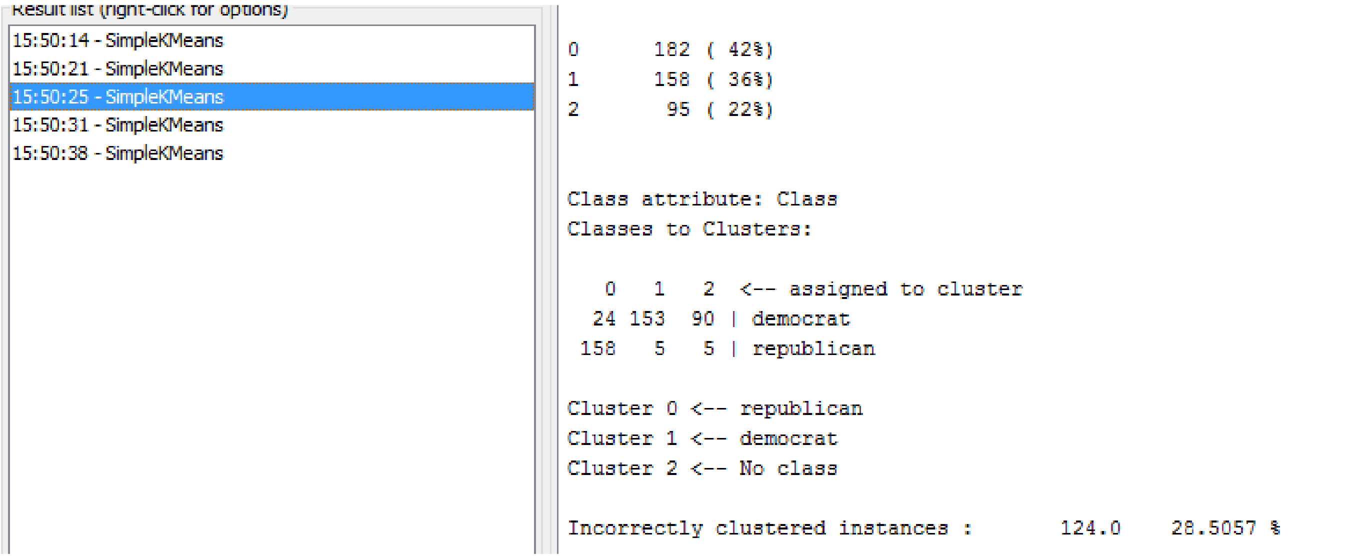


Cluster mode
○ Use training set
○ Supplied test set
○ Percentage split % 66
● Classes to clusters evaluation
(Nom) Class
☑ Store clusters for visualization

Result list (right-click for options)
15:50:14 - SimpleKMeans
15:50:21 - SimpleKMeans
15:50:25 - SimpleKMeans
15:50:31 - SimpleKMeans
15:50:38 - SimpleKMeans

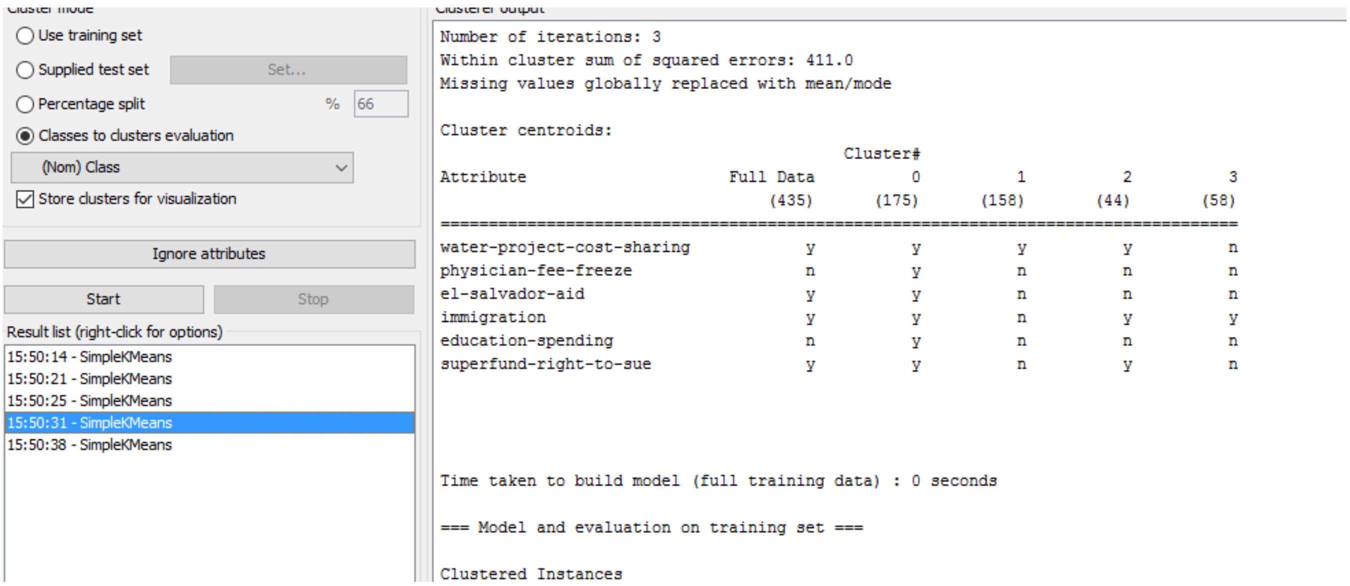
Clusterer output

```
Number of iterations: 3
Within cluster sum of squared errors: 472.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                         Cluster#
Attribute                   Full Data          0          1          2
                                (435)      (182)      (158)       (95)
========================================================================
water-project-cost-sharing          y          y          n          y
physician-fee-freeze                n          y          n          n
el-salvador-aid                     y          y          n          n
immigration                         y          y          n          y
education-spending                  n          y          n          n
superfund-right-to-sue              y          y          n          y




Time taken to build model (full training data) : 0 seconds
```



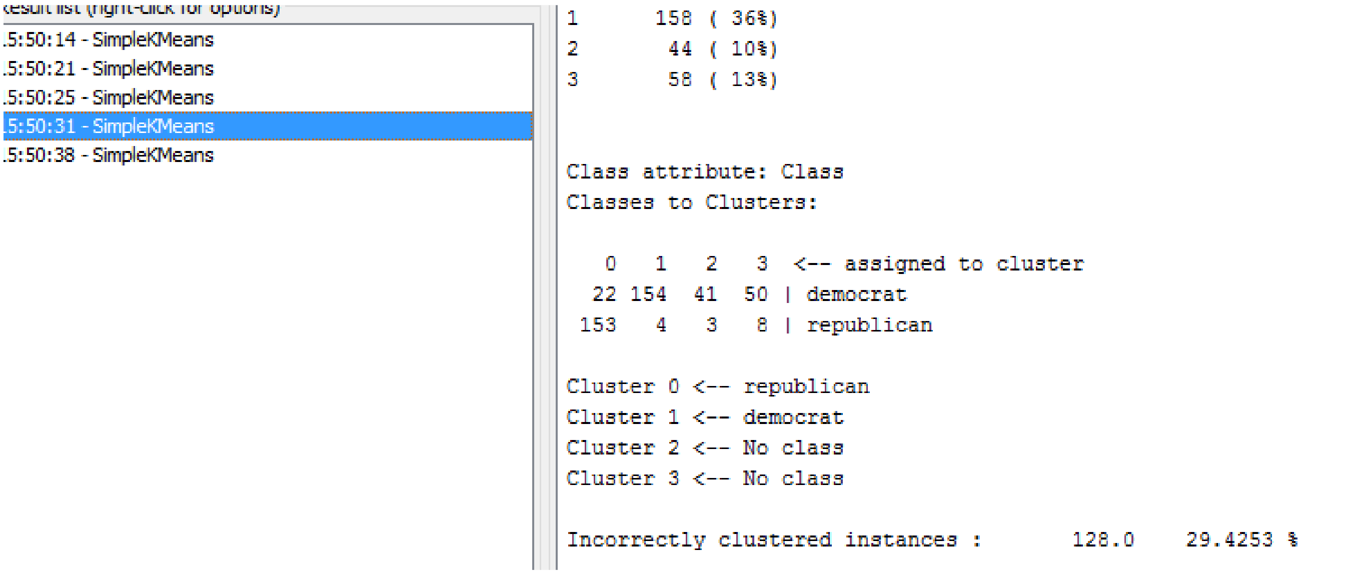
```
0       182 ( 42%)
1       158 ( 36%)
2        95 ( 22%)


Class attribute: Class
Classes to Clusters:

   0   1   2  <-- assigned to cluster
  24 153  90 | democrat
 158   5   5 | republican

Cluster 0 <-- republican
Cluster 1 <-- democrat
Cluster 2 <-- No class

Incorrectly clustered instances :       124.0    28.5057 %
```

When number of clusters = 4



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

```
Number of iterations: 3
Within cluster sum of squared errors: 411.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                       Cluster#
Attribute                  Full Data          0          1          2          3
                               (435)      (175)      (158)       (44)       (58)
==================================================================================
water-project-cost-sharing         y          y          y          y          n
physician-fee-freeze               n          y          n          n          n
el-salvador-aid                    y          y          n          n          n
immigration                        y          y          n          y          y
education-spending                 n          y          n          n          n
superfund-right-to-sue             y          y          n          y          n




Time taken to build model (full training data) : 0 seconds

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

Clustered Instances
```

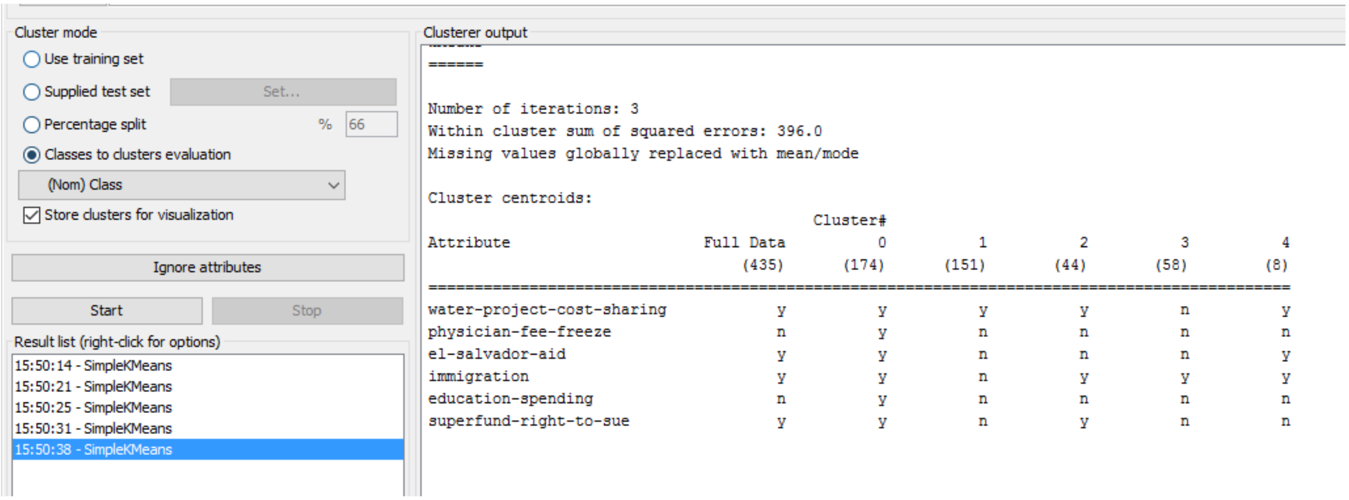


Result list (right-click for options)

```
.5:50:14 - SimpleKMeans
.5:50:21 - SimpleKMeans
.5:50:25 - SimpleKMeans
.5:50:31 - SimpleKMeans
.5:50:38 - SimpleKMeans
```

```
1       158 ( 36%)
2        44 ( 10%)
3        58 ( 13%)


Class attribute: Class
Classes to Clusters:

   0   1   2   3  <-- assigned to cluster
  22 154  41  50 | democrat
 153   4   3   8 | republican

Cluster 0 <-- republican
Cluster 1 <-- democrat
Cluster 2 <-- No class
Cluster 3 <-- No class

Incorrectly clustered instances :       128.0    29.4253 %
```

When number of clusters = 5



Cluster mode

○ Use training set

○ Supplied test set [Set...]

○ Percentage split % 66

● Classes to clusters evaluation

(Nom) Class

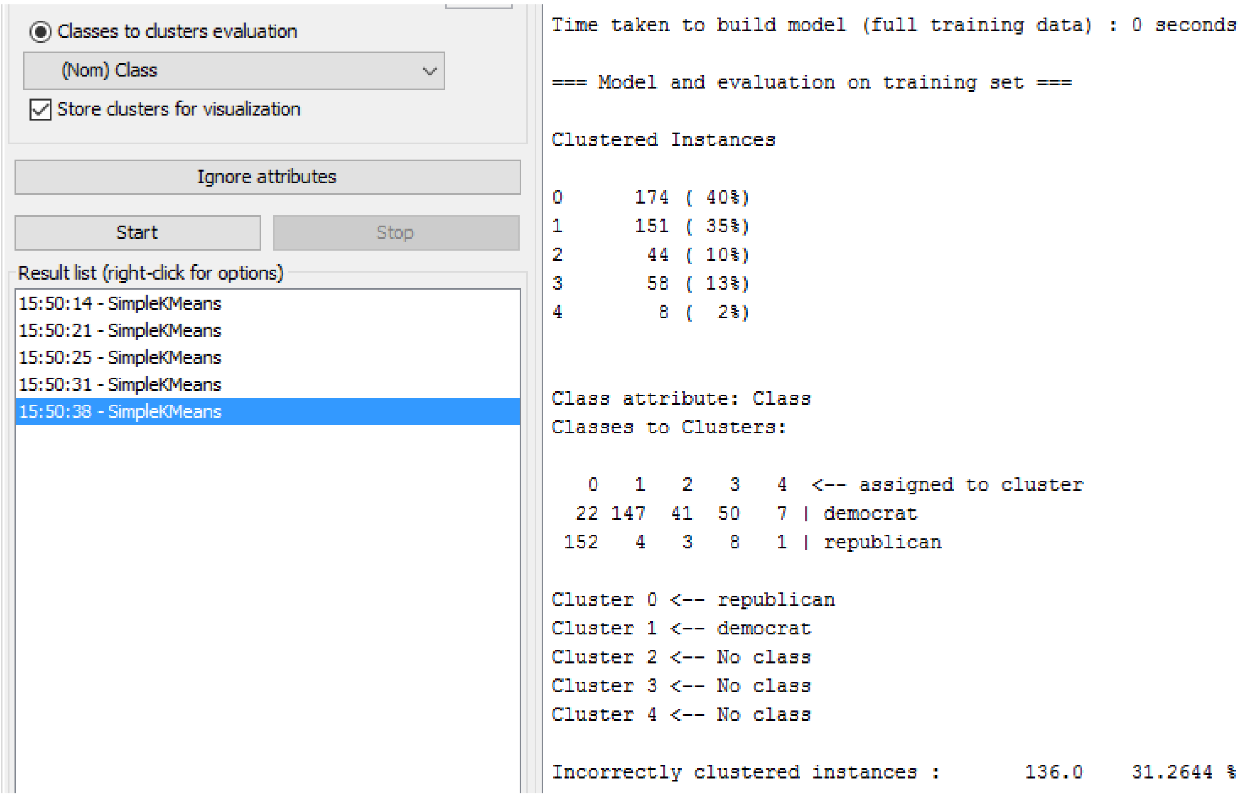
☑ Store clusters for visualization

Result list (right-click for options)

```
15:50:14 - SimpleKMeans
15:50:21 - SimpleKMeans
15:50:25 - SimpleKMeans
15:50:31 - SimpleKMeans
15:50:38 - SimpleKMeans
```

Clusterer output

```
======

Number of iterations: 3
Within cluster sum of squared errors: 396.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                          Cluster#
Attribute                     Full Data          0          1          2          3          4
                                  (435)      (174)      (151)       (44)       (58)        (8)
==============================================================================================
water-project-cost-sharing            y          y          y          y          n          y
physician-fee-freeze                  n          y          n          n          n          n
el-salvador-aid                       y          y          n          n          n          y
immigration                           y          y          n          y          y          y
education-spending                    n          y          n          n          n          n
superfund-right-to-sue                y          y          n          y          n          n
```



```
Time taken to build model (full training data) : 0 seconds

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

Clustered Instances

0       174 ( 40%)
1       151 ( 35%)
2        44 ( 10%)
3        58 ( 13%)
4         8 (  2%)


Class attribute: Class
Classes to Clusters:

   0   1   2   3   4  <-- assigned to cluster
  22 147  41  50   7 | democrat
 152   4   3   8   1 | republican

Cluster 0 <-- republican
Cluster 1 <-- democrat
Cluster 2 <-- No class
Cluster 3 <-- No class
Cluster 4 <-- No class

Incorrectly clustered instances :       136.0    31.2644 %
```

Inference from the above screenshots:

Maximum NO. of Iterations in clustering (default)	Actual No. of Iterations during clustering	Number of Clusters (Constant)	SSE	Number of Incorrectly Clustered Instances
50	1	1	1304	168
50	3	2	599	79
50	3	3	472	124
50	3	4	411	128
50	3	5	396	136

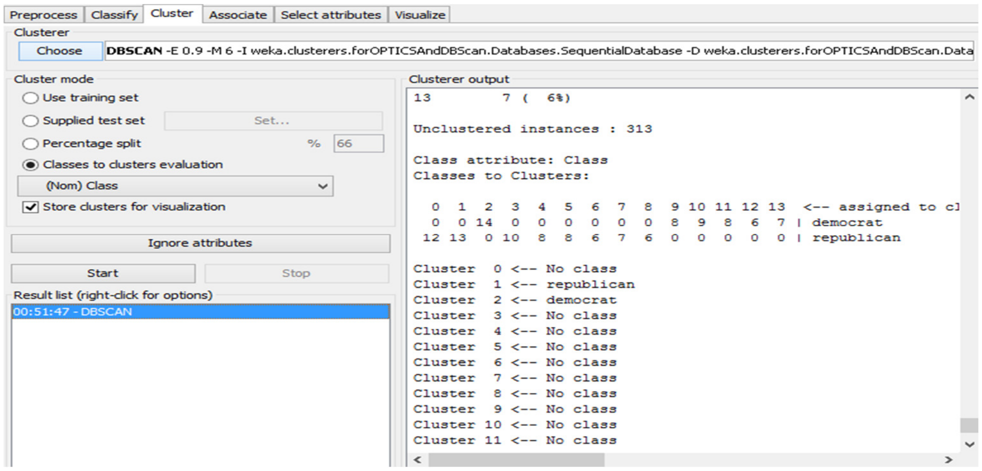
Inferences:

Number of clusters increases, SSE decreases. The optimum point is reached when SSE is less and number of incorrect clustered instances is also less.

In our case, the second run gives the optimum SSE along with less number of incorrectly clustered instances.

Vote Dataset with DBScan Clustering Algorithm:

Execution of DBScan algorithm on Vote Dataset with default values:



DBSCAN -E 0.9 -M 6 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAndDBScan.Data

○ 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)
00:51:47 - DBSCAN

Clusterer output

```
13         7 (  6%)

Unclustered instances : 313

Class attribute: Class
Classes to Clusters:

  0  1  2  3  4  5  6  7  8  9 10 11 12 13  <-- assigned to cl
  0  0 14  0  0  0  0  0  0  8  9  8  6  7 | democrat
 12 13  0 10  8  8  6  7  6  0  0  0  0  0 | republican

Cluster  0 <-- No class
Cluster  1 <-- republican
Cluster  2 <-- democrat
Cluster  3 <-- No class
Cluster  4 <-- No class
Cluster  5 <-- No class
Cluster  6 <-- No class
Cluster  7 <-- No class
Cluster  8 <-- No class
Cluster  9 <-- No class
Cluster 10 <-- No class
Cluster 11 <-- No class
```

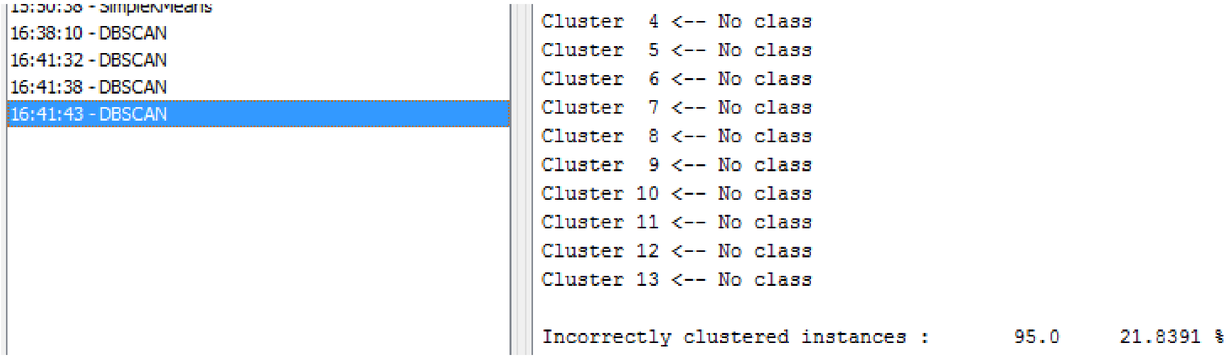
The above fig, shows the execution of the DB Scan algorithm changing the various parameters such as “epislon”,”Minpts”.

Inference is arrived by changing the epislon and minpts values at each run and how it affects the “ Number of Clustered instances”, “Number of UnClustered instances and “Number of Incorrectly clustered Instances”.

Default Parameters:

The parameters database_Type and database_distanceType are set with default values.

Running DB Scan algorithm by changing various parameters.



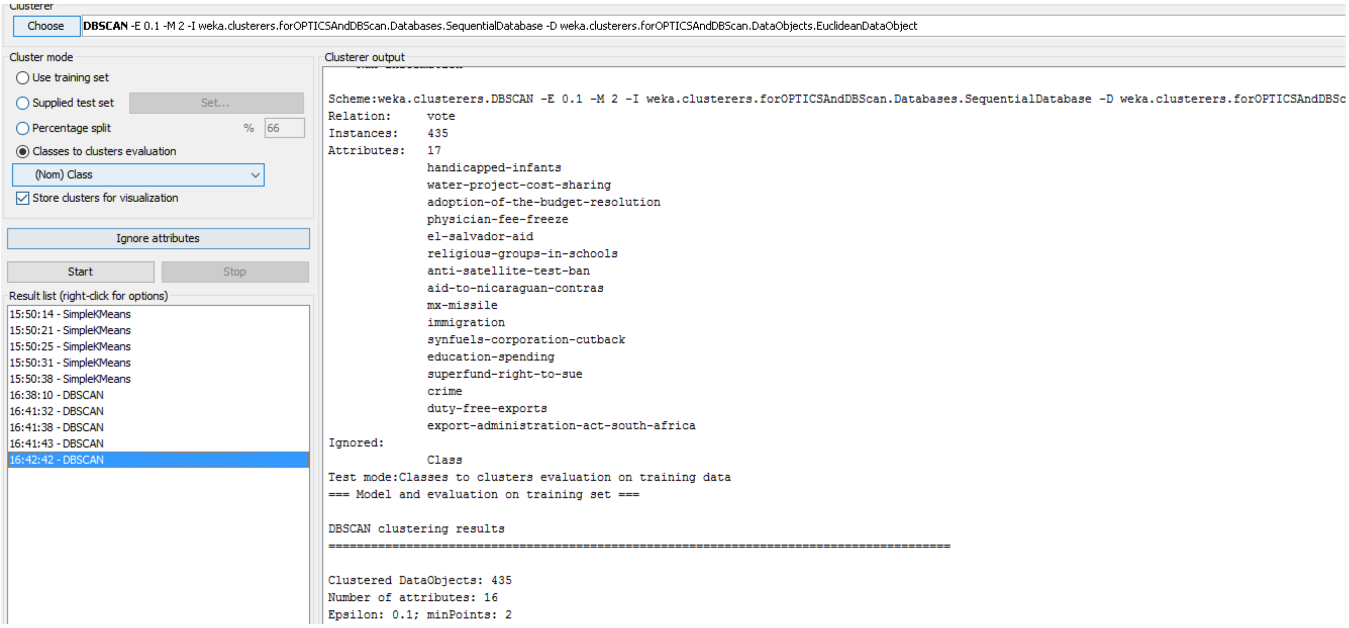
15:50:38 - SimpleKMeans
16:38:10 - DBSCAN
16:41:32 - DBSCAN
16:41:38 - DBSCAN
16:41:43 - DBSCAN

```
Cluster  4 <-- No class
Cluster  5 <-- No class
Cluster  6 <-- No class
Cluster  7 <-- No class
Cluster  8 <-- No class
Cluster  9 <-- No class
Cluster 10 <-- No class
Cluster 11 <-- No class
Cluster 12 <-- No class
Cluster 13 <-- No class

Incorrectly clustered instances :       95.0     21.8391 %
```

The above picture shows the execution of the DB Scan algorithm changing the various parameters such as “epislon”,”Minpts”.

Impact of Changing the epislon and minpts parameter in DBScan algorithm over Vote Dataset:

When epsilon = 0.1, minPoints = 2



Clusterer

Choose **DBSCAN** -E 0.1 -M 2 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAndDBScan.DataObjects.EuclideanDataObject

Cluster mode

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

Ignore attributes

Start Stop

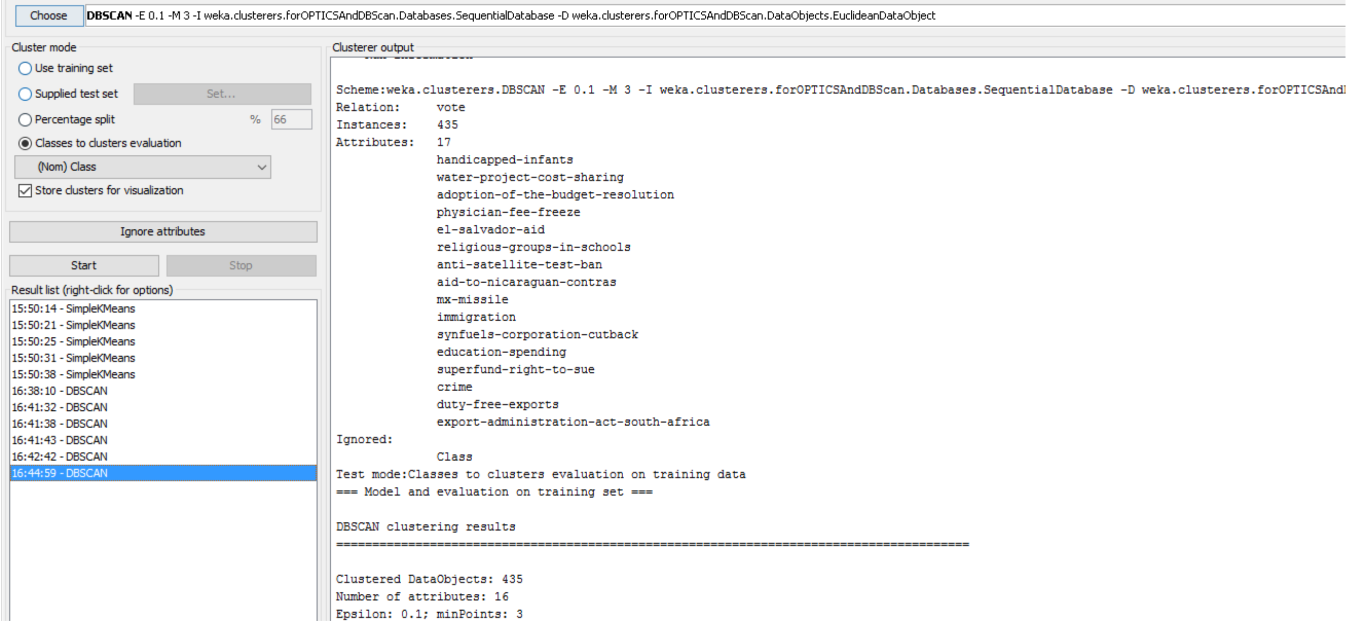
Result list (right-click for options)

15:50:14 - SimpleKMeans
15:50:21 - SimpleKMeans
15:50:25 - SimpleKMeans
15:50:31 - SimpleKMeans
15:50:38 - SimpleKMeans
16:38:10 - DBSCAN
16:41:32 - DBSCAN
16:41:38 - DBSCAN
16:41:43 - DBSCAN
16:42:42 - DBSCAN

Clusterer output

```
Scheme:weka.clusterers.DBSCAN -E 0.1 -M 2 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAndDBSc
Relation:     vote
Instances:    435
Attributes:   17
              handicapped-infants
              water-project-cost-sharing
              adoption-of-the-budget-resolution
              physician-fee-freeze
              el-salvador-aid
              religious-groups-in-schools
              anti-satellite-test-ban
              aid-to-nicaraguan-contras
              mx-missile
              immigration
              synfuels-corporation-cutback
              education-spending
              superfund-right-to-sue
              crime
              duty-free-exports
              export-administration-act-south-africa
Ignored:
              Class
Test mode:Classes to clusters evaluation on training data
=== Model and evaluation on training set ===

DBSCAN clustering results
========================================================================================

Clustered DataObjects: 435
Number of attributes: 16
Epsilon: 0.1; minPoints: 2
```

When epsilon = 0.1, minPoints = 3



Choose **DBSCAN** -E 0.1 -M 3 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAndDBScan.DataObjects.EuclideanDataObject

Cluster mode

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

Ignore attributes

Start Stop

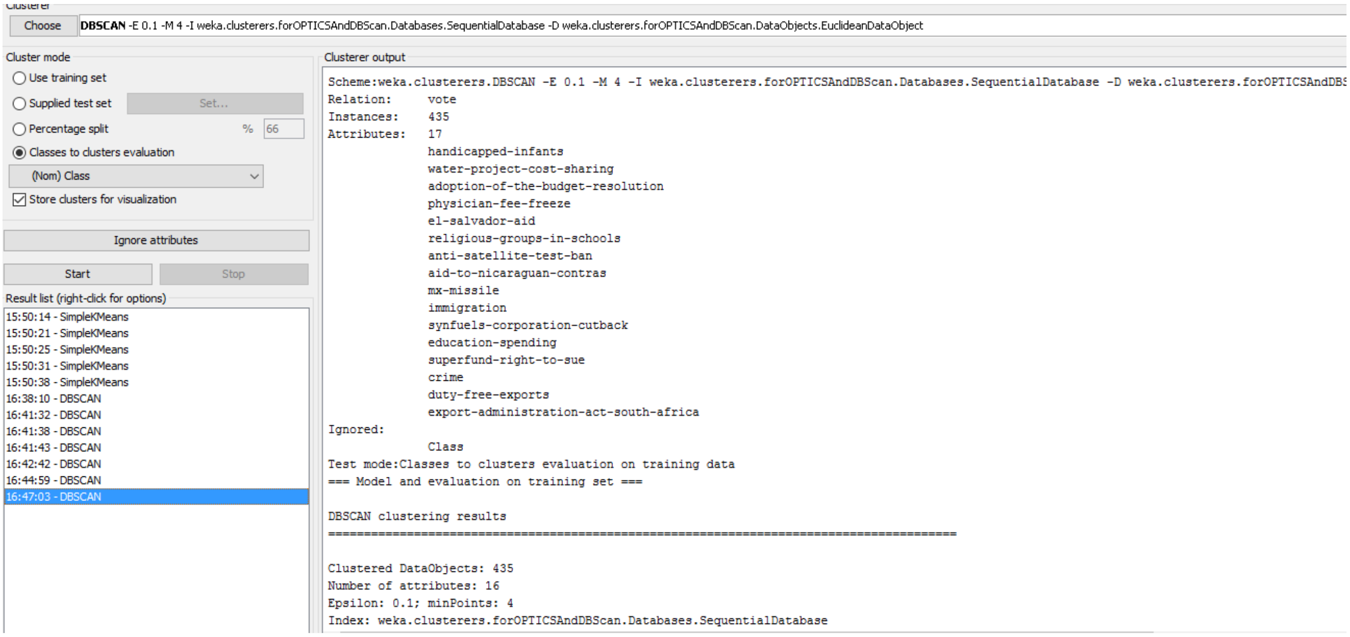
Result list (right-click for options)

15:50:14 - SimpleKMeans
15:50:21 - SimpleKMeans
15:50:25 - SimpleKMeans
15:50:31 - SimpleKMeans
15:50:38 - SimpleKMeans
16:38:10 - DBSCAN
16:41:32 - DBSCAN
16:41:38 - DBSCAN
16:41:43 - DBSCAN
16:42:42 - DBSCAN
16:44:59 - DBSCAN

Clusterer output

```
Scheme:weka.clusterers.DBSCAN -E 0.1 -M 3 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAnd
Relation:     vote
Instances:    435
Attributes:   17
              handicapped-infants
              water-project-cost-sharing
              adoption-of-the-budget-resolution
              physician-fee-freeze
              el-salvador-aid
              religious-groups-in-schools
              anti-satellite-test-ban
              aid-to-nicaraguan-contras
              mx-missile
              immigration
              synfuels-corporation-cutback
              education-spending
              superfund-right-to-sue
              crime
              duty-free-exports
              export-administration-act-south-africa
Ignored:
              Class
Test mode:Classes to clusters evaluation on training data
=== Model and evaluation on training set ===

DBSCAN clustering results
========================================================================================

Clustered DataObjects: 435
Number of attributes: 16
Epsilon: 0.1; minPoints: 3
```

When epsilon = 0.1, minPoints = 4



Clusterer: DBSCAN -E 0.1 -M 4 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAndDBScan.DataObjects.EuclideanDataObject

Cluster mode

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

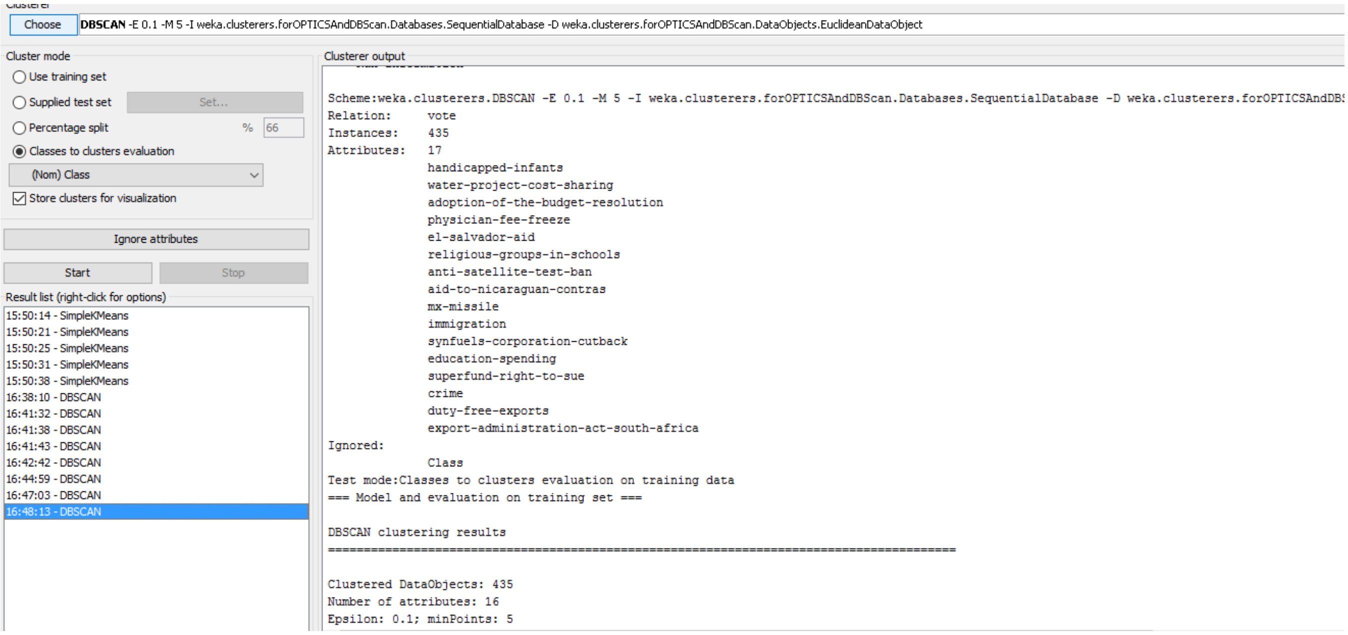
Result list (right-click for options)

```
15:50:14 - SimpleKMeans
15:50:21 - SimpleKMeans
15:50:25 - SimpleKMeans
15:50:31 - SimpleKMeans
15:50:38 - SimpleKMeans
16:38:10 - DBSCAN
16:41:32 - DBSCAN
16:41:38 - DBSCAN
16:41:43 - DBSCAN
16:42:42 - DBSCAN
16:44:59 - DBSCAN
16:47:03 - DBSCAN
```

Clusterer output

```
Scheme:weka.clusterers.DBSCAN -E 0.1 -M 4 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAndDBS
Relation:     vote
Instances:    435
Attributes:   17
              handicapped-infants
              water-project-cost-sharing
              adoption-of-the-budget-resolution
              physician-fee-freeze
              el-salvador-aid
              religious-groups-in-schools
              anti-satellite-test-ban
              aid-to-nicaraguan-contras
              mx-missile
              immigration
              synfuels-corporation-cutback
              education-spending
              superfund-right-to-sue
              crime
              duty-free-exports
              export-administration-act-south-africa
Ignored:
              Class
Test mode:Classes to clusters evaluation on training data
=== Model and evaluation on training set ===

DBSCAN clustering results
========================================================================================

Clustered DataObjects: 435
Number of attributes: 16
Epsilon: 0.1; minPoints: 4
Index: weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase
```

When epsilon = 0.1, minPoints = 5



Clusterer: DBSCAN -E 0.1 -M 5 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAndDBScan.DataObjects.EuclideanDataObject

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)

```
15:50:14 - SimpleKMeans
15:50:21 - SimpleKMeans
15:50:25 - SimpleKMeans
15:50:31 - SimpleKMeans
15:50:38 - SimpleKMeans
16:38:10 - DBSCAN
16:41:32 - DBSCAN
16:41:38 - DBSCAN
16:41:43 - DBSCAN
16:42:42 - DBSCAN
16:44:59 - DBSCAN
16:47:03 - DBSCAN
16:48:13 - DBSCAN
```

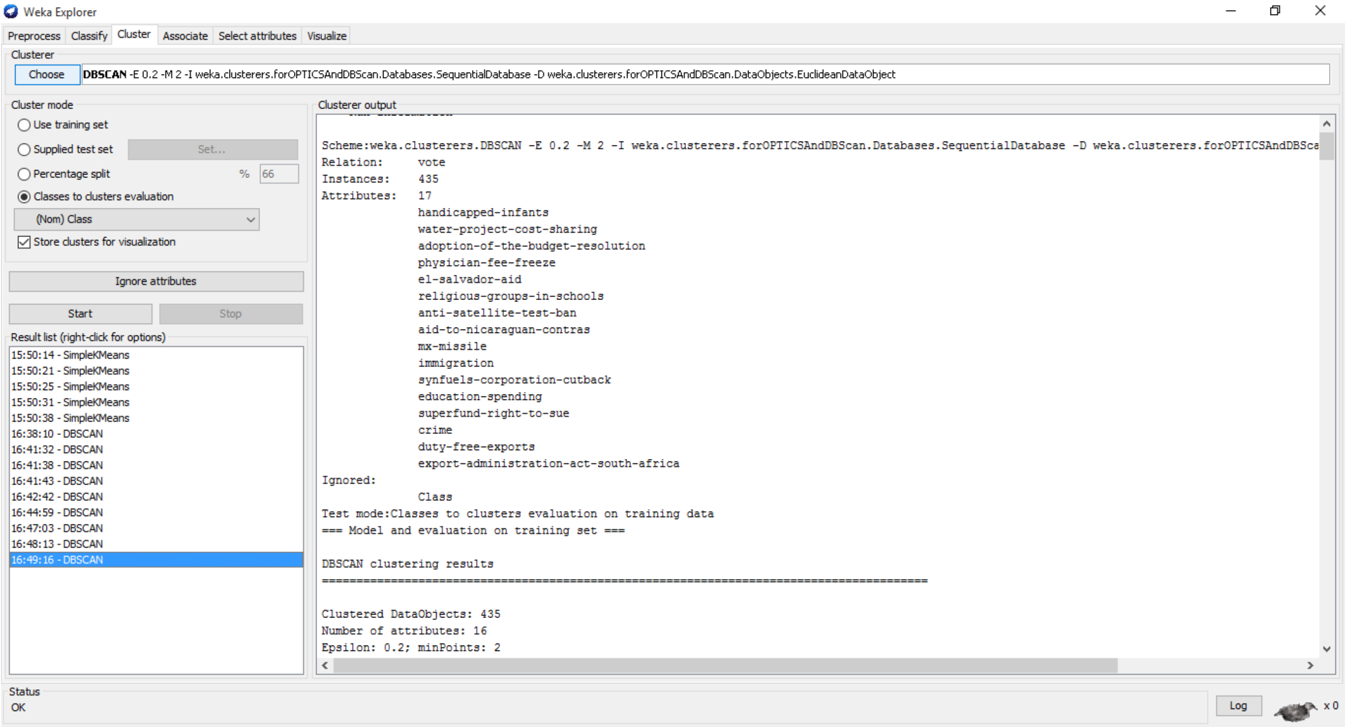
Clusterer output

```
Scheme:weka.clusterers.DBSCAN -E 0.1 -M 5 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAndDBS
Relation:     vote
Instances:    435
Attributes:   17
              handicapped-infants
              water-project-cost-sharing
              adoption-of-the-budget-resolution
              physician-fee-freeze
              el-salvador-aid
              religious-groups-in-schools
              anti-satellite-test-ban
              aid-to-nicaraguan-contras
              mx-missile
              immigration
              synfuels-corporation-cutback
              education-spending
              superfund-right-to-sue
              crime
              duty-free-exports
              export-administration-act-south-africa
Ignored:
              Class
Test mode:Classes to clusters evaluation on training data
=== Model and evaluation on training set ===

DBSCAN clustering results
========================================================================================

Clustered DataObjects: 435
Number of attributes: 16
Epsilon: 0.1; minPoints: 5
```

ASSIGNMENT 5: CLUSTER ANALYSIS

PREPARED BY: RAJARAJESWARI VAIDYANATHAN **CWID**: A20362220 **COURSE**: CS422 – DATA MINING

When epsilon = 0.2, minPoints = 2



DBSCAN -E 0.2 -M 2 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAndDBScan.DataObjects.EuclideanDataObject

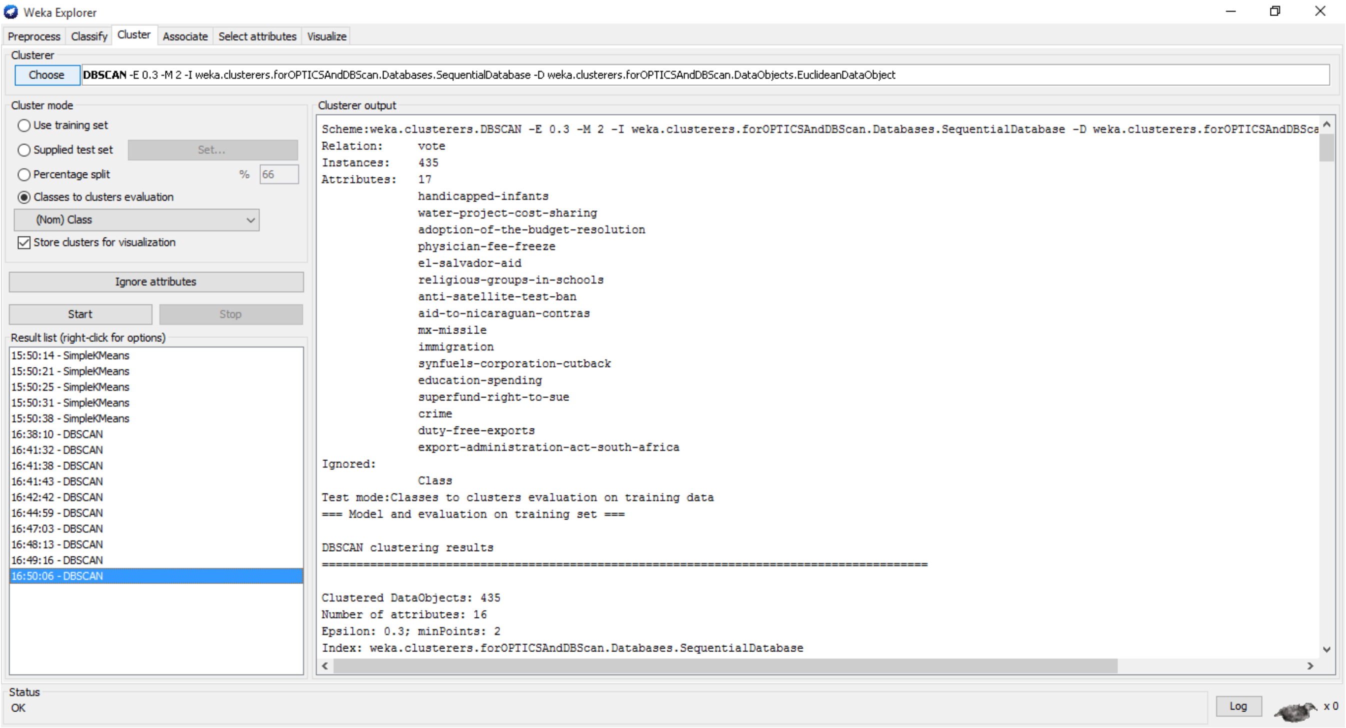
○ Use training set
○ Supplied test set
○ Percentage split % 66
● Classes to clusters evaluation
(Nom) Class
☑ Store clusters for visualization

```
15:50:14 - SimpleKMeans
15:50:21 - SimpleKMeans
15:50:25 - SimpleKMeans
15:50:31 - SimpleKMeans
15:50:38 - SimpleKMeans
16:38:10 - DBSCAN
16:41:32 - DBSCAN
16:41:38 - DBSCAN
16:41:43 - DBSCAN
16:42:42 - DBSCAN
16:44:59 - DBSCAN
16:47:03 - DBSCAN
16:48:13 - DBSCAN
16:49:16 - DBSCAN
```

```
Scheme:weka.clusterers.DBSCAN -E 0.2 -M 2 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAndDBSca
Relation:     vote
Instances:    435
Attributes:   17
              handicapped-infants
              water-project-cost-sharing
              adoption-of-the-budget-resolution
              physician-fee-freeze
              el-salvador-aid
              religious-groups-in-schools
              anti-satellite-test-ban
              aid-to-nicaraguan-contras
              mx-missile
              immigration
              synfuels-corporation-cutback
              education-spending
              superfund-right-to-sue
              crime
              duty-free-exports
              export-administration-act-south-africa
Ignored:
              Class
Test mode:Classes to clusters evaluation on training data
=== Model and evaluation on training set ===

DBSCAN clustering results
========================================================================================

Clustered DataObjects: 435
Number of attributes: 16
Epsilon: 0.2; minPoints: 2
```

Status
OK

When epsilon = 0.3, minPoints = 2



DBSCAN -E 0.3 -M 2 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAndDBScan.DataObjects.EuclideanDataObject

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

```
15:50:14 - SimpleKMeans
15:50:21 - SimpleKMeans
15:50:25 - SimpleKMeans
15:50:31 - SimpleKMeans
15:50:38 - SimpleKMeans
16:38:10 - DBSCAN
16:41:32 - DBSCAN
16:41:38 - DBSCAN
16:41:43 - DBSCAN
16:42:42 - DBSCAN
16:44:59 - DBSCAN
16:47:03 - DBSCAN
16:48:13 - DBSCAN
16:49:16 - DBSCAN
16:50:06 - DBSCAN
```

```
Scheme:weka.clusterers.DBSCAN -E 0.3 -M 2 -I weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D weka.clusterers.forOPTICSAndDBSca
Relation:     vote
Instances:    435
Attributes:   17
              handicapped-infants
              water-project-cost-sharing
              adoption-of-the-budget-resolution
              physician-fee-freeze
              el-salvador-aid
              religious-groups-in-schools
              anti-satellite-test-ban
              aid-to-nicaraguan-contras
              mx-missile
              immigration
              synfuels-corporation-cutback
              education-spending
              superfund-right-to-sue
              crime
              duty-free-exports
              export-administration-act-south-africa
Ignored:
              Class
Test mode:Classes to clusters evaluation on training data
=== Model and evaluation on training set ===

DBSCAN clustering results
========================================================================================

Clustered DataObjects: 435
Number of attributes: 16
Epsilon: 0.3; minPoints: 2
Index: weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase
```

Status
OK

Epsilon	Minpts	Number of Incorrectly clustered Instances	Number of unclustered Instances
0.1	2	171	237
0.1	3	137	271
0.1	4	122	286
0.1	5	110	298
0.2	2	171	237
0.2	3	137	271
0.2	4	122	286
0.2	5	110	298
0.3	2	171	237
0.3	3	137	271
0.3	4	122	286
0.3	5	110	298

Inferences:

- For each incrementally increments, the epsilon values and minpts are changed, the number of incorrectly clustered instances shows a pattern as seen from the above tabular.

- This pattern seems to be similar when epsilon = 0.1 or 0.2 or 0.3 (AND) when minPoints = 2, the density space is uniform.

- Initially the epsilon value is kept constant and the minpts is increased ,we could infer that the Number of Incorrectly clustered instances decreases and Number of Unclustered Instances increases.

- To have an optimistic clustering we need to minimize the number of Incorrectly clustered Instances and less importance is given to the number of unclustered instances because the unclustered instances may be outliers or noise. So we consider only the Incorrectly clustered instances which we expect it be low.

- Now we could infer that for each epsilon value say 0.1 and 0.2 and corresponding minpts values 2, 3, 4,5 throws the same output in Number of Incorrectly clustered Instances and Number of Unclustered Instances. The pattern repeats in the above table.

- Hence the above inference shows that the density of the data points are uniform and consistent .The cluster density is also uniform and consistent.

Impact on including only the specific attributes for Clustering using DBSCAN algorithm:

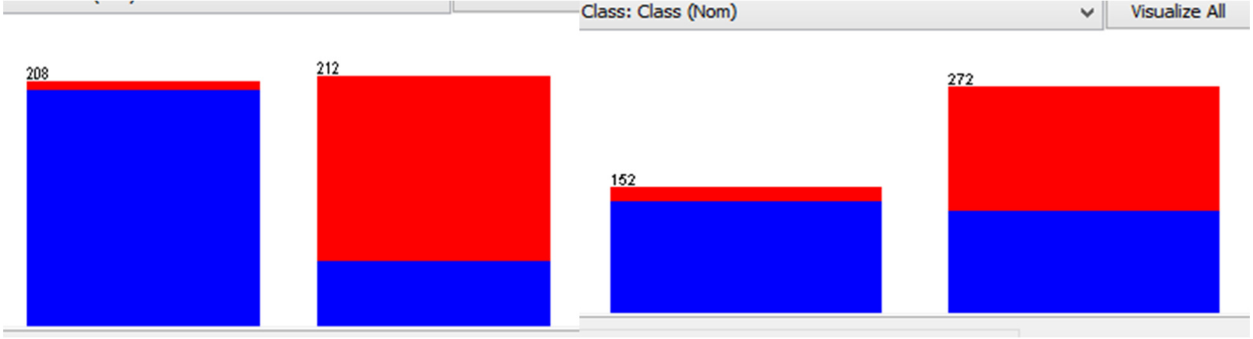
Attributes selected for Clustering	Number of Incorrectly Clustered Instances	Number of UnClustered Instances
Water-project cost sharing, immigration	302	192
education, spending superfund right to use	145	111
Physician fee freeze ,el salvor aid	80	64

The above table shows the execution of the DBScan clustering using only some of the attributes / dimensions in the dataset.

Inference:

- Selecting specific attributes results in good clustering with minimum number of Unclustered instances and minimum number of incorrectly clustered instances.
- The global optimum function Number of Incorrectly clustered instances is used to predict the optimistic clustering. Now the clusters produced using specific attributes is visualized and explained how the Number of Incorrectly clustered instances is obtained for those cluster.

Clusters using attributes Water-project cost sharing, immigration:

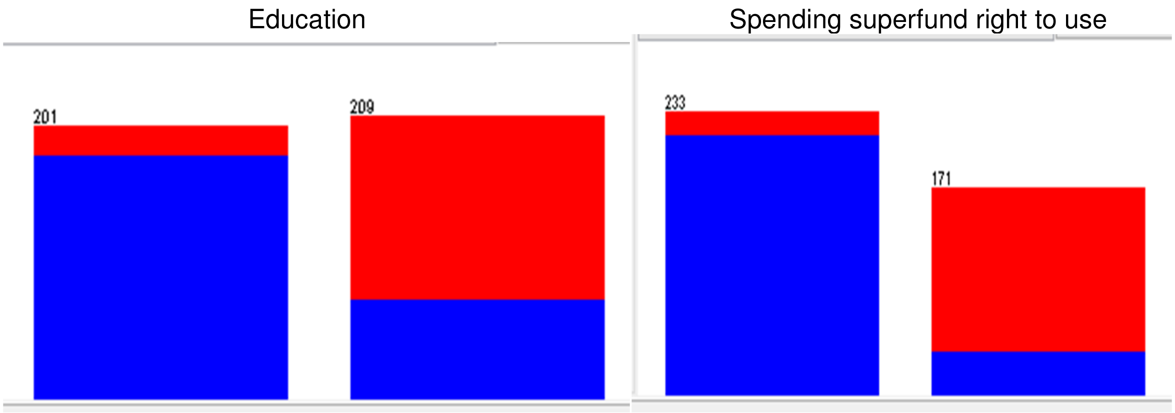

Water-project cost sharing

Immigration

Inferences:

- The above figure shows the clusters formed using the attributes Water-project cost sharing and immigration.
- It doesn't represent the both the classes and the distribution is not uniform hence the Number of Incorrectly clustered instances is very high.
- The data points in the cluster are not uniform and distorted.
- Hence the Number of Incorrectly clustered instances is larger.

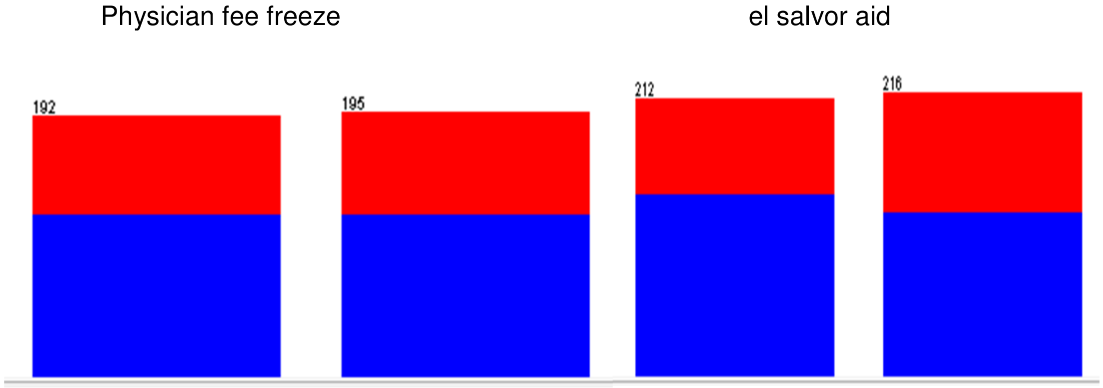
Cluster using attributes education, spending superfund right to use:


Education

Spending superfund right to use

This figure shows the distribution of the clusters in education, spending superfund right to use attributes, here the distribution is uniform and hence the clustering has less Number of Incorrectly clustered instances when compared to previous run.

Cluster using attributes Physician fee freeze ,el salvor aid:


Physician fee freeze

el salvor aid

In this figure the distribution of the instances is uniform, Hence the attributes Physician fee freeze ,el salvor aid yields to the optimistic Number of Incorrectly clustered instances and incorrectly clustered instances.

CONCLUSION/SUMMARY:

- The number of clusters is increased and various possibility of SSE (Sum of Squared Error) and Number of Incorrectly observed instances are observed in 2 attributes.
- As the number of clusters increases the SSE decreases, this is because as the number of clusters increases, the points in the dataset are placed with more appropriate clusters i.e. data point moves to the closest cluster centroid which in turn causes the SSE of each cluster to reduce.
- As the number of cluster increases, the Incorrectly clustered instances initially decreases , when the number of cluster further increases it leads to outliers and missing data values fall into those clusters and there by the Incorrectly clustered instances increases.
- Default parameters, it is inferred that there are many incorrectly clustered instances and the number of iterations is more.
- For IRIS dataset, optimum number of clusters is 3 as highlighted above as the SSE is also less and number of incorrect instances is also less when compared to the previous runs.
- Also, though we try increasing the maximum number of iterations, the cycle stops at 6 and does not iterate further. This implies that the centroid has been achieved and cannot be iterated further.
- When Number of clusters = 2, (default value) without depending on the Class variable the clusters formed are very poor as it has got high SSE as shown below.
- The number of clusters is increased and more possible chances of SSE and incorrect instances being reported.
- When the number of cluster increases, the misplaced clustered instances decreases.
- When the number of clusters further increases and eventually leads to outliers and missing data thereby increasing the incorrect instances.
- The points inside the clusters are not uniform and are distorted, leading to a total SSE which is larger.
- Selecting specific attributes results in good clustering with min SSE and min Number of incorrect instances.
- If the attribute which are selected randomly represents the data well then they are to be called good clusters.
- For optimistic clustering , minimize the number of Incorrectly clustered Instances and less importance is given to the number of unclustered instances. Clustering when the epsilon value is low and minpts is high. If the attribute selected represents the data well. Good clusters are created.
- For each incrementally increments epsilon value and corresponding value ,the Number of Incorrectly clustered instance shows a pattern.The pattern is because the density space is uniform.If the attribute selected represents the data well. Good clusters are created.