

k- means cluster

Aim: To implement and design kmeans cluster algorithm using weka.

Algorithm:

- Determine root node
- Calculate entropy for classes
- Calculate entropy after split for each attributes.
- Calculate information gain
- Perform split
- Perform further split
- Compute kmeans cluster algorithm

Output:

The screenshot displays the Weka GUI with the SimpleKMeans clustering algorithm applied to the Iris dataset. The 'Clusterer' tab is active, showing the command: `SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 2 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1`. The 'Cluster mode' section has 'Use training set' selected. The 'Clusterer output' pane shows the following results:

Within cluster sum of squared errors: 62.1436882815797

Initial starting points (random):

Cluster 0: 6.1,2.9,4.7,1.4,Iris-versicolor
Cluster 1: 6.2,2.9,4.3,1.3,Iris-versicolor

Missing values globally replaced with mean/mode

Final cluster centroids:

Attribute	Full Data (150.0)	Cluster# 0 (100.0)	1 (50.0)
sepal.length	5.8433	6.262	5.006
sepal.width	3.054	2.872	3.418
petal.length	3.7597	4.906	1.464
petal.width	1.1997	1.676	0.244
class	Iris-setosa Iris-versicolor Iris-setosa		

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

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

Clustered Instances

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

The bottom status bar shows 'Status OK' and 'Log' button. The system tray at the bottom indicates '26°C Partly cloudy' and the date '05-02-2023'.