Deep Panchal
TY - 9 - B - 38

Experiment no : 8

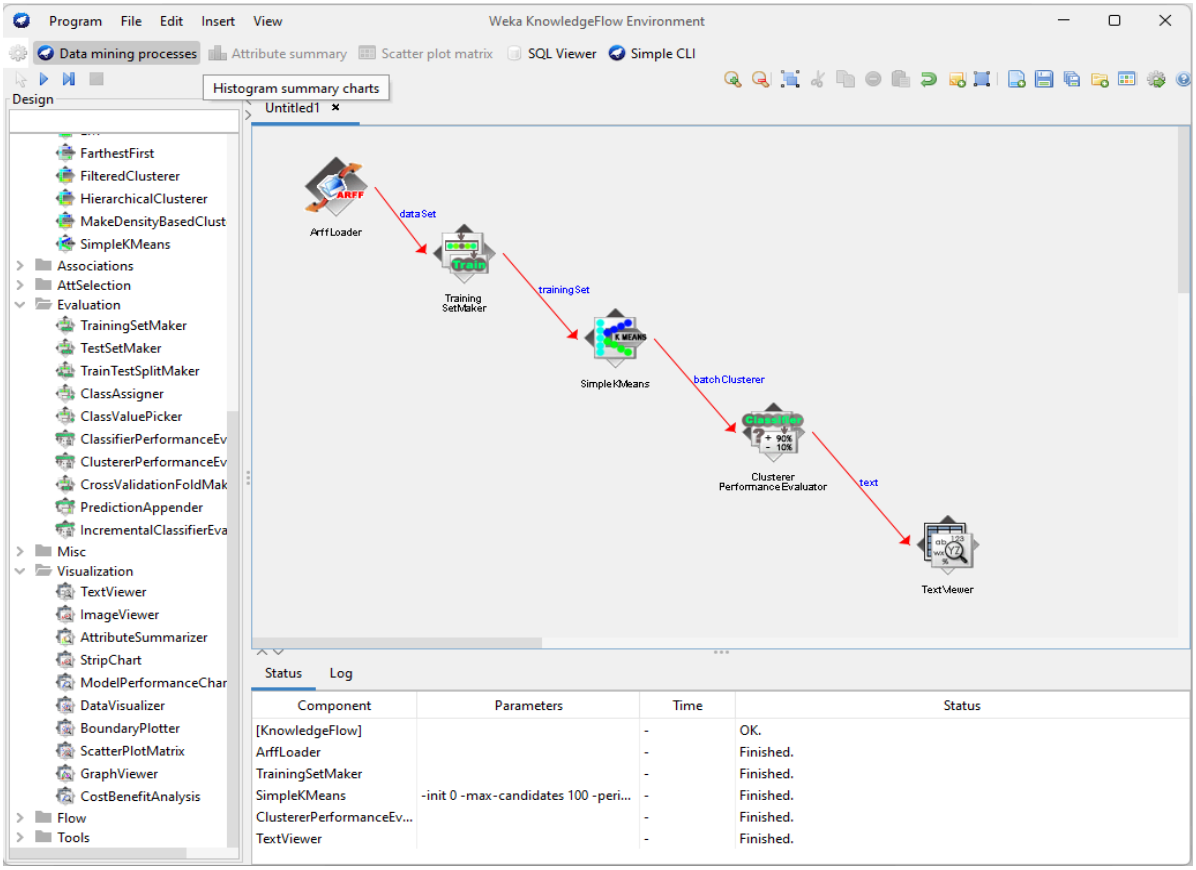
Aim: Perform data Pre-processing task and Demonstrate Classification algorithm on data sets using data mining tools (WEKA)

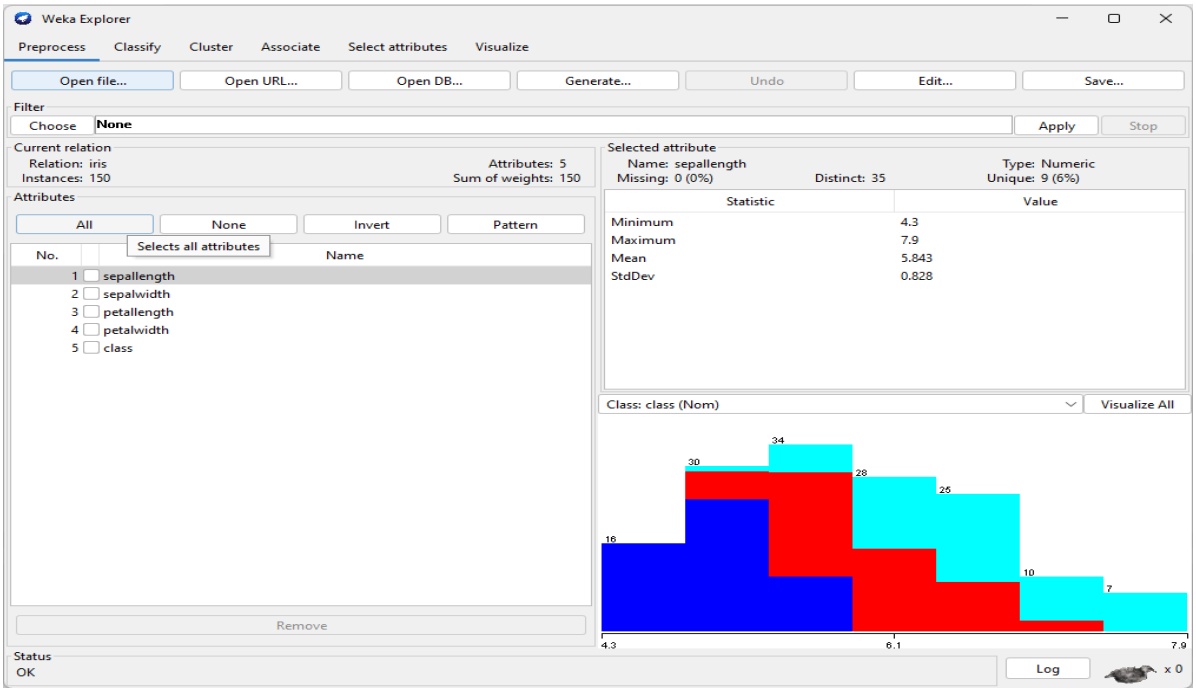
What is WEKA

Weka is an open-source software suite for machine learning and data mining. It offers tools for data preprocessing, classification, regression, clustering, and visualization. It provides a user-friendly graphical interface and supports a wide range of algorithms like decision trees, and k-means. Weka is widely used for educational and research purposes due to its ease of use and extensibility.

1. Install Weka.
2. Load a Dataset into Weka.
3. Preprocess the Data (cleaning, feature selection, normalization).
4. Choose an Algorithm (classification, clustering, etc.).
5. Train the Model using cross-validation or split data.
6. Evaluate the Model (accuracy, confusion matrix, etc.).
7. Visualize the model if applicable (e.g., decision trees).
8. Save the Model or make predictions on new data.

OUTPUT:


Program File Edit Insert View
Weka KnowledgeFlow Environment
Data mining processes
Attribute summary
Scatter plot matrix
SQL Viewer
Simple CLI
Histogram summary charts
Design
Untitled1
FarthestFirst
FilteredClusterer
HierarchicalClusterer
MakeDensityBasedClust
SimpleKMeans
Associations
AttSelection
Evaluation
TrainingSetMaker
TestSetMaker
TrainTestSplitMaker
ClassAssigner
ClassValuePicker
ClassifierPerformanceEv
ClustererPerformanceEv
CrossValidationFoldMak
PredictionAppender
IncrementalClassifierEva
Misc
Visualization
TextViewer
ImageViewer
AttributeSummarizer
StripChart
ModelPerformanceChar
DataVisualizer
BoundaryPlotter
ScatterPlotMatrix
GraphViewer
CostBenefitAnalysis
Flow
Tools
ArffLoader
data Set
Training SetMaker
training Set
SimpleKMeans
batch Clusterer
Clusterer PerformanceEvaluator
text
TextViewer
Status
Log
Component
Parameters
Time
Status
[KnowledgeFlow]
OK.
ArffLoader
Finished.
TrainingSetMaker
Finished.
SimpleKMeans
-init 0 -max-candidates 100 -peri...
Finished.
ClustererPerformanceEv...
Finished.
TextViewer
Finished.


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Open file...
Open URL...
Open DB...
Generate...
Undo
Edit...
Save...
Filter
Choose
None
Apply
Stop
Current relation
Relation: iris
Instances: 150
Attributes: 5
Sum of weights: 150
Attributes
All
None
Invert
Pattern
Selects all attributes
No.
Name
1 sepallength
2 sepalwidth
3 petallength
4 petalwidth
5 class
Remove
Selected attribute
Name: sepallength
Missing: 0 (0%)
Distinct: 35
Type: Numeric
Unique: 9 (6%)
Statistic
Value
Minimum 4.3
Maximum 7.9
Mean 5.843
StdDev 0.828
Class: class (Nom)
Visualize All
34
30
28
26
16
10
7
4.3
6.1
7.9
Status
OK
Log
x 0

Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize

Clusterer

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

Cluster mode

- (•) Use training set
- ( ) Supplied test set — Set...
- ( ) Percentage split — % 66
- ( ) Classes to clusters evaluation
  - (Nom) class
- [x] Store clusters for visualization

Ignore attributes

Start

Ignore attributes during clustering

Result list (right-click for options)

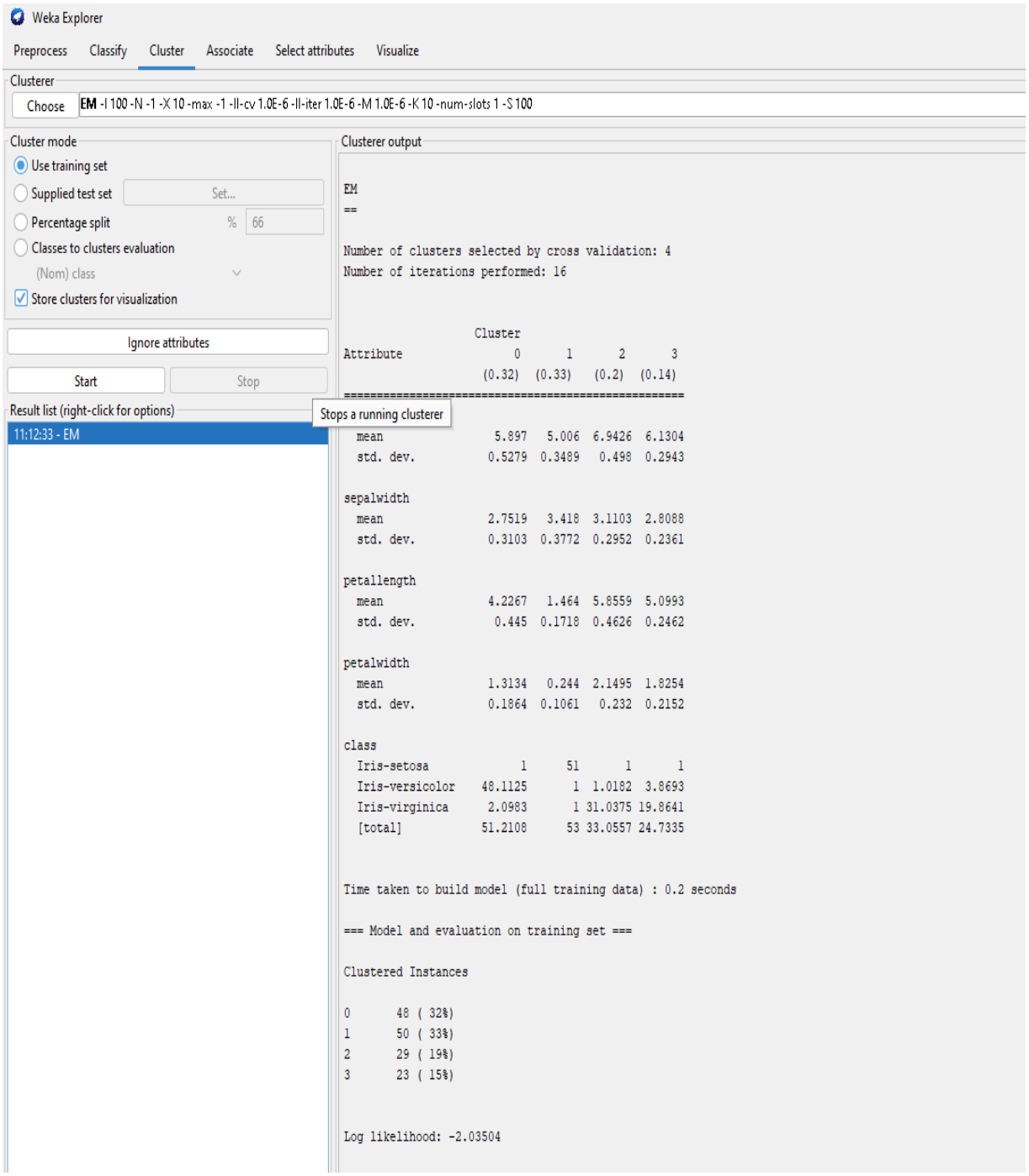
11:12:33 - EM

Clusterer output

```
=== Run information ===

Scheme:       weka.clusterers.EM -I 100 -N -1 -X 10 -max -1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100
Relation:     iris
Instances:    150
Attributes:   5
              sepallength
              sepalwidth
              petallength
              petalwidth
              class
Test mode:    evaluate on training data


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


EM
==

Number of clusters selected by cross validation: 4
Number of iterations performed: 16


                  Cluster
Attribute               0      1      2      3
                   (0.32)  (0.33)   (0.2)  (0.14)
=================================================
sepallength
  mean              5.897   5.006  6.9426  6.1304
  std. dev.         0.5279  0.3489   0.498  0.2943

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

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

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

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

Status

OK


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Clusterer
Choose
EM -I 100 -N -1 -X 10 -max -1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100
Cluster mode
Use training set
Supplied test set
Set...
Percentage split
%
66
Classes to clusters evaluation
(Nom) class
Store clusters for visualization
Ignore attributes
Start
Stop
Result list (right-click for options)
11:12:33 - EM
Stops a running clusterer
Clusterer output
EM
==
Number of clusters selected by cross validation: 4
Number of iterations performed: 16
Cluster
Attribute 0 1 2 3
(0.32) (0.33) (0.2) (0.14)
sepallength
mean 5.897 5.006 6.9426 6.1304
std. dev. 0.5279 0.3489 0.498 0.2943
sepalwidth
mean 2.7519 3.418 3.1103 2.8088
std. dev. 0.3103 0.3772 0.2952 0.2361
petallength
mean 4.2267 1.464 5.8559 5.0993
std. dev. 0.445 0.1718 0.4626 0.2462
petalwidth
mean 1.3134 0.244 2.1495 1.8254
std. dev. 0.1864 0.1061 0.232 0.2152
class
Iris-setosa 1 51 1 1
Iris-versicolor 48.1125 1 1.0182 3.8693
Iris-virginica 2.0983 1 31.0375 19.8641
[total] 51.2108 53 33.0557 24.7335
Time taken to build model (full training data) : 0.2 seconds
=== Model and evaluation on training set ===
Clustered Instances
0 48 (32%)
1 50 (33%)
2 29 (19%)
3 23 (15%)
Log likelihood: -2.03504

Conclusion

Using WEKA, we performed data preprocessing and applied classification algorithms on a dataset. Proper preprocessing improved model performance. This demonstrates how essential clean data and the right algorithm are for successful classification in data mining.

GITHUB : https://github.com/panchaldeep1123/dwm