EXPERIMENT NO. 6

Aim: Perform data preprocessing task and demonstrate Classification, Clustering, Association algorithm on data sets using data mining tool(WEKA / R tool)

Softwares used: WEKA

Theory:

Waikato Environment for Knowledge Analysis (Weka) is a popular suite of machine learning software written in Java, developed at the University of Waikato, New Zealand. It is free software licensed under the GNU General Public License.

Weka is a workbench that contains a collection of visualization tools and algorithms for data analysis and predictive modeling, together with graphical user interfaces for easy access to these functions.

This original version was primarily designed as a tool for analyzing data from agricultural domains, but the more recent fully Java-based version (Weka 3), for which development started in 1997, is now used in many different application areas, in particular for educational purposes and research.

Advantages of Weka include:

- Free availability under the GNU General Public License.Portability, since it is fully implemented in the Java programming language and thus runs on almost any modern computing platform.
- A comprehensive collection of data preprocessing and modeling techniques.
- Ease of use due to its graphical user interfaces.

Weka supports several standard data mining tasks, more specifically, data preprocessing, clustering, classification, regression, visualization, and feature selection. All of Weka's techniques are predicated on the assumption that the data is available as one flat file or relation, where each data point is described by a fixed number of attributes (normally, numeric or nominal attributes, but some other attribute types are also supported). Weka provides access to SQL databases using Java Database Connectivity and can process the result returned by a database query. It is not capable of multi-relational data mining, but there is separate software for converting a collection of linked database tables into a single table that is suitable for processing using Weka.

Weka's main user interface is the Explorer, but essentially the same functionality can be accessed through the component-based Knowledge Flow interface and from the command line. There is also the Experimenter, which allows the systematic comparison of the predictive performance of Weka's machine learning algorithms on a collection of datasets.

The Explorer interface features several panels providing access to the main components of the workbench:

The Preprocess panel has facilities for importing data from a database, a comma-separated values (CSV) file, etc., and for preprocessing this data using a so-called filtering algorithm. These filters can be used to transform the data (e.g., turning numeric attributes into discrete ones) and make it possible to delete instances and attributes according to specific criteria.

The Classify panel enables applying classification and regression algorithms (indiscriminately called classifiers in Weka) to the resulting dataset, to estimate the accuracy of the resulting predictive model, and to visualize erroneous predictions, receiver operating characteristic (ROC) curves, etc., or the model itself (if the model is amenable to visualization like, e.g., a decision tree).

The Associate panel provides access to association rule learners that attempt to identify all important interrelationships between attributes in the data.

The Cluster panel gives access to the clustering techniques in Weka, e.g., the simple k-means algorithm. There is also an implementation of the expectation maximization algorithm for learning a mixture of normal distributions.

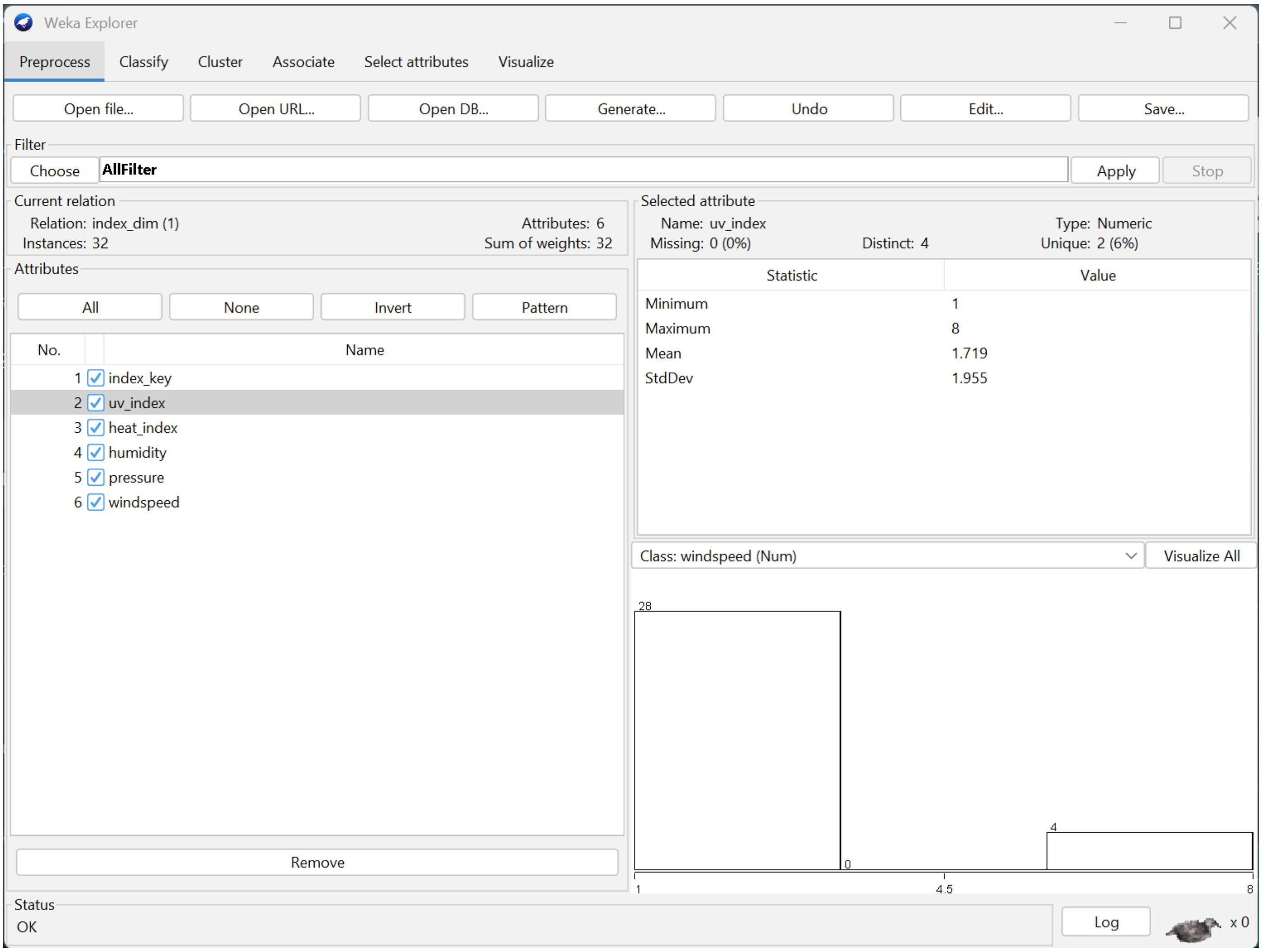
The Select attributes panel provides algorithms for identifying the most predictive attributes in a dataset.

The Visualize panel shows a scatter plot matrix, where individual scatter plots can be selected and enlarged, and analyzed further using various selection operators.

Preprocessing in WEKA

Selecting or Filtering Attributes

In the "Filter" panel, click on the "Choose" button. This will show a popup window with a list available filters. Scroll down the list and select the "weka.filters.unsupervised.attribute.Remove" filter as shown in Figure



Filter: AllFilter

Current relation
Relation: index_dim (1)
Attributes: 6
Instances: 32
Sum of weights: 32

Attributes

No.		Name
1	☑	index_key
2	☑	uv_index
3	☑	heat_index
4	☑	humidity
5	☑	pressure
6	☑	windspeed

Selected attribute
Name: uv_index
Type: Numeric
Missing: 0 (0%)
Distinct: 4
Unique: 2 (6%)

Statistic	Value
Minimum	1
Maximum	8
Mean	1.719
StdDev	1.955

Class: windspeed (Num)

Status
OK

Classification using WEKA:

This experiment illustrates the use of naïve bayes classifier in weka. Consider the sample data set “employee”data available at arff format. This document assumes that appropriate data pre processing has been performed.

Steps involved in this experiment:

1. Begin the experiment by loading the data (employee.arff) into weka.

Step2: Next we select the “classify” tab and click “choose” button to select the “Naïve Bayes”classifier.

Step3: Now specify the various parameters. These can be specified by clicking in the text box to the right of the chose button. In this example, accept the default values his default version does perform some pruning but does not perform error pruning.

Step4: Under the “text “options in the main pane l. select the 10-fold cross validation as our evaluation approach. Since we don’t have separate evaluation data set, this is necessary to get a reasonable idea of accuracy of generated model.

Step-5: now click”start”to generate the model .the ASCII version of the tree as well as evaluation statistic will appear in the right panel when the model construction is complete.

Step-6: Note that the classification accuracy ofmodel is about 69%.this indicates that we may find more work. (Either in preprocessing or in selecting current parameters for the classification)

Step-7: Now weka also lets us a view a graphical version of the classification tree. This

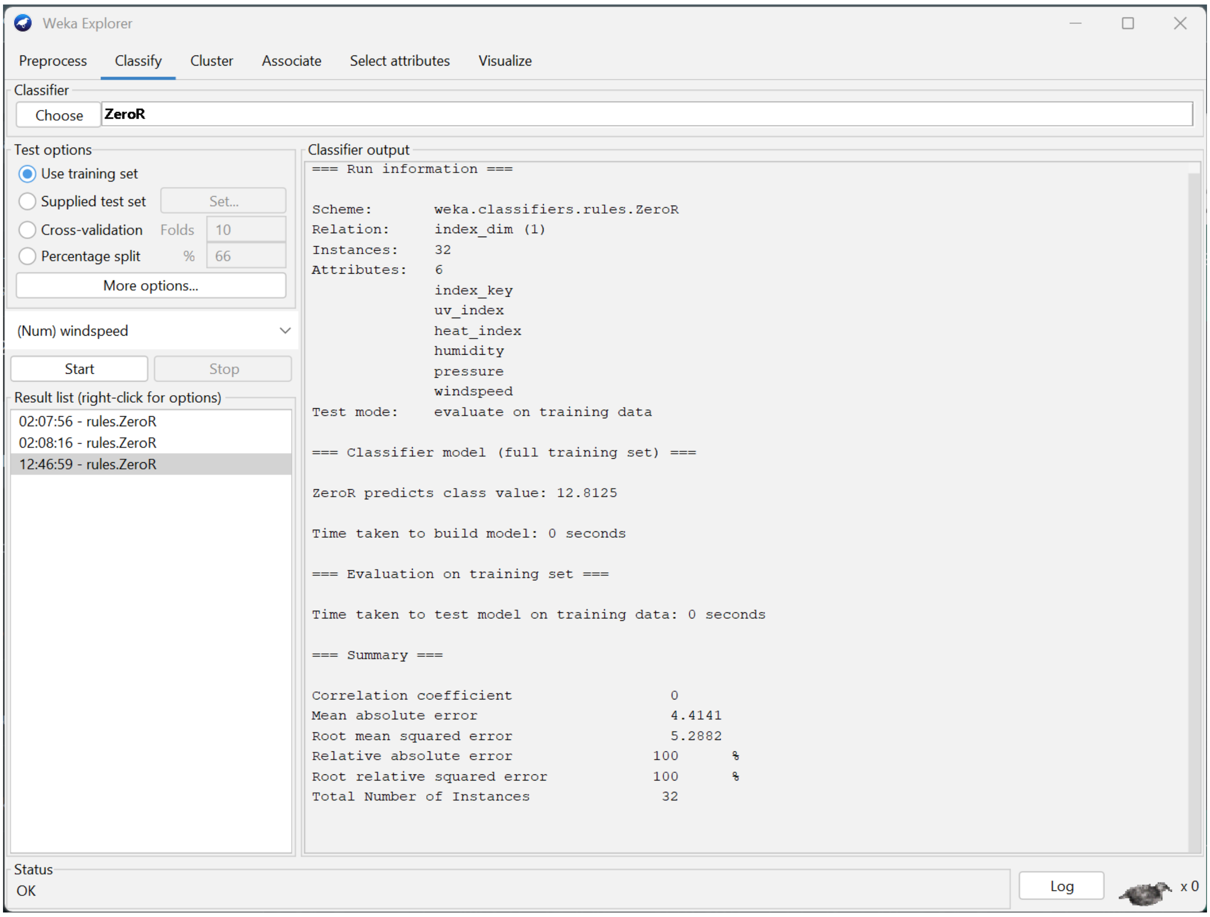
can be done by right clicking the last result set and selecting “visualize tree” from the

pop-up menu.

Step-8: Use the model to classify the new instances.

Step-9: In the main panel under “text “options click the “supplied test set” radio button and then click the “set” button. This will show pop-up window which will allow you to open the file containing test instances.

The following screenshot shows the classification rules that were generated when naive bayes algorithm is applied on the given dataset



ZeroR

● Use training set
○ Supplied test set   Set...
○ Cross-validation   Folds 10
○ Percentage split   % 66

(Num) windspeed

02:07:56 - rules.ZeroR
02:08:16 - rules.ZeroR
12:46:59 - rules.ZeroR

Classifier output

```
=== Run information ===

Scheme:       weka.classifiers.rules.ZeroR
Relation:     index_dim (1)
Instances:    32
Attributes:   6
              index_key
              uv_index
              heat_index
              humidity
              pressure
              windspeed
Test mode:    evaluate on training data

=== Classifier model (full training set) ===

ZeroR predicts class value: 12.8125

Time taken to build model: 0 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0 seconds

=== Summary ===

Correlation coefficient                  0
Mean absolute error                      4.4141
Root mean squared error                  5.2882
Relative absolute error                100      %
Root relative squared error            100      %
Total Number of Instances               32
```

Status
OK

Clustering Using WEKA:

This experiment illustrates the use of simple k-mean clustering with Weka explorer. The sample data set used for this example is based on the iris data available in ARFF format. This document assumes that appropriate preprocessing has been performed. This iris dataset includes 150 instances.

Steps involved in this Experiment

Step 1: Run the Weka explorer and load the data file iris.arff in preprocessing interface.

Step 2: In order to perform clustering select the ‘cluster’ tab in the explorer and click on the choose button. This step results in a dropdown list of available clustering algorithms.

Step 3 : In this case we select ‘simple k-means’.

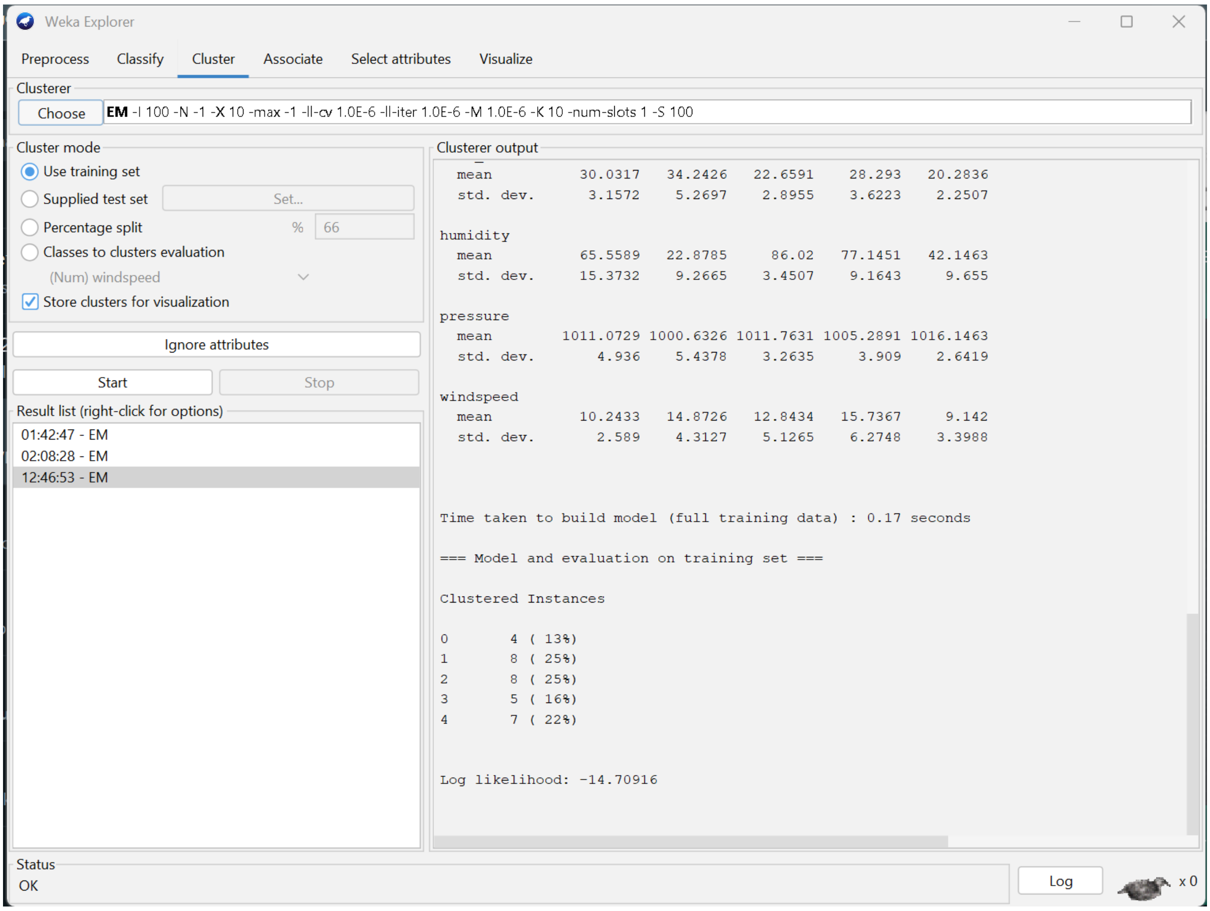
Step 4: Next click in text button to the right of the choose button to get popup window shown in the screenshots. In this window we enter six on the number of clusters and we leave the value of the seed on as it is. The seed value is used in generating a random number which is used for making the internal assignments of instances of clusters.

Step 5 : Once of the option have been specified. We run the clustering algorithm there we must make sure that they are in the ‘cluster mode’ panel. The use of training set option is selected and then we click ‘start’ button. This process and resulting window are shown in the following screenshots.

Step 6 : The result window shows the centroid of each cluster as well as statistics on the number and the percent of instances assigned to different clusters. Here clusters centroid are means vectors for each clusters. This clusters can be used to characterized the cluster. For eg, the centroid of cluster1 shows the class iris.versicolor mean value of the sepal length is 5.4706, sepal width 2.4765, petal width 1.1294, petal length 3.7941.

Step 7: Another way of understanding characterstics of each cluster through visualization ,we can do this, try right clicking the result set on the result. List panel and selecting the visualize cluster assignments.

The following screenshot shows the clustering rules that were generated when simple k means algorithm is applied on the given dataset.



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

Cluster mode

● Use training set

○ Supplied test set

○ Percentage split % 66

○ Classes to clusters evaluation

(Num) windspeed

☑ Store clusters for visualization

Result list (right-click for options)

01:42:47 - EM
02:08:28 - EM
12:46:53 - EM

Clusterer output

```
  mean          30.0317   34.2426   22.6591    28.293   20.2836
  std. dev.      3.1572    5.2697    2.8955    3.6223    2.2507

humidity
  mean          65.5589   22.8785     86.02   77.1451   42.1463
  std. dev.     15.3732    9.2665    3.4507    9.1643     9.655

pressure
  mean        1011.0729 1000.6326 1011.7631 1005.2891 1016.1463
  std. dev.       4.936    5.4378    3.2635     3.909    2.6419

windspeed
  mean          10.2433   14.8726   12.8434   15.7367     9.142
  std. dev.       2.589    4.3127    5.1265    6.2748    3.3988



Time taken to build model (full training data) : 0.17 seconds

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

Clustered Instances

0       4 ( 13%)
1       8 ( 25%)
2       8 ( 25%)
3       5 ( 16%)
4       7 ( 22%)


Log likelihood: -14.70916
```

Status

OK

Association Rule Mining in WEKA:

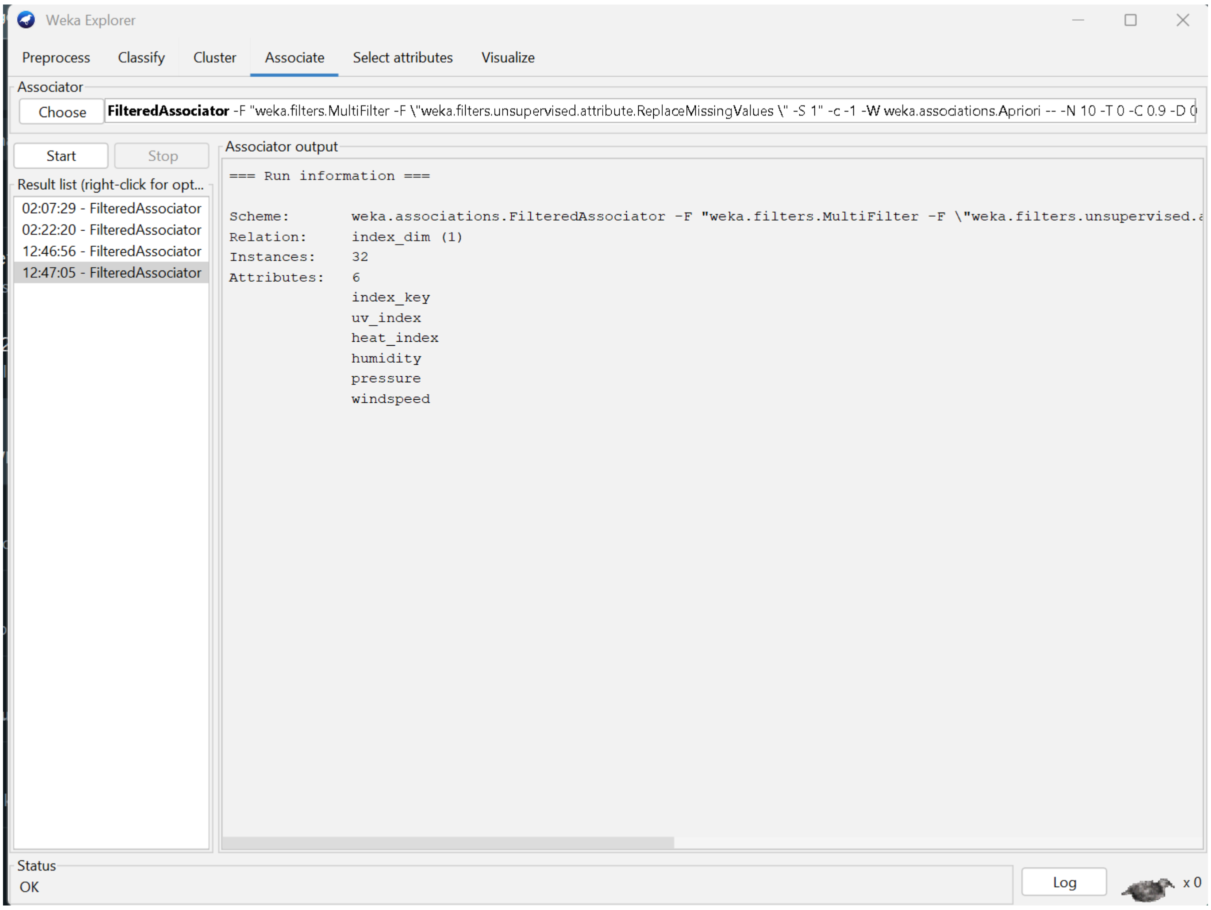
This experiment illustrates some of the basic elements of asscociation rule mining using WEKA. The sample dataset used for this example is test.arff

Step1: Open the data file in Weka Explorer. It is presumed that the required data fields have been discretized. In this example it is age attribute.

Step2: Clicking on the associate tab will bring up the interface for association rule algorithm.

Step3: Use apriori algorithm..

Step4: Inorder to change the parameters for the run (example support, confidence etc) we click on the text box immediately to the right of the choose button.



FilteredAssociator -F "weka.filters.MultiFilter -F \"weka.filters.unsupervised.attribute.ReplaceMissingValues \" -S 1" -c -1 -W weka.associations.Apriori -- -N 10 -T 0 -C 0.9 -D 0

02:07:29 - FilteredAssociator
02:22:20 - FilteredAssociator
12:46:56 - FilteredAssociator
12:47:05 - FilteredAssociator

```
=== Run information ===

Scheme:       weka.associations.FilteredAssociator -F "weka.filters.MultiFilter -F \"weka.filters.unsupervised.a
Relation:     index_dim (1)
Instances:    32
Attributes:   6
              index_key
              uv_index
              heat_index
              humidity
              pressure
              windspeed
```

OK

CONCLUSION:

The different mining algorithms of data mining were studied and the need for association mining algorithm was recognized and understood. Thus, we perform data .Pre-processing task and Demonstrate performing Classification, Clustering, Association algorithm on data sets using data mining using WEKA tool.

SIGN AND REMARK

R1 **(3 M)**	**R2** **(3 M)**	**R3** **(3 M)**	**R4** **(3 M)**	**R5** **(3 M)**	**Total**	**Sign**

DATE