Introduction to Weka- A Toolkit for Machine Learning

1. Introduction

Weka is open source software under the GNU General Public License. System is developed at the University of Waikato in New Zealand. "Weka" stands for the Waikato Environment for Knowledge Analysis. The software is freely available at http://www.cs.waikato.ac.nz/ml/weka. The system is written using object oriented language Java. There are several different levels at which Weka can be used. Weka provides implementations of state-of-the-art data mining and machine learning algorithms. Weka contains modules for data preprocessing, classification, clustering and association rule extraction.

Main features of Weka include:

- 49 data preprocessing tools
- 76 classification/regression algorithms
- 8 clustering algorithms
- 15 attribute/subset evaluators + 10 search algorithms for feature selection.
- 3 algorithms for finding association rules
- 3 graphical user interfaces
  - "The Explorer" (exploratory data analysis)
  - "The Experimenter" (experimental environment)
  - "The KnowledgeFlow" (new process model inspired interface)

1.1 Weka: Download and Installation

- Download Weka (the stable version) from http://www.cs.waikato.ac.nz/ml/weka/
  - Choose a self-extracting executable (including Java VM)
  - (If you are interested in modifying/extending weka there is a developer version that includes the source code)
- After download is completed, run the self extracting file to install Weka, and use the default set-ups.

1.2 Start the Weka

- From windows desktop,
  - click "Start", choose "All programs", Choose "Weka 3.6" to start Weka
  - Then the first interface window appears:
- Weka **GUI Chooser (Fig. 1)**.

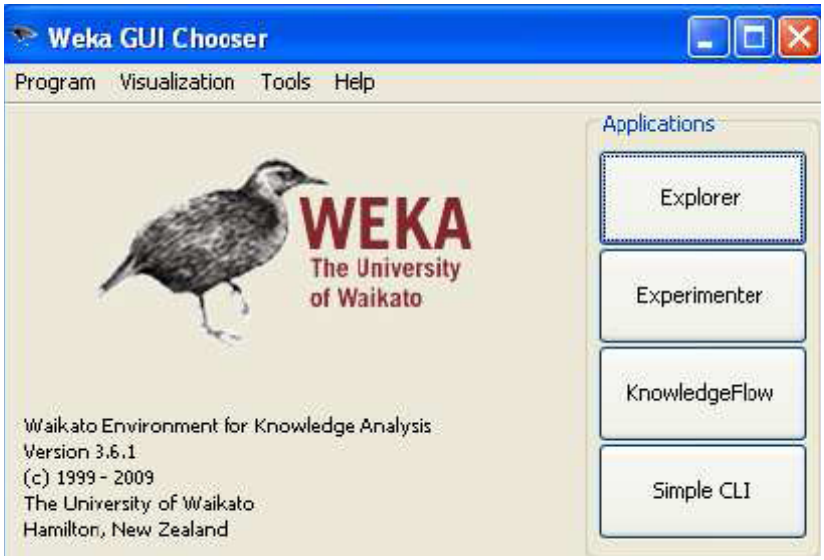

Fig. 1: Weka GUI

1.3 Weka Application Interfaces

- Explorer
  - preprocessing, attribute selection, learning, visualiation
- Experimenter
  - testing and evaluating machine learning algorithms
- Knowledge Flow
  - visual design of KDD process
  - Explorer
- Simple Command-line
  - A simple interface for typing commands

1.4 WEKA data formats

Attribute Relation File Format (ARFF) is the default file type for data analysis in weka but data can also be imported from various formats.

- ARFF (Attribute Relation File Format) has two sections:
  - the Header information defines attribute name, type and relations.
  - the Data section lists the data records.
- CSV: Comma Separated Values (text file)
- Data can also be read from a database using ODBC connectivity.

1.4.1 Attribute Relation File Format (arff)

ARFF format of weather dataset from sample data in weka is presented here. Attribute type is specified in the header tag. Nominal attribute have the distinct values of attribute in curly

brackets along with attribute name. Numeric attribute is specified by the keyword real along with attribute name.

```
@relation weather

@attribute outlook {sunny, overcast, rainy}
@attribute temperature real
@attribute humidity real
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}

@data
sunny,85,85,FALSE,no
sunny,80,90,TRUE,no
overcast,83,86,FALSE,yes
rainy,70,96,FALSE,yes
rainy,68,80,FALSE,yes
rainy,65,70,TRUE,no
overcast,64,65,TRUE,yes
sunny,72,95,FALSE,no
sunny,69,70,FALSE,yes
rainy,75,80,FALSE,yes
sunny,75,70,TRUE,yes
overcast,72,90,TRUE,yes
overcast,81,75,FALSE,yes
rainy,71,91,TRUE,no
```

2. WEKA Explorer

- Click the Explorer on Weka GUI Chooser
- On the Explorer window, click button “Open File” to open a data file from the folder where your data files stored.
- Then select the desired module (Preprocess, Classify, Cluster, Association etc) from the upper tabs.

3. Load Data in Weka from Other File Formats

Weka expects data file it to be in ARFF format, because it is necessary to have type information about each attribute which cannot be automatically deduced from the attribute values. Before you can apply any algorithm to your data, is must be converted to ARFF form. This can be done very easily. Most spreadsheet and database programs allow you to export your data into a file in comma separated format—as a list of records where the items are separated by commas. Once this has been done, you need only load the file into a text editor or a word processor; add the dataset’s name using the `@relation` tag, the attribute information using `@attribute`, and a `@data` line; save the file as raw text. Following example presents conversion of data to arff format from a Microsoft Excel spreadsheet. From the excel spreadsheet, save the data in .CSV format. In weka, On the ***Preprocess*** tab, select ***Open file…***Then select the Dataset.csv file (Fig. 2). Make sure that you have selected files of type csv, or you won’t see the dataset that we want to open.

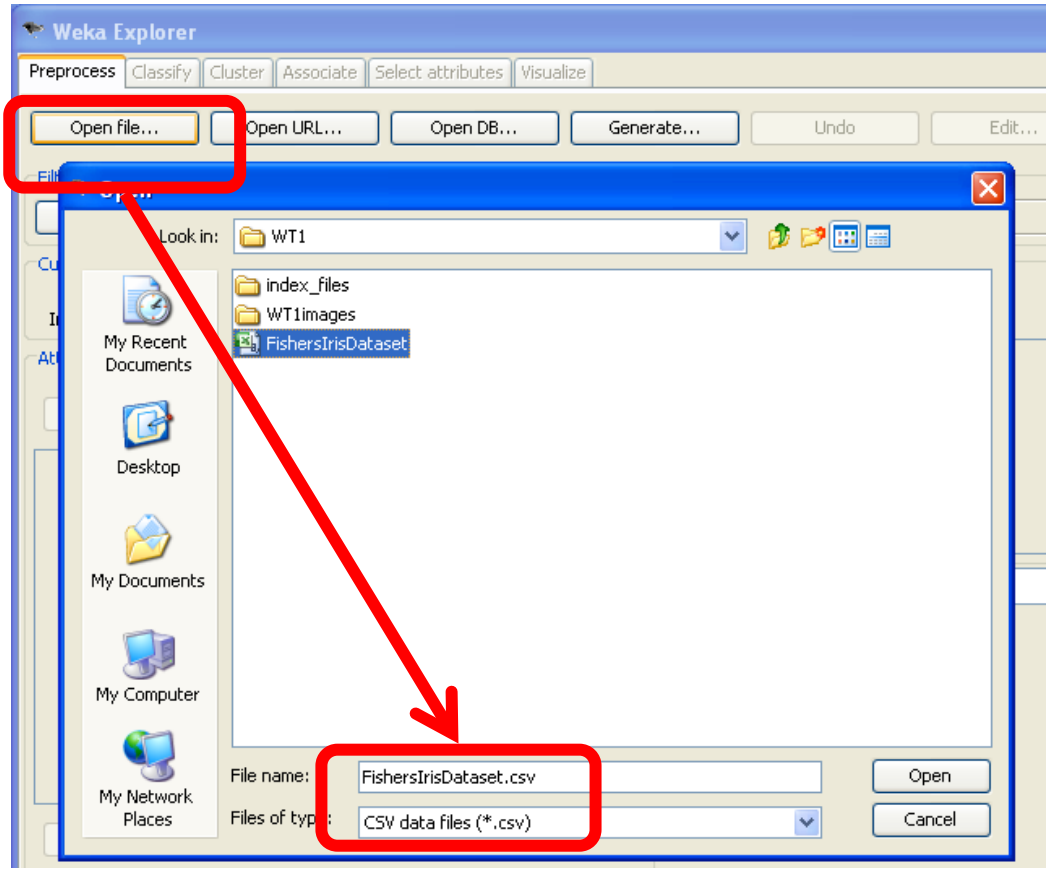

Fig. 2: Open a CSV file

4. Data Preprocessing

Some attributes may not be required in the analysis, and then those attributes can be removed from the dataset before analysis. For example, attribute instance number of iris dataset is not required in analysis. This attribute can be removed by selecting it in the ***Attributes*** check box, and clicking ***Remove*** (Fig. 3). Resulting dataset then can be stored in arff file format.

4.1 Selecting or Filtering Attributes

In case some attributes needs to be removed before the data mining step, this can be done using the Attribute filters in WEKA. 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 4. Next, click on text box immediately to the right of the "Choose" buttom. In the resulting dialog box enter the index of the attribute to be filtered out (this can be a range or a list separated by commas). In this case, we enter 1 which is the index of the "id" attribute (see the left panel). Make sure that the "invertSelection" option is set to false (otherwise everything except attribute 1 will be filtered) (Fig 5). Then click "OK"

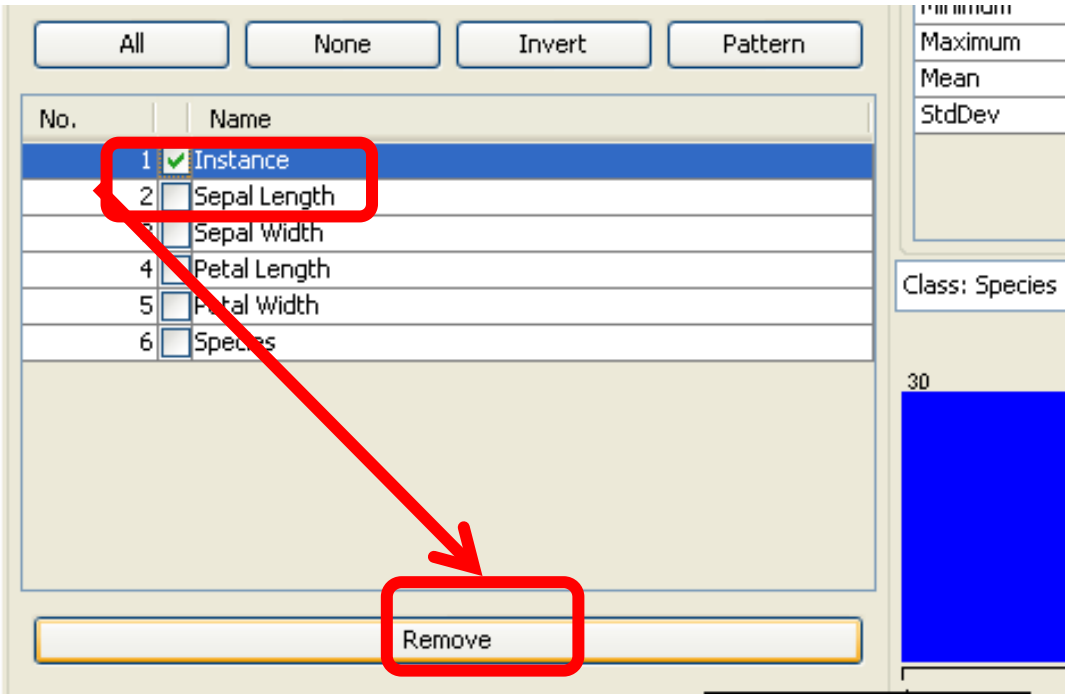

Fig. 3: Remove an attribute

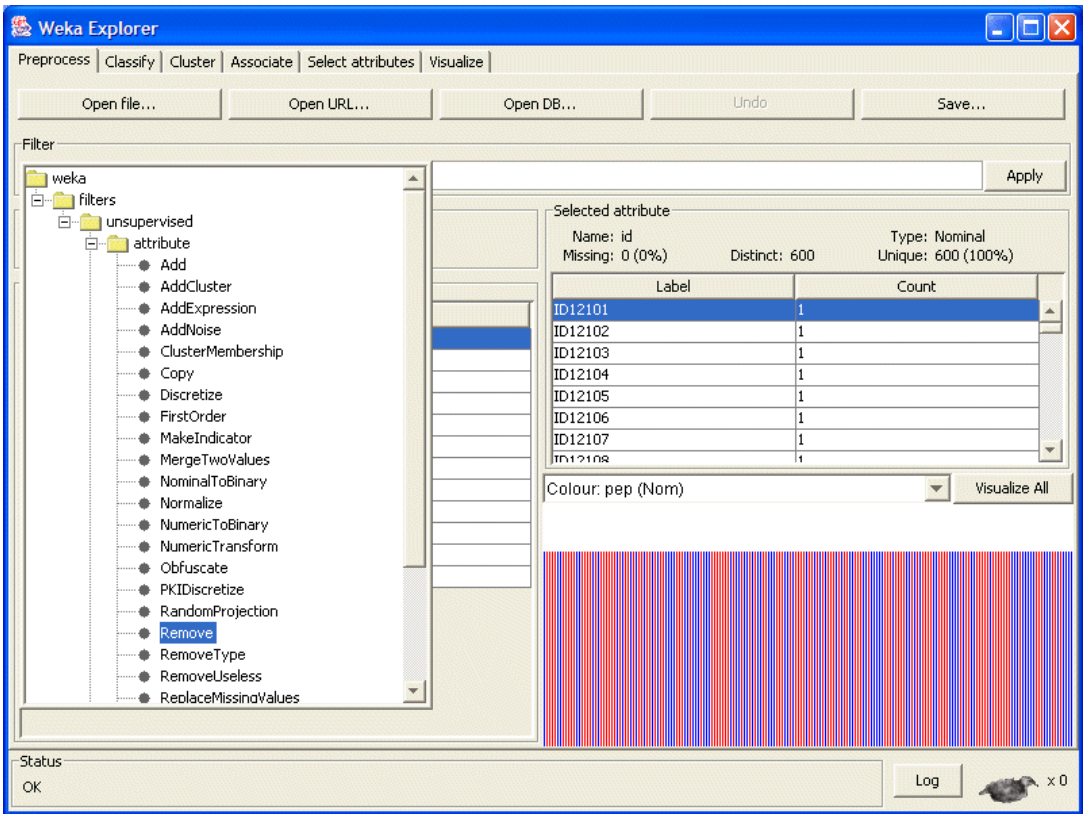

Fig. 4: Filter an attribute

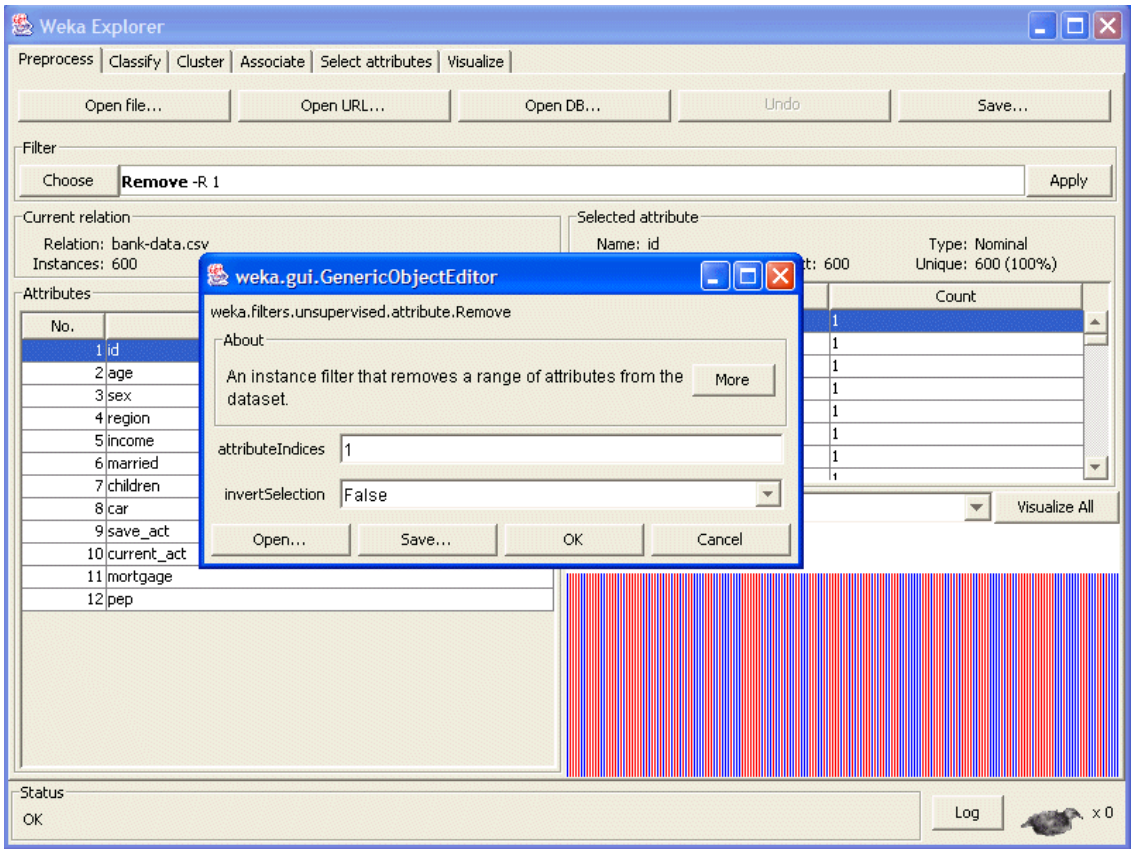

Fig. 5: Options for filtering an attribute

4.2 Discretization

Some techniques require performing discretization on numeric or continuous attributes before applying data mining task. The WEKA discretization filter, can divide the ranges blindly, or used various statistical techniques to automatically determine the best way of partitioning the data. Discretization is represented here with the help of simple binning method. Click the Filter dialog box and select "weka.filters.unsupervised.attribute.Discretize" from the list (Fig. 6). Enter the index for the attributes to be discretized. In this case we enter 1 corresponding to attribute "age". We also enter 3 as the number of bins (note that it is possible to discretize more than one attribute at the same time (by using a list of attribute indeces). Since we are doing simple binning, all of the other available options are set to "false" (Fig 7).

You can observe that WEKA has assigned its own labels to each of the value ranges for the discretized attribute. For example, the lower range in the "age" attribute is labeled "(-inf-34.333333]" (enclosed in single quotes and escape characters), while the middle range is labeled "(34.333333-50.666667]", and so on. These labels now also appear in the data records where the original age value was in the corresponding range.

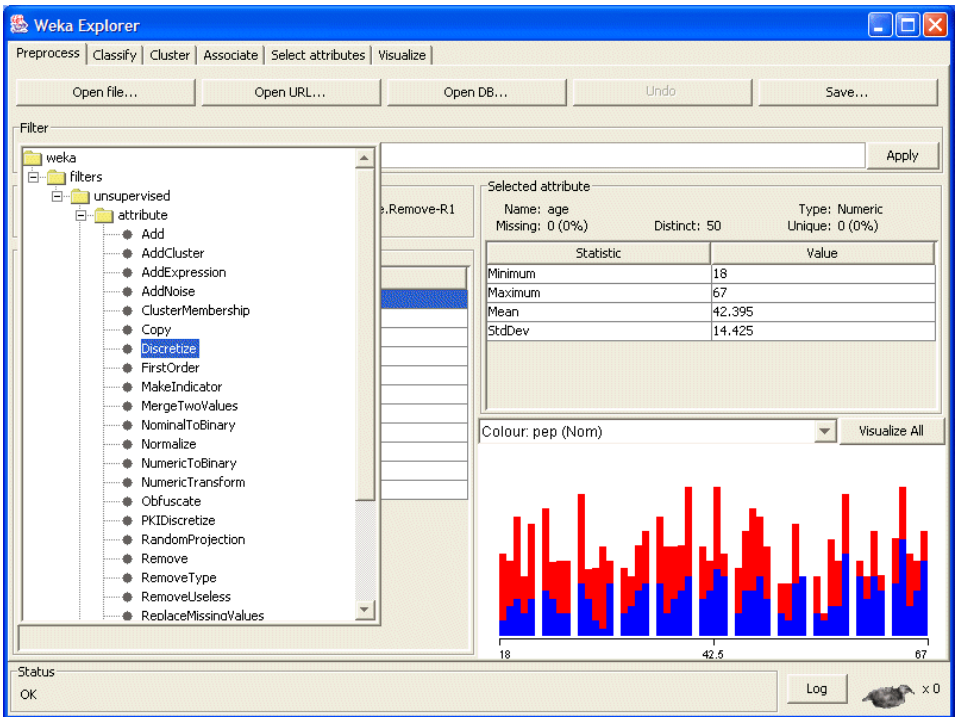

Fig. 6: Discretization Filter

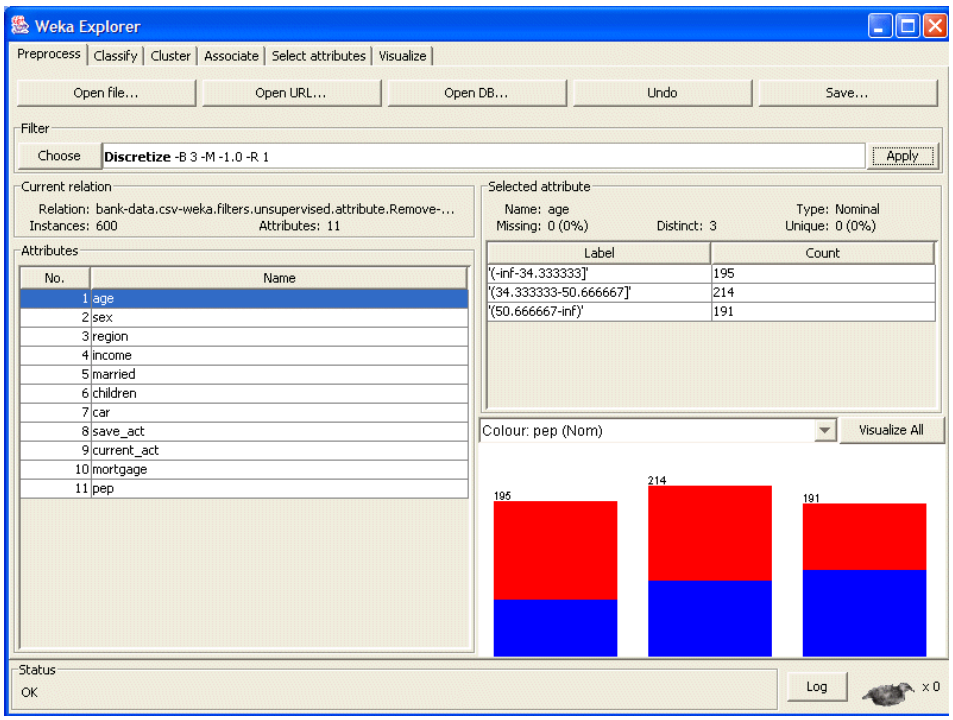

Fig. 7: Discretization options for attribute

5. ID3 Classifier Example with Weka Explorer

Decision Tree is a "divide-and-conquer" approach to the problem of learning from a set of independent instances and leads naturally to a style of representation called a *decision tree*. Nodes in a decision tree involve testing a particular attribute. Usually, the test at a node compares an attribute value with a constant. However, some trees compare two attributes

with each other, or use some function of one or more attributes. Leaf nodes give a classification that applies to all instances that reach the leaf, or a set of classifications, or a probability distribution over all possible classifications. To classify an unknown instance, it is routed down the tree according to the values of the attributes tested in successive nodes, and when a leaf is reached the instance is classified according to the class assigned to the leaf. ID3 is the basic decision tree classifier. Following is the example of ID3 on weather data from sample datasets of weka (Fig. 8).

- Select the Classify tab from the upper tabs.
- There are many classifiers available in the Weka
- Select ID3 from the tree class
- You can select the cross-validation or percentage split of the data
- Other options like selection of variables for analysis
- By default algorithm considers last attribute as class attribute, user can define any other attribute as class attribute too.
- Click on start to run the algorithm.

Interpretation of obtained results:

The first two columns are the TP Rate (True Positive Rate) and the FP Rate (False Positive Rate). For the first level where 'play=yes' TP Rate is the ratio of play cases predicted correctly cases to the total of positive cases (eg: 8 out of 9 is predicted correctly =8/9=0.88).

The FP Rate is then the ratio no play cases incorrectly predicted as play yes cases to the total of play no cases. 1 play no case was wrongly predicted as play yes. So the FP Rate is 1/5=0.2

The next two columns are terms related to information retrieval theory. When one is conducting a search for relevant documents, it is often not possible to get to the relevant documents easily or directly. In many cases, a search will yield lots results many of which will be irrelevant. Under these circumstances, it is often impractical to get all results at once but only a portion of them at a time. In such cases, the terms recall and precision are important to consider.

Recall is the ratio of relevant documents found in the search result to the total of all relevant documents. Thus, higher recall values imply that relevant documents are returned more quickly. A recall of 30% at 10% means that 30% of the relevant documents were found with only 10% of the results examined. Precision is the proportion of relevant documents in the results returned. Thus a precision of 0.75 means that 75% of the returned documents were relevant.

In our example, such measures are not very applicable…the recall in this case just corresponds to the TP Rate, as we are always looking at 100% of test sample and precision is just the proportion of low and normal weight cases in the test sample.
the F-measure is a way of combining recall and precision scores into a single measure of performance. The formula for it is:

2*recall*precision / recall+ precision

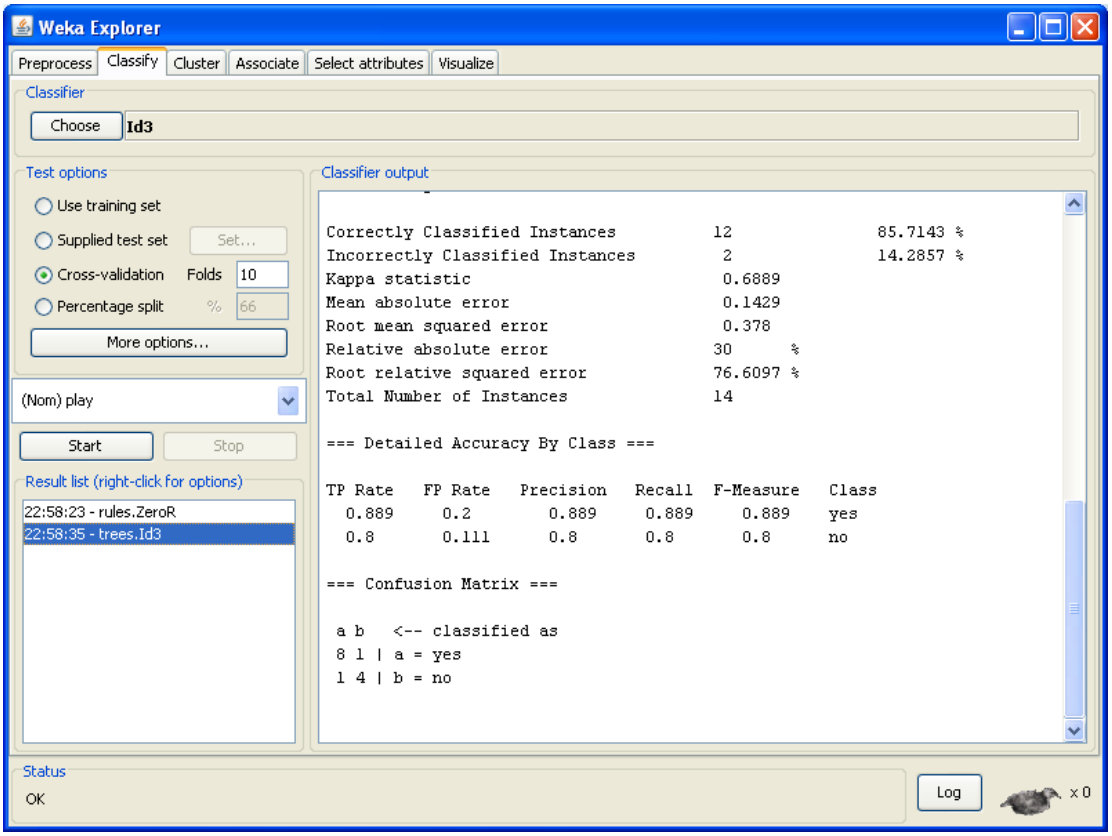

Fig. 8: ID3 algorithm in weka

Confusion matrix specifies the classes of obtained results. For example, class a has majority of objects (8 objects) from yes category, hence a is treated as class of "yes" group. Similarly b has majority of objects (4) from no category, hence b is treated as class of "no" group. Hence one object each from both the classes is misclassified, which leads to misclassified instance as 2. User can see the plot of tree too.

6. K Means Clustering Example with Weka Explorer

K-means is the most popularly used algorithm for clustering. User need to specify the number of clusters (k) in advance. Algorithm randomly selects k objects as cluster mean or center. It works towards optimizing square error criteria function, defined as:

$\sum_{i=1}^{k} \sum_{x \in C_i} \|x - m_i\|^2$, where m_i is the mean of cluster C_i.

Main steps of k-means algorithm are:

1) Assign initial means m_i

2) Assign each data object x to the cluster C_i for the closest mean

3) Compute new mean for each cluster

4) Iterate until criteria function converges, that is, there are no more new assignments.

Following is the example of K means on weather data from sample datasets of weka (Fig. 9).

- Select the Cluster tab from the upper tabs.
- Select Kmeans from the choose tab.
- You can select the attributes for clustering.
- If class attribute is known, then user can select that attribute for "classes to cluster evaluation" to check for accuracy of results.
- In order to store the results, select "Store cluster for visualization"
- Click on start to run the algorithm.
- Right click on the result and select visualize cluster assignment.
- Click on Save button to store the results in arff file format.

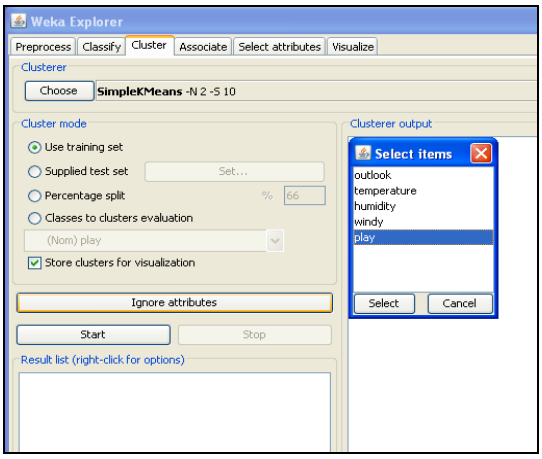

Fig. 9: K means clustering in weka

Figure 10 shows the results of k means on weather data. Confusion matrix specifies the classes of obtained results as we have selected the classes to cluster evaluation. For example, cluster0 has total 9 objects, out of which majority of objects (6) are from yes category, hence this cluster is treated as cluster of "yes". Similarly, cluster1 has total of 5 objects, out of which 3 objects are from "no" category, hence it is considered as cluster of no category.

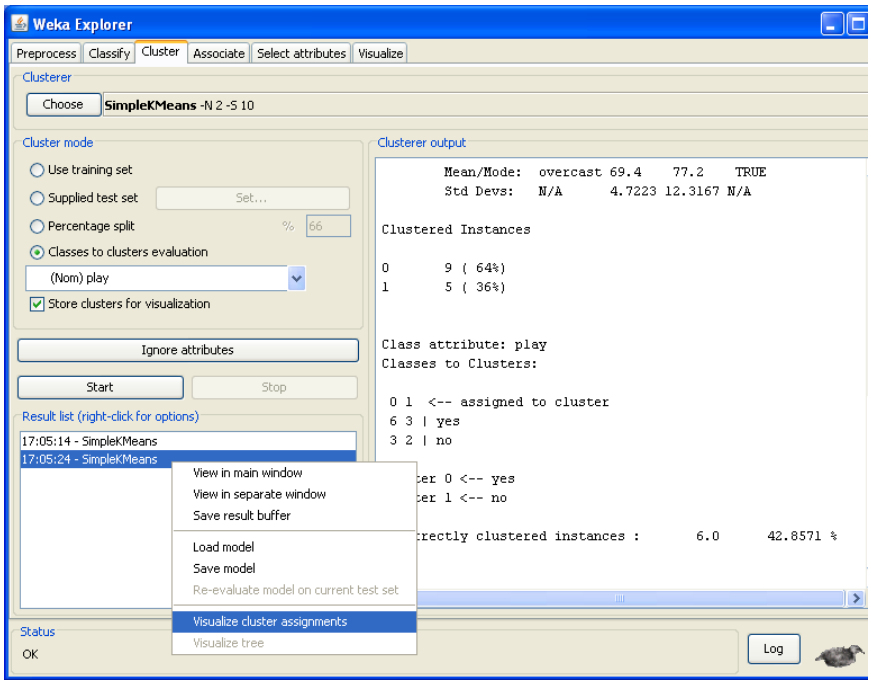

Fig. 10: Results of K means on weather data

References:

Holmes, A. Donkin, I. H. Witten, WEKA: A Machine Learning Workbench, *In Proceedings of the Second Australian and New Zealand Conference on Intelligent Information Systems*, 357-361,1994.

Software, available at: http://www.cs.waikato.ac.nz/~ml/

I. H. Witten, E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementation*, Morgan Kaufmann publishers, 1999.

http://webpages.uncc.edu/~wjiang3/TA/weka/practiceWEKA.pdf

www.cs.waikato.ac.nz/ml/weka/index_documentation.html

http://www.cs.utexas.edu/users/ml/tutorials/Weka-tut/index.htm

http://www.cs.ru.nl/~peterl/teaching/DM/weka.pdf