Practical 2 Week 3

**Introduction to WEKA**

# What are we doing?

* Download an open source machine learning tool “WEKA” and explore the main features of this tool.
* Understand and practice the basic data pre-processing operations that can be performed using WEKA.

**Submission:**

You are required to submit one .arff file (after completing the practical task as instructed in this prac document) via the weekly-practical submission box (available on CP1407 LearnJCU)

# What is WEKA?

The WEKA (The Waikato Environment for Knowledge Analysis) is a machine learning toolkit developed at the University of Waikato in Hamilton, New Zealand. The software provides many machine learning statistics and other data mining solutions for various types of data mining task, such as classification, cluster detection, association rule discovery and attribute selection. The software is also equipped with data pre-processing and post-processing tools and visualisation tools so that complete data mining projects can be conducted via a number of different styles of user interface. The toolkit is written in Java and can, therefore, run on various platforms, such as Linux, Windows and Macintosh. It is an open-source software and distributed under the terms and conditions of the GNU General Public License.

# Launching and Starting WEKA

You are provided two materials (on LearnJCU) for the start of this practical:

* Prac 2 (Week 3) Weka Introduction.ppt
* WekaManual.pdf

Firstly, start with going through the Power Point Slides (“Prac 2 (Week 3) Weka Introduction.ppt”) to get fundamental understanding about the WEKA and its basic features.

Following the instructions guided in the slides, download and install WEKA on your computer. (For internal mode students: your computer lab should have WEKA installed)

For further details, the WEKA manual document (“WekaManual.pdf”) may be helpful. At this stage, Chapter 2, 5, 6, 8 and 9 of the manual are recommended to be read through.

# Data Pre-Processing using WEKA

This example illustrates some of the basic data preprocessing operations that can be performed using WEKA. The sample data set used for this example, unless otherwise indicated, is the "bank data" available in comma-separated format on LearnJCU, called **Bank Data.csv**

The data contains the following fields

|  |  |
| --- | --- |
| **id** | a unique identification number |
| **age** | age of customer in years (numeric) |
| **sex** | MALE / FEMALE |
| **region** | inner\_city/rural/suburban/town |
| **income** | income of customer (numeric) |
| **married** | is the customer married (YES/NO) |
| **children** | number of children (numeric) |
| **car** | does the customer own a car (YES/NO) |
| **save\_acct** | does the customer have a saving account (YES/NO) |
| **current\_acct** | does the customer have a current account (YES/NO) |
| **mortgage** | does the customer have a mortgage (YES/NO) |
| **pep** | did the customer buy a PEP (Personal Equity Plan) after the last mailing (YES/NO) |

### **Loading the Data**

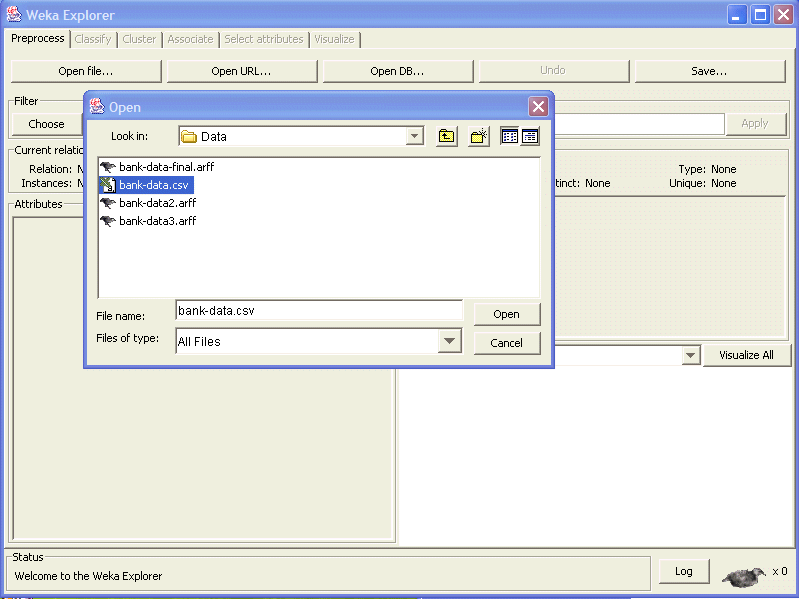
In addition to the native ARFF data file format, WEKA has the capability to read in ".csv" format files. This is fortunate since many databases or spreadsheet applications can save or export data into flat files in this format. A usual Microsoft Excel worksheet can be saved as a CSV file and opened by WEKA. The first row of the spreadsheet is used to name the attributes and the data types for the attributes are derived automatically but not always accurately. Once opened, you can save the data set into an ARFF file in WEKA (by clicking “Save” in the Preprocess tab).

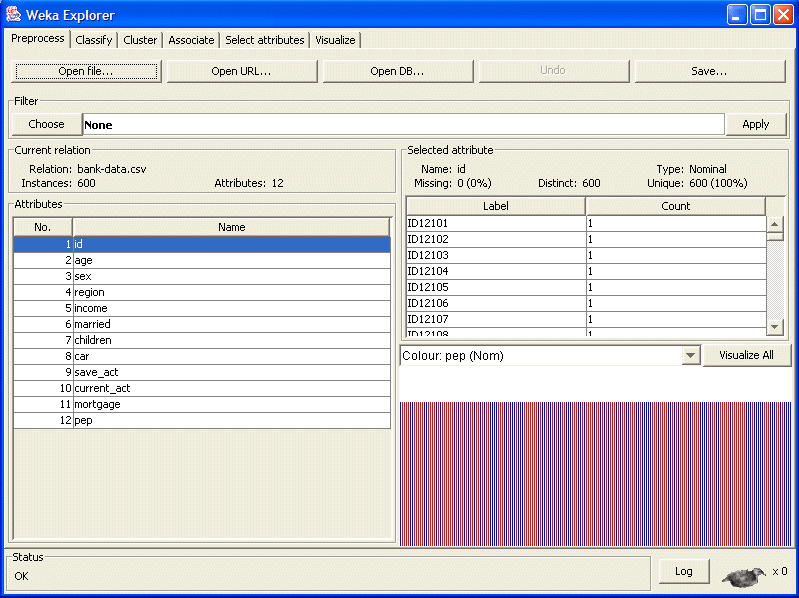
In this example, we load the data set into WEKA, perform a series of operations using WEKA's attribute and discretization filters. While all of these operations can be performed from the command line, we use the GUI interface for WEKA Explorer.

Initially (in the Preprocess tab) click "open" and navigate to the directory containing the data file (which is something like bank-data.csv). This is shown in [Figure 1].

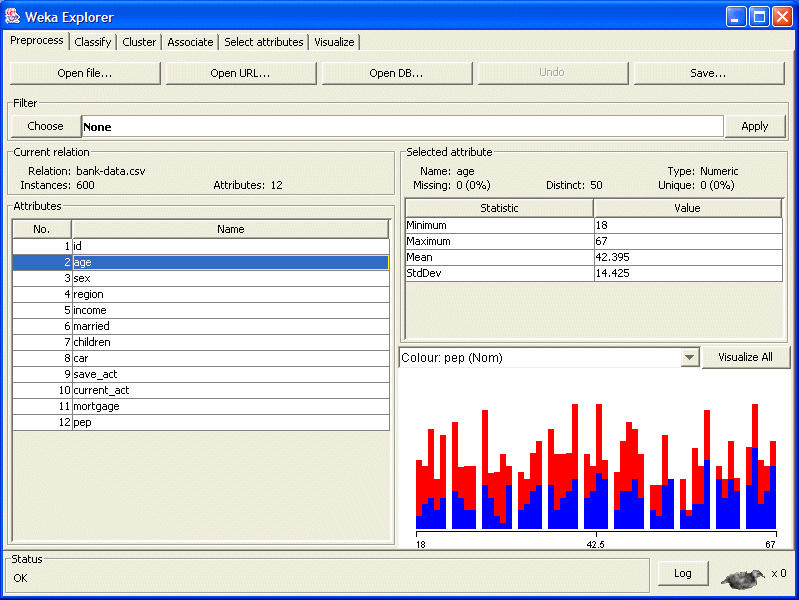
Once the data is loaded, WEKA will recognize the attributes and during the scan of the data will compute some basic statistics on each attribute. The left panel in [Figure 2] shows the list of recognized attributes, while the top panels indicate the names of the base relation (or table) and the current working relation (which are the same initially).

**Note:** The recent version of WEKA has an additional tab named “Edit” under Preprocess menu to view the current contents of the dataset under working. Whenever you apply any filter in WEKA, you can see the updated contents via this viewer facility. (Alternatively, you can use the “Arff Viewer” tool included in WEKA. Refer to the WEKA manual document for further details)

 [Figure 1]

[Figure 2]

Clicking on any attribute in the left panel will show the basic statistics on that attribute. For categorical attributes, the frequency for each attribute value is shown, while for continuous attributes we can obtain min, max, mean, standard deviation, etc. As an example, see the [Figure 3] below which show the results of selecting the “age” attribute.

 [Figure 3]

### **Selecting or Filtering Attributes**

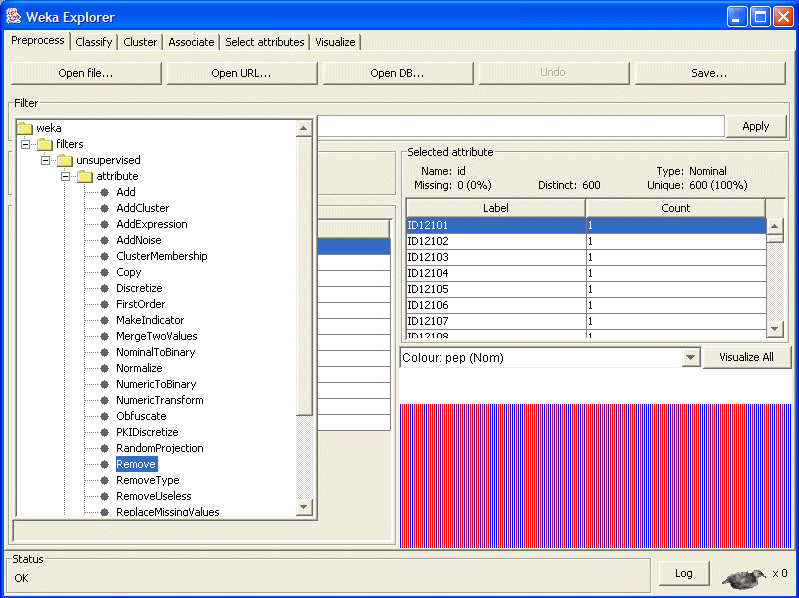
In our sample data file, each record is uniquely identified by a customer id (the "id" attribute). We need to remove this attribute before the data mining step (as this attribute is not necessary). We can do this by using the Attribute filters in WEKA.

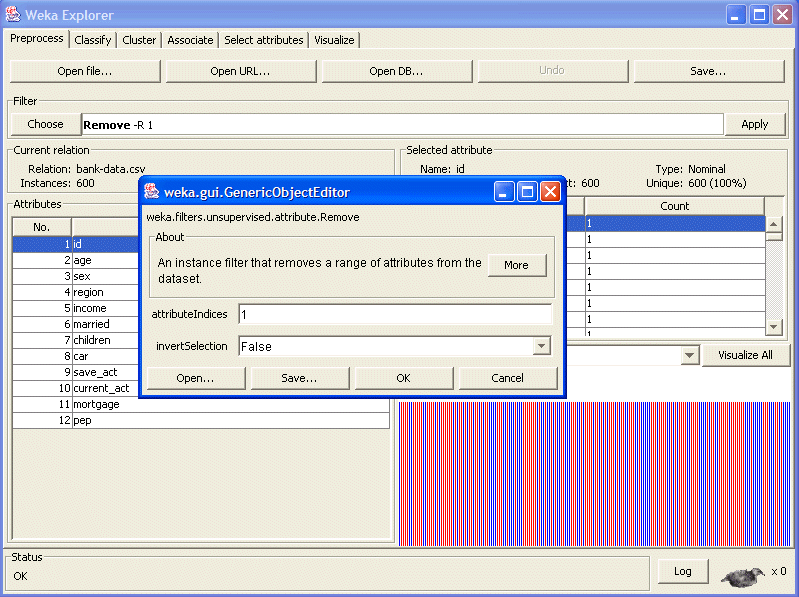
In the "Filter" panel, click on the "Choose" button.

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].

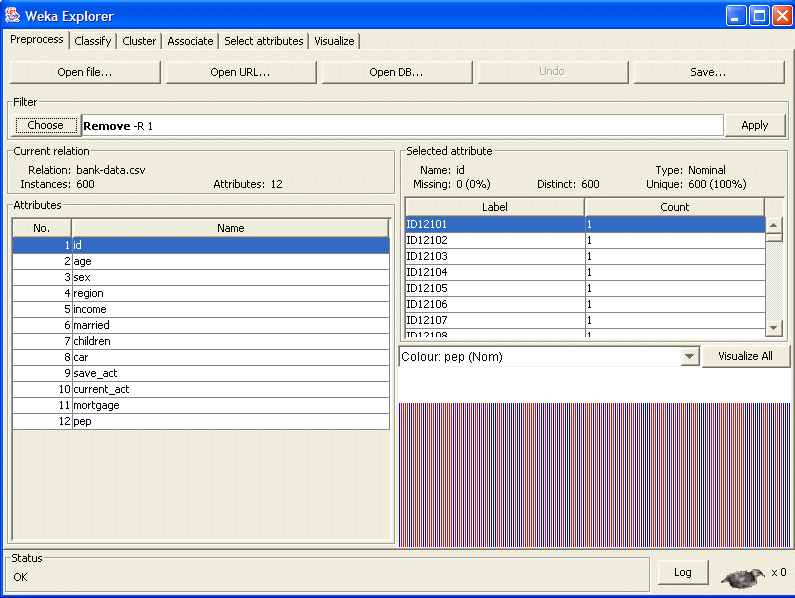
Next, click on text box immediately to the right of the "Choose" button.

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). Then click "OK" (See [Figure 5]). Now, in the filter box you will see "Remove -R 1" (see [Figure 6]).

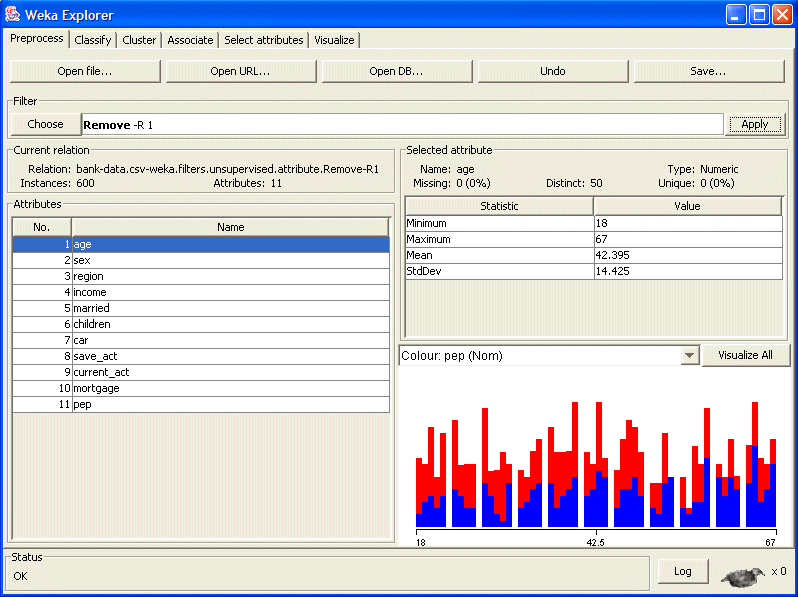
 [Figure 4]

 [Figure 5]

[Figure 6]

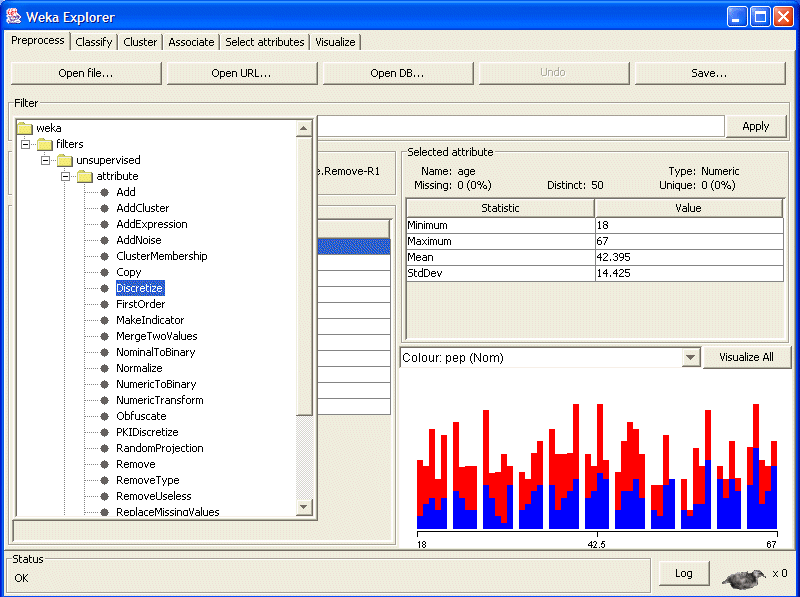


Click the "Apply" button to apply this filter to the data. This will remove the "id" attribute and create a new working relation (whose name now includes the details of the filter that was applied). The result is depicted in [Figure 7].

 [Figure 7]

### **Discretization**

Some techniques, such as association rule mining, can only be performed on categorical data. This requires performing discretization on numeric or continuous attributes. (There are 3 such attributes in this data set: "age", "income", and "children"). Click on the “age” attribute. Again we activate the Filter dialog box, but this time, we will select "Discretize" filter from the list. (see [Figure 8]).

 [Figure 8]

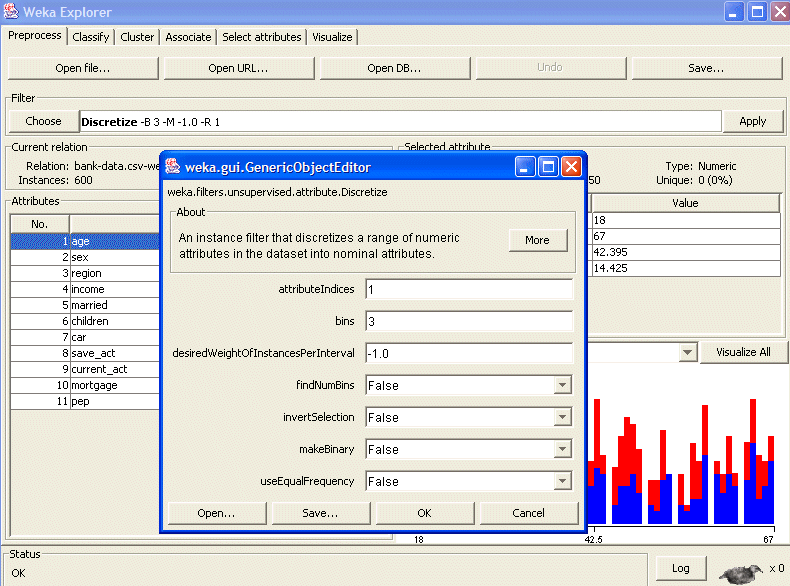
Next, to change the defaults for this filter, click on the box to the right of the "Choose" button. This will open the Discretize Filter dialog box.

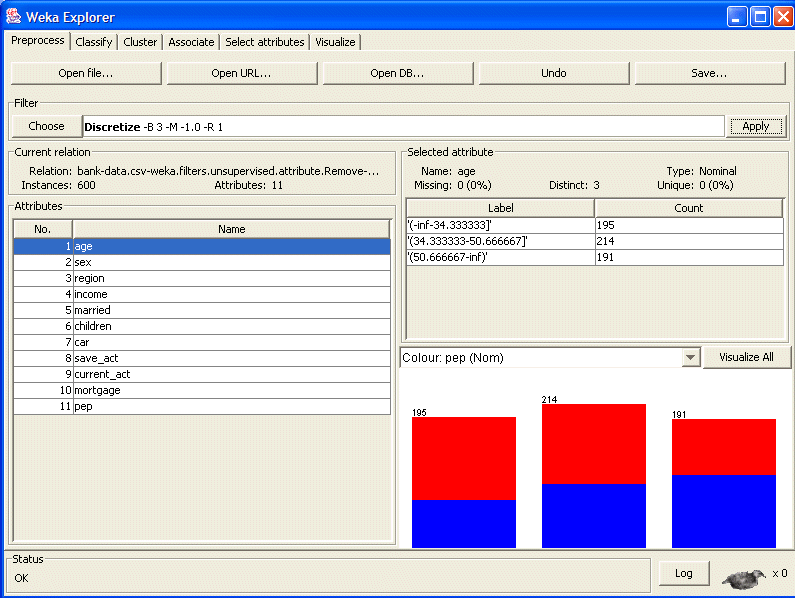
We enter the index for the the attributes to be discretized. In this case we enter 1 corresponding to attribute "age". We also enter 3 as the number of bins (note that it is possible to discretize more than one attribute at the same time (by using a list of attribute indexes). Since we are doing simple binning, all of the other available options are set to "false". The dialog box is shown in [Figure 9].

Click "Apply" in the Filter panel. This will result in a new working relation with the selected attribute partitioned into 3 bins (shown in Figure 10).

Finally, save the file as something like "bank-data-final.arff".

**Submit this final filtered arff file to prove your work for this weekly practical.**

[Figure 9]

[Figure 10]

### **Other Useful Filters in WEKA**

There are more useful preprocessing filters provided in WEKA in addition to filters we tried in this exercise. The following is briefs of some among them. You are recommended to refer to WEKA manual for further details and have a try to apply some to bank data for your own exercise.

In WEKA, data pre-processing is done using attribute or instance filters that can operate supervised or unsupervised. Attribute filters are applied to attributes (columns) and instance filters are applied to data objects (rows). Supervised filters perform with consideration of a class attribute whereas unsupervised filters do not.

(Many unsupervised filters have a supervised counterpart. Supervised filters must be used with care for classification tasks; test examples must be pre-processed in the same way as the training examples.)

The many other filters for data pre-processing have not been described here due to limitations of space. Filters in WEKA are continuously developed and new filters are constantly added in new versions.

**Add attribute filter**

Using “Add” filter, we can create a new attribute (with empty value as default) and specify the location, name and labels of the new attribute. Once created, the value of the new attribute can be entered manually in the viewer window for data objects.

New numeric features can be added with the “AddExpression” filter, which applies a mathematical expression based on the values of other attributes.

**Numeric transformation attribute filters**

The “MathExpression” filter allows transformation with a valid mathematical expression that uses arithmetic operators and built-in functions, such as absolute (abs), logarithm (log), square root (sqrt), etc.

The “NumericTransform” filter only allows transformations by methods supported by the Java math library. Unlike AddExpression, these filters do not create new attributes but replace the current values with the transformed values.

**Transformation attribute filters**

The “Normalize” filter converts the values of all numeric attributes in the loaded data set to those within a common range. The default range is [0.1]. The user can change the normal range if needed.

The “Standardize” filter standardizes all numeric attributes to have zero mean and unit variance.

**ReplaceMissingValues filter**

This rudimentary filter fills in missing values; numeric values are replaced with the sample mean and nominal values are replaced with the sample mode. The user can also fill in missing values manually in the viewer window (using “Edit” menu). For numeric attributes, the user may enter any value. For nominal attributes, the user can only select one of the nominal labels that already exists in the attribute domain. If the label does not exist (for instance, it is a special code indicating unknown), the label can be added into the attribute domain by using “AddValues” filter.

**Resample instance filter**

This filter selects a random sample of a certain percentage (SampleSizePercent parameter) of the loaded data set, with or without replacement (to sample without replacement, set the noReplacement parameter to True). The unsupervised Resample filter draws the sample from the entire data set reflecting the real distribution of attribute values including class values; the supervised Resample filter draws samples according to either the real distribution of classes (set the biasToUniformClass parameter to 0) or a uniform distribution of classes (set the biasToUniformClass parameter to 1).