DEPAUL UNIVERSITY Computer Science, Telecommunications, and Information Systems

ECT 584 - Web Data Mining

Data Preprocessing in WEKA

The following guide is based WEKA version 3.4.1 Additional resources on WEKA, including sample data sets can be found from the official WEKA Web site.

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 (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. As can be seen in the sample data file, the first row contains the attribute names (separated by commas) followed by each data row with attribute values listed in the same order (also separated by commas). In fact, once loaded into WEKA, the data set can be saved into ARFF format. If, however, you are interested in conveting a ".csv" file into WEKA's native ARFF using the commandline, this can be accomplished using the following command:

```
java weka.core.converters.CSVLoader filename.csv > filename.arff
```

In this example, we load the data set into WEKA, perform a series of operations using WEKA's attribute and discretization filters, and then perform association rule mining on the resulting data set. 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 (.csv or .arff). In this case we will open the above data file. This is shown in Figure p1.

Figure p1

Since the data is not in ARFF format, a dialog box will prompt you to use the convertor, as in Figure p2. You can click on "Use Covertor" button, and click OK in the next dialog box that appears (See Figure p3).

Figure p2
Figure p3

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 p4 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).

Figure p4

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 Figures p5 and p6 below which show te results of selecting the "age" and "married" attributes, respectively.

Figure p5
Figure p6

Note that the visualization in the right bottom panel is a form of cross-tabulation across two attributes. For example, in Figure p6 above, the default visualization panel cross-tabulates "married" with the "pep" attribute (by default the second attribute is the last column of the data file). You can select another attribute using the drop down list.

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. 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 p7.

Figure p7

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). Then click "OK" (See Figure p8). Now, in the filter box you will see "Remove -R 1" (see Figure p9).

Figure p8
Figure p9

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 p10:

Figure p10

It is possible now to apply additional filters to the new working relation. In this example, however, we will save our intermediate results as separate data files and treat each step as a separate WEKA session. To save the new working relation as an ARFF file, click on save button in the top panel. Here, as shown in the "save" dialog box (see Figure p11), we will save the new relation in the file "bank-data-R1.arff".

Figure p11

Figure p12 shows the top portion of the new generated ARFF file (in TextPad).

Figure p12

Note that in the new data set, the "id" attribute and all the corresponding values in the records have been removed. Also, note that Weka has automatically determined the correct types and values associated with the attributes, as listed in the Attributes section of the ARFF file.

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". In the case of the "children" attribute the range of possible values are only 0, 1, 2, and 3. In this case, we have opted for keeping all of these values in the data. This means we can simply discretize by removing the keyword "numeric" as the type for the "children" attribute in the ARFF file, and replacing it with the set of discrete values values. We do this directly in our text editor as seen in Figure p13. In this case, we have saved the resulting relation in a separate file "bank-data2.arff".

Figure p13

We will rely on WEKA to perform discretization on the "age" and "income" attributes. In this example, we divide each of these into 3 bins (intervals). The WEKA discretization filter, can divide the ranges blindly, or used various statistical techniques to automatically determine the best way of partitioning the data. In this case, we will perform simple binning.

First we will load our filtered data set into WEKA by opening the file "bank-data2.arff". The "open" dialog box in depicted in Figure p14.

Figure p14

If we select the "children" attribute in this new data set, we see that it is now a categorical attribute with four possible discrete values. This is depicted in Figure p15.

Figure p15

Now, once again we activate the Filter dialog box, but this time, we will select "weka.filters.unsupervised.attribute.Discretize" from the list (see Figure p16).

Figure p16

Next, to change the defaults for this filter, click on the box immediately 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 indeces). Since we are doing simple binning, all of the other available options are set to "false". The dialog box is depicted in Figure p17.

Figure p17

Click "Apply" in the Filter panel. This will result in a new working relation with the selected attribute partitioned into 3 bins (see Figure p18). To examine the results, we save the new working relation in the file "bank-data3.arff" as depicted in Figure p19.

Figure p18
Figure p19

Let us now examine the new data set using our text editor (in this case, TextPad). The top portion of the data is shown in Figure p19. 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.

Next, we apply the same process to discretize the "income" attribute into 3 bins. Again, Weka automatically performs the binning and replaces the values in the "income" column with the appropriate automatically generated labels. We save the new file into "bank-data3.arff", replacing the older version.

Clearly, the WEKA labels, while readable, leave much to be desired as far as naming conventions go. We will thus use the global search/replace functions in TextPad to replace these labels with more succinct and readable ones. Fortunately, TextPad has a powerful regular expression pattern matching capability which allows us to do this efficiently. Figure p20 shows the TextPad search/replace dialog box for replacing the age label "(-inf-34.333333]" with the label "0_34". Note that the "regular expression" option is selected. In the "Find what" box we have entered the full label '\'(-inf-34.333333]\" (including the back-slashes and single quotes). Furthermore, back-slashes are escaped with another back-slash so that in the regular expression patterns matching they are treated as literals (resulting in: '\\'(-inf-34.333333]\\". In the "Replace with" box we enter "0_34".

Figure p20

Now we click on the "Replace All" button to replace all instances of the old patterns with the new one. The result of this operation is depicted in Figure p21.

Figure p21

Note that the new label now appears in place of the old one both in the attribute section of the ARFF file as well as in the relevant data records. We repeat this manual re-labeling process with all of the WEKA-assigned labels for the "age" and the "income" attributes. Figure p22 shows the final result of the transformation and the newly assigned labels for these attribute values.

Figure p22

We now also change the relation name in the ARFF file to "bank-data-final" and save the file as "bank-data-final.arff".

Return to Main Page

Copyright © 2005-2006, *Bamshad Mobasher*, School of CTI, DePaul University.