CO544 Machine Learning and Data Mining Labs

Classification, Prediction, Clustering and Association Learning

Part 1: Classification using WEKA

Aim: The aim of this part of the lab is to provide students hand on experience in classification using WEKA data mining toolkit.

Objectives : At the end of the lab, students should be able to

- Perform simple preprocessing tasks using filters available in WEKA.
- Use various classification algorithms with different test options for a given classification problem.
- Analyze the output of classification algorithm and interpret the results.

1. Load the **Zoo** dataset. Observe attributes and their values.

2. Build the C4.5 decision tree (`J48` in WEKA) using default parameters and test options. Observe the output of the algorithm.

3. Visualize the output of C4.5 by right-clicking on the experiment in the result list and then choosing the **Visualize tree** option. Explore different error estimates and record the classification accuracy of C4.5 algorithm. Examine the true positive (TP) rates, the false positive (FP) rates and the confusion matrix. Explain misclassifications observed in the confusion matrix.

4. Evaluate C4.5 algorithm using the following testing options:
   a) the training set and b) 10-fold cross validation.
Record the classification accuracies using both the methods. Which one provides more realistic future performance? Why?

5. Can you apply **ID3** learning algorithm on this dataset? Explain your answer.

6. Remove **Instances, animal name** and **legs** attributes from the dataset using **Remove** filter available in **preprocess** tab. Left click on **Remove** and insert attribute indices to remove from the dataset. You can save the modified dataset using **Save** button.

7. Build the **ID3** decision tree. Examine the output. Record the 10-fold cross validation accuracy.

8. Use **OneR** algorithm and explain the classifier output. Record the 10-fold cross validation accuracy.

9. Use another classification algorithm of your choice and observe the output of the algorithm. Compare the results of the chosen algorithm with previous outputs.

Part 2: Predicting Class Values

Aim: The aim of this part of the lab is to provide students hand on experience in using WEKA data mining toolkit to predict the given dataset according to the training dataset.

Objectives: At the end of the lab, students should be able to

- Create a model learned under the classification learning using a training data set and make predictions for a test set.
- Analyze the output of model and interpret the results.

In WEKA tool, after a model has been learned under the classification learning, one can make predictions for a test set, whether that set contains valid class values or not. If the class values are present in a test set the output will contain both the actual and predicted class values. If the class values are missing from a test set, the actual class label for each instance will not contain useful information, but the predicted class label will. We will demonstrate how to do class prediction using WEKA for a test set (which includes the class values) using ***zoo_train.arff*** and ***zoo_test.arff*** files.

The steps are as follows.

1. Load the ***zoo_train.arff*** data set. Observe the attributes and their values.

2. Build the C4.5 decision tree (J48 in WEKA) with '**Use training set**' test option.

3. Select '**Supplied test set**' test option. Click on the '**Set...**' button to the right side of that option. Then a separate window named '**Test Instances**' will pop out. Click on '**Open file...**' button and browse and select the ***zoo_test.arff*** file. Then close the 'Test Instances window'.

4. Click on '**More options...**' button. Then a separate window named '**Classifier evaluation options**' will pop out. Select '**Output predictions option**' and click '**OK**'. Now you are ready to predict the class values for test instances.

5. Right click on the result buffer in the '**Result list**', which corresponds to your model and select the option '**Re-evaluate model on current test set**' option from the dropdown list. After this, the classifier output will contain the predictions for separate test instances together other useful information respective to the re-evaluation. A part of the predictions is illustrated in **Figure 1**.

```
=== Predictions on test split ===

inst#,     actual, predicted, error, probability distribution
     1 7:inverteb 7:inverteb          0      0      0      0      0      0.125 *0.875
     2     4:fish     4:fish          0      0      0     *1      0      0      0
     3     2:bird     2:bird          0     *1      0      0      0      0      0
     4   1:mammal   1:mammal         *1      0      0      0      0      0      0
     5 7:inverteb 7:inverteb          0      0      0      0      0      0.125 *0.875
     6     4:fish     4:fish          0      0      0     *1      0      0      0
     7     2:bird     2:bird          0     *1      0      0      0      0      0
     8   6:insect 7:inverteb      +   0      0      0      0      0      0.125 *0.875
     9 5:amphibia 5:amphibia          0      0      0      0     *1      0      0
    10  3:reptile 7:inverteb      +   0      0      0      0      0      0.125 *0.875
    11  3:reptile 5:amphibia      +   0      0      0      0     *1      0      0
    12     4:fish     4:fish          0      0      0     *1      0      0      0
    13   1:mammal   1:mammal         *1      0      0      0      0      0      0
    14   1:mammal   1:mammal         *1      0      0      0      0      0      0
    15     2:bird     2:bird          0     *1      0      0      0      0      0
    16   1:mammal   1:mammal         *1      0      0      0      0      0      0
    17   6:insect   6:insect          0      0      0      0      0     *1      0
    18   1:mammal   1:mammal         *1      0      0      0      0      0      0
    19 7:inverteb 7:inverteb          0      0      0      0      0      0.125 *0.875
    20     2:bird     2:bird          0     *1      0      0      0      0      0
```

Figure 1

You can interpret the above results as follows.

- First column - instance numbers.
- Second column - actual class value of each test instance.
  If class is not present a '?' symbol will be displayed.
  If class is present the class number with its class value will be displayed. For example in the first instance actual class value is displayed as '7:inverteb'. This means instance one is predicted to be of class 7, whose value is invertebrate (abbreviated as inverteb).
- Third column - predicted class value of each test instance.
- Fourth column - whether the predicted class value mismatches with the actual class value.
- If it is an error a '+' sign will be displayed. Else nothing will be displayed.
- Fifth column - estimation of probabilities for each instance actually belongs to a class. For example in the first instance there are probabilities 0, 0, 0, 0, 0, 0.125, 0.875 where each value estimates the probability of first instance actually belongs to classes 1 , 2, 3, 4, 5, 6, 7 respectively. The '*' sign represents the highest probability.

Note: Here we first built the model using the train set and used it for predictions at the same time. If you want, you can build the model and save it for future uses. Then load it at a later time for predictions.

Exercise:

1. Observe the output of the algorithm with the training set. Explore different error estimates and record the percentages of classifications and misclassifications.2. Observe the output of the algorithm with the test set. Explore different error estimates and record the percentages of classifications and misclassifications.

3. Comment on the two results you obtained in 1 and 2 above.

4. Do the predictions using ***zoo_test_classmissing.arff*** as the test set. This file has the same data set with missing class values. Comment on your results with respect to the results you obtained in step 2 above.

Part 3: Clustering

Aim: The aim of this part of the lab is to provide students hands-on experience in clustering using WEKA data mining toolkit.

Objectives: At the end of the lab, students should be able to

- Use the **K-Means** algorithm for a given dataset in WEKA.
- Analyze the output of **K-Means** algorithm and interpret the results.

1. Load the '**iris.arff**' data set into the WEKA tool and observe the attributes and their data types. The dataset is intended to use for classification. Let's make the dataset into an unlabeled dataset by removing the class attribute from the dataset. Use the preprocess tab remove the attribute '**class**' from the dataset.

2. Goto '**Cluster**' tab and choose '**SimpleKMeans**' as the algorithm under the '**Cluster**' panel. Left click on the '**SimpleKMeans**' label and obtain the '**Generic Object Editor**' window. Click the '**More**' button and identify what is indicated by each parameter in the '**Generic Object Editor**'.

3. Briefly describe what is meant by the term '**seed**' in the '**Generic Object Editor**'. Describe the use of seed with the KMeans algorithm.

4. Input the number of clusters as 2 and keep the other fields with default values. In the cluster mode panel select the option '**Use training set**'. Apply the **SimpleKMeans** with the above values. Observe the cluster assignments and describe the values in each cluster. Record the sum of squared error and the proportions of instances assigned to each cluster.

5. Right click on the result list and select '**Visualize cluster assignments**' option from the drop down list. Choose suitable labels for '**X**', '**Y**' and '**Colour**' fields and try to discover a description for each cluster. Briefly describe your observations.

6. To save each instance along with its assigned cluster, click '**Save**' button in the visualization window and provide the file name as '**iris-kemeans-noofclusters-2.arff**'. Briefly describe the contents of the ARFF file.

7. Repeat the above process for different values of k ($2 \leq k \leq 5$) and suggest a suitable value for k. Justify your answer.

8. Now reload the '**iris.arrf**' file into the WEKA tool. Go to '**Cluster**' tab. Select the '**Classes to clusters evaluation**' option under the '**Cluster mode**' panel and select the '**(Nom)class**' option from the drop down list. In the '**Generic Object Editor**' window keep the default values. For the value of the '**numClusters**' input the value of k which you decided in step 7 above and apply '**SimpleKMeans**' algorithm. Observe the results and briefly comment on the results with respect to assigned class value for each cluster.

Part 4: Association Rule Learning

Aim: The aim of this part of the lab is to provide students hand on experience in association rule mining using WEKA data mining toolkit.

Objectives: At the end of the lab, students should be able to

- Use the **Apriori algorithm** in WEKA to discover association rules.
- Analyze the output of **Apriori algorithm** and interpret the results.

1. Load the '**mini_supermarket.arff**' dataset into the WEKA tool and observe each transaction and its respective attribute values.

2. Go to the '**Associate**' tab and choose '**Apriori**' as the algorithm under the '**Associator**' panel. Left click on the '**Apriori**' label and obtain the '**Generic Object Editor**' window. Click the '**More**' button and identify what is indicated by each parameter in the '**Generic Object Editor**'.

3. Briefly describe *lowerBoundMinSupport*, *upperBoundMinSupport*, *delta*, *numRules* and **Confidence** *metricType* parameters in the '**Generic Object Editor**'.

4. Apply the '**Apriori**' learner with the default values. Describe the meaning of each rule with its corresponding support and confidence.

5. Apply the '**Apriori**' learner with the two different numbers of rules: one less and one greater than the default number of rules. Comment on the three different set of rules.

6. Suppose you are working in a supermarket which sells the above set of products and your company wants to increase the sales of the product milk. Using '**Apriori**' build a set of rules that predict in which situations the product milk will be sold or not. From the rule set select the best way to arrange the placement of the products in the supermarket to increase the sales of the milk. (Hint: set appropriate parameters for *car* and *classIndex*)

7. Now let us examine a real-world dataset '**supermarket.arff**', which is provided with the WEKA tool. This dataset has been collected from an actual New Zealand supermarket. Take a look at this file using a text editor to verify that you understand the structure. The main point of this section is to show you how difficult it is to find any interesting patterns in this type of data. Experiment with '**Apriori**' and learn a suitable set of association rules. Write a brief description of the main findings of your investigation.

Submission:

Compile reports for each part separately (clearly mention the part number). Name the report as `e17xxx`_wekalab_partX.pdf (here XXX is your registration number using 3 digits and X is part number) and submit to the given link in FEeLS.