Practical 5: Introduction to Weka for Classification
Nadarajen Veerapen and Gabriela Ochoa

Introduction to Weka

1. Download weather.nominal.arff. A small dataset with attributes describing weather conditions, and a decision of whether it is desirable to play outdoor or not.
2. Open Weka and choose Explorer.
3. Load weather.nominal.arff (Open file... button)
4. Have a look at the different attributes. In *Current relation*, we can see that there are 14 instances and 5 attributes in the dataset. Click on each attribute to see its properties in *Selected attribute* and a graph of the distribution of the values of the attribute. The colours in the graph each correspond to a class. Pay attention to the type of the attributes. In this dataset all the attributes are *Nominal*: the values indicate different distinct categories that describe the attribute. An attribute could also be *Numeric*: the values are numbers that measure the attribute. Notice that *play* has been suggested as the class attribute, that is the one that is predicted from the other attributes.

5. Have a look at the data (*Edit…* button). We can see all the data in this window. Each of the rows corresponds to an instance and the columns are the attributes.

6. We are now going to build a decision tree. On the *Classify* tab, the default classifier is *ZeroR* so click on *Choose* to select the *Id3* classifier from the trees.
**ID3** is one of the simplest decision tree classifiers. Clicking on the classifier name text box, in this case *Id3*, will bring up a window providing a very short description of the classifier. Click on *More* for a bit more details and on *Capabilities* to know the kinds of attributes and classes the classifier can handle.

This information tells us that the ID3 algorithm can only handle nominal attributes and cannot deal with missing values. We can therefore apply it to our data. Note that classifiers that are not compatible with the data are greyed out and cannot be selected.

7. The **Test options** allow us to choose how to train the classifier.
   a. *Use training set* will use all the data for both the training and the test sets.
   b. *Supplied test set* allows you to provide a separate test set.
   c. *Cross-validation* will perform cross-validation according to the number of folds provided. This means that the data will be split into *k* subsets of equal size. For each value of *i* in \{1, 2, ..., *k*\} the classifier will be tested on the *i*th subset after being trained on all the other data. The *k* results are then averaged to describe the performance of the classifier.
   d. *Percentage split* will train the classifier on the indicated percentage of the data and test it on the rest.

8. The dropdown menu allows us to choose the class attribute. Here *play* has already been appropriately suggested. Clicking *Start* will execute the training and evaluation process. For the moment, let us just try *Use training set* and click *Start*.

9. The results are displayed in the **Classifier output** panel.
   a. The decision tree is given in text form:
   ```
   outlook = sunny
   |  humidity = high: no
   |  humidity = normal: yes
   outlook = overcast: yes
   outlook = rainy
   |  windy = TRUE: no
   |  windy = FALSE: yes
   ```
   and corresponds to the representation on the right.

   Unfortunately, in Weka, we cannot see a visualisation of a tree produced by ID3. However, this is possible for the *J48* classifier, which is an implementation of the **C4.5** algorithm. To visualise a tree, right-click on the corresponding result in the *Result list* and choose *Visualize tree*.

   b. A summary of the evaluation gives information such as the percentage of correctly and incorrectly classified instances.
c. Information about the accuracy is given. TP and FP refer to True Positives and False Positives respectively.

d. The confusion matrix contains information about the prediction in terms of true and false positives and true and false negatives.

<table>
<thead>
<tr>
<th>Actual value</th>
<th># true positives</th>
<th># false negatives</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

You can find lots of information about classifier accuracy and confusion matrices online, for example [http://www.dataschool.io/simple-guide-to-confusion-matrix-terminology](http://www.dataschool.io/simple-guide-to-confusion-matrix-terminology)

### Classification

1. Open the [weather.nominal](#) dataset.
   a. Compare the accuracy of the J48 classifier when tested using the Use training set option and when using different values of Percentage split.
   b. Suggest an explanation for the difference in accuracy. Why is it not good practice to use the same instances for training and testing?

2. Open the [bank-data](#) dataset which contains data about a bank’s customers and whether or not they bought a Personal Equity Plan (PEP is the class).
   a. Inspect the actual data in the dataset (Edit button on Preprocess tab), paying attention to the type of the attributes and to attributes where each instance has a different value.
   b. Can ID3 be used as a classifier on this dataset? Why?
   c. Which attribute is useless to predict the class? Select it and remove it (Preprocess tab).
   d. Use the J48 classifier with a Percentage split of 50%. Examine the results and generate the visualisation of the tree. If the visualisation window is too small, maximise it, right-click in the window and choose Fit to Screen.
   e. Based on the prediction from the decision tree:
      i. Is an unmarried customer without children and no mortgage likely to buy a PEP?
      ii. What about a customer with more than one child and an income of less than £30000?