Data Mining and Machine Learning: Assessed Coursework 1

Handed Out: 2nd October 2017
What must be submitted (all students): A report in PDF format.
To be Submitted Monday October 16th 2017 -- Submit via Vision
Worth: 30% of the marks for the module
This is **individual** coursework and you should submit your own individual report.

Marking scheme: 80% for doing what I ask really well; I will look at reasoned argument, presentation, understanding. 20% 'wow factor' for insightful additional feature: e.g. dazzle me with a particularly interesting and useful way of presenting results, amaze me with some additional work using an appropriate method to gain more insight into the datasets and showing that you understand these methods, discuss your findings in depth and with insight.

We maintain strict **length limits** on this coursework. Marks will be subtracted for going over the limits. See the end of this document for details.

You may be subject to a brief viva in October/November – this is mainly for me to ensure it is your own work.

The aim: I want to make sure you get experience with handling data; this includes applying and understanding methods that explore individual attributes of a dataset, so that you will be able to make sense of the information that Weka provides about them. In the process you will become familiar with Weka's Explorer interface and comfortable navigating the tools that Weka offers for pre-processing and filtering files and visualisation.

WHAT TO DO

Level 10 students: (BSc, and 3rd or 4th yr MEng): PART 1, 2, 3, 4 only. Marks are divided evenly between the 4 sections (25% each).

Level 11 students: (MSc and final-year MEng): PART 1, 2, 3, 4, 5 Marks are divided evenly between the 5 sections (20% each).

For this assignment you will explore two datasets

- *iris.arff*
- *glass.arff*

We have discussed *iris.arff* dataset in class. The glass identification dataset *glass.arff* is from the US Forensic Science Service. contains data on six types of glass. This dataset has been made available on the UCI Machine Learning Repository and also in .arff format from Weka. You can inspect each of the files with a text editor to see descriptions of the data and the Weka metadata.

1. Getting started

Download both datasets from Vision. For each dataset, *iris.arff* and *glass.arff* in turn:

- Load the dataset into the Explorer interface. Explore the attributes and instances. How many attributes and instances are there in each dataset? What are the attribute names? What type and range of values does each attribute have? What is the class attribute and how is it distributed? Conduct a similar exploration of any relevant instance information. Is there missing data and if so how is it handled?
- Use the **Visualise** panel to explore the attributes. Do any of the attributes seem highly correlated with each other? Which attributes seem to contribute most to a successful classification, and why?

For each dataset write a paragraph summarising the answers to the questions above. Support your arguments with any relevant screenshots of Weka visualisations. Section 1 should take up to 1 side of A4, including any relevant screenshots.

2. Running Nearest Neighbour Learning

IBk is Weka's nearest neighbour classifier. You will find it in the Explorer under the Classifiers tab (*classifiers.lazy.IBk*). Run the Nearest Neighbour classification algorithm on the *glass.arff* dataset. Start with the number of nearest neighbours (KNN) set to 1 (this is the default).

(a) Use cross-validation to test its performance, with the number of folds set to the default value of 10. What is the accuracy of IBk?

(b) Run IBk again, but increase the number of neighbour instances to 5 by setting the KNN field to 5. Use cross-validation to test its performance, with the number of folds set to the default value of 10. What is the accuracy of IBk?

(c) Run Nearest Neighbour classification twice on the *iris.arff* dataset, again with the number of neighbours (KNN) set to 1 and then to 5. How does this affect accuracy? Explain the difference in accuracy made by changing the number of neighbours for *iris.arff* compared to *glass.arff.*

(d) After classification has run, Weka displays a number of different accuracy measures for each class, including True and False Positive rates; Precision, Recall and F-Measure; ROC Area; and a Confusion Matrix. What does each of these measures tell you about properties of the data set? Give some technical/logical justification of your answer based on definitions of these measures, and taking into account the above experiments.

Section 2 should take up approximately 1 side of A4, including any appropriately chosen screenshots.

3. Selecting Attributes

We want to investigate which subset of attributes produces the best classification accuracy for the nearest-neighbour algorithm on the *glass.arff* dataset.

(a) To begin with we will do this by hand, using a process called Backward Elimination. Backward Elimination consists of dropping each attribute individually from the full dataset, and then running the learning algorithm with cross-validation on the remaining attributes.
    - When you have determined the best eight-attribute dataset, repeat the procedure with the best eight-attribute dataset to find the best seven-attribute dataset, and so on.
    - Repeat until you have no attributes except the class attribute.
    - Record in a table the accuracy for IBk on the different attribute subsets

Number of attributes	Names of attributes in the "best" subset	Classification Accuracy
9		
8		
7		
6		
5		
4		
3		
2		
1		
0		

(b) What is the best set of attributes?
(c) Is the best accuracy an *unbiassed* estimate of accuracy on future data? (Hint: for an unbiassed estimate on future data, we must not look at the test data at all when producing the classification model)
(d) Automated attribute selection methods can be divided into *filter* and *wrapper* methods. Filter methods apply a heuristic to measure the quality of a subset of attributes. Use the Explorer interface's Select Attributes panel and choose the default CfsSubsetEval, which evaluates subsets of attributes. CfsSubsetEval aims to identify a subset of attributes that are highly correlated with the class attribute and not highly correlated with each other. Use the default settings to choose a set of attributes. How does this compare to the set of attributes you found by hand in Part b?

Section 3 should take up approximately 1 side of A4 including tables and analysis.

4. Pre-processing: Exploring the effects of Discretization

There are two types of discretization techniques, unsupervised ones (which do not take the class attribute into account) and supervised techniques.

Weka's main unsupervised method is *weka.filters.unsupervised.Discretize*. This can be set to perform either *equal-width* or *equal-frequency* discretization.

(a) Load the *glass.arff* dataset into the Explorer and apply the unsupervised discretization filter in each of the two modes, equal-width and equal-frequency. Compare the histograms for attributes with each of these methods. Why is the equal-frequency histogram skewed for some of the attributes?

Supervised discretization tries to create intervals which have a consistent class distribution across the interval, although the distributions may be different from one interval to the next. The main supervised discretization technique is *weka.filters.supervised.attribute.Discretize*.

(b) Load the *iris.arff* data and apply supervised discretization to it. Look at the histograms. Which attributes do you think would be most predictive?

(c) Do the same for the *glass.arff* dataset. Which attributes do you think would be most predictive?

(d) Some attributes have only a single bar in the histogram. What does this mean?

Section 4 should take up approximately 1 side of A4, including any appropriately chosen screenshots to support your analysis.

5. MSC Additional work: Class Noise and Nearest-Neighbour Learning

Nearest-neighbour learning is sensitive to noise in the training data set. Weka can inject different amounts of noise into the dataset and observe the effects on the classifier. The filter AddNoise (in *weka,filters.unsupervised.attribute*) will flip a certain percentage of the class labels in the data to a randomly chosen value. For this experiment only the training data should be affected by noise, and the test data should not. To do this, use a meta-learner called FilteredClassifier in *weka.classifiers.meta*. FilteredClassifier applies the filter to the data before running the learning algorithm. Configure FilteredClassifier so that it uses IBk as the classifier and AddNoise as the filter.

(a) For the *glass.arrf* dataset, create a table of the cross-validated accuracy estimate of IBk for10 different percentages of class noise and for neighbourhood sizes k = 1, k =3, k = 5 (by setting the value of KNN in the k-nearest-neighbour classifier).

Percentage noise	k=1	k=3	k=5
0%			
10%			
20%			
30%			
40%			
50%			
60%			
70%			
80%			
90%			
100%			

(b) Describe the effects of increasing the amount of class noise.
(c) Describe the effects of altering the value of *k*.

Section 4 should take up to 1 side of A4 including table and analysis.

THE REPORT

- Your report should contain clear presentations of the answers to the each of the questions, numbered appropriately. You may add extra discussion as appropriate.
- Your report should contain the tables specified.
- You may add relevant histograms or other plots.
- Your document should use an 11 point font with 2cm margins on all sides.
- This must all be done within **4 sides** of A4 for Level 10 students**, 5 sides** of A4 for Level 11. If a submission has *n* more than the required number of pages, you will automatically lose 10**n* marks out of 100. It will be quite possible to get negative marks for this assignment.

AN IMPORTANT NOTE

Before you start completing the above tasks, create folders on your computer to store classifiers, Weka settings, screenshots and results of all your experiments.
As part of your coursework marking, you may be asked to re-run all your experiments in the lab. So please store all of this data safely in a way that will allow you to re-produce all your results on request.