

CSCE 5380
Project Assignment 1
Date Issued: 09/21
Date Due: 10/21
Total Points: 100

This assignment is based on the Weka data mining tool which is an open source software tool which contains suites of algorithms and visualization/graphical API's, among other tools that can be used to analyze and process various datasets.

More information about Weka is available at: <http://www.cs.waikato.ac.nz/ml/weka/>

The TA's office hours are Fridays 12:00pm to 4:00 p.m. at F205.

Weka Introduction:

If you are downloading Weka on your machine, you would need a recent version of Java (v1.5 or v1.6 should be okay) installed for Weka to work correctly. Weka is available for download on Windows, Linux and MacOS. For Windows and MacOS, there is an executable file (.exe) and a disk image file (.dmg) available respectively, for Linux; it is in the form of a zip file which you would need to unzip, change into that directory and type:

```
j ava -j ar weka. j ar
```

This should start the Weka GUI chooser. The GUI chooser has 4 interfaces: Explorer, Experimenter, Knowledge-Flow and a command line interface. If you click on "Explorer", it will show you an interface, on the top toolbar of which there will be tabs for classification, clustering, data visualization, etc. Weka has a set of datasets which you can experiment with; these datasets can be accessed by clicking on Open File -> data. Try opening a dataset and running a classification algorithm on it like one for categorical prediction like Naive Bayes or a numeric prediction decision tree algorithm like J48 or any other algorithm you might have learned in class or from the textbook. Explore the different options available like cross-validation (with a variable number of folds) or specifying a percentage split between the training and test data sets.

The native data format of Weka is ".arff" (Attribute Relation File Format) and all the sample datasets in Weka are in the ARFF format. Although Weka supports CSV (Comma Separated Values), analyzing the data is much easier if you convert the dataset into the ARFF format before running any algorithm on it. (The reason for this is that CSV files do not contain any information about the attribute labels, and Weka needs to generate/determine attribute labels by itself which can create problems such as inconsistencies between the training and test datasets). You can use the following command for converting from CSV format to ARFF format:

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

In this assignment, we focus on classification, and use a few algorithms in Weka for classifying and visualizing datasets. There are two tasks you need to do:

Task 1:

This task requires you to test the Segment-test dataset (segment-test.arff) in Weka on two classification algorithms – Naïve-Bayes and Random-Tree Decision Tree (RT).

Select the Segment-test data set and keep the default values for pre-processing.

Problem 1 (15 points): Cross-validation -

Run the Naïve-Bayes classifier by selecting 3 different numbers of folds (for example, do 3 different runs by selecting 6, 9, 14 folds in each run respectively).

Now, run the Random-Tree classifier by selecting the same number of folds selected for the Naïve-Bayes classifier (for example, if you selected 6, 9, 14 folds for Naïve-Bayes, use the same 3 numbers for Random-Tree).

1. Do the number of folds have any correlation with the number and percentage of correctly classified instances within the same model (For example, 6 folds and 9 folds in NB and RT respectively)? Explain the results.
2. Do the same number of folds when applied to different models have any effect on the number and percentage of correctly classified instances (For example, 6 folds and 9 folds in NB and RT)? Explain the results.
3. Select 1 set of results generated for each classifier. For example, if you performed a test by selecting 9 folds, select the results you obtained for 9 folds for both – NB and RT. Considering all classes in the dataset; calculate the accuracy and error rate from the confusion matrix provided by Weka. Show the formula and explain the steps in calculating the accuracy and error-rate from the confusion matrix.

Problem 2: (15 points): Percentage Split -

Run the NB classifier by selecting 3 different percentages of training data [percentage split ratio] (for example, run 3 different runs by selecting a testing-training split of 42%-58%, 54%-46%, 65%-35%, etc. in each run).

Run the RT classifier by selecting the same set of training set ratios selected for the NB classifier. Does the percentage of training data affect the classifier accuracy?

1. Does the percentage of training data affect the classifier accuracy? How and why?
2. If the same percentage of testing data is used for both classifiers (e.g. 42% for both NB and RT, does the classification accuracy vary from one classifier to another? Why?
3. Select any one set of results generated for each classifier. For example, if you performed a test by selecting 42% training data, select the results you obtained for 42% for both – NB and RT. Considering all classes in the dataset, calculate the accuracy and error rate from the confusion matrix provided by Weka for the results of NB and RT. Show the formula and explain the steps in calculating the accuracy and error-rate from the matrix.

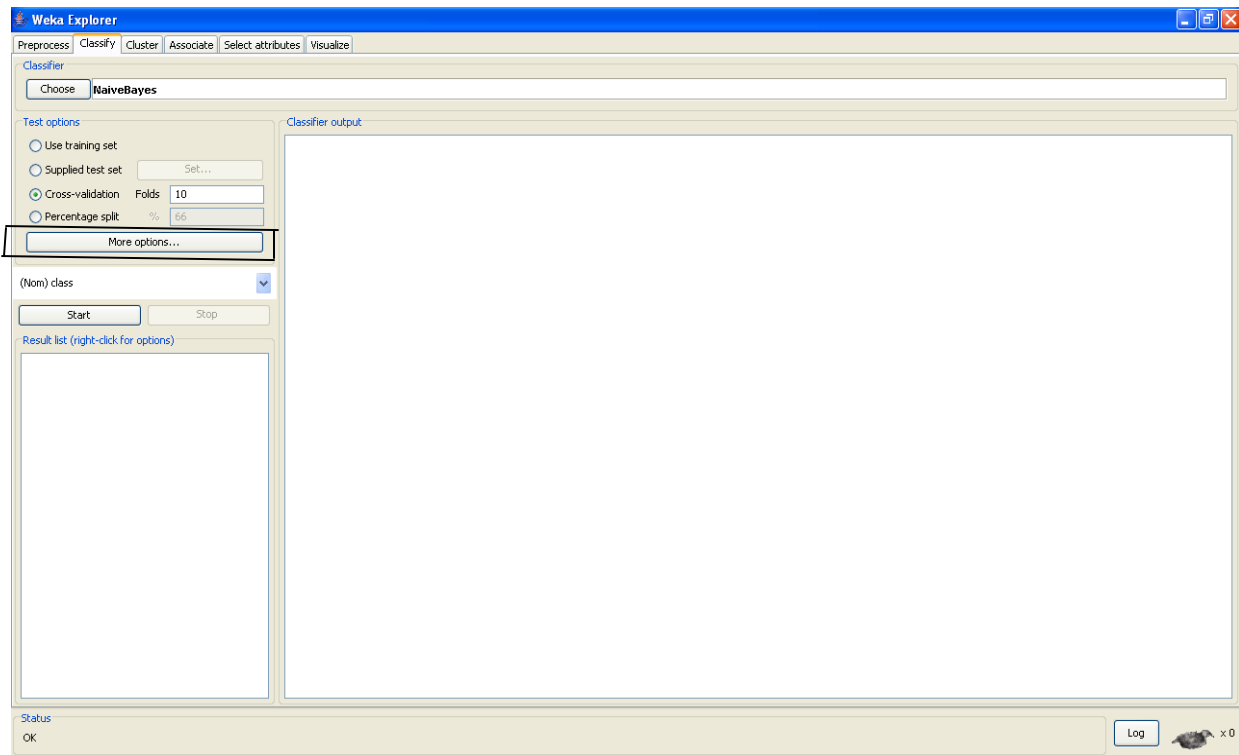
Task 2:

This task requires you to test the Soybean dataset in Weka software on two classification algorithms – Naïve-Bayes and Random-Tree.

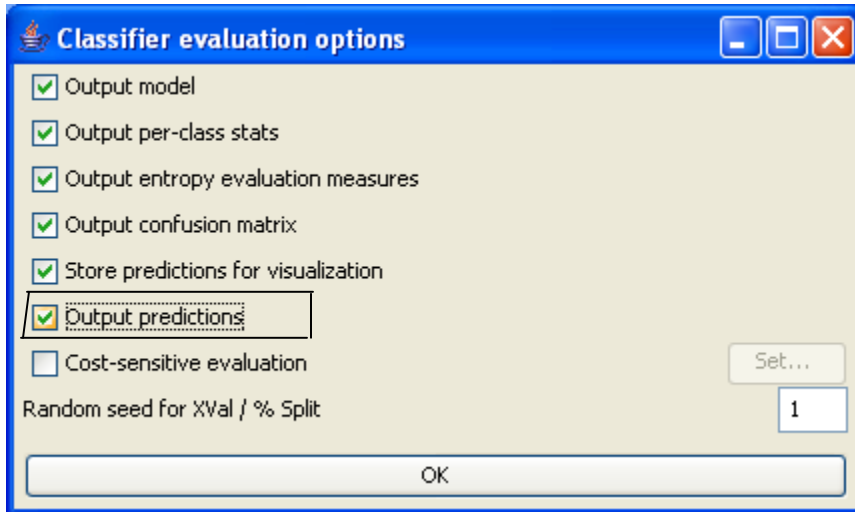
Select the Soybean dataset and keep the default values for pre-processing.

Problem 3: (15 points)

Run each of the NB and RT classifier once by selecting a certain value for number of folds for cross-validation (e.g. 6 folds). Before running the classifier, click on “More options” in the Weka Window.



From the window shown by Weka, select “Output Predictions”



Run both the classifiers with this option enabled.

For the results obtained, generate a ROC curve for the class “brown-spot” for both the classifiers. In addition to the curve itself, you should also show the TP/FP/TN/FN/TPR/FPR table that is used for constructing the ROC curve.

Problem 4: (15 points):

Run each of the NB and RT classifier once by selecting a certain percentage of training data (e.g. 42%). [Similar to previous problem, before running the classifier, click on “More options” in the Weka window and from the window, select “Output predictions”]

For the results obtained, generate a ROC curve for the class “frog-eye-leaf-spot” for both the classifiers. In addition to the curve itself, you should also show the TP/FP/TN/FN/TPR/FPR table that is used for constructing the ROC curve.

Task 3:

From the Weka web-page (<http://www.cs.waikato.ac.nz/ml/weka/>), download the UCI repository dataset which contains a set of 37 datasets for classification problems -- “datasets-UCI.jar”. Open the “audiology.arff”, “anneals.arff”, and “autos.arff” datasets in Weka (If the UCI datasets do not load or show up immediately in Weka, you might have to play around with it a bit).

Run the J48 Decision Tree classifier on it with a 70% and 30% split of training and test data respectively.

Problem 5: (40 points)

1. Calculate and report the error-rate and accuracy from the confusion matrix provided by Weka for each of the datasets. Which of the 3 datasets has the highest number of correctly classified instances?
2. Which of the three datasets has the smallest and largest J48 decision trees? Explain why you think the size of the decision trees varies.

Submission instruction:

Please turn in a hard copy containing the answers to the five problems before class on the due date.