CS725: Assignment 3

20 Marks, Report (approx 5 pages) due on November 1^{st} and to be uplpaded via moodle, vivas on November 5^{th} and 6^{th}.

This assignment can be done in groups of at most 3.

- Consider the 20 News groups dataset `http://people.csail.mit.edu/jrennie/20Newsgroups/`. This is a text classification dataset. In this assignment, we will look at implementing classification, clustering and itemset mining (apriori style) algorithms for this dataset. Go through this dataset and

    1. You should try as hard as possible to get the most accurate classifiers of each type below. Use the best smoothing you can think of, or whatever is the best possible impurity function in (a) or the the probabilistic model you think is best in (b).

        (a) Code up any of the non-probabilistic classifiers we have discussed so far (such as some decision tree classifier) or any of the non-probabilistic classifiers we will be discussing (such as support vector machine). Write program to train it using the 20 Newsgroups training data, and program to evaluate it on the test data. In your report, present the confusion matrix[1] on the test data as well and from this matrix, report the test accuracy. In a similar manner, report also the confusion matrix on the training data as well as the train accuracy. **Copying any piece of code from the web or from other teams will amount to a direct FF grade**.

        (b) Code up any of the probabilistic classifiers (along with its training algorithm and its evalution code) discussed so far or any of the probabilistic classifiers we will be discussing. Report all train and test confusion matrices as well as the train and test accuracies. **Copying any piece of code from the web or from other teams will amount to a direct FF grade**.

[1] `http://en.wikipedia.org/wiki/Confusion_matrix`

Report your comparison of the numbers obtained in (a) and (b) above and in general, the results. Compare your accuracies against that you get from Weka[2] or Rainbow [3] or any other existing implementation of the classifiers you chose in (a) and (b) and report how your implementation compares with other implementations on accuracy and speed.

2. Now ignore the class labels on the documents.
   (a) Try as much as you can to get as high accuracy a hierarchical clustering algorithm as possible on the training documents (ignoring their class labels). You can test what fraction of the documents in each cluster belong to the class which majority of the members of the cluster belong to and compute the accuracy of clustering. Report the accuracies and your general observations. **Copying any piece of code from the web or from other teams will amount to a direct FF grade**.
   (b) Do the same as above for a non-hierarchical clustering algorithm. **Copying any piece of code from the web or from other teams will amount to a direct FF grade**.
3. Find frequent feature sets using the apriori algorithm (for your choice of support threshold according to what you think helps) and use these new features along with (first setting) and without (second setting) the original set of features in both the classification and both the clusterting algorithms. Do your training/test accuracies improve in either of the two settings? Report the accuracies and your general observations. **Copying any piece of code from the web or from other teams will amount to a direct FF grade**.

Report any other general observations.

[2] `http://www.cs.waikato.ac.nz/ml/weka/`

[3] `http://people.csail.mit.edu/jrennie/20Newsgroups/`