CSC475 MIR Assignment 3, Spring 2016 (10 pts)

The goal of this assignment is to familiarize you with data mining/machine learning in the context of music information retrieval.

Hope you find it interesting, George Tzanetakis

1 Classification using Audio Features (5 points)

Download the 1.2 GB genre classification dataset from:
`http://marsyas.info/downloads/datasets.html`
You will only need 1.2 GB of space for download but after that you can pick any three genres out of the 10 genres for your experiments. Alternatively if you don't have enough space you can download individual files for 3 genres (at least 20 tracks for each genre) from:
`http://marsyas.cs.uvic.ca/sound/genres/`
Read the instructions in Chapter 3 of the Marsyas User Manual (Tour - Command Line Tools) and use the **bextract** command-line program to extract features for the 3 genres you selected. Load the extracted .arff file into Weka and report on the classification accuracy of the following classifiers: ZeroR, NaiveBayesSimple, J48, and SMO.

Your deliverable will be the list of command you used and the classification accuracy + confusion matrix for each classifier for the 3-genre experiment. (**) **(2 points)**

Now use Weka to convert the .arff to the .libsvm format that is supported by scikit-learn. Do a similar experiment using scikit-learn i.e 3 classifiers and report accuracy and confusion matrix. Provide a listing of the relevant code (**) **(3 points)**.

2 Movie review categorization using Naive Bayes (5pts)

Text categorization is the task of assigning a given document to one of a fixed set of categories, on the basis of text it contains. Naive Bayes models are often used for this task. In these models, the query variable is the document category, and the "effect" variables are the presence/absence of each word in the language; the assumption is that words occur independently in documents within a given category (condititional independence), with frequencies determined by document category. Download the following file:
`http://www.cs.cornell.edu/People/pabo/movie-review-data/review\_polarity.tar.gz`
containing a dataset that has been used for text mining consisting of movie reviews classified into negative and positive. You will see that there are two folders for the positivie and negative category and they each contain multiple text files with the reviews. You can find more information about the dataset at: `http://www.cs.cornell.edu/People/pabo/movie-review-data/`

Our goal will be to build a simple Naive Bayes classifier for this dataset. More complicated approaches using term frequency and inverse document frequency weighting and many more words are possible but the basic concepts are the same. The goal is to understand the whole process so don't use existing machine learning packages but rather build the classifier from "scratch".

Our feature vector representation for each text file will be simply a binary vector that shows which of the following words are present in the text file:

```
Awful
Bad
Boring
Dull
Effective
Enjoyable
Great
Hilarious
```

For example the text file *cv996_11592.txt* would be represented as $(0, 0, 0, 0, 1, 0, 1, 0)$ because it contains *Effective* and *Great* but none of the other words.

- (**) Write code that parses the text files and calculates the probabilities for each dictionary word given the review polarity **(1pt)**.

- (**) Explain how these probability estimates can be combined to form a Naive Bayes classifier. Calculate the classification accuracy and confusion matrix that you would obtain using the whole data set for both training and testing. **(1pt)**

- (**) Check the associated README file and see what convention is used for the 10-fold cross-validation. Calculate the classification accuracy and confusion matrix using the recommended 10-fold cross-validation. **(1pt)**

- (***) One can consider the Naive Bayes classifier a generative model that can generate binary feature vectors using the associated probabilities from the training data. The idea is similar to how we do direct sampling in Bayesian Networks and depends on generating random number from a discrete distribution (the unifying underlying theme of this assignment). Describe how you would generate random "movie" reviews consisting solely of the words from the dictionary using your model. Show 5 examples of randomly generated positive reviews and 5 examples of randomly generated negative reviews. Each examples should consist of a subset of the words in the dictionary. **(2pt)**