Two zip archives have been associated with this assignment on Blackboard. Archive digits_train_f.zip contains a subset of the MIT/TIDIGITS spoken digit corpus (Leonard and Doddington, 1993). In this subset, a set of female speakers are saying the digits zero through nine, and the word “oh,” a frequent pronunciation for zero.

The second archive ps02_skeleton.zip contains skeleton code for your assignment. It includes a solution to the classes from the previous assignment. Please make sure that you understand these if you had problems with the previous assignment. New functions are skeleton code with signatures. You must comply with the specified signatures in the skeleton code.

Your task for this assignment is to create functions and classes to construct spectrograms and conduct PCA. A couple of new functions and classes are given to you and do not need to be implemented, but you should read and understand the code:

- mydsp.utils.get_corpus(dir) – Examine a given directory recursively for wav files and return a list of files.
- mydsp.multifileaudioframes.MultifFileAudioFrames(file_list, adv_ms, len_ms) – Generates audio frames in the same manner as the AudioFrames class, but spans all files in the list.
- PCA – The class discussed in lecture.

The following classes and functions described below need to be written. Arguments are only briefly described as there is further documentation in the skeleton code.

1. (20 points) Complete class mydsp.dftstream.DFTStream. Like other stream objects that we have used, it takes a frame stream as its source.
2. (40 points) Write the following functions in mydsp.plots:
   a. spectrogram – Given a matrix of intensity values and the time and frequency labels, plot the spectrogram. matplotlib.pyplot functions pcolormesh and colorbar will be useful here. Note that the libraries have a spectrogram function, you should not use this or look at the source code although you are welcome to look at and reuse code from the lectures.
   b. pca_gram – Given a matrix of magnitudes from a PCA projection and a time-axis, display the values in a manner similar to a spectrogram.
   c. concatenated_spectrogram – Given a list of files, create a spectrogram showing the contents of all the files.
   d. pca_variance_captured – Given a PCA object, plot the number of components versus the total amount of variance captured.
3. (40 points) driver function. Your driver should do the following:
   a. Find all of the speech files using get_corpus().
b. Plot two spectrograms of every other one of the first 22 files, e.g. file 0, 2, 4, … in a single plot. This should be all 11 spoken digits. One spectrogram should use narrow-band analysis and the other wide-band.

c. Compute the PCA of the entire dataset.

d. Determine and plot how much variance is captured.

e. Print how many components are needed for each decile of the variance (e.g. $N_1$ components to capture $\geq 10\%$, $N_2$ components to capture $\geq 20\%$, etc.)

f. Transform the digit spectra from part b into PCA space using the number of components needed to capture 70% of the variance (arbitrary number). Create a plot of the values in PCA space (pca_gram).

Turn in your program output and plots. All plots should have informative captions. Print your spectrograms in landscape mode or they may be too squished to be useful.

4. (10 points) Draw (by hand) a set of two-dimensional feature vectors such that one could reasonably assume that they have negative covariance.

5. (10 points) Will a high capacity model be likely to have higher bias or variance? Justify your answer.

References

http://ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC93S10.