COMPSCI304 Spring 2022 Project 3   Due date: Feb 7th 9pm Beijing time

Note: In this assignment, you are encouraged to reuse much of the code you have already written. Your feature computation code can be used to derive features from data.

In this assignment, you will be required to record digits many times and build DTW and HMM-based isolated word recognition systems. In total, you will need to record each of the digits 0,1,..9 ten times each. Each recording must be isolated, with no more than half a second of preceding and trailing silence. If you're working in a team, you may want to split the task of recording across team members. The details of the problem are below.

Problem 1

The first problem is on DTW-based recognition.

- Write a routine to perform DTW between a template feature vector sequence and a "data" feature vector sequence. The feature vectors must be 39-dimensional features (cepstra/delta cepstra/double delta) that are obtained with the code you wrote for assignment 2.(needs to perform mean subtraction and variance normalization)
- Record one instance each of digits zero, one, two, three, four, five, six, seven, eight, nine as templates. You may use your code from assignment 1 to record the data. Compute feature vector sequences from them to act as templates.
- Record a further five instances each of the same digits (isolated words) as test data. Compute feature vectors from them and recognize them using the DTW algorithm. Report recognition accuracy.
- Redo the earlier step with time-synchronous DTW.
- Redo the above with pruning. Use relative pruning. Plot pruning threshold as a function of recognition accuracy.
- Record *yet another* set of four recordings of each of the digits. Including the first set of recordings you will now have five templates for each digit. Repeat the recognition experiment above using *multiple* templates for each word. Plot how the recognition accuracy changes as a function of the number of templates of the words (1 through 5). Do this using time-synchronous DTW, without pruning. Repeat it using a pruning threshold determined from the plot in the earlier experiment.

Problem 2

Use the segmental K-means procedure to train an HMM for each of the digits (using the 5 "training" recordings you have for them). Assume each state to have a single Gaussian distribution, and the HMM for each digit to have 5 states. Recognize the 5 test utterances using the HMM models and report recognition accuracy.

Problem 3: Repeat the segmental K-means to train HMMs with mixtures of 4 Gaussians per state and test the performance again (K-means algorithm can be used to replace GMM).