Name: Ng Yi Ming

Email: E0486563@u.nus.edu

Student ID: A0211008B

OVERVIEW In this assignment, you will implement the Baum-Welch algorithm to find the unknown parameters of 1-dimensional Gaussian hidden Markov models (HMMs). References: Lecture 8

For this assignment, we are tasked to create a HMM with Baum-Welch algorithm, which consist of an E and M step respectively.

The model is first initialized by passing the state distribution vector and transition matrix through the softmax transformation to ensure all probabilities sum to 1, then KNN is used to provide starting Gaussian Process clusters.

**E step**

Then the forward backwark algorithm is used to compute alpha (steps up to t) and beta (steps t+1 onwards) for each trial which is then in turn used to compute gamma which is the marginal posterior distribution of the probabilities for starting at each state.

To find the first step probability, we take the initial pi ( initial guess of starting state distribution) and product it with the emission probability of observing the state. Then alphas are calculated recursively as the sum product of the previous state, relevant transition probability, and the sampled emission probability ( chance of observing the trial given the current state ) which is scaled.

Beta is calculated as the likelihood of observing all future data points from t+1 onwards which is the sum product for the current state for time t+1 with the emission probability of observing the given sequence step which is scaled as well.

**M step**

With the information collected in the E-step, we then update our initial guesses (parameters pi, A, and phi) to be more in line with the observed data. Pi is estimated by averaging the first gamma values from all sequences giving us a posterior probability for each state is to be the starting state. The updated transition matrix A is then estimated via considering the xi values, which tell us how likely it is to transition from one state to another.

Finally we re-estimate the mean and variance for the gaussian clusters.

**Fit\_hmm**

The workhorse of the algorithm. First the initial posterior guesses are initialized using KNN and softmax as elaborated earlier, then iteratively, E and M step are applied. For each iteration, the model is checked if the current iteration of parameters lead to an improved loglikelihood score for early breaking or up to the maximum defined number of iterations, whichever is earlier.

The updated posterior distributions and parameters are then returned