Exercise for the Expectation-Maximization Algorithm (Using MATLAB)

(a) Generate a set of X of N=1000 2-dimensional points that obtained from the following *pdf:* $\mathrm{p(x)} = \sum_{j=1}^{3} P_j p(x|j)$ where the *p(x|j)*'s, *j=1,2,3* are normal distribution with mean values $m_1 = [1,1]^T$, $m_2 = [3,3]^T, m_3 = [2,6]^T$ and covariance matrices $S_1 = 0.1\mathrm{I}$, $S_2 = 0.2\mathrm{I}$, $S_3 = 0.3\mathrm{I}$, respectively (where I is the 2x2 identity matrix). In addition, $P_1 = 0.4, P_2 = 0.4$, $P_3 = 0.2$. Then pretend that we do not know how they were generated. Use EM algorithm to estimate the unknown parameters of $\mathrm{p(x)}$. You can consider the following sets of initial parameter estimates:

(1) $\mathrm{J} = 3, m_{1,ini} = [0,2]^T$, $m_{2,ini} = [5,2]^T$, $m_{3,ini} = [5,5]^T$, $S_{1,ini} = 0.15\mathrm{I}$, $S_{2,ini} = 0.27\mathrm{I}, S_{3,ini} = 0.4\mathrm{I}$, $P_{1,ini} = P_{2,ini} = P_{3,ini} = 1/3$.

(2) $\mathrm{J} = 3, m_{1,ini} = [1.6,1.4]^T$, $m_{2,ini} = [1.4,1.6]^T$, $m_{3,ini} = [1.3,1.5]^T$, $S_{1,ini} = 0.2\mathrm{I}$, $S_{2,ini} = 0.4\mathrm{I}, S_{3,ini} = 0.3\mathrm{I}$, $P_{1,ini} = 0.2, P_{2,ini} = 0.4, P_{3,ini} = 0.4$.

(3) $\mathrm{J} = 2, m_{1,ini} = [1.6,1.4]^T$, $m_{2,ini} = [1.4,1.6]^T$, $m_{3,ini} = [5,5]^T$, $S_{1,ini} = 0.2\mathrm{I}$, $S_{2,ini} = 0.4\mathrm{I}$, $P_{1,ini} = P_{2,ini} = 0.5$.

(4) Discuss your results from the observation.

(b) The E-M algorithm is used in a classification application.

(1) **The training data set**: assume a 2-class problem with N=1000 2-dimensiuonal vectors is considered. 500 vectors obtained from class w_1, which is modeled as $p_1(\mathrm{x}) = \sum_{j=1}^{3} P_{1j} p_1(x|j), where\ \ p_1\ (x|j), j = 1,2,3$ are normal distributions with mean values $m_{11} = [1.25,1.25]^T, m_{12} = [2.75,2.75]^T, m_{13} = [2,1.6]^T$, and covariance matrices $S_{1j} = \sigma_{1j}^2 I, j = 1,2,3$, where $\sigma_{11}^2 = 0.1, \sigma_{12}^2 = 0.2, \sigma_{13}^2 = 0.3$, respectively. The mixing probabilities are $P_{11} = 0.4, P_{12} = 0.4, and\ \ P_{13} = 0.2$.
The other 500 data vectors obtained from class w_2, which is modeled as $p_2(\mathrm{x}) = \sum_{j=1}^{3} P_{2j} p_2(x|j), where\ \ p_2\ (x|j), j = 1,2,3$ are normal distributions with mean values $m_{21} = [1.25,2.75]^T, m_{22} = [2.75,1.25]^T, m_{23} = [4,6]^T$, and covariance matrices $S_{2j} = \sigma_{2j}^2 I, j = 1,2,3$, where $\sigma_{21}^2 = 0.1, \sigma_{22}^2 = 0.2, \sigma_{23}^2 = 0.3$, respectively. The mixing probabilities are $P_{21} = 0.2, P_{22} = 0.3, and\ \ P_{23} = 0.5$.
The testing data set: generated an additional data set X of 1000 data vectors, such that 500 from $p_1(\mathrm{x})$ and the rest obtained from $p_2(\mathrm{x})$
Pretend we do not know how to the training data set generated. **We assume from the data set that we have a priori information about the number of dense regions in each class, so we adopt a mixture model with three Gaussian components to model the *pdf* in each class.** Use the concept run in (a) to estimate $p_1(\mathrm{x})$ and $p_2(x)$ by properly initial parameter estimates. The a priori class probability $\mathrm{P}(w_i), i = 1,2$ for each class is estimated as the number of vectors in the respective class divided by the total number of vectors.

(2) With the pdf and other parameters estimates for each class obtained in (1), the Bayesian classifier is used to classify **the testing data set X**. In classification task, when the number of summed distributions in the mixture model is not known, the task is run a number of times with different values of J; the values that results in the lowest classification error over the test set is adopted.

Exercise for the HMM Algorithm (Using MATLAB)

(A) Write a function named *Viterbi-HMM* that take as input (a) a column vector of initial state probabilities *pi_init*, (b) the transition matrix A, whose *(i,j)* elements is the probability of transition from state *i* to state *j*, (c) the matrix of the emission probabilities B, whose *(i, j)* elements is the probability to emit the i-th alphabet symbol from state *j*, and (d) a row vector O, which contains a sequence of the code numbers of discrete symbols. It returns (1) the best state sequence and (b) the respective probability produced when the HMM, defined by *pi_init, A, B,* is applied to the sequence of symbols contained in O using the Viterbi method. Assume that if the alphabet symbols are say, *s1, s2, s3,…,sq*, the corresponding code numbers are *1, 2,..q.*

(B) Assume two coins A, and B are used for a coin tossing experiment, The probability that A returns heads is 0.6, and B returns heads is 0.4. A individual standing behind a curtain decides which coin to toss as follows: the first coin to be tossed is always coin A, the probability that A is re-tossed is 0.4, and the coin B re-tossed is 0.6. A observer can only have access to the outcome sequence of heads or tails of the experiment. (1) Model the its HMM. (2) Use the function in (A) to find the best state sequence and respective path probability, for the following observation sequences: {H, T, T, T, H} and {T, T, T, H, H, H, H}.

Show your program and results before June 27,2019