E9 205 – Machine Learning for Signal Processing

Homework # 2
Due date: Sept. 26, 2018

(3:45 PM) for analytical and end of the day for coding part.
Coding assignment submitted to mlsp18 doT iisc aT gmail doT com

September 14, 2018

1. **Maximum Likelihood Classification** Consider a generative classification model with K classes defined by prior probabilities $p(C_k) = \pi_k$ and class-conditional densities $p(\boldsymbol{\phi}|C_k)$ where $\boldsymbol{\phi}$ is the input feature vector. Suppose that a training data is given $\{\boldsymbol{\phi}_n, \boldsymbol{t}_n\}$ for $n = 1, .., N$ and $\boldsymbol{t}_n$ denotes a binary target vector of dimension K with components $t_{nj} = \delta_{j,k}$ if input pattern $\boldsymbol{\phi}_n$ belongs to class k. Assuming that the data points are drawn independently, show that the ML solution for prior probabilities is given by,

$$\pi_k = \frac{N_k}{N}$$

where N_k is the number of points belonging to class k. (**Points** 10)

2. **Maximum Likelihood Linear Regression** - Kiran is doing his PhD on acoustic channel estimation. In order to estimate the channel characteristics, he designs an experiment in which he records the ultra sound signal at the source as well as at the output of the acoustic channel. Let $\mathbf{x}_i, i = 1, \ldots, N$ and $\mathbf{y}_i, i = 1, \ldots, N$ denote the feature sequence corresponding to the source and channel outputs.

   (a) He begins with an assumption of a linear model for the channel,

   $$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{b} + \boldsymbol{\epsilon}$$

   where the source features $\mathbf{x}$ are assumed to be non-random and $\boldsymbol{\epsilon}$ represents i.i.d. channel noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^{\mathbf{2}}\mathbf{I})$. Given this model, his advisor Mohan recommends the maximum likelihood (ML) method to estimate the parameters of the channel $\big(\mathbf{A}, \mathbf{b}, \sigma\big)$. How will you solve the problem if you were Kiran ? (**Points** 10)

   (b) While Kiran is successful in estimating the parameters of his model, Mohan is unhappy with the results when the model is used to approximate a cell phone transmission. Mohan proposes a more complex model where the source ultrasound data $\mathbf{x}_i, i = 1, \ldots, N$ is modeled as a Gaussian mixture model (GMM).

   $$\mathbf{x} \sim \sum_{m=1}^{M} \alpha_m \mathcal{N}(\mathbf{x}, \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$$

   Further, the channel is modeled as a linear transformation of the GMM mean components $\hat{\boldsymbol{\mu}}_m = \mathbf{A}\boldsymbol{\mu}_m + \mathbf{b}$. The covariances are not affected in this model. With the channel ouptuts $\mathbf{y}_i, i = 1, \ldots, N$, how will you help Kiran achieve his PhD faster by solving for the channel parameters $\mathbf{A}_m, \mathbf{b}_m, m = 1, \ldots, M$ assuming that the source signal GMM is already estimated. Simplify your result. (**Points** 15)

3. **Implementing GMM** - A set of training and test examples of music and speech are provided.
http://www.leap.ee.iisc.ac.in/sriram/teaching/MLSP_18/assignments/speech_music_data.tar.gz

   Using these examples,

   a Generate spectrogram features - Use the log magnitude spectrogram as before with a 64 component magnitude FFT (NFFT). In this case, the spectrogram will have dimension 32 times the number of frames (using 25 ms with a shift of 10 ms).

   b Train two GMM models with K-means initialization (for each class) separately each with (i) 2 mixtures with diagonal covariance, (ii) 2 mixtures with full covariance and (iii) 5-mixture components with diagonal/full covariance respectively on this data. Plot the log-likelihood as a function of the EM iteration.

   c Classify the test samples using the built classifiers and report the performance in terms of error rate (percentage of mis-classified samples) on the text data.

   e Discuss the impact on the performance for different number of mixture components, diagonal versus full covariance ?

   (**Points** 30)

4. **Implementing HMM** - Using the same data and features as above

   a Write a code to implement the likelihood computation using the forward variable after assuming a uniform flat start based initialization with 3 and 5 states per HMM and GMM with 2 mixture components per state.

   b Write a code to implement the Viterbi algorithm to decode the best state sequence using the existing model.

   c Use the Baum-Welch reestimation method to train HMMs with examples from music and speech features.

   d Classify the test examples and report the performance. How does the performance change for different number of states per HMM, different number of mixture components per GMM ? Will diagonal or full covariance GMM be a good choice ?

   Hint - Use a flat-start to initialize the model and use scaling technique in implementation. (**Points** 35)