18-698 / 42-632
Neural Signal Processing
Prof. Byron Yu Spring 2023

Problem Set 6

This problem set is due on **Tuesday, April 11, 11:59pm**.

Be sure to show your work and include all **Matlab code** and plots. **Any Matlab questions without code will receive no credit.**

If you have questions, please post them on the Piazza Q&A webpage, rather than emailing the course staff. This will allow other students with the same question to see the response and any ensuing discussion.

Please submit your work as a **single PDF file** on Gradescope, which is linked from Canvas. When preparing your solutions, please complete each problem on a **separate page**. Gradescope will ask you select the pages that contain the solution to each problem.

Submissions can be written in LaTeX or they can be handwritten and photocopied using a scanner or smartphone camera. Handwritten work should be clearly labeled and legible.

The dataset for the following problems can be found on Canvas under "Files → Data sets → ps6_data.mat". When you load the .mat file, you will find the following variables:

`Spikes`: a 31×552 matrix of spike snippets[1], where `Spikes(:,n)` is the nth threshold crossing snippet ($n = 1, \ldots, 552$). Values are in μV.

`InitParams1`: a structure containing initialization parameters for a Gaussian mixture model, with the following fields

- `mu` is a 31×3 matrix, where the kth column is the initialization of the kth cluster center, $\boldsymbol{\mu}_k$ ($k = 1, 2, 3$).
- `Sigma` is a 31×31 covariance matrix. Assume that all cluster covariances Σ_k are initialized to the same covariance matrix.
- `pi` is a 1×3 vector, where the kth element is the intialization of π_k, the prior cluster probability for cluster k.

`InitParams2`: a structure of the same form as `InitParams1`.

[1] The neural data have been generously provided by the laboratory of Prof. Krishna Shenoy at Stanford University. The data are to be used exclusively for educational purposes in this course.

1. In class, we derived the EM algorithm for the Gaussian mixture model by taking derivatives of the data likelihood $P(\{\mathbf{x}\} \mid \theta)$, where $\{\mathbf{x}\}$ represents the training data $\mathbf{x}_1, \ldots, \mathbf{x}_N$ and θ represents all model parameters $\boldsymbol{\mu}_k$, Σ_k, and π_k $(k = 1, \ldots, K)$. An alternate approach is to apply the general EM algorithm (shown on p.440–441 in *PRML*) to the Gaussian mixture model. Both approaches should yield the same update equations, which we will show in this problem.

   The Gaussian mixture model is defined by the prior probability of each mixture component indexed by z

$$P(z = k) = \pi_k$$

   and the conditional distribution of the data point $\mathbf{x}$ given the mixture component

$$P(\mathbf{x} \mid z = k) = \mathcal{N}\left(\mathbf{x} \mid \boldsymbol{\mu}_k, \Sigma_k\right),$$

   where $k = 1, \ldots, K$. We will denote the nth data point as $\mathbf{x}_n$ and its corresponding latent variable as z_n, where $n = 1, \ldots, N$.

   (a) **(10 points)** In the **E-step** of the general EM algorithm, we evaluate $P(z_n = k \mid \mathbf{x}_n)$. For the Gaussian mixture model, show that

$$P(z_n = k \mid \mathbf{x}_n) = \frac{\mathcal{N}\left(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \Sigma_k\right)\pi_k}{\sum_{j=1}^{K} \mathcal{N}\left(\mathbf{x}_n \mid \boldsymbol{\mu}_j, \Sigma_j\right)\pi_j}.$$

   (b) **(25 points)** In the **M-step** of the general EM algorithm, we maximize the expected log joint distribution (summed across all N data points)

$$\begin{aligned} \mathcal{Q}(\theta) &= \sum_{n=1}^{N} E\Big[\log P(\mathbf{x}_n, z_n \mid \theta)\Big] \\ &= \sum_{n=1}^{N}\sum_{k=1}^{K} P(z_n = k \mid \mathbf{x}_n) \log P(\mathbf{x}_n, z_n = k \mid \theta) \end{aligned}$$

   with respect to the model parameters θ. Note that the expectation above is taken with respect to the distribution found in part (a). We seek to find

$$\theta^{\text{new}} = \underset{\theta}{\operatorname{argmax}}\ \mathcal{Q}(\theta).$$

   For the Gaussian mixture model, let $\gamma_{nk} = P(z_n = k \mid \mathbf{x}_n)$ for notational simplicity. By maximizing $\mathcal{Q}(\theta)$ with respect to each of the model parameters, show that

$$\begin{aligned}\boldsymbol{\mu}_k^{\text{new}} &= \frac{1}{N_k}\sum_{n=1}^{N}\gamma_{nk}\,\mathbf{x}_n \\ \Sigma_k^{\text{new}} &= \frac{1}{N_k}\sum_{n=1}^{N}\gamma_{nk}\left(\mathbf{x}_n-\boldsymbol{\mu}_k^{\text{new}}\right)\left(\mathbf{x}_n-\boldsymbol{\mu}_k^{\text{new}}\right)^{\mathrm{T}} \\ \pi_k^{\text{new}} &= \frac{N_k}{N},\end{aligned}$$

where $N_k = \sum_{n=1}^{N}\gamma_{nk}$.

(Hint: The distribution found in the E-step, $P(z_n = k \mid \mathbf{x}_n)$, should be treated as a fixed distribution (with no parameters) in the M-step. In other words, the γ_{nk} should be treated as constants in $\mathcal{Q}(\theta)$. The model parameters should only appear in the joint distribution $P(\mathbf{x}_n, z_n = k \mid \theta)$.)

2. Implement the EM algorithm for the Gaussian mixture model in MATLAB, and use it to determine the neuron responsible for each recorded spike stored in `Snippets`. (*Hint: Use log probabilities to avoid numerical underflow.*)

   Treat each snippet as a point $\mathbf{x}_n \in \mathbf{R}^D$ $(n = 1, ..., N)$, where $D = 31$ is the number of samples in each snippet, and N is the number of detected spikes. In this problem, we will assume that there are $K = 3$ neurons contributing spikes to the recorded waveform. Initialize the model parameters using `InitParams1`.

   (a) **(25 points)** Run the EM algorithm for 100 iterations. Plot the data log likelihood $\log P(\{\mathbf{x}\} \mid \theta)$ versus EM iteration number. (Hint: The data log likelihood should increase monotonically until convergence. After convergence, the data log likelihood may *decrease* by values on the order of 1e-10 due to floating point rounding by Matlab, as discussed in class.)

   (b) **(10 points)** Once EM has converged, you will have a set of parameter estimates $\boldsymbol{\mu}_k$, Σ_k, and π_k $(k = 1, 2, 3)$. What are the π_k?

   (c) **(20 points)** For each cluster $(k = 1, 2, 3)$, create a separate "voltage versus time" plot containing the following:

   - a solid red waveform trace for the cluster center $\boldsymbol{\mu}_k$ (i.e., the prototypical action potential for the kth neuron),
   - a dotted red trace for $\boldsymbol{\mu}_k + \sqrt{\text{diag}\{\Sigma_k\}}$ and another dotted red trace for $\boldsymbol{\mu}_k - \sqrt{\text{diag}\{\Sigma_k\}}$, where $\text{diag}\{\Sigma_k\}$ is a $D \times 1$ vector containing the diagonal elements of Σ_k.

     These dotted traces show the one-standard-deviation spread of the snippets. This indicates how similar the snippets assigned to the kth neuron are at each timepoint.

- all of the waveform snippets assigned to the kth neuron.

  Note that the EM algorithm, in itself, does not provide cluster assignments; the EM algorithm only provides parameter estimates. Here, we define the following cluster assignment rule. A snippet $\mathbf{x}_n$ is assigned to the kth neuron if

  $$k = \underset{j}{\operatorname{argmax}} \; P(z_n = j \mid \mathbf{x}_n).$$

  (Hint: This will involve running one additional E-step once the EM algorithm has converged.)

3. **(10 points)** Run the EM algorithm on the same data as in Problem 2, but now initializing the model parameters using `InitParams2`. This should lead to a MATLAB error. Why did the error occur?