

Instructions: Please put all answers in a single PDF with your name and NetID and upload to SAKAI before class on the due date (there is a LaTeX template on the course web site for you to use). Definitely consider working in a group; please include the names of the people in your group and write up your solutions separately. If you look at any references (even wikipedia), cite them. If you happen to track the number of hours you spent on the homework, it would be great if you could put that at the top of your homework to give us an indication of how difficult it was.

Problem 1

EM and K-means

- (a) We wish to compute the M-step in an E-M algorithm for a Gaussian mixture model. Starting with the expected complete data log likelihood (auxiliary) function:

$$Q(\theta, \theta^{(t-1)}) = \sum_{i=1}^n \sum_{k=1}^K p(z_i = k | x_i, \theta^{(t-1)}) \log[\pi_k] + \sum_{i=1}^n \sum_{k=1}^K p(z_i = k | x_i, \theta^{(t-1)}) \log[p(x_i | \theta_k)]$$

where $\theta = \{\mu_k, \pi_k, \Sigma_k\}$
(based on Equation 11.26 on page 351 in Murphy's Book)

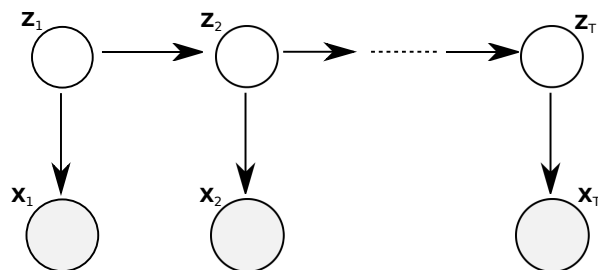
derive the maximum likelihood estimations of μ_k and π_k .

As defined in equations 11.1 and 11.2 on pages 338 – 9 in Murphy's book: each base distribution in the mixture model is a multivariate Gaussian with mean vector μ_k and covariance matrix Σ_k . And π are the mixing weights satisfying $0 \leq \pi_k \leq 1$ and $\sum_{k=1}^K \pi_k = 1$.

- (b) Compare these maximum likelihood updates for π and μ to that of K-means (Section 11.4.2.5 on page 352 in Murphy's book). How are the updates similar and how do they differ? Use 3 – 4 sentences to explain your answer.

Problem 2

Hidden Markov Model



In this problem we will be using a first-order Hidden Markov Model (HMM). Where our latent variables are denoted as $Z \in \{Z_1, \dots, Z_T\}$ and our observed variables are $X \in \{X_1, \dots, X_T\}$ as shown in the figure above. Assume we have observed data $\mathbb{D} = \{x_1, \dots, x_n\}$, where $x_i \in \{0, 1\}$. Assume the emission and transition distributions are:

$$\begin{aligned} X_t | Z_t &\sim \text{Ber}(\mu_{Z_t}) \\ Z_t | Z_{t-1} &\sim \text{Ber}(\pi_{Z_{t-1}}) \end{aligned}$$

Note: that Z_t and Z_{t-1} are subscript notations of the parameters μ and π in the Bernoulli distributions above. Our goals are to estimate parameters $\theta = \{\mu, \pi\}$, and also update our estimates as we observe more data.

- Write out the observation likelihood function ($P(X|\theta)$).
- Write out the complete data log likelihood ($\ell_c(\theta) = \log P(X, Z|\theta)$).
- Write out the expected complete data log likelihood ($Q(\theta, \theta^{t-1}) = \mathbb{E}[\ell_c(\theta) | \mathbb{D}, \theta^{t-1}]$).
- Explicitly write out the maximum likelihood updates for π and μ (the M-step), assuming the E-step is as we discussed in class.
- Clearly outline and write out the steps for updating the parameters using the Expectation - Maximization algorithm in pseudocode (but you do not need to implement it).

Problem 3

Conceptual clustering algorithm

We have seen in class that we can use the K-means algorithms to identify K clusters in a dataset. Now consider a case where we are given a data set $X \in \mathbb{R}^{1000 \times 1}$ (download from Sakai) and do not know the appropriate pre-defined cluster number, K . We propose to cluster the data, and determine the number of clusters K from the data, using the following method:

(Note that η_k is the centroid of cluster k , and K is the number of clusters, each with their own centroid: $\eta = \{\eta_1, \dots, \eta_K\}$)

Adaptive K-means algorithm:

- Initialize by labeling all of the data in one cluster (all X_i 's are assigned to one cluster and call η_0 is their centroid). Store η_0 .
- Within the range of the data (i.e., based on centroid η_0), randomly generate a new centroid, which we will call η_{K+1} .

3. Assign each X_i to one of $K+1$ clusters based on which η_k minimizes the Euclidean distance:

$$z_i = \operatorname{argmin}_k ||x_i - \eta_k||^2$$

4. Keep all centroids, η_k , that have at least one X_i assigned to it. Update the value of K based on the total number of labelled centroids. Update the retained centroids based on assigned data points, as in the standard K-means algorithm.
5. Repeat Steps 2 – 4 until you have reached a reasonable stopping place.

Question:

1. Conceptually, how many clusters do we expect to have? In other words, what value of K should we expect? Answer this before implementing.
2. When you implement this method in R, how many clusters do we end up with (please show code)?
3. How do you suggest we modify our method to prevent overfitting? Try this approach in your code and report the estimated K . Did it work?