STAT 37710/CAAM 37710/CMSC 35400 Homework 1

University of Chicago, Spring 2025 due Thursday, April 10 at 11:59pm

1. **(25 points)** In this problem, we illustrate the Expectation-Maximization (EM) algorithm using concrete examples. Suppose that the complete dataset consists of $\mathcal{Z} = (\mathcal{X}, \mathcal{Y})$, where $\mathcal{X}$ is observed but $\mathcal{Y}$ is unobserved. The log likelihood for $\mathcal{Z}$ is then denoted by $l(\theta; \mathcal{X}, \mathcal{Y})$, where θ determines the unknown parameter vector. We repeat the E-Step and M-Step below until the sequence of θ_{new} 's converges (which can be guaranteed for some cases for local maximum).

   E-Step (Expectation Step): We compute the expected value of $l(\theta; \mathcal{X}, \mathcal{Y})$, using (a) the information gained from the observed data $\mathcal{X}$, and (b) the current parameter estimate, θ_{old} . More precisely, let

$$Q\left(\theta; \theta_{\text{old}}\right) := \mathbb{E}\left[l(\theta; \mathcal{X}, \mathcal{Y}) \mid \mathcal{X}, \theta_{\text{old}}\right] = \int l(\theta; \mathcal{X}, y) p\left(y \mid \mathcal{X}, \theta_{\text{old}}\right) \mathrm{d}y. \tag{1}$$

   where $p\left(\cdot \mid \mathcal{X}, \theta_{\text{old}}\right)$ is the conditional density of $\mathcal{Y}$ given the observed data $\mathcal{X}$.

   M-Step (Maximization Step): We maximize θ over the conditional expectation (1). We simply set $\theta_{\text{new}} := \max_\theta Q\left(\theta; \theta_{\text{old}}\right)$, and afterwards, let $\theta_{\text{old}} = \theta_{\text{new}}$.

   (a) We now derive the algorithm above. Let $p(\cdot \mid \cdot)$ denote an arbitrary conditional probability density function. Show that

$$l(\theta; \mathcal{X}) = \ln p(\mathcal{X} \mid \theta) = \ln \int p(\mathcal{X}, y \mid \theta) \, \mathrm{d}y \geq Q\left(\theta; \theta_{\text{old}}\right) - \mathbb{E}\left[\ln p\left(\mathcal{Y} \mid \mathcal{X}, \theta_{\text{old}}\right) \mid \mathcal{X}, \theta_{\text{old}}\right]. \tag{2}$$

   (b) Denote the rightmost side of (2) by $g\left(\theta \mid \theta_{\text{old}}\right)$. It is clear that $l(\theta; \mathcal{X}) \geq g\left(\theta \mid \theta_{\text{old}}\right)$. Prove that we have equality when $\theta = \theta_{\text{old}}$. Why does this imply that the EM algorithm is reasonable for maximizing likelihood?

   (c) Now, consider the multinomial distribution with four classes $\text{Mult}\left(n, \pi_\theta\right)$ where

$$\pi_\theta = \left(\frac{1}{2} + \frac{1}{4}\theta, \frac{1}{4}(1-\theta), \frac{1}{4}(1-\theta), \frac{1}{4}\theta\right).$$

   let $\mathbf{x} := (x_1, x_2, x_3, x_4)$ be a sample from this distribution. Write down the likelihood $L(\theta; \mathbf{x})$, and log-likelihood $l(\theta; \mathbf{x})$, for sample $\mathbf{x}$.

   (d) We will maximize $l(\theta; \mathbf{x})$ over θ using the EM algorithm (other algorithms will receive no marks), as a toy example. To use EM, we assume that the complete data $\mathcal{Z}$ is given by $\mathbf{y} := (y_1, y_2, y_3, y_4, y_5)$ and that $\mathbf{y}$ has a 5-class $\text{Mult}\left(n, \pi^*_\theta\right)$ distribution where

$$\pi^*_\theta = \left(\frac{1}{2}, \frac{1}{4}\theta, \frac{1}{4}(1-\theta), \frac{1}{4}(1-\theta), \frac{1}{4}\theta\right).$$

   However, instead of observing $\mathbf{y}$ directly, we are only able to observe $\mathbf{x} = (y_1 + y_2, y_3, y_4, y_5)$. Therefore, we let $\mathcal{X} = (y_1 + y_2, y_3, y_4, y_5)$ and $\mathcal{Y} = y_2$, where $\mathcal{Y}$ remains unobserved. Write down the E-Step and M-Step update equations, with derivations.

2. **(15 points)** As we saw in class, k-means clustering minimizes the average square distance distortion

$$J_{\text{avg}^2} = \sum_{j=1}^{k} \sum_{\mathbf{x} \in C_j} d(\mathbf{x}, \boldsymbol{m}_j)^2, \tag{3}$$

where $d(\mathbf{x}, \mathbf{x}') = \|\mathbf{x} - \mathbf{x}'\|$, and C_j is the set of points belonging to cluster j. Another distortion function that we mentioned is the intra-cluster sum of squared distances,

$$J_{\text{IC}} = \sum_{j=1}^{k} \frac{1}{|C_j|} \sum_{\mathbf{x} \in C_j} \sum_{\mathbf{x}' \in C_j} d(\mathbf{x}, \mathbf{x}')^2.$$

(a) Given that in k-means, $\boldsymbol{m}_j = \frac{1}{|C_j|} \sum_{\mathbf{x} \in C_j} \mathbf{x}$, show that $J_{\text{IC}} = 2\, J_{\text{avg}^2}$.

(b) Let $\gamma_i \in \{1, \ldots, k\}$ be the cluster assignment of the i'th data point $\mathbf{x}_i$, and let n be the total number of data points. Then

$$J_{\text{avg}^2}(\gamma_1, \ldots, \gamma_n, \boldsymbol{m}_1, \ldots, \boldsymbol{m}_k) = \sum_{i=1}^{n} d\big(\mathbf{x}_i, \boldsymbol{m}_{\gamma_i}\big)^2.$$

Recall that k-means clustering alternates the following two steps:

1. Update the cluster assignments:

$$\gamma_i \leftarrow \arg \min_{j \in \{1, \ldots, k\}} d(\mathbf{x}_i, \boldsymbol{m}_j) \quad \forall\, i = 1, \ldots, n.$$

2. Update the centroids:

$$\boldsymbol{m}_j \leftarrow \frac{1}{|C_j|} \sum_{i:\, \gamma_i = j} \mathbf{x}_i \quad j = 1, \ldots, k.$$

Show that step 1 minimizes J_{avg^2} w.r.t. the assignments (holding $\{\boldsymbol{m}_j\}$ fixed), and step 2 minimizes J_{avg^2} w.r.t. the centroids (holding the assignments fixed).

3. **(10 points)** Implement the k-means algorithm in a language of your choice, initializing the cluster centers randomly. The algorithm terminates when no further change in cluster assignments or centroids occurs.

(a) Use the toy dataset `toydata.txt` (500 points in $\mathbb{R}^2$, from 3 well-separated clusters). Plot the final clustering assignments (by color or symbol) and also, on a separate figure, plot the distortion value vs. iteration for 20 separate runs. Comment on whether you get the "correct" clusters each time, and on the variability of results across runs.

(b) Implement k-means++ initialization and repeat part (a). Compare convergence (speed and final distortion) to the original random initialization.

(c) Run both the original and k-means++ algorithms on the MNIST dataset (images are 28×28 pixels, i.e. 784-dimensional vectors). Compare how they converge and how results differ for $k = 10$ vs. $k = 16$. You can download MNIST via

```
from torchvision import datasets
mnist_trainset = datasets.MNIST(root='./data', train=True,
                                download=True, transform=None)
mnist_testset  = datasets.MNIST(root='./data', train=False,
                                download=True, transform=None)
```

Explain any differences you observe in speed, distortion, or cluster quality.

4. **(50 points)** Recall the Gaussian mixture model for clustering

$$p(\mathbf{x}, z) \;=\; \pi_z\, \mathcal{N}\big(\mathbf{x};\, \boldsymbol{\mu}_z,\, \boldsymbol{\Sigma}_z\big),$$

with parameters $\theta = \big\{\{\pi_j\},\, \{\boldsymbol{\mu}_j\},\, \{\boldsymbol{\Sigma}_j\}\big\}_{j=1}^{k}$.

(a) Given an i.i.d. sample $\{(\mathbf{x}_i, z_i)\}_{i=1}^n$ from the model, write down the complete-data log-likelihood $\ell(\theta)$, ignoring additive constants that do not affect optimization.

(b) Let $p_{i,j} = P(z_i = j \mid \mathbf{x}_i)$. Give an expression for $p_{i,j}$ in terms of the mixture parameters.

(c) Derive the expected complete-data log-likelihood $\overline{\ell}_{\theta_{\text{old}}}(\theta)$ with respect to these posterior probabilities $p_{i,j}$.

(d) Show that maximizing $\overline{\ell}_{\theta_{\text{old}}}(\theta)$ under the constraint $\sum_j \pi_j = 1$ gives

$$\pi_j \leftarrow \frac{1}{n} \sum_{i=1}^{n} p_{i,j}.$$

(e) Similarly, derive the updates for $\boldsymbol{\mu}_j$ and $\boldsymbol{\Sigma}_j$.

(f) Compare these updates to the k-means updates from Question 2.

(g) Apply the mixture-of-Gaussians EM algorithm to the toy data and comment on how it clusters the points vs. k-means (both accuracy and convergence speed).

5. **(Extra Credit up to 20 points)**

Create a dataset for which k-means++ leads to solutions whose final distortion is at least 10 times better (on average) than random initialization. Provide the code used to generate the data.