Lecture 26: Expectation Maximization (EM algorithm)

April 12, 2016

**AIM:** Suppose we get only partial observations/samples from a parameterized population, then how can we perform efficient maximum likelihood parameter estimation?

**Applications:**

1. Machine Learning
2. Clustering (Unsupervised learning)
3. Bio-informatics, Genomics, Speech processing (Baum-Welch algorithm)

1 Estimating Mixtures of Gaussians (MoG)

The MoG model is a joint distribution on \((x, z)\) with \(x \in \mathbb{R}^d, z \in [k]\) and \(z\) has multinomial distribution,

\[ z \sim \text{Multinomial}(\phi) \]

i.e., \(\text{Multinomial}[[\phi_1, \phi_2, \ldots, \phi_k]^T]\) with \(\phi_i \geq 0 ; \sum_{j=1}^{k} \phi_j = 1\). Given \(z = j\), the random vector \(x\) is Gaussian distributed \(x|z=j \sim \mathcal{N}(\mu_j, \Sigma_j)\). Here, \(\phi\) is the mixture distribution, \(\{\mu_j\}\) is the cluster center and \(\{\Sigma_j\}\) is the cluster size.

**Example 1.1.** For \(d = k = 2\), let

\[ \mu_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} ; \mu_2 = \begin{bmatrix} -1 \\ -1 \end{bmatrix} , \]

\[ \Sigma_1 = \Sigma_2 = I_2, \text{ and } \phi = [0.5 \ 0.5] . \]

Here, cluster concentration is uniform as seen in Fig. 1, and roughly centers of clusters are \(\mu_1\) and \(\mu_2\).
Example 1.2. For $d = k = 2$, let

$$
\mu_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad \mu_2 = \begin{bmatrix} -1 \\ -1 \end{bmatrix},
$$

$$
\Sigma_1 = \Sigma_2 = I_2, \quad \phi = \begin{bmatrix} 0.25 & 0.75 \end{bmatrix}.
$$

Since the distribution is non-uniform, cluster density is also different (see Fig. 2).

Let us define parameter

$$
\theta \equiv (\phi, \mu_1, \mu_2, \ldots, \mu_k, \Sigma_1, \Sigma_2, \ldots, \Sigma_k).
$$

Suppose we only observe $x_1, x_2, \ldots, x_m \in \mathbb{R}^d$ where $(x_i, z_i) \sim$ mixture of Gaussians with parameter $\theta$ (here, $z_i$ is called “latent variable”). The goal is to find a
“maximum likelihood” estimate of $\theta$. 

$$\theta_{\text{MLE}} = \arg \max_{\theta = \phi, \mu, \Sigma} \sum_{i=1}^{m} \log p(x_i | \phi, \mu, \Sigma)$$ (2) 

$$= \arg \max_{\{\phi, \mu, \Sigma\}} \sum_{i=1}^{m} \log \sum_{z_i \in [k]} p(x_i, z_i | \phi, \mu, \Sigma)$$ (3) 

$$= \arg \max_{\{\phi, \mu, \Sigma\}} \sum_{i=1}^{m} \log \sum_{z_i=1}^{k} \phi(z_i) f(x_i | (z = z_i))$$ (4) 

where $x_i | (z = z_i) \sim \mathcal{N}(\mu_{z_i}, \Sigma_{z_i})$. This optimization is impossible to solve in closed form over $\{\phi, \mu, \Sigma\}$. However, MLE solution is easy if $\{z_i\}_{i=1}^{m}$ were observed.
Case: \( \{ z_i \}_{i=1}^m \) are observed

In this case,

\[
\hat{\theta}_{\text{MLE}} = \arg \max_{\{\phi, \mu, \Sigma\}} \sum_{i=1}^m \log p(x_i, z_i | \phi, \mu, \Sigma)
\]

\[
= \arg \max_{\{\phi, \mu, \Sigma\}} \sum_{i=1}^m \left[ \log \phi(z_i) + \log f(x_i | z_i) \right]
\]

\[
= \arg \max_{\{\phi, \mu, \Sigma\}} \sum_{i=1}^m \sum_{j=1}^k \mathbb{1}_{\{z_i = j\}} \left[ \log \phi(j) + \log f(x_i | z_i = j) \right]
\]

\[
= \arg \max_{\{\phi, \mu, \Sigma\}} \left[ \sum_{j=1}^k \log \phi(j) \sum_{i=1}^m \mathbb{1}_{\{z_i = j\}} + \sum_{j=1}^k \sum_{i=1}^m \mathbb{1}_{\{z_i = j\}} \log f(x_i | z_i = j) \right]
\]

\[
= \{ \tilde{\phi}, \tilde{\mu}, \tilde{\Sigma} \}
\]

where,

\[
\tilde{\mu}_j = \frac{\sum_{i=1}^m \mathbb{1}_{\{z_i = j\}} x_i}{\sum_{i=1}^m \mathbb{1}_{\{z_i = j\}}} \quad (10)
\]

\[
\tilde{\Sigma}_j = \frac{1}{\sum_{i=1}^m \mathbb{1}_{\{z_i = j\}}} \sum_{i=1}^m \mathbb{1}_{\{z_i = j\}} (x_i - \tilde{\mu}_j)(x_i - \tilde{\mu}_j)^T \quad (11)
\]

\[
\tilde{\phi}_j = \frac{1}{m} \sum_{i=1}^m \mathbb{1}_{\{z_i = j\}} \quad (12)
\]

Thus if \( z_1, z_2, ..., z_m \) are observed, we have an efficient way to solve this problem. This observation leads us to an algorithm that solves the ML parameter estimation problem efficiently.

2 EM algorithm

EM algorithm is an iterative algorithm involving two steps in every iteration. In the first step which is called the “E-step”, an arbitrary value for \( \theta = (\phi, \mu, \Sigma) \) is assumed to guess the values for the latent variables \( (z_1, z_2, ..., z_m) \). In the next step which is called the M-step, the guessed values for \( (z_1, z_2, ..., z_m) \) are used to find the MLE solution for \( (\phi, \mu, \Sigma) \) which is easy to find as seen in the previous section. The EM-algorithm is described in Algo. 1.

In the next section we try to answer 2 fundamental questions related EM-algorithm:
Algorithm 1 EM algorithm

1: Initialize \((\phi, \mu, \Sigma)\) arbitrarily.
2: \textbf{while} not converged \textbf{do}
3: \hspace{1em} \textbf{E-step:}
4: \hspace{2em} \(w_{ij} = \mathbb{P}[z_i = j|x_i, \phi, \mu, \Sigma], \forall i \in [m], j \in [k].\)
5: \hspace{1em} \textbf{M-step: Update}
6: \hspace{2em} \(\forall j \in [k].\)
7: \hspace{3em} \(\mu_j = \frac{1}{m} \sum_{i=1}^{m} \left( \frac{1}{\sum_{i=1}^{m} w_{ij}} w_{ij} x_i \right), \Sigma_j = \frac{1}{m} \sum_{i=1}^{m} \left( \frac{1}{\sum_{i=1}^{m} w_{ij}} w_{ij} (x_i - \mu_j)(x_i - \mu_j)^T \right),\)
8: \hspace{3em} \(\phi_j = \frac{1}{m} \sum_{i=1}^{m} w_{ij}.\)
9: \hspace{1em} Output: \(\{\mu_j, \Sigma_j, \phi_j\}\)

1. Is there a deeper principle behind EM algorithm?
2. Does it converge?

3 General EM-algorithm

Before getting into the details of the General EM-algorithm, let’s review the Jensen’s inequality which is the tool used in this algorithm.

**Definition 3.1.** Jensen’s Inequality If \(X\) is a random variable and \(f()\) is a convex function, then
\[
 f(\mathbb{E}[X]) \leq \mathbb{E}[f(X)].
\]
(f() is a convex function if \(\forall \lambda \in [0, 1] f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y)\)

Suppose we have observations \(x_1, x_2, ..., x_m\) where \((x_i, z_i) \sim f(x, z|\theta), \theta \in \Theta,\) MLE of \(\theta\) given \(x\) is,
\[
 \hat{\theta}_{MLE} = \arg \max_{\theta \in \Theta} \log L_{\theta}(x)
 = \arg \max_{\theta \in \Theta} \sum_{i=1}^{m} \log p(x_i|\theta)
 = \arg \max_{\theta \in \Theta} \sum_{i=1}^{m} \log \sum_{z_i} p(x_i, z_i|\theta)
\]
However, the MLE is easy with observed \( z = (z_1, z_2, \ldots, z_m) \), then \( \text{EM-algorithm’s strategy} \) is to construct an “easy” uniform lower bound for \( L_\theta(x) \) across \( \theta \in \Theta \) and maximize it.

For each \( i \in [m] \), let \( Q_i \) be some distribution for \( Z \). Consider,

\[
\log L_\theta(x) = \sum_{i=1}^{m} \log \sum_{z_i} p(x_i, z_i|\theta)
\]

\[
= \sum_{i=1}^{m} \log \sum_{z_i} Q(z_i) \frac{p(x_i, z_i|\theta)}{Q(z_i)}
\]

\[
\geq \sum_{i=1}^{m} \sum_{z_i} Q(z_i) \log \left[ \frac{p(x_i, z_i|\theta)}{Q(z_i)} \right] \quad \text{(By Jensen’s inequality)}.
\]

This uniform lower bound for \( \log L_\theta(x) \) is valid for any choice of \( Q_1, Q_2, \ldots, Q_m \). Suppose we choose \( Q_1, Q_2, \ldots, Q_m \) such that the lower bound is tight at some \( \theta \in \Theta \). This can be achieved, if the random variable in Jensen’s inequality is constant, which in turn implies,

\[
\forall i \in [m], \quad \frac{p(x_i, z_i|\theta)}{Q_i(z_i)} = C, \quad \text{(constant not depending on } z_i) \]

\[
Q_i(z_i) = \frac{p(x_i, z_i|\theta)}{C},
\]

\[
Q_i(z_i) = \frac{p(x_i, z_i|\theta)}{\sum_{z_i} p(x_i, z_i|\theta)}, \quad \forall z_i
\]

\[
= \frac{p(x_i, z_i|\theta)}{p(x_i|\theta)},
\]

\[
= p(z_i|x_i, \theta_t),
\]

which is the posterior probability of \( z_i \) given \( x_i \) under pdf defined by \( \theta \). The \textit{General EM-algorithm} is described in Algo. 2.

3.1 Convergence of EM-algorithm

\textit{Claim:} Suppose \( \theta_t \in \Theta \) and \( \theta_{t+1} \in \Theta \) are parameters that are the outputs of 2 successive EM iterations. Then,

\[
\log L_{\theta_t}(x) \leq \log L_{\theta_{t+1}}(x).
\]

\textit{Proof.} Consider starting at \( \theta_t \in \Theta \). Then, E-step chooses

\[
Q_i^{(t)}(z_i) = p(z_i|x_i, \theta_t).
\]
Algorithm 2 General EM algorithm

1: Initialize $\theta \in \Theta$ arbitrarily.
2: while not converged do
3:     E-step:
4:     $Q_i(z_i) = p(z_i|x_i, \theta), \forall i \in [m], \forall z_i$
5:     M-step:
6:     $\hat{\theta} = \arg\max_{\theta \in \Theta} \sum_{i=1}^{m} \sum_{z_i} Q(z_i) \log \left[ \frac{p(x_i, z_i|\theta)}{Q(z_i)} \right]$
7: Output: $\hat{\theta}$

This makes Jensen’s inequality tight at $\theta_t$. Let

$$\log L_{\theta_t}(x) = \sum_{i=1}^{m} \sum_{z_i} Q_i^{(t)}(z_i) \log \left[ \frac{p(x_i, z_i|\theta_t)}{Q_i^{(t)}(z_i)} \right] = g(\theta_t).$$

$\theta_{t+1}$ is simply the maximizer of $g()$ over $\theta \in \Theta$. Therefore, we must have

$$\log L_{\theta_{t+1}}(x) \overset{\text{Jensen’s}}{\geq} \sum_{i=1}^{m} \sum_{z_i} Q_i^{(t)}(z_i) \log \left[ \frac{p(x_i, z_i|\theta_{t+1})}{Q_i^{(t)}(z_i)} \right] = g(\theta_{t+1}) \geq g(\theta_t) = \log L_{\theta_t}(x).$$

Since $\log L_{\theta_t}(x)$ is a monotonically increasing sequence, the algorithm converges to a maximum (local) at infinity.