1 One Dimensional Mixture of Two Gaussians

Suppose we have a mixture of two Gaussians in \( \mathbb{R} \) that can be described by a pair of random variables \((X, Z)\) where \(X\) takes values in \( \mathbb{R} \) and \(Z\) takes value in the set \(1, 2\). The joint-distribution of the pair \((X, Z)\) is given to us as follows:

\[
Z \sim \text{Bernoulli}(\beta), \\
(X|Z = k) \sim \mathcal{N}(\mu_k, \sigma_k^2), \quad k \in 1, 2
\]

We use \(\theta\) to denote the set of all parameters \(\beta, \mu_1, \sigma_1, \mu_2, \sigma_2\).

(a) Write down the expression for the joint likelihood \(p_\theta(X = x_i, Z_i = 1)\) and \(p_\theta(X = x_i, Z_i = 2)\). What is the marginal likelihood \(p_\theta(X = x_i)\)?

**Solution:**

Joint likelihood:

\[
p_\theta(X = x_i, Z_i = 1) = p_\theta(X = x_i|Z_i = k)p(Z_i = 1) = \beta \mathcal{N}(x_i|\mu_1, \sigma_1^2)
\]

\[
p_\theta(X = x_i, Z_i = 2) = p_\theta(X = x_i|Z_i = 2)p(Z_i = 2) = (1 - \beta) \mathcal{N}(x_i|\mu_2, \sigma_2^2)
\]

Marginal likelihood:

\[
p_\theta(X = x_i) = \sum_k p_\theta(X = x_i, Z_i = k)
\]

\[
= \sum_k p_\theta(X = x_i|Z_i = k)p(Z_i = k)
\]

\[
= \beta \mathcal{N}(x_i|\mu_1, \sigma_1^2) + (1 - \beta) \mathcal{N}(x_i|\mu_2, \sigma_2^2)
\]

(b) What is the log-likelihood \(\ell_\theta(x)\)? Why is this hard to optimize?

**Solution:**

Log-likelihood:

\[
\ell_\theta(x) = \log(p_\theta(X = x_1, \ldots, X = x_n))
\]
\[ \sum_{i=1}^{n} \log(p_{\theta}(X = x_i)) \]
\[ = \sum_{i=1}^{n} \log [\beta \mathcal{N}(x_i | \mu_1, \sigma_1^2) + (1 - \beta) \mathcal{N}(x_i | \mu_2, \sigma_2^2)] \]

Taking the derivative with respect to \( \mu_1 \), for example, would give:
\[ \frac{\partial \ell_{\theta}(x)}{\partial \mu_1} = \sum_{i=1}^{n} \frac{\beta \mathcal{N}(x_i | \mu_1, \sigma_1^2)}{\beta \mathcal{N}(x_i | \mu_1, \sigma_1^2) + (1 - \beta) \mathcal{N}(x_i | \mu_2, \sigma_2^2)} \frac{x_i - \mu_1}{\sigma_1^2} \]

This ratio of exponentials and linear terms makes it difficult to solve for a maximum likelihood expression. Recall that there is no rule for splitting up \( \log(a + b) \) which prevents us from applying the log to the exponential.

(c) (Optional) You’d like to optimize the log likelihood: \( \ell_{\theta}(x) \). However, we just saw this can be hard to solve for an MLE closed form solution. Show that a lower bound for the log likelihood is \( \ell_{\theta}(x_i) \geq \mathbb{E}_q \left[ \log \left( \frac{p_{\theta}(X = x_i, Z_i = k)}{q_{\theta}(Z_i = k | X = x_i)} \right) \right] \).

**Solution:**
\[ \ell_{\theta}(x_i) = \log \left( \sum_k p_{\theta}(X = x_i, Z_i = k) \right) \text{ Marginalizing over possible Gaussian labels} \]
\[ = \log \left( \sum_k q_{\theta}(Z_i = k | X = x_i) p_{\theta}(X = x_i, Z_i = k) \right) \text{ Introducing arbitrary distribution q} \]
\[ = \log \left( \mathbb{E}_q \left[ \frac{p_{\theta}(X = x_i, Z_i = k)}{q_{\theta}(Z_i = k | X = x_i)} \right] \right) \text{ Rewriting as expectation} \]
\[ \geq \mathbb{E}_q \left[ \log \left( \frac{p_{\theta}(X = x_i, Z_i = k)}{q_{\theta}(Z_i = k | X = x_i)} \right) \right] \text{ Using Jensen’s inequality} \]

where Jensen’s inequality says \( \phi(\mathbb{E}[X]) \leq \mathbb{E}[\phi(X)] \) for convex function \( \phi \).

(d) (Optional) The EM algorithm first initially starts with two randomly placed Gaussians \((\mu_1, \sigma_1)\) and \((\mu_2, \sigma_2)\), which are both particular realizations of \( \theta \).

- **E-step:** \( q_{i+1}^{(k)} = p_{\theta}(Z_i = k | X = x_i) \). For each data point, determine which Gaussian generated it, being either \((\mu_1, \sigma_1)\) or \((\mu_2, \sigma_2)\).
- **M-step:** \( \theta_{i+1} = \arg\max_{\theta} \sum_{i=1}^{n} \mathbb{E}_q \left[ \log(p_{\theta}(X = x_i, Z_i = k)) \right] \). After labeling all data-points in the E-step, adjust \((\mu_1, \sigma_1)\) and \((\mu_2, \sigma_2)\).
Why does alternating between the E-step and M-step result in maximizing the lower bound?

**Solution:** To show the M-step (so-called because we are maximizing with respect to the parameters) is maximizing the lower bound:

\[
\mathbb{E}_q \left[ \log \left( \frac{p_\theta(X = x_i, Z_i = k)}{q_\theta(Z_i = k | X = x_i)} \right) \right] = \mathbb{E}_q \left[ \log(p_\theta(X = x_i)) \right] - \mathbb{E}_q \left[ \log(q_\theta(Z_i = k | X = x_i)) \right]
\]

The M-step is maximizing the first term.

To show the E-step is maximizing the bound we can rewrite the lower bound as:

\[
\mathbb{E}_q \left[ \log \left( \frac{p_\theta(X = x_i)p_\theta(Z_i = k | X = x_i)}{q_\theta(Z_i = k | X = x_i)} \right) \right] = \mathbb{E}_q \left[ \log(p_\theta(X = x_i)) \right] - \mathbb{E}_q \left[ \log \left( \frac{q_\theta(Z_i = k | X = x_i)}{p_\theta(Z_i = k | X = x_i)} \right) \right]
\]

This expression is minimized if the second term is 0, which occurs when \( q_\theta(Z_i = k | X = x_i) = P(Z_i = k | X = x_i) \).

(e) E-step: What are expressions for probabilistically imputing the classes for all the datapoints, i.e. \( q_{i,1}^{(t+1)} \) and \( q_{i,2}^{(t+1)} \)?

**Solution:**

\[
q_{i,1}^{(t+1)} = P(Z = 1 | X = x_i; \theta^t) = \frac{P(x_i | Z = 1; \theta^t)P(Z = 1)}{P(x_i | Z = 1; \theta^t)P(Z = 1) + P(x_i | Z = 2; \theta^t)P(Z = 2)}
\]

\[
q_{i,2}^{(t+1)} = P(Z = 2 | X = x_i; \theta^t) = \frac{P(x_i | Z = 2; \theta^t)P(Z = 2)}{P(x_i | Z = 1; \theta^t)P(Z = 1) + P(x_i | Z = 2; \theta^t)P(Z = 2)}
\]

where \( P(x_i | Z = 1) = \frac{1}{\sqrt{2\pi \sigma_1^2}} \exp \left( -\frac{(x_i - \mu_1)^2}{2\sigma_1^2} \right) \).

To be clear, you would have to compute \( nC \) such \( q_{i,k} \) values at each time step where \( C \) is the number of classes. Here, \( C = 2 \).

(f) What is the expression for \( \mu_1^{(t+1)} \) for the M-step?

**Solution:** From Homework 10, we know that

\[
\mu_1^{(t+1)} = \frac{\sum_{i=1}^{n} q_{i,1}^{(t+1)} x_i}{\sum_{i=1}^{n} q_{i,1}^{(t+1)}}
\]

\[
\mu_2^{(t+1)} = \frac{\sum_{i=1}^{n} q_{i,2}^{(t+1)} x_i}{\sum_{i=1}^{n} q_{i,2}^{(t+1)}}
\]

\[
\sigma_1^{(t+1)} = \frac{\sum_{i=1}^{n} q_{i,1}^{(t+1)} (x_i - \mu_1^{(t+1)})^2}{\sum_{i=1}^{n} q_{i,1}^{(t+1)}}
\]
Figure 1: EM examples in 1D for two clusters (yellow and blue). The shadings of the datapoints (circles) indicate the respective estimated probabilities of coming from either the yellow or blue cluster.
\[
(\sigma_2^2)^{(t+1)} = \frac{\sum_{i=1}^n q_{i,2}^{t+1}(x_i - \mu_2^{t+1})^2}{\sum_{i=1}^n q_{i,2}^{t+1}}
\]

We show how to obtain \(\mu_1^{t+1}\) as an example:

\[
\sum_{i=1}^n E_{q_i} \left[ \log(p_\theta(X = x_i, Z_i = k)) \right] = \sum_{i=1}^n \left[ q_{i,1}^{t+1} \log(\beta \mathcal{N}(x_i | \mu_1, \sigma_1^2)) + q_{i,2}^{t+1} \log ((1 - \beta) \mathcal{N}(x_i | \mu_2, \sigma_2^2)) \right] = \sum_{i=1}^n \left[ q_{i,1}^{t+1} \left( \log(\beta) - \frac{(x_i - \mu_1)^2}{2\sigma_1^2} - \log(\sigma_1) \right) + q_{i,2}^{t+1} \left( \log(1 - \beta) - \frac{(x_i - \mu_2)^2}{2\sigma_2^2} - \log(\sigma_2) \right) \right] + \text{constants}
\]

Taking a derivative with respect to \(\mu_1\) and setting to 0 to obtain the maximum gives:

\[
\sum_{i=1}^n q_{i,1}^{t+1} \left( \frac{x_i - \mu_1}{\sigma_1^2} \right) = 0
\]

\[
\sum_{i=1}^n q_{i,1}^{t+1} x_i - \sum_{i=1}^n q_{i,1}^{t+1} \mu_1 = 0
\]

\[
\mu_1 = \frac{\sum_{i=1}^n q_{i,1}^{t+1} x_i}{\sum_{i=1}^n q_{i,1}^{t+1}}
\]

(g) Compare and contrast k-means, soft k-means, and mixture of Gaussians fit with EM.

**Solution:** For k-means, we implicitly assume clusters are spherical and so this doesn’t work for complex geometrical shaped data. Additionally, it uses hard assignment, meaning the \(q_{i,1}\) probabilities are 0 or 1. This can be easier to interpret, but doesn’t incorporate information from all data points to update each centroid. K-means will also usually have trouble with clusters that have large overlap (see Figure 2).

For soft k-means and EM we have soft assignments. For soft k-means, the weighted mean amounts to

\[
\begin{align*}
    r_{i,1} &= \frac{\exp\{-B||x_i - \mu_1||^2\}}{\exp\{-B||x_i - \mu_1||^2\} + \exp\{-B||x_i - \mu_2||^2\}} \\
    r_{i,2} &= \frac{\exp\{-B||x_i - \mu_2||^2\}}{\exp\{-B||x_i - \mu_1||^2\} + \exp\{-B||x_i - \mu_2||^2\}}
\end{align*}
\]
Figure 2: K-means for two clusters in 1D. 'x' points indicate coming from the $\mu_1$ while 'o' indicates points coming from $\mu_2$. The colors blue and green indicate the predicted clustering. Black dots indicate the true means, while red indicates the predicted means.

$$
\mu_{t+1}^1 = \frac{\sum_{i=1}^{n} r_{i,1}^{t+1} x_i}{\sum_{i=1}^{n} r_{i,1}^{t+1}} \\
\mu_{t+1}^2 = \frac{\sum_{i=1}^{n} r_{i,2}^{t+1} x_i}{\sum_{i=1}^{n} r_{i,2}^{t+1}}
$$

where we have a stiffness parameter $\beta$, which can be interpreted as the inverse variance. In cases where the clusters have different geometry, one might resort to EM. Note that EM is not unrelated to LDA/QDA. The setup is similar in that we probabilistically determine the probabilities of coming from cluster $k$, but LDA/QDA does hard classification, EM probabilistic performs soft classification.