Mixture Models and EM algorithm

- Parameter Estimation (2)

Dr. Prakash Andugula
CSE, RAIT

Difficulty in Parameter Estimation in mixture models

Issues in parameter estimation

- Unidentifiabity (Covered in Module 3_Lecture 1)
- Non-Convex MAP Estimate

Non-Convex MAP Estimate

Consider the log-likelihood for an LVM: $\log p(\mathcal{D}|\boldsymbol{\theta}) = \sum_i \log \left[\sum_{\mathbf{z}_i} p(\mathbf{x}_i, \mathbf{z}_i|\boldsymbol{\theta}) \right]$

suppose the joint probability distribution $p(\mathbf{z}_i, \mathbf{x}_i|\boldsymbol{\theta})$ is in the exponential family $p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \exp[\boldsymbol{\theta}^T \phi(\mathbf{x}, \mathbf{z})]$

With this assumption, the **complete data log likelihood** can be written as follows:

$$\ell_c(\boldsymbol{\theta}) = \sum_i \log p(\mathbf{x}_i, \mathbf{z}_i|\boldsymbol{\theta}) = \boldsymbol{\theta}^T (\sum_i \phi(\mathbf{x}_i, \mathbf{z}_i)) - NZ(\boldsymbol{\theta}) \quad (11.14)$$

The first term is clearly linear in $\boldsymbol{\theta}$. One can show that $Z(\boldsymbol{\theta})$ is a convex function (Boyd and Vandenberghe 2004), so the overall objective is concave (due to the minus sign), and hence has a unique maximum.

Non-Convex MAP Estimate..

Now consider what happens when we have missing data. The **observed data log likelihood** is given by

$$\ell(\boldsymbol{\theta}) = \sum_i \log \sum_{\mathbf{z}_i} p(\mathbf{x}_i, \mathbf{z}_i | \boldsymbol{\theta}) = \sum_i \log \left[\sum_{\mathbf{z}_i} e^{\boldsymbol{\theta}^T \boldsymbol{\phi}(\mathbf{z}_i, \mathbf{x}_i)} \right] - N \log Z(\boldsymbol{\theta}) \tag{11.15}$$

One can show that the log-sum-exp function is convex (Boyd and Vandenberghe 2004), and we know that $Z(\boldsymbol{\theta})$ is convex. However, the difference of two convex functions is not, in general, convex. So the objective is neither convex nor concave, and has local optima.

The EM algorithm

Expectation-Maximization (EM) algorithm

- Expectation-Maximization (EM) algorithm is a powerful optimization technique used in machine learning and statistics to find the ---
- Maximum Likelihood (ML) or Maximum A Posteriori (MAP) estimates of parameters in probabilistic models, especially when the data contains latent (hidden) variables or missing information.
- The problem in many models is that if we had **complete data**, optimizing the likelihood function would be straightforward. However, when part of the data is **hidden (latent)**, computing the likelihood becomes challenging.
- The EM algorithm overcomes this by iteratively:
- E-step (Expectation Step): Estimate the missing data (or latent variables) based on the current parameter values.
- M-step (Maximization Step): Maximize the expected complete-data log-likelihood to update the model parameters.

Expectation-Maximization (EM) algorithm..

This iterative approach gradually converges to a local maximum of the likelihood function, making it useful for models like:

- **Gaussian Mixture Models (GMMs)**: Where the cluster assignment (hidden variable) is unknown.
- **Hidden Markov Models (HMMs)**: Where the state sequence is hidden.
- **Factor Analysis**: Where the latent factors are unobserved.
- **Mixture of Experts Models**: Where the responsible expert (decision-maker) is hidden

Challenge of Maximizing Log-Likelihood
Why We Need EM?

Consider a dataset $D = \{x_1, x_2, \dots, x_N\}$ with:

- **Observed data:** x_i
- **Hidden (latent) variables:** z_i
- **Model parameters:** θ

The likelihood function is expressed as:

- $\ell(\theta) = \sum_{i=1}^{N} \log p(x_i|\theta)$

Since z_i is hidden, the likelihood can be written as a marginal probability:

- $\ell(\theta) = \sum_{i=1}^{N} \log\left(\sum_{z_i} p(x_i, z_i|\theta)\right)$

However, <u>maximizing this log-likelihood</u> is difficult because:

- The logarithm is outside the sum, making direct maximization complex.
- The latent variable z_i is unknown.
- The EM algorithm resolves this by introducing an auxiliary function that separates the hidden variable estimation from parameter maximization.

Goal of the EM algorithm

The goal of the EM algorithm is to iteratively

- **E-step:** Estimate the expected value of the hidden data z_i given the current parameters $\theta^{(t-1)}$.
- **M-step:** Maximize the expected complete-data log-likelihood with respect to θ.

Goal of the EM algorithm

The goal of the EM algorithm is to iteratively

- **E-step:** Estimate the expected value of the hidden data z_i given the current parameters $\theta^{(t-1)}$.
- **M-step:** Maximize the expected complete-data log-likelihood with respect to θ.

The E-Step (Expectation Step)

In the E-step, compute the expected value of the complete-data log-likelihood under the current parameters $\theta^{(t-1)}$.

Step 1: Define the Complete-Data Log-Likelihood

If we had complete data (x_i, z_i), the complete log-likelihood would be:

- $\ell_c(\theta) = \sum_{i=1}^{N} \log p(x_i, z_i | \theta)$

However, z_i is hidden. So, instead compute the **expected log-likelihood** with respect to the latent variable distribution:

- $Q(\theta, \theta^{(t-1)}) = \mathbb{E}_{z|x,\theta^{(t-1)}}[\log p(x, z|\theta)]$

In simpler terms, :

- Use the current parameters $\theta^{(t-1)}$.
- Estimate the probability distribution of the hidden variable z.
- Take the expected log-likelihood of the complete data.

The E-Step (Expectation Step)..

Step 2: Calculate the Posterior Probability (Responsibilities)

The **posterior probability** (responsibility) that point x_i belongs to cluster k:

- $r_{ik} = p(z_i = k | x_i, \theta^{(t-1)})$

By applying **Bayes' Theorem**:

- $r_{ik} = \frac{\pi_k^{(t-1)} p(x_i | \mu_k^{(t-1)}, \Sigma_k^{(t-1)})}{\sum_{j=1}^{K} \pi_j^{(t-1)} p(x_i | \mu_j^{(t-1)}, \Sigma_j^{(t-1)})}$
  - π_k: Mixing coefficient (prior probability of cluster k)
  - $p(x_i | \mu_k, \Sigma_k)$: Gaussian probability of x_i given cluster k
  - r_{ik}: The probability that x_i belongs to cluster k

This step effectively **fills in the missing data** by assigning probabilistic responsibilities for each point.

The E-Step (Expectation Step)..

Step 3: Compute the Expected Complete-Data Log-Likelihood

- The expected complete-data log-likelihood is now:
- $Q(\theta, \theta^{(t-1)}) = \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik}\left[\log \pi_k + \log p(x_i|\mu_k, \Sigma_k)\right]$

The next step, **M-step** is used to maximize this function.

The M-Step (Maximization Step)

In the **M-step**, we maximize the expected complete-data log-likelihood $Q(\theta, \theta^{(t-1)})$ w.r.t the parameters θ.

Update Mixing Coefficient π_k

- The mixing coefficient π_k: $\pi_k = \frac{\sum_{i=1}^{N} r_{ik}}{N}$
  - $r_k = \sum_{i=1}^{N} r_{ik}$: The effective number of points in cluster k.

Update Mean μ_k

  - The new mean for cluster k: $\mu_k = \frac{\sum_{i=1}^{N} r_{ik} x_i}{r_k}$
  - This is the **weighted average** of the data points assigned to cluster k.

Update Covariance Σ_k

  - The covariance matrix for cluster k: $\Sigma_k = \frac{\sum_{i=1}^{N} r_{ik}(x_i - \mu_k)(x_i - \mu_k)^T}{r_k}$
  - This is the **weighted covariance matrix** using the posterior responsibilities.

Iteration of EM Algorithm

Iterate Until Convergence

- Repeat the E-step using the updated parameters θ.
- Repeat the M-step to maximize the likelihood.
- Continue until convergence when: $\ell(\theta^{(t)}) - \ell(\theta^{(t-1)}) < \epsilon$

  where ϵ is a small threshold indicating convergence.

Step	Description
E-step	Estimate the missing data by computing the posterior probability r_{ik}.
M-step	Maximize the expected complete-data log-likelihood by updating π_k, μ_k, Σ_k.
Iterate	Repeat until the log-likelihood converges.

References

- Murphy, Kevin P. *Machine learning: a probabilistic perspective*. MIT press, 2012.
  - Chapter 11