annotated version

Machine Learning Course - CS-433

Expectation-Maximization Algorithm

Nov 29, 2023

Martin Jaggi
Last updated on: November 28, 2023

credits to Mohammad Emtiyaz Khan & Rüdiger Urbanke

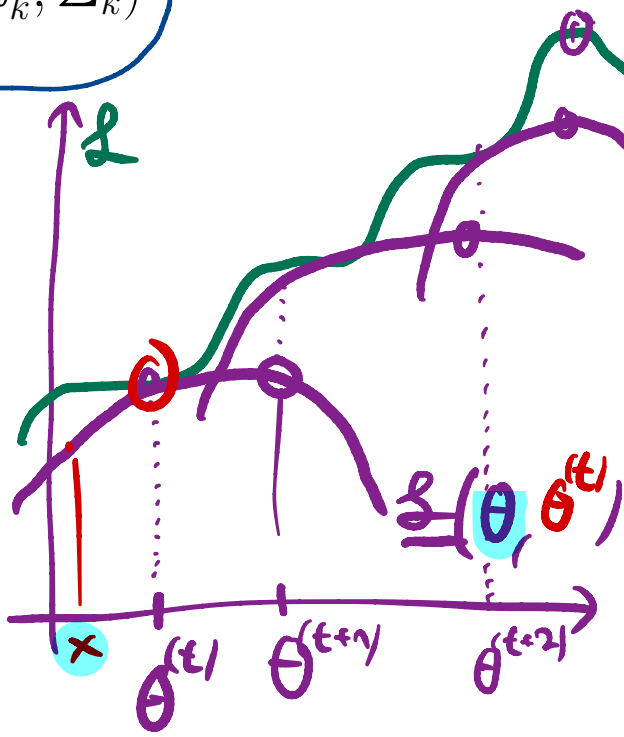
EPFL

Motivation

Computing maximum likelihood for Gaussian mixture model is difficult due to the log outside the sum.

$$\max_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) := \sum_{n=1}^{N} \log \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Expectation-Maximization (EM) algorithm provides an elegant and general method to optimize such optimization problems. It uses an iterative two-step procedure where individual steps usually involve problems that are easy to optimize.



EM algorithm: Summary

Start with $\boldsymbol{\theta}^{(1)}$ and iterate:

1. Expectation step: Compute a lower bound to the cost such that it is tight at the previous $\boldsymbol{\theta}^{(t)}$:

$$\mathcal{L}(\boldsymbol{\theta}) \geq \underline{\mathcal{L}}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(t)}) \text{ and}$$
$$\mathcal{L}(\boldsymbol{\theta}^{(t)}) = \underline{\mathcal{L}}(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}^{(t)}).$$

2. Maximization step: Update $\boldsymbol{\theta}$:

$$\boldsymbol{\theta}^{(t+1)} = \arg\max_{\boldsymbol{\theta}} \underline{\mathcal{L}}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(t)}).$$

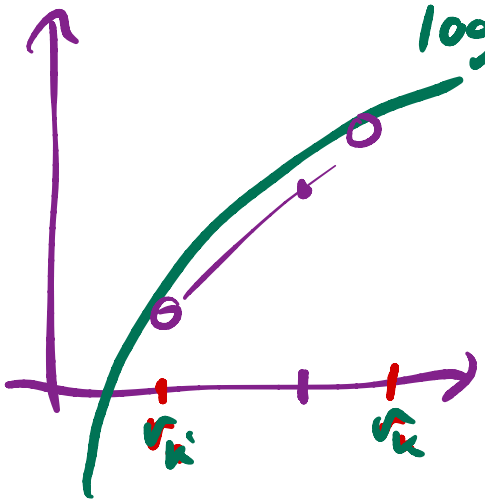
- log is convex

Concavity of log

Jensen's inequality

Given non-negative weights q s.t. $\sum_k q_k = 1$, the following holds for any $r_k > 0$:

$$\log\left(\sum_{k=1}^{K} q_k r_k\right) \geq \sum_{k=1}^{K} q_k \log r_k$$



The expectation step

$=: \mathcal{L}_n(\theta, \theta^{(t)})$

$$\log \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \geq \sum_{k=1}^{K} q_{kn} \log \frac{\pi_k \mathcal{N}(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{q_{kn}}$$

$\mathcal{L}_n$ r_k

with equality when,

$$q_{kn} = \frac{\pi_k^{(t)} \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)})}{\sum_{k=1}^{K} \pi_k^{(t)} \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)})}$$

- $\sum_k q_{kn} = 1$ ✓
- $\mathcal{L}_n(\theta^{(t)}, \theta^{(t)}) \overset{?}{=} \mathcal{L}_n(\theta^{(t)})$

This is not a coincidence.

$$= \sum_k q_{kn}^{(t)} \log(r_k)$$

$$= \sum_k \frac{\pi_k \mathcal{N}(x|\mu,\Sigma)}{\sum_k \pi_k \mathcal{N}(x|\mu,\Sigma)} \log \frac{\pi_k \mathcal{N}(x_n|\mu,\Sigma)}{\sum_k \pi_k \mathcal{N}(x_n|\mu,\Sigma)}$$

$$= \log \sum_k \pi_k \mathcal{N}(x_n|\mu_k,\Sigma_k) = \mathcal{L}_n(\theta^t)$$

The maximization step

Maximize the lower bound w.r.t. $\boldsymbol{\theta}$.

$$\max_{\boldsymbol{\theta}} \sum_{n=1}^{N} \sum_{k=1}^{K} q_{kn}^{(t)} \left[\log \pi_k + \log \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right]$$

Differentiating w.r.t. $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k^{-1}$, we can get the updates for $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$.

$$\boldsymbol{\mu}_k^{(t+1)} := \frac{\sum_n q_{kn}^{(t)} \mathbf{x}_n}{\sum_n q_{kn}^{(t)}}$$

$$\boldsymbol{\Sigma}_k^{(t+1)} := \frac{\sum_n q_{kn}^{(t)} (\mathbf{x}_n - \boldsymbol{\mu}_k^{(t+1)})(\mathbf{x}_n - \boldsymbol{\mu}_k^{(t+1)})^\top}{\sum_n q_{kn}^{(t)}}$$

For π_k, we use the fact that they sum to 1. Therefore, we add a Lagrangian term, differentiate w.r.t. π_k and set to 0, to get the following update:

$$\pi_k^{(t+1)} := \frac{1}{N} \sum_{n=1}^{N} q_{kn}^{(t)}$$

Summary of EM for GMM

Initialize $\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)}, \boldsymbol{\pi}^{(1)}$ and iterate between the E and M step, until $\mathcal{L}(\boldsymbol{\theta})$ stabilizes.

1. E-step: Compute assignments $q_{kn}^{(t)}$:

$$q_{kn}^{(t)} := \frac{\pi_k^{(t)} \mathcal{N}(\mathbf{x}_n \mid \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)})}{\sum_{k=1}^{K} \pi_k^{(t)} \mathcal{N}(\mathbf{x}_n \mid \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)})}$$

2. Compute the marginal likelihood (cost).

$$\mathcal{L}(\boldsymbol{\theta}^{(t)}) = \sum_{n=1}^{N} \log \sum_{k=1}^{K} \pi_k^{(t)} \mathcal{N}(\mathbf{x}_n \mid \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)})$$

3. M-step: Update $\boldsymbol{\mu}_k^{(t+1)}, \boldsymbol{\Sigma}_k^{(t+1)}, \pi_k^{(t+1)}$.

$$\boldsymbol{\theta}^{(t+1)} = \begin{cases} \boldsymbol{\mu}_k^{(t+1)} := \dfrac{\sum_n q_{kn}^{(t)} \mathbf{x}_n}{\sum_n q_{kn}^{(t)}} \\ \boldsymbol{\Sigma}_k^{(t+1)} := \dfrac{\sum_n q_{kn}^{(t)} (\mathbf{x}_n - \boldsymbol{\mu}_k^{(t+1)})(\mathbf{x}_n - \boldsymbol{\mu}_k^{(t+1)})^\top}{\sum_n q_{kn}^{(t)}} \\ \pi_k^{(t+1)} := \dfrac{1}{N} \sum_n q_{kn}^{(t)} \end{cases}$$

If we let the covariance be diagonal i.e. $\boldsymbol{\Sigma}_k := \sigma^2 \mathbf{I}$, then EM algorithm is same as K-means as $\sigma^2 \to 0$.

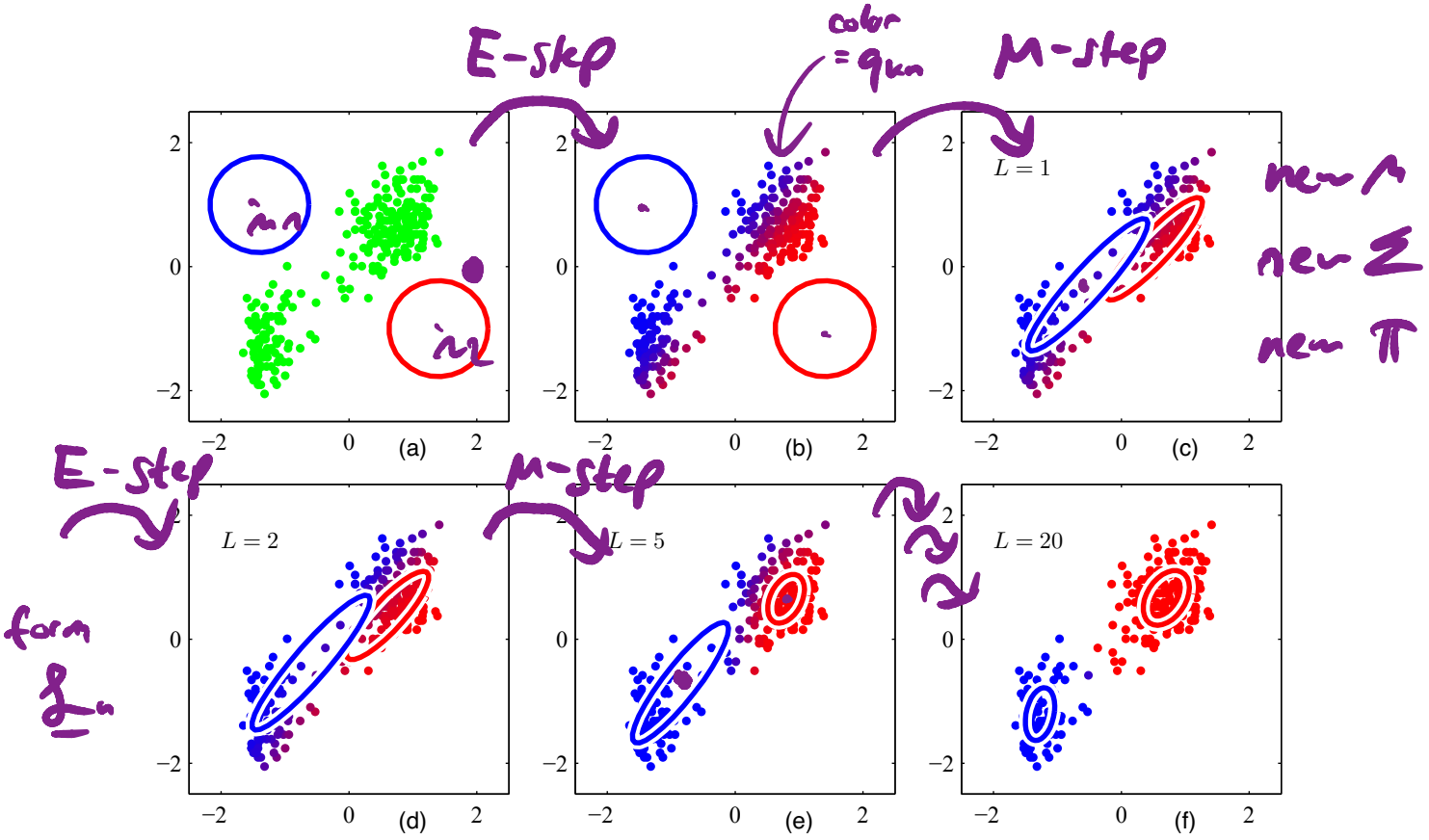


Figure 1: EM algorithm for GMM

Posterior distribution

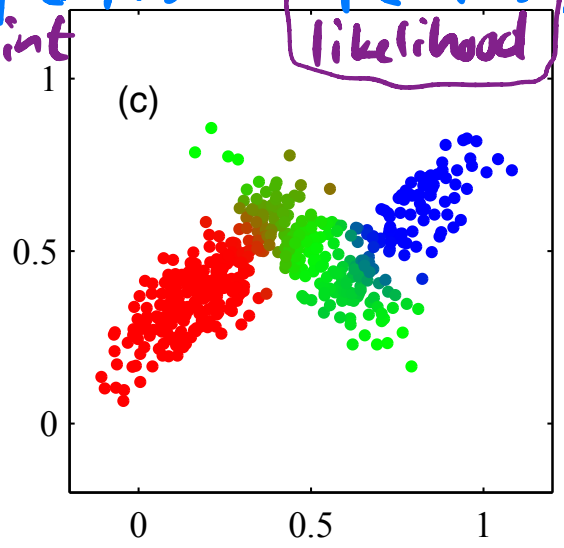
We now show that $q_{kn}^{(t)}$ is the posterior distribution of the latent variable, i.e. $q_{kn}^{(t)} = p(z_n = k \,|\, \mathbf{x}_n, \boldsymbol{\theta}^{(t)})$

$$p(\mathbf{x}_n, z_n|\boldsymbol{\theta}) = p(\mathbf{x}_n|z_n, \boldsymbol{\theta})p(z_n|\boldsymbol{\theta}) = p(z_n|\mathbf{x}_n, \boldsymbol{\theta})p(\mathbf{x}_n|\boldsymbol{\theta})$$

$p(A,B) = p(A|B)\,p(B) = p(B|A)\,p(A)$!!

joint — likelihood — prior — posterior — marginal likelihood

$\sum_{k=1}^{K} p(x_n|z_n)\,p(z_n=k)$



$$p(z_n=k|x_n,\theta) = \frac{\text{prior} \cdot \text{likelihood}}{\text{marginal LH.}} = \frac{p(z_n=k)\cdot p(x_n|z_n=k,\theta)}{\sum_k p(z)\cdot p(x|z_n=k)} = \sum_k \pi_k \mathcal{N}(x_n|\mu_k,\Sigma_k) = q_{kn}$$

EM in general

Given a general joint distribution $p(\mathbf{x}_n, z_n|\boldsymbol{\theta})$, the marginal likelihood can be lower bounded similarly:

The EM algorithm can be compactly written as follows:

$$\boldsymbol{\theta}^{(t+1)} := \arg\max_{\boldsymbol{\theta}} \sum_{n=1}^{N} \mathbb{E}_{p(z_n|\mathbf{x}_n,\boldsymbol{\theta}^{(t)})}\big[\log p(\mathbf{x}_n, z_n|\boldsymbol{\theta})\big]$$

Another interpretation is that part of the data is missing, i.e. $(\mathbf{x}_n, z_n)$ is the "complete" data and z_n is missing. The EM algorithm averages over the "unobserved" part of the data.