Machine Learning in Communications
Lecture 7: Density Estimation using GMM and Expectation Maximization

Harpreet S. Dhillon

Wireless@VT, Bradley Department of Electrical & Computer Engineering
Virginia Tech, Blacksburg, VA

https://www.dhillon.ece.vt.edu
hdhillon@vt.edu

JTG/IEEE Information Theory Society Summer School
IIT Kanpur

Lecture Objectives

- Today, we will capture clustering in data by modeling it using GMMs.
- Once we *model* the data using a GMM, the problem reduces to determining the parameters of the model.
- These parameters are determined using extremely useful idea of expectation maximization.
- Along the way, we will also connect GMMs to the k-means algorithm.
- Reference: Kevin Murphy's MLPP.

Density Estimation: Problem Setup

- Problem: For a given dataset: $\mathbf{X} = \{\boldsymbol{x}_i\}_{i=1}^n$, fit $\boldsymbol{x}_i \sim p(\boldsymbol{x}_i)$.
- We *model* $p(\boldsymbol{x}_i)$ as a mixture of Gaussians. Most of our discussion in this lecture will also be applicable to general mixture models.
- Let's recall our notation from the last module:
  - Latent variable: $z_i \in \{1, 2, \ldots, k\}$ with $p(z_i = j) = \pi_j$.
  - Likelihood: $\boldsymbol{x}_i | \{z_i = j\} \sim \mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$ is the pdf of the j^{th} Gaussian.

$$p(\boldsymbol{x}_i | z_i = j) = \frac{1}{\sqrt{(2\pi)^d \, |\boldsymbol{\Sigma}_j|}} \exp\left(-\frac{1}{2} (\boldsymbol{x}_i - \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_j) \right)$$

- Objective: Estimate the following parameters:

$$\boldsymbol{\theta} = \{\boldsymbol{\pi}, \{\boldsymbol{\mu}_j\}_{j=1}^k, \{\boldsymbol{\Sigma}_j\}_{j=1}^k\}$$

- Let's discuss maximum likelihood estimation for this problem next.

Gaussian Mixture Model: Maximum Likelihood Estimation

- The ML estimate for this problem is:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log p(\boldsymbol{x}_i|\boldsymbol{\theta})$$

$$= \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log \sum_{j=1}^{k} p(\boldsymbol{x}_i, z_i = j|\boldsymbol{\theta})$$

  - The log-sum form is problematic. We don't get nice factors as before.
- In order to understand this, let's assume for the sake of the argument that we had labelled data $\{(\boldsymbol{x}_i, z_i)\}$, i.e., z_i is not "latent". The ML estimate in this case is:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log p(\boldsymbol{x}_i|z_i, \boldsymbol{\theta}) + \sum_{i=1}^{n} \log p(z_i|\boldsymbol{\theta})$$

  - Problem becomes easier when we have "complete data". We will use this fact.

One Way to Get Rid of Log-Sum Form: Hard Assignment

- Let's assume hard assignment of points to clusters (like in k-means):

$$p(\boldsymbol{x}_i, z_i = j) = \begin{cases} p(\boldsymbol{x}_i, z_i = c_i) & j = c_i \\ 0 & o.w. \end{cases}$$

- In order to make the connection of GMM with k-means even more concrete, let's consider the following two additional assumptions:
  - $p(z_i = j) = \pi_j = \frac{1}{k}, \forall j$
  - Assume "spherical" Gaussians:

$$\boldsymbol{\Sigma}_j = \boldsymbol{\Sigma} = \begin{bmatrix} \sigma^2 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma^2 \end{bmatrix}$$

- This gives us

$$p(\boldsymbol{x}_i | z_i = j) = \frac{1}{(2\pi\sigma^2)^{\frac{d}{2}}} \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{\mu}_j\|^2}{2\sigma^2} \right)$$

Log Likelihood under these Assumptions

- ML estimate for this case is:

$$\begin{aligned}\boldsymbol{\theta}^* &= \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log p(\boldsymbol{x}_i|\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log \sum_{j=1}^{k} p(\boldsymbol{x}_i, z_i = j|\boldsymbol{\theta}) \\ &= \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log p(\boldsymbol{x}_i, z_i = c_i|\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log p(\boldsymbol{x}_i|z_i = c_i, \boldsymbol{\theta})\pi_{c_i} \\ &= \arg\max_{\boldsymbol{\theta}} -\frac{1}{n} \sum_{i=1}^{n} \frac{\|\boldsymbol{x}_i - \boldsymbol{\mu}_{c_i}\|^2}{2\sigma^2}\end{aligned}$$

- Recall that this is just the (negative of) average distortion that we approximately minimized using k-means. Now you know the implicit assumptions made while using k means algorithm.

Expectation Maximization: General Idea

- Very useful algorithm for density estimation.
- No need to make any assumptions along the lines of what we just did for k-means.
- It is also applicable to general mixture models (beyond GMM).
- Here is the general idea:
  - E step: Find soft assignment of points to Gaussians, $p(z_i = j|\boldsymbol{x}_i, \boldsymbol{\theta})$, so that we can write Expected Complete Data Log Likelihood.
    - This just assigns soft labels to the data. This is said to "complete" the data.
  - M step: Maximize this expected log likelihood to update parameters.
- It is natural to wonder why do we take expectation of the log here. We will understand this soon.

Expectation Maximization: Steps

- Let's first understand how EM works.
- Step 0: Initialize $\boldsymbol{\theta}$ as $\boldsymbol{\theta}^{(0)} = \left\{ \boldsymbol{\pi}^{(0)}, \{\boldsymbol{\mu}_j^{(0)}\}_{j=1}^k, \{\boldsymbol{\Sigma}_j^{(0)}\}_{j=1}^k \right\}$.
- E step: At time t, we have $\boldsymbol{\theta}^{(t)}$. Given this, find the assignment probability of i^{th} point to the j^{th} Gaussian:

$$a_{ij}^{(t)} = p(z_i = j | \boldsymbol{x}_i, \boldsymbol{\theta}^{(t)}) = \frac{p(\boldsymbol{x}_i, z_i = j | \boldsymbol{\theta}^{(t)})}{p(\boldsymbol{x}_i | \boldsymbol{\theta}^{(t)})}$$
$$= \frac{p(z_i = j | \boldsymbol{\theta}^{(t)}) p(\boldsymbol{x}_i | z_i = j, \boldsymbol{\theta}^{(t)})}{\sum_{j=1}^k p(z_i = j | \boldsymbol{\theta}^{(t)}) p(\boldsymbol{x}_i | z_i = j, \boldsymbol{\theta}^{(t)})}$$

- Since $p(z_i = j | \boldsymbol{\theta}^{(t)}) = \pi_j$ and $p(\boldsymbol{x}_i | z_i = j, \boldsymbol{\theta}^{(t)})$ is $\mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$, we have everything that we need to compute $a_{ij}^{(t)}$ for a given $\boldsymbol{\theta}^{(t)}$.
- This gives us *soft* assignments of points to Gaussians and thus completes our data.

Expectation Maximization: Steps

- M step: From the E-step, we get the expected (complete data) log likelihood. Assuming the assignments to be fixed $a_{ij}^{(t)}$, we determine $\boldsymbol{\theta}^{(t+1)}$ by maximizing the expected log likelihood as:

$$\boldsymbol{\theta}^{(t+1)} = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \sum_{j=1}^{k} a_{ij}^{(t)} \log p(\boldsymbol{x}_i, z_i = j | \boldsymbol{\theta})$$

- This reduces the problem to MLE with Gaussians:

$$\hat{\pi}_j^{(t+1)} = \frac{\sum_{i=1}^{n} a_{ij}^{(t)}}{\sum_{i=1}^{n} \sum_{j=1}^{k} a_{ij}^{(t)}} = \frac{1}{n} \sum_{i=1}^{n} a_{ij}^{(t)}$$

$$\hat{\boldsymbol{\mu}}_j^{(t+1)} = \frac{\sum_{i=1}^{n} a_{ij}^{(t)} \boldsymbol{x}_i}{\sum_{i=1}^{n} a_{ij}^{(t)}}$$

$$\hat{\boldsymbol{\Sigma}}_j^{(t+1)} = \frac{\sum_{i=1}^{n} a_{ij}^{(t)} (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_j^{(t+1)})(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_j^{(t+1)})^T}{\sum_{i=1}^{n} a_{ij}^{(t)}}$$

Expected Log Likelihood is a (Special) Lower Bound

- Recall the M step from the previous slide:

$$\boldsymbol{\theta}^{(t+1)} = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \sum_{j=1}^{k} \underbrace{p(z_i = j | \boldsymbol{x}_i, \boldsymbol{\theta}^{(t)})}_{a_{ij}^{(t)}} \log p(\boldsymbol{x}_i, z_i = j | \boldsymbol{\theta}^{(t)})$$

- Now, let's revisit our original objective function:

$$\boldsymbol{\theta}^{*} = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log \sum_{j=1}^{k} p(\boldsymbol{x}_i, z_i = j | \boldsymbol{\theta})$$

Applying Jensen's Inequality

Consider a generic distribution $\mathbf{q}_i = \{q_i(j)\}$ and express the original objective as

$$\begin{aligned}\boldsymbol{\theta}^* &= \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log \sum_{j=1}^{k} p(\boldsymbol{x}_i, z_i = j|\boldsymbol{\theta}) \frac{q_i(j)}{q_i(j)} \\ &= \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \underbrace{\log \mathbb{E}_{\mathbf{q_i}} \frac{p(\boldsymbol{x}_i, z_i = j|\boldsymbol{\theta})}{q_i(j)}}_{\geq \mathbb{E}_{\mathbf{q_i}} \log \frac{p(\boldsymbol{x}_i, z_i = j|\boldsymbol{\theta})}{q_i(j)} \text{ by Jensen's Inequality}}\end{aligned}$$

Let's look at this lower bound carefully next. The idea is select $\mathbf{q}_i = \{q_i(j)\}$ that provides the tightest lower bound.

Lower Bound

Here is the lower bound on the original likelihood that we derived using Jensen's inequality:

$$\sum_{i=1}^{n}\sum_{j=1}^{k} q_i(j)\log\frac{p(\boldsymbol{x}_i, z_i = j|\boldsymbol{\theta})}{q_i(j)}$$

$$= \underbrace{\sum_{i=1}^{n}\sum_{j=1}^{k} q_i(j)\log\frac{p(z_i = j|\boldsymbol{x}_i, \boldsymbol{\theta})}{q_i(j)}}_{-\mathrm{KL}(q_i \| p(z_i|\boldsymbol{x}_i, \boldsymbol{\theta}))} + \underbrace{\sum_{i=1}^{n}\sum_{j=1}^{k} q_i(j)\log p(\boldsymbol{x}_i|\boldsymbol{\theta})}_{\sum_{i=1}^{n}\log p(\boldsymbol{x}_i|\boldsymbol{\theta}) \quad \text{Original Objective}}$$

The first term goes to zero when we select $q_i(j) = p(z_i = j|\boldsymbol{x}_i, \boldsymbol{\theta}) = a_{ij}$. This will give us the tightest lower bound, which will touch the original objective.

Lower Bound is Expected Complete Data Log Likelihood

Our lower bound from the previous slide was

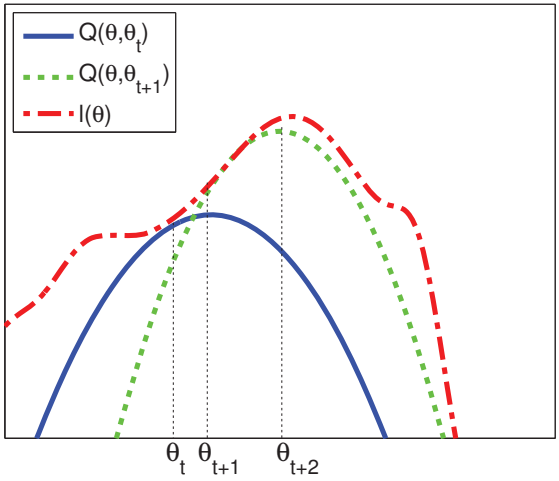
$$\sum_{i=1}^{n}\sum_{j=1}^{k} q_i(j)\log\frac{p(\boldsymbol{x}_i, z_i = j|\boldsymbol{\theta})}{q_i(j)}$$

$$= \sum_{i=1}^{n}\sum_{j=1}^{k} q_i(j)\log p(\boldsymbol{x}_i, z_i = j|\boldsymbol{\theta}) - \sum_{i=1}^{n}\sum_{j=1}^{k} q_i(j)\log q_i(j)$$

The second term can be ignored when we do arg max since it does not depend upon $\boldsymbol{\theta}$. The first term is our Expected Log Likelihood when we substitute $q_i(j) = a_{ij}$ from the previous slide. This recovers the M step:

$$\boldsymbol{\theta}^{(t+1)} = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n}\sum_{j=1}^{k} a_{ij}\log p(\boldsymbol{x}_i, z_i = j|\boldsymbol{\theta})$$

Expectation Maximization: Illustration of the Lower Bound



[MLPP Figure 11.16] Illustration of EM.

Summary

With this lecture, we conclude our discussion of the basics of unsupervised learning. We will go over a case study on distributed learning in wireless networks in the next lecture. In this lecture, we have covered:

- Gaussian mixture models.
- Interpretation of k means in terms of a specific GMM.
- Density estimation using expectation maximization.