

## Lecture 11: Density Estimation with Gaussian Mixture Models

Yi, Yung (이웅)

Mathematics for Machine Learning  
<https://yung-web.github.io/home/courses/mathml.html>  
KAIST EE

April 7, 2021

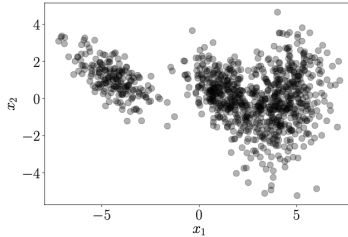
Please watch this tutorial video by Luis Serrano on Gaussian Mixture Model.

<https://www.youtube.com/watch?v=q71Niz856KE>

- (1) Gaussian Mixture Model
- (2) Parameter Learning: MLE
- (3) Latent-Variable Perspective for Probabilistic Modeling
- (4) EM Algorithm

- (1) **Gaussian Mixture Model**
- (2) Parameter Learning: MLE
- (3) Latent-Variable Perspective for Probabilistic Modeling
- (4) EM Algorithm

- Represent data compactly using a density from a parametric family, e.g., Gaussian or Beta distribution
- Parameters of those families can be found by MLE and MAPE
- However, there are many cases when simple distributions (e.g., just Gaussian) fail to approximate data.



L11(1)

April 7, 2021 5 / 26

- More expressive family of distribution
- Idea: Let's mix! A **convex combination** of  $K$  "base" distributions

$$p(\mathbf{x}) = \sum_{k=1}^K \pi_k p_k(\mathbf{x}), \quad 0 \leq \pi_k \leq 1, \quad \sum_{k=1}^K \pi_k = 1$$

- Multi-modal distributions: Can be used to describe datasets with multiple clusters
- Our focus: Gaussian mixture models
- Want to finding the parameters using MLE, but **cannot have the closed form** solution (even with the mixture of Gaussians) → some iterative methods needed

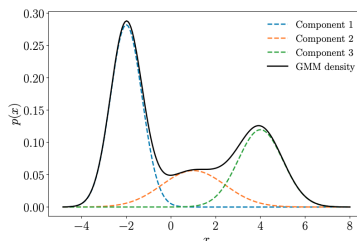
L11(1)

April 7, 2021 6 / 26

$$p(\mathbf{x}|\boldsymbol{\theta}) = \sum_{k=1}^K \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \quad 0 \leq \pi_k \leq 1, \quad \sum_{k=1}^K \pi_k = 1,$$

where the parameters  $\boldsymbol{\theta} := \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k : k = 1, \dots, K\}$

- **Example.**  $p(x|\boldsymbol{\theta}) = 0.5\mathcal{N}(x|-2, 1/2) + 0.2\mathcal{N}(x|1, 2) + 0.3\mathcal{N}(x|4, 1)$



L11(1)

April 7, 2021 7 / 26

- (1) Gaussian Mixture Model
- (2) **Parameter Learning: MLE**
- (3) Latent-Variable Perspective for Probabilistic Modeling
- (4) EM Algorithm

L11(2)

April 7, 2021 8 / 26

- Given a iid dataset  $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ , the log-likelihood is:

$$\mathcal{L}(\theta) = \log p(\mathcal{X}|\theta) = \sum_{n=1}^N \log p(\mathbf{x}_n|\theta) = \sum_{n=1}^N \log \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n|\mu_k, \Sigma_k)$$

- $\theta_{\text{ML}} = \arg \min_{\theta} (-\mathcal{L}(\theta))$
- Necessary condition for  $\theta_{\text{ML}}$ :  $\left. \frac{d\mathcal{L}}{d\theta} \right|_{\theta_{\text{ML}}} = 0$
- However, the closed-form solution of  $\theta_{\text{ML}}$  does not exist, so we rely on an iterative algorithm (also called EM algorithm).
- We show the algorithm first, and then discuss how we get the algorithm.

- Definition. Responsibilities.** Given  $n$ -th data point  $\mathbf{x}_n$  and the parameters  $(\mu_k, \Sigma_k, \pi_k : k = 1, \dots, K)$ ,

$$r_{nk} = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\mu_k, \Sigma_k)}{\sum_j \pi_j \mathcal{N}(\mathbf{x}_n|\mu_j, \Sigma_j)}$$

- How much is each component  $k$  responsible, if the data  $\mathbf{x}_n$  is sampled from the current mixture model?
- $\mathbf{r}_n = (r_{nk} : k = 1, \dots, K)$  is a probability distribution, so  $\sum_{k=1}^K r_{nk} = 1$
- Soft assignment of  $\mathbf{x}_n$  to the  $K$  mixture components

L11(2)

April 7, 2021 9 / 26

L11(2)

April 7, 2021 10 / 26

## EM Algorithm: MLE in Gaussian Mixture Models

## Example: EM Algorithm

## EM for MLE in Gaussian Mixture Models

- S1.** Initialize  $\mu_k, \Sigma_k, \pi_k$
- S2. E-step:** Evaluate responsibilities  $r_{nk}$  for every data point  $\mathbf{x}_n$  using the current  $\mu_k, \Sigma_k, \pi_k$ :

$$r_{nk} = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\mu_k, \Sigma_k)}{\sum_j \pi_j \mathcal{N}(\mathbf{x}_n|\mu_j, \Sigma_j)}, \quad N_k = \sum_{n=1}^N r_{nk}$$

- S3. M-step:** Reestimate parameters  $\mu_k, \Sigma_k, \pi_k$  using the current responsibilities  $r_{nk}$ :

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^N r_{nk} \mathbf{x}_n, \quad \Sigma_k = \frac{1}{N_k} \sum_{n=1}^N r_{nk} (\mathbf{x}_n - \mu_k)(\mathbf{x}_n - \mu_k)^T, \quad \pi_k = \frac{N_k}{N},$$

and go to **S2**.

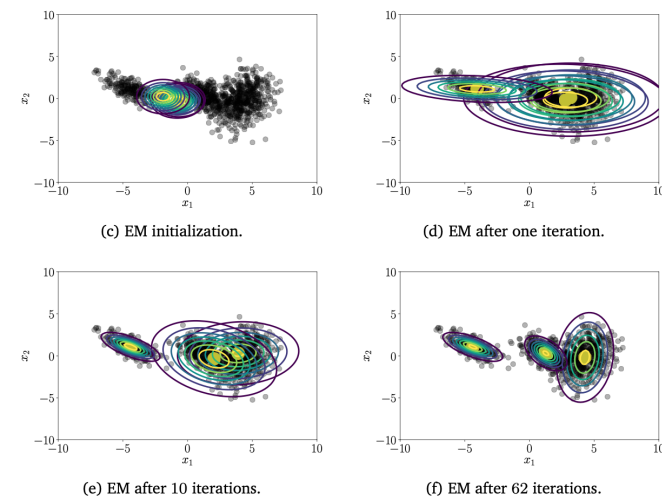
- The update equation in **M-step** is still mysterious, which will be covered later.

L11(2)

April 7, 2021 11 / 26

L11(2)

April 7, 2021 12 / 26



- Given  $\mathcal{X}$  and  $r_{nk}$  from E-step, the new updates of  $\mu_k$ ,  $\Sigma_k$ ,  $\pi_k$  should be made, such that the followings are satisfied:

$$\frac{\partial \mathcal{L}}{\partial \mu_k} = 0^T \iff \sum_{n=1}^N \frac{\partial \log p(\mathbf{x}_n | \theta)}{\partial \mu_k} = 0^T$$

$$\frac{\partial \mathcal{L}}{\partial \Sigma_k} = 0 \iff \sum_{n=1}^N \frac{\partial \log p(\mathbf{x}_n | \theta)}{\partial \Sigma_k} = 0$$

$$\frac{\partial \mathcal{L}}{\partial \pi_k} = 0 \iff \sum_{n=1}^N \frac{\partial \log p(\mathbf{x}_n | \theta)}{\partial \pi_k} = 0$$

- Nice thing: the new updates of  $\mu_k$ ,  $\Sigma_k$ ,  $\pi_k$  are all expressed by the responsibilities  $[r_{nk}]$
- Let's take a look at them one by one!

$$\mu_k^{\text{new}} = \frac{\sum_{n=1}^N r_{nk} \mathbf{x}_n}{\sum_{n=1}^N r_{nk}}, k = 1, \dots, K$$

•

$$\Sigma_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^N r_{nk} (\mathbf{x}_n - \mu_k)(\mathbf{x}_n - \mu_k)^T, k = 1, \dots, K$$

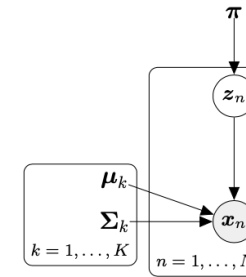
•

$$\pi_k^{\text{new}} = \frac{\sum_{n=1}^N r_{nk}}{N}, k = 1, \dots, K$$

•

- (1) Gaussian Mixture Model
- (2) Parameter Learning: MLE
- (3) Latent-Variable Perspective for Probabilistic Modeling
- (4) EM Algorithm

- Justify some ad hoc decisions made earlier
- Allow for a concrete interpretation of the responsibilities as **posterior distributions**
- Iterative algorithm for updating the model parameters can be derived in a principled manner



L11(3)

April 7, 2021 17 / 26

L11(3)

April 7, 2021 18 / 26

## Generative Process

- Latent variable  $\mathbf{z}$ :** One-hot encoding random vector  $\mathbf{z} = [z_1, \dots, z_K]^T$  consisting of  $K - 1$  many 0s and exactly one 1.
- An indicator rv  $z_k = 1$  represents whether  $k$ -th component is used to generate the data sample  $\mathbf{x}$  or not.
- $p(\mathbf{x}|z_k = 1) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$
- Prior for  $\mathbf{z}$  with  $\pi_k = p(z_k = 1)$

$$p(\mathbf{z}) = \boldsymbol{\pi} = [\pi_1, \dots, \pi_K]^T, \quad \sum_{k=1}^K \pi_k = 1$$

- Sampling procedure
  - Sample which component to use  $z^{(i)} \sim p(\mathbf{z})$
  - Sample data according to  $i$ -th Gaussian  $\mathbf{x}^{(i)} \sim p(\mathbf{x}|z^{(i)})$

L11(3)

April 7, 2021 19 / 26

## Joint Distribution, Likelihood, and Posterior (1)

- Joint distribution

$$p(\mathbf{x}, \mathbf{z}) = \begin{pmatrix} p(\mathbf{x}, z_1 = 1) \\ \vdots \\ p(\mathbf{x}, z_K = 1) \end{pmatrix} = \begin{pmatrix} p(\mathbf{x}|z_1 = 1)p(z_1 = 1) \\ \vdots \\ p(\mathbf{x}|z_K = 1)p(z_K = 1) \end{pmatrix} = \begin{pmatrix} \pi_1 \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) \\ \vdots \\ \pi_K \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_K, \boldsymbol{\Sigma}_K) \end{pmatrix}$$

- Likelihood for an arbitrary single data  $\mathbf{x}$ : By summing out all latent variables<sup>1</sup>,

$$p(\mathbf{x}|\boldsymbol{\theta}) = \sum_{\mathbf{z}} p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{z})p(\mathbf{z}|\boldsymbol{\theta}) = \sum_{k=1}^K p(\mathbf{x}|\boldsymbol{\theta}, z_k = 1)p(z_k = 1|\boldsymbol{\theta}) = \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- For all the data samples  $\mathcal{X}$ , the log-likelihood is:

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

Compare: Page 7

<sup>1</sup>In probabilistic PCA,  $\mathbf{z}$  was continuous, so we integrated them out.

L11(3)

April 7, 2021 20 / 26

- Posterior for the  $k$ -th  $z_k$ , given an arbitrary single data  $\mathbf{x}$ :

$$p(z_k = 1|\mathbf{x}) = \frac{p(z_k = 1)p(\mathbf{x}|z_k = 1)}{\sum_{j=1}^K p(z_j = 1)p(\mathbf{x}|z_j = 1)} = \frac{\pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

- Now, for all data samples  $\mathcal{X}$ , each data  $\mathbf{x}_n$  has  $\mathbf{z}_n = [z_{n1}, \dots, z_{nK}]^T$ , but with the same prior  $\pi$ .

$$p(z_{nk} = 1|\mathbf{x}_n) = \frac{p(z_{nk} = 1)p(\mathbf{x}_n|z_{nk} = 1)}{\sum_{j=1}^K p(z_{nj} = 1)p(\mathbf{x}_n|z_{nj} = 1)} = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} = r_{nk}$$

- Responsibilities are mathematically interpreted as **posterior distributions**.

- (1) Gaussian Mixture Model
- (2) Parameter Learning: MLE
- (3) Latent-Variable Perspective for Probabilistic Modeling
- (4) **EM Algorithm**

L11(3)

April 7, 2021 21 / 26

L11(4)

April 7, 2021 22 / 26

## Revisiting EM Algorithm for MLE

## Other Issues

**S1.** Initialize  $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k$

**S2. E-step:**

$$r_{nk} = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_j \pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

**S3. M-step:** Update  $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k$  using  $r_{nk}$  and go to **S2**.

- E-step. Expectation** over  $\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^{(t)}$ : Given the current  $\boldsymbol{\theta}^{(t)} = (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k)$ , calculates the expected log-likelihood

$$\begin{aligned} Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) &= \mathbb{E}_{\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^{(t)}}[\log p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})] \\ &= \int \log p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^{(t)}) d\mathbf{z} \end{aligned}$$

- M-step. Maximization** of the computation results in E-step for the new model parameters.

- Only guarantee of just local-optimum because the original optimization is not necessarily a convex optimization.

L7(4)

- Model selection for finding a good  $K$ , e.g., using nested cross-validation
- Application: Clustering
  - K-means: Treat the means in GMM as cluster centers and ignore the covariances.
  - K-means: hard assignment, GMM: soft assignment
- EM algorithm: Highly generic in the sense that it can be used for parameter learning in general latent-variable models
- Standard criticism for MLE exists such as overfitting. Also, fully-Bayesian approach assuming some priors on the parameters is possible, but not covered in this notes.
- Other density estimation methods
  - Histogram-based method: non-parametric method
  - Kernel-density estimation: non-parametric method

L11(4)

April 7, 2021 23 / 26

L11(4)

April 7, 2021 24 / 26

Questions?

1)