Introduction to Machine Learning

Lecture 9 Representation and Clustering - Gaussian Mixture Models and EM Algorithm

Hongteng Xu

中国人民大學 RENMIN UNIVERSITY OF CHINA

高瓴人工智能学院 Gaoling School of Artificial Intelligence

Outline

Review

- Kmeans
- Spectral clustering and its connections to Kmeans and manifold learning
- Evaluation of clustering results

Outline

Review

- Kmeans
- Spectral clustering and its connections to Kmeans and manifold learning
- Evaluation of clustering results

Today

- Generative modeling and Gaussian mixture model
- EM algorithm

Generative Model v.s. Discriminative Model

Denote X as the random variable of samples, optionally Y as the random variable of labels.

- **Generative Model:**
  - **Functionality:** Capture the mechanism of generating data

Generative Model v.s. Discriminative Model

Denote X as the random variable of samples, optionally Y as the random variable of labels.

- **Generative Model:**
  - **Functionality:** Capture the mechanism of generating data
  - **Principle:** Model the data distribution $P(X)$ or the joint distribution $P(X, Y)$

Generative Model v.s. Discriminative Model

Denote X as the random variable of samples, optionally Y as the random variable of labels.

- **Generative Model:**
  - **Functionality:** Capture the mechanism of generating data
  - **Principle:** Model the data distribution $P(X)$ or the joint distribution $P(X, Y)$
- **Discriminative Model:**
  - **Functionality:** Capture the difference between different data points

Generative Model v.s. Discriminative Model

Denote X as the random variable of samples, optionally Y as the random variable of labels.

- **Generative Model:**
  - **Functionality:** Capture the mechanism of generating data
  - **Principle:** Model the data distribution $P(X)$ or the joint distribution $P(X, Y)$
- **Discriminative Model:**
  - **Functionality:** Capture the difference between different data points
  - **Principle:** Model the conditional probability $P(Y|X)$

Generative Model v.s. Discriminative Model

Denote X as the random variable of samples, optionally Y as the random variable of labels.

- **Generative Model:**
  - **Functionality:** Capture the mechanism of generating data
  - **Principle:** Model the data distribution $P(X)$ or the joint distribution $P(X, Y)$
- **Discriminative Model:**
  - **Functionality:** Capture the difference between different data points
  - **Principle:** Model the conditional probability $P(Y|X)$
- How to categorize the models we learned before?

Generative Model v.s. Discriminative Model

Denote X as the random variable of samples, optionally Y as the random variable of labels.

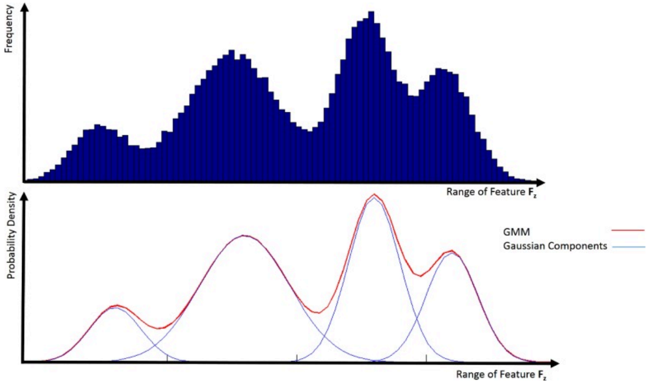
- **Generative Model:**
  - **Functionality:** Capture the mechanism of generating data
  - **Principle:** Model the data distribution $P(X)$ or the joint distribution $P(X, Y)$
- **Discriminative Model:**
  - **Functionality:** Capture the difference between different data points
  - **Principle:** Model the conditional probability $P(Y|X)$
- How to categorize the models we learned before?
- Which one is harder?

Let's design the two modeling strategies in practice

- **Data:** Given $\{x_n, y_n\}_{n=1}^{N}$, where x's represent education years, and y's represent yearly incomes.
- **Task:** Learn an estimator predicting yearly incomes based on education years.

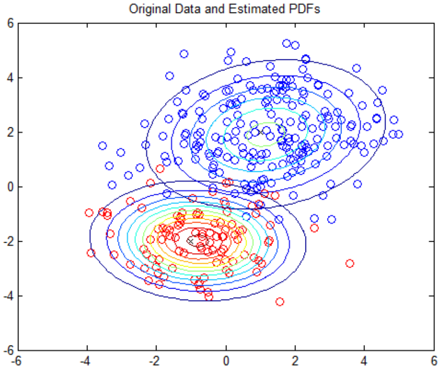
Gaussian Mixture Models

A generative model for the data with clustering structures



Gaussian Mixture Models

A generative model for the data with clustering structures



GMM: Generative Mechanism

Suppose that there are K Gaussian distributions defined on the sample space $\mathcal{X} \subset \mathbb{R}^D$.

- $\boldsymbol{w} = [w_k] \in \Delta^{K-1}$
- $\{\mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\}_{k=1}^{K}$

GMM: Generative Mechanism

Suppose that there are K Gaussian distributions defined on the sample space $\mathcal{X} \subset \mathbb{R}^D$.

- $\boldsymbol{w} = [w_k] \in \Delta^{K-1}$
- $\{\mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\}_{k=1}^K$

Generative process:

1. Determine the cluster: $k \sim \text{Categorical}(\boldsymbol{w})$
2. Determine the sample based on the cluster: $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$.

GMM: Generative Mechanism

Suppose that there are K Gaussian distributions defined on the sample space $\mathcal{X} \subset \mathbb{R}^D$.

- $\boldsymbol{w} = [w_k] \in \Delta^{K-1}$
- $\{\mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\}_{k=1}^K$

Generative process:

1. Determine the cluster: $k \sim \text{Categorical}(\boldsymbol{w})$
2. Determine the sample based on the cluster: $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$.

How to achieve the above sampling processes?

The MLE Framework of GMM

What is the probability of x? (Derive it)

The MLE Framework of GMM

What is the probability of $\boldsymbol{x}$? (Derive it)

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} p(k) p(\boldsymbol{x}|k)$$

The MLE Framework of GMM

What is the probability of $\boldsymbol{x}$? (Derive it)

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} p(k) p(\boldsymbol{x}|k) = \sum_{k=1}^{K} w_k \det(2\pi \boldsymbol{\Sigma}_k)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_k)\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_k)\right) \tag{1}$$

The MLE Framework of GMM

What is the probability of $\boldsymbol{x}$? (Derive it)

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} p(k)p(\boldsymbol{x}|k) = \sum_{k=1}^{K} w_k \det(2\pi\boldsymbol{\Sigma}_k)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_k)\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_k)\right) \tag{1}$$

Given $\{\boldsymbol{x}_n\}_{n=1}^{N}$, what is the likelihood function of model parameters?

The MLE Framework of GMM

What is the probability of $\boldsymbol{x}$? (Derive it)

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} p(k)p(\boldsymbol{x}|k) = \sum_{k=1}^{K} w_k \det(2\pi\boldsymbol{\Sigma}_k)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_k)\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_k)\right) \tag{1}$$

Given $\{\boldsymbol{x}_n\}_{n=1}^N$, what is the likelihood function of model parameters?

$$L(\boldsymbol{\theta}) = \prod_{n=1}^{N} p(\boldsymbol{x}_n; \boldsymbol{\theta}) \tag{2}$$

The Challenges of GMM's MLE

Phenomenon in formulation:

- $\log L(\boldsymbol{\theta})$ falls into a log-sum-exp formulation.

The Challenges of GMM's MLE

Phenomenon in formulation:

- $\log L(\boldsymbol{\theta})$ falls into a log-sum-exp formulation.
- Hard to optimize (Could you derive its gradient?)

The Challenges of GMM's MLE

Phenomenon in formulation:

- $\log L(\boldsymbol{\theta})$ falls into a log-sum-exp formulation.
- Hard to optimize (Could you derive its gradient?)

Reason in Statistics:

- Hidden/unobserved generative process — we don't know which cluster is each sample from?

The Challenges of GMM's MLE

Phenomenon in formulation:

- $\log L(\boldsymbol{\theta})$ falls into a log-sum-exp formulation.
- Hard to optimize (Could you derive its gradient?)

Reason in Statistics:

- Hidden/unobserved generative process — we don't know which cluster is each sample from?
- What if we know it?

Learning GMM: An EM Algorithm

In practice, we often apply an **expectation-maximization (EM)** algorithm to learn GMM.

- **E-step:** Given current parameters $\{w_k^{(t)}, \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)}\}_{k=1}^K$, calculate **responsibility** (the posterior distribution of clusters given a sample.)

$$p^{(t)}(k|\boldsymbol{x}_n) = \frac{w_k^{(t)} p(\boldsymbol{x}_n; \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)})}{\sum_{i=1}^K w_i^{(t)} p(\boldsymbol{x}_n; \boldsymbol{\mu}_i^{(t)}, \boldsymbol{\Sigma}_i^{(t)})}, \quad \forall n = 1, ..., N,\ k = 1, ..., K$$

Learning GMM: An EM Algorithm

In practice, we often apply an **expectation-maximization (EM)** algorithm to learn GMM.

- **E-step:** Given current parameters $\{w_k^{(t)}, \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)}\}_{k=1}^K$, calculate **responsibility** (the posterior distribution of clusters given a sample.)

$$p^{(t)}(k|\boldsymbol{x}_n) = \frac{w_k^{(t)} p(\boldsymbol{x}_n; \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)})}{\sum_{i=1}^K w_i^{(t)} p(\boldsymbol{x}_n; \boldsymbol{\mu}_i^{(t)}, \boldsymbol{\Sigma}_i^{(t)})}, \quad \forall n = 1, ..., N,\ k = 1, ..., K \tag{3}$$

How to explain the denominator and the numerator?

Learning GMM: An EM Algorithm

In practice, we often apply an **expectation-maximization (EM)** algorithm to learn GMM.

- **E-step:** Given current parameters $\{w_k^{(t)}, \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)}\}_{k=1}^K$, calculate **responsibility** (the posterior distribution of clusters given a sample.)

$$p^{(t)}(k|\boldsymbol{x}_n) = \frac{w_k^{(t)} p(\boldsymbol{x}_n; \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)})}{\sum_{i=1}^K w_i^{(t)} p(\boldsymbol{x}_n; \boldsymbol{\mu}_i^{(t)}, \boldsymbol{\Sigma}_i^{(t)})}, \quad \forall n = 1, ..., N, \ k = 1, ..., K \tag{3}$$

How to explain the denominator and the numerator?

- **M-step:** Update model parameters based on the responsibilities

$$w_k^{(t+1)} = \frac{1}{N} \sum_{n=1}^N p^{(t)}(k|\boldsymbol{x}_n),$$

Learning GMM: An EM Algorithm

In practice, we often apply an **expectation-maximization (EM)** algorithm to learn GMM.

- **E-step:** Given current parameters $\{w_k^{(t)}, \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)}\}_{k=1}^K$, calculate **responsibility** (the posterior distribution of clusters given a sample.)

$$p^{(t)}(k|\boldsymbol{x}_n) = \frac{w_k^{(t)} p(\boldsymbol{x}_n; \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)})}{\sum_{i=1}^K w_i^{(t)} p(\boldsymbol{x}_n; \boldsymbol{\mu}_i^{(t)}, \boldsymbol{\Sigma}_i^{(t)})}, \quad \forall n = 1, ..., N,\ k = 1, ..., K \tag{3}$$

How to explain the denominator and the numerator?

- **M-step:** Update model parameters based on the responsibilities

$$w_k^{(t+1)} = \frac{1}{N} \sum_{n=1}^N p^{(t)}(k|\boldsymbol{x}_n), \quad \boldsymbol{\mu}_k^{(t+1)} = \frac{\sum_{n=1}^N p^{(t)}(k|\boldsymbol{x}_n)\boldsymbol{x}_n}{\sum_{n=1}^N p^{(t)}(k|\boldsymbol{x}_n)}$$

Learning GMM: An EM Algorithm

In practice, we often apply an **expectation-maximization (EM)** algorithm to learn GMM.

- **E-step:** Given current parameters $\{w_k^{(t)}, \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)}\}_{k=1}^K$, calculate **responsibility** (the posterior distribution of clusters given a sample.)

$$p^{(t)}(k|\boldsymbol{x}_n) = \frac{w_k^{(t)} p(\boldsymbol{x}_n; \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)})}{\sum_{i=1}^K w_i^{(t)} p(\boldsymbol{x}_n; \boldsymbol{\mu}_i^{(t)}, \boldsymbol{\Sigma}_i^{(t)})}, \quad \forall n = 1, ..., N,\ k = 1, ..., K \tag{3}$$

How to explain the denominator and the numerator?

- **M-step:** Update model parameters based on the responsibilities

$$\begin{aligned} w_k^{(t+1)} &= \frac{1}{N}\sum_{n=1}^N p^{(t)}(k|\boldsymbol{x}_n), \quad \boldsymbol{\mu}_k^{(t+1)} = \frac{\sum_{n=1}^N p^{(t)}(k|\boldsymbol{x}_n)\boldsymbol{x}_n}{\sum_{n=1}^N p^{(t)}(k|\boldsymbol{x}_n)} \\ \boldsymbol{\Sigma}_k^{(t+1)} &= \frac{\sum_{n=1}^N p^{(t)}(k|\boldsymbol{x}_n)(\boldsymbol{x}_n - \boldsymbol{\mu}_k^{(t+1)})(\boldsymbol{x}_n - \boldsymbol{\mu}_k^{(t+1)})^T}{\sum_{n=1}^N p^{(t)}(k|\boldsymbol{x}_n)} \end{aligned} \tag{4}$$

An Optimization Viewpoint of The EM Algorithm

Original MLE: Denote $\boldsymbol{\theta} = \{w_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$, the MLE of GMM corresponds to

$$\min_{\boldsymbol{\theta}} \sum_{n=1}^N -\log p(\boldsymbol{x}_n; \boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \underbrace{\sum_{n=1}^N -\log\left(\sum_{k=1}^K w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right)}_{L(\boldsymbol{\theta})}.$$

An Optimization Viewpoint of The EM Algorithm

Original MLE: Denote $\boldsymbol{\theta} = \{w_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$, the MLE of GMM corresponds to

$$\min_{\boldsymbol{\theta}} \sum_{n=1}^N -\log p(\boldsymbol{x}_n; \boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \underbrace{\sum_{n=1}^N -\log\left(\sum_{k=1}^K w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right)}_{L(\boldsymbol{\theta})}. \tag{5}$$

- $L(\theta)$ is a convex function.

An Optimization Viewpoint of The EM Algorithm

Original MLE: Denote $\boldsymbol{\theta} = \{w_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$, the MLE of GMM corresponds to

$$\min_{\boldsymbol{\theta}} \sum_{n=1}^{N} -\log p(\boldsymbol{x}_n; \boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \underbrace{\sum_{n=1}^{N} -\log\left(\sum_{k=1}^{K} w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right)}_{L(\boldsymbol{\theta})}. \tag{5}$$

- $L(\theta)$ is a convex function.
- Apply **Jensen's inequality** — instead of minimizing $L(\theta)$, we minimize its upper bound $Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)})$ instead.

An Optimization Viewpoint of The EM Algorithm

Original MLE: Denote $\boldsymbol{\theta} = \{w_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$, the MLE of GMM corresponds to

$$\min_{\boldsymbol{\theta}} \sum_{n=1}^{N} -\log p(\boldsymbol{x}_n; \boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \underbrace{\sum_{n=1}^{N} -\log \left(\sum_{k=1}^{K} w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right)}_{L(\boldsymbol{\theta})}. \tag{5}$$

- $L(\theta)$ is a convex function.
- Apply **Jensen's inequality** — instead of minimizing $L(\theta)$, we minimize its upper bound $Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)})$ instead.

$$L(\boldsymbol{\theta}) = -\sum_{n=1}^{N} \log \left(\sum_{k=1}^{K} w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right)$$

An Optimization Viewpoint of The EM Algorithm

Original MLE: Denote $\boldsymbol{\theta} = \{w_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$, the MLE of GMM corresponds to

$$\min_{\boldsymbol{\theta}} \sum_{n=1}^N -\log p(\boldsymbol{x}_n; \boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \underbrace{\sum_{n=1}^N -\log\left(\sum_{k=1}^K w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right)}_{L(\boldsymbol{\theta})}. \tag{5}$$

- $L(\theta)$ is a convex function.
- Apply **Jensen's inequality** — instead of minimizing $L(\theta)$, we minimize its upper bound $Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)})$ instead.

$$\begin{aligned} L(\boldsymbol{\theta}) &= -\sum_{n=1}^N \log\left(\sum_{k=1}^K w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right) \\ &= -\sum_{n=1}^N \log\left(\sum_{k=1}^K \frac{p^{(t)}(k|\boldsymbol{x}_n)}{p^{(t)}(k|\boldsymbol{x}_n)} w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right) \end{aligned}$$

An Optimization Viewpoint of The EM Algorithm

Original MLE: Denote $\boldsymbol{\theta} = \{w_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$, the MLE of GMM corresponds to

$$\min_{\boldsymbol{\theta}} \sum_{n=1}^N -\log p(\boldsymbol{x}_n; \boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \underbrace{\sum_{n=1}^N -\log\left(\sum_{k=1}^K w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right)}_{L(\boldsymbol{\theta})}. \tag{5}$$

- $L(\theta)$ is a convex function.
- Apply **Jensen's inequality** — instead of minimizing $L(\theta)$, we minimize its upper bound $Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)})$ instead.

$$\begin{aligned} L(\boldsymbol{\theta}) &= -\sum_{n=1}^N \log\left(\sum_{k=1}^K w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right) \\ &= -\sum_{n=1}^N \log\left(\sum_{k=1}^K \frac{p^{(t)}(k|\boldsymbol{x}_n)}{p^{(t)}(k|\boldsymbol{x}_n)} w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right) \\ &\le -\sum_{n=1}^N \sum_{k=1}^K p^{(t)}(k|\boldsymbol{x}_n) \log \frac{w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{p^{(t)}(k|\boldsymbol{x}_n)} = Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) \end{aligned} \tag{6}$$

An Optimization Viewpoint of The EM Algorithm

The optimization problem in the t-th step:

$$\min_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) = \min_{\boldsymbol{\theta}} - \sum_{n=1}^{N} \sum_{k=1}^{K} p^{(t)}(k|\boldsymbol{x}_n) \log \frac{w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{p^{(t)}(k|\boldsymbol{x}_n)}.$$

An Optimization Viewpoint of The EM Algorithm

The optimization problem in the t-th step:

$$\min_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) = \min_{\boldsymbol{\theta}} -\sum_{n=1}^{N} \sum_{k=1}^{K} p^{(t)}(k|\boldsymbol{x}_n) \log \frac{w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{p^{(t)}(k|\boldsymbol{x}_n)}. \tag{7}$$

Then, in the M-step:

$$\begin{aligned} &\min_{\boldsymbol{w} \in \Delta^{K-1}} -\sum_{n=1}^{N} \sum_{k=1}^{K} p^{(t)}(k|\boldsymbol{x}_n) \log w_k \\ = &\min_{\boldsymbol{w} \in \Delta^{K-1}} -\sum_{k=1}^{K} \underbrace{\frac{\sum_{n=1}^{N} p^{(t)}(k|\boldsymbol{x}_n)}{N}}_{p_k^{(t)}} \log w_k \end{aligned}$$

An Optimization Viewpoint of The EM Algorithm

The optimization problem in the t-th step:

$$\min_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) = \min_{\boldsymbol{\theta}} - \sum_{n=1}^{N} \sum_{k=1}^{K} p^{(t)}(k|\boldsymbol{x}_n) \log \frac{w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{p^{(t)}(k|\boldsymbol{x}_n)}. \tag{7}$$

Then, in the M-step:

$$\begin{aligned} &\min_{\boldsymbol{w} \in \Delta^{K-1}} - \sum_{n=1}^{N} \sum_{k=1}^{K} p^{(t)}(k|\boldsymbol{x}_n) \log w_k \\ = &\min_{\boldsymbol{w} \in \Delta^{K-1}} - \sum_{k=1}^{K} \underbrace{\frac{\sum_{n=1}^{N} p^{(t)}(k|\boldsymbol{x}_n)}{N}}_{p_k^{(t)}} \log w_k = \min_{\boldsymbol{w}} \mathrm{KL}(\boldsymbol{p}^{(t)} \| \boldsymbol{w}) \end{aligned}$$

An Optimization Viewpoint of The EM Algorithm

The optimization problem in the t-th step:

$$\min_{\boldsymbol{\theta}} Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(t)}) = \min_{\boldsymbol{\theta}} -\sum_{n=1}^{N}\sum_{k=1}^{K} p^{(t)}(k|\boldsymbol{x}_n)\log\frac{w_k p(\boldsymbol{x}_n;\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)}{p^{(t)}(k|\boldsymbol{x}_n)}. \tag{7}$$

Then, in the M-step:

$$\begin{aligned}&\min_{\boldsymbol{w}\in\Delta^{K-1}} -\sum_{n=1}^{N}\sum_{k=1}^{K} p^{(t)}(k|\boldsymbol{x}_n)\log w_k\\ =&\min_{\boldsymbol{w}\in\Delta^{K-1}} -\sum_{k=1}^{K}\underbrace{\frac{\sum_{n=1}^{N} p^{(t)}(k|\boldsymbol{x}_n)}{N}}_{p_k^{(t)}}\log w_k = \min_{\boldsymbol{w}} \mathrm{KL}(\boldsymbol{p}^{(t)}\|\boldsymbol{w}) \quad\Rightarrow\quad \boldsymbol{w}^{(t+1)} = \boldsymbol{p}^{(t)}\end{aligned} \tag{8}$$

How to update μ_k and Σ_k?

An Optimization Viewpoint of The EM Algorithm

The optimization problem in the t-th step:

$$\min_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) = \min_{\boldsymbol{\theta}} -\sum_{n=1}^{N} \sum_{k=1}^{K} p^{(t)}(k|\boldsymbol{x}_n) \log \frac{w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{p^{(t)}(k|\boldsymbol{x}_n)}. \tag{7}$$

Then, in the M-step:

$$\begin{aligned} &\min_{\boldsymbol{w}\in\Delta^{K-1}} -\sum_{n=1}^{N} \sum_{k=1}^{K} p^{(t)}(k|\boldsymbol{x}_n) \log w_k \\ =& \min_{\boldsymbol{w}\in\Delta^{K-1}} -\sum_{k=1}^{K} \underbrace{\frac{\sum_{n=1}^{N} p^{(t)}(k|\boldsymbol{x}_n)}{N}}_{p_k^{(t)}} \log w_k = \min_{\boldsymbol{w}} \mathrm{KL}(\boldsymbol{p}^{(t)} \| \boldsymbol{w}) \quad \Rightarrow \quad \boldsymbol{w}^{(t+1)} = \boldsymbol{p}^{(t)} \end{aligned} \tag{8}$$

How to update μ_k and Σ_k?

$$\min_{\boldsymbol{\mu}_k} -\sum_{n=1}^{N} p^{(t)}(k|\boldsymbol{x}_n) \log p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k^{(t)})$$

An Optimization Viewpoint of The EM Algorithm

The optimization problem in the t-th step:

$$\min_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) = \min_{\boldsymbol{\theta}} - \sum_{n=1}^{N} \sum_{k=1}^{K} p^{(t)}(k|\boldsymbol{x}_n) \log \frac{w_k p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{p^{(t)}(k|\boldsymbol{x}_n)}. \tag{7}$$

Then, in the M-step:

$$\begin{aligned} &\min_{\boldsymbol{w}\in\Delta^{K-1}} - \sum_{n=1}^{N} \sum_{k=1}^{K} p^{(t)}(k|\boldsymbol{x}_n) \log w_k \\ = &\min_{\boldsymbol{w}\in\Delta^{K-1}} - \sum_{k=1}^{K} \underbrace{\frac{\sum_{n=1}^{N} p^{(t)}(k|\boldsymbol{x}_n)}{N}}_{p_k^{(t)}} \log w_k = \min_{\boldsymbol{w}} \mathrm{KL}(\boldsymbol{p}^{(t)} \| \boldsymbol{w}) \quad \Rightarrow \quad \boldsymbol{w}^{(t+1)} = \boldsymbol{p}^{(t)} \end{aligned} \tag{8}$$

How to update $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$?

$$\begin{aligned} &\min_{\boldsymbol{\mu}_k} - \sum_{n=1}^{N} p^{(t)}(k|\boldsymbol{x}_n) \log p(\boldsymbol{x}_n; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k^{(t)}) \\ &\frac{\partial Q}{\partial \boldsymbol{\mu}_k} = \mathbf{0} \quad \Rightarrow \quad \boldsymbol{\mu}_k^{(t+1)} = \frac{\sum_{n=1}^{N} p^{(t)}(k|\boldsymbol{x}_n)\boldsymbol{x}_n}{\sum_{n=1}^{N} p^{(t)}(k|\boldsymbol{x}_n)} \quad \text{(Derive It.)} \end{aligned} \tag{9}$$

Revisit Kmeans from The Viewpoint of EM Algorithm

Kmeans learns K centroids, which also provides us with a **generative model**.

1. Initialize K centroids randomly from the observed data, e.g., $\{\mathbf{c}_k\}_{k=1}^{K}$.

Revisit Kmeans from The Viewpoint of EM Algorithm

Kmeans learns K centroids, which also provides us with a **generative model**.

1. Initialize K centroids randomly from the observed data, e.g., $\{\mathbf{c}_k\}_{k=1}^K$.
2. Repeat the following steps till converge
   a. Assign each data to the nearest cluster: $\forall\, \boldsymbol{x}_n$

$$\boldsymbol{x}_n \in \mathcal{C}_k, \quad \text{if } k = \arg\min_{k \in \{1,\ldots,K\}} d(\boldsymbol{x}_n, \mathbf{c}_k).$$

Revisit Kmeans from The Viewpoint of EM Algorithm

Kmeans learns K centroids, which also provides us with a **generative model**.

1. Initialize K centroids randomly from the observed data, e.g., $\{\mathbf{c}_k\}_{k=1}^K$.
2. Repeat the following steps till converge
   a. Assign each data to the nearest cluster: $\forall\, \boldsymbol{x}_n$

$$\boldsymbol{x}_n \in \mathcal{C}_k, \quad \text{if } k = \arg\min_{k \in \{1,\dots,K\}} d(\boldsymbol{x}_n, \mathbf{c}_k). \tag{10}$$

   b. Update the centriods:

$$\mathbf{c}_k = \frac{1}{|\mathcal{C}_k|} \sum\nolimits_{x_n \in \mathcal{C}_k} \boldsymbol{x}_n.$$

Revisit Kmeans from The Viewpoint of EM Algorithm

Kmeans learns K centroids, which also provides us with a **generative model**.

1. Initialize K centroids randomly from the observed data, e.g., $\{\mathbf{c}_k\}_{k=1}^{K}$.
2. Repeat the following steps till converge
   a. Assign each data to the nearest cluster: $\forall\, \boldsymbol{x}_n$

$$\boldsymbol{x}_n \in \mathcal{C}_k, \quad \text{if } k = \arg\min_{k \in \{1,\dots,K\}} d(\boldsymbol{x}_n, \mathbf{c}_k). \tag{10}$$

   b. Update the centriods:

$$\mathbf{c}_k = \frac{1}{|\mathcal{C}_k|} \sum_{x_n \in \mathcal{C}_k} \boldsymbol{x}_n. \tag{11}$$

- $w_k = p(k) = \frac{|\mathcal{C}_k|}{N}$, and

$$p(\boldsymbol{x}; \mathbf{c}_k) = \begin{cases} \frac{\Gamma(\frac{D}{2}+1)}{\pi^{D/2} r}, & \|\boldsymbol{x} - \mathbf{c}_k\|_2 \leq r, \\ 0, & \text{Otherwise.} \end{cases}$$

Revisit Kmeans from The Viewpoint of EM Algorithm

Kmeans learns K centroids, which also provides us with a **generative model**.

1. Initialize K centroids randomly from the observed data, e.g., $\{\mathbf{c}_k\}_{k=1}^K$.
2. Repeat the following steps till converge
   a. Assign each data to the nearest cluster: $\forall\, \boldsymbol{x}_n$

$$\boldsymbol{x}_n \in \mathcal{C}_k, \quad \text{if } k = \arg\min_{k\in\{1,\dots,K\}} d(\boldsymbol{x}_n, \mathbf{c}_k). \tag{10}$$

   b. Update the centriods:

$$\mathbf{c}_k = \frac{1}{|\mathcal{C}_k|} \sum\nolimits_{x_n\in\mathcal{C}_k} \boldsymbol{x}_n. \tag{11}$$

- $w_k = p(k) = \frac{|\mathcal{C}_k|}{N}$, and

$$p(\boldsymbol{x};\mathbf{c}_k) = \begin{cases} \frac{\Gamma(\frac{D}{2}+1)}{\pi^{D/2} r}, & \|\boldsymbol{x}-\mathbf{c}_k\|_2 \leq r, \\ 0, & \text{Otherwise}. \end{cases} \tag{12}$$

Recall nonparametric kernel functions:)

Revisit Kmeans from The Viewpoint of EM Algorithm

Original MLE: Denote $\boldsymbol{\theta} = \{w_k, \mathbf{c}_k, r_k\}_{k=1}^K$, the MLE of GMM corresponds to

$$\min_{\boldsymbol{\theta}} \sum_{n=1}^{N} -\log p(\boldsymbol{x}_n; \boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \underbrace{\sum_{n=1}^{N} -\log \left(\sum_{k=1}^{K} w_k p(\boldsymbol{x}_n; \mathbf{c}_k, r_k) \right)}_{L(\boldsymbol{\theta})}.$$

Revisit Kmeans from The Viewpoint of EM Algorithm

Original MLE: Denote $\boldsymbol{\theta} = \{w_k, \mathbf{c}_k, r_k\}_{k=1}^{K}$, the MLE of GMM corresponds to

$$\min_{\boldsymbol{\theta}} \sum_{n=1}^{N} -\log p(\boldsymbol{x}_n; \boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \underbrace{\sum_{n=1}^{N} -\log\left(\sum_{k=1}^{K} w_k p(\boldsymbol{x}_n; \mathbf{c}_k, r_k)\right)}_{L(\boldsymbol{\theta})}. \tag{13}$$

EM Algorithm:

- **E-step:** $\exists r > 0$ for $\boldsymbol{x}_n$ and $k = 1, ..., K$

$$p^{(t)}(k|\boldsymbol{x}_n) = \frac{w_k^{(t)} p(\boldsymbol{x}_n; \mathbf{c}_k^{(t)}, r_k^{(t)})}{\sum_{i=1}^{K} w_i^{(t)} p(\boldsymbol{x}_n; \mathbf{c}_i^{(t)}, r_i^{(t)})} = \begin{cases} 1 & \|\boldsymbol{x}_n - \mathbf{c}_k^{(t)}\|_2 \leq r \\ 0 & \text{Otherwise} \end{cases}$$

Revisit Kmeans from The Viewpoint of EM Algorithm

Original MLE: Denote $\boldsymbol{\theta} = \{w_k, \mathbf{c}_k, r_k\}_{k=1}^{K}$, the MLE of GMM corresponds to

$$\min_{\boldsymbol{\theta}} \sum_{n=1}^{N} -\log p(\boldsymbol{x}_n; \boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \underbrace{\sum_{n=1}^{N} -\log \left(\sum_{k=1}^{K} w_k p(\boldsymbol{x}_n; \mathbf{c}_k, r_k) \right)}_{L(\boldsymbol{\theta})}. \tag{13}$$

EM Algorithm:

- **E-step:** $\exists r > 0$ for $\boldsymbol{x}_n$ and $k = 1, \ldots, K$

$$p^{(t)}(k|\boldsymbol{x}_n) = \frac{w_k^{(t)} p(\boldsymbol{x}_n; \mathbf{c}_k^{(t)}, r_k^{(t)})}{\sum_{i=1}^{K} w_i^{(t)} p(\boldsymbol{x}_n; \mathbf{c}_i^{(t)}, r_i^{(t)})} = \begin{cases} 1 & \|\boldsymbol{x}_n - \mathbf{c}_k^{(t)}\|_2 \leq r \\ 0 & \text{Otherwise} \end{cases} \tag{14}$$

- **M-step:**

$$\min_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) = \min_{\boldsymbol{\theta}} -\sum_{n=1}^{N} \sum_{k=1}^{K} p^{(t)}(k|\boldsymbol{x}_n) \log \frac{w_k p(\boldsymbol{x}_n; \mathbf{c}_k)}{p^{(t)}(k|\boldsymbol{x}_n)}. \tag{15}$$

A Generalized Formulation of The EM Algorithm

- Essentially, the EM algorithm is used to find **local** maximum likelihood parameters of a statistical model that involve **unobserved latent variables**

A Generalized Formulation of The EM Algorithm

- Essentially, the EM algorithm is used to find **local** maximum likelihood parameters of a statistical model that involve **unobserved latent variables**
- In GMMs, the unobserved latent variable, its the cluster ID of each sample.

A Generalized Formulation of The EM Algorithm

- Essentially, the EM algorithm is used to find **local** maximum likelihood parameters of a statistical model that involve **unobserved latent variables**
- In GMMs, the unobserved latent variable, its the cluster ID of each sample.
- Denote $\boldsymbol{X}$ as observed data, and $\boldsymbol{Z}$ as latent variables of the data, $\boldsymbol{\theta}$ as the model parameters.

A Generalized Formulation of The EM Algorithm

- Essentially, the EM algorithm is used to find **local** maximum likelihood parameters of a statistical model that involve **unobserved latent variables**
- In GMMs, the unobserved latent variable, its the cluster ID of each sample.
- Denote $\boldsymbol{X}$ as observed data, and $\boldsymbol{Z}$ as latent variables of the data, $\boldsymbol{\theta}$ as the model parameters.
- $L(\boldsymbol{\theta}; \boldsymbol{X}, \boldsymbol{Z}) = p(\boldsymbol{X}, \boldsymbol{Z} | \boldsymbol{\theta})$: a likelihood function.

A Generalized Formulation of The EM Algorithm

- Essentially, the EM algorithm is used to find **local** maximum likelihood parameters of a statistical model that involve **unobserved latent variables**
- In GMMs, the unobserved latent variable, its the cluster ID of each sample.
- Denote $\boldsymbol{X}$ as observed data, and $\boldsymbol{Z}$ as latent variables of the data, $\boldsymbol{\theta}$ as the model parameters.
- $L(\boldsymbol{\theta}; \boldsymbol{X}, \boldsymbol{Z}) = p(\boldsymbol{X}, \boldsymbol{Z} | \boldsymbol{\theta})$: a likelihood function.
- The MLE of $\boldsymbol{\theta}$ given observed $\boldsymbol{X}$: maximizing the marginal probability of $\boldsymbol{X}$:

$$L(\boldsymbol{\theta}; \boldsymbol{X}) = p(\boldsymbol{X} | \boldsymbol{\theta})$$

A Generalized Formulation of The EM Algorithm

- Essentially, the EM algorithm is used to find **local** maximum likelihood parameters of a statistical model that involve **unobserved latent variables**
- In GMMs, the unobserved latent variable, its the cluster ID of each sample.
- Denote $\boldsymbol{X}$ as observed data, and $\boldsymbol{Z}$ as latent variables of the data, $\boldsymbol{\theta}$ as the model parameters.
- $L(\boldsymbol{\theta}; \boldsymbol{X}, \boldsymbol{Z}) = p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})$: a likelihood function.
- The MLE of $\boldsymbol{\theta}$ given observed $\boldsymbol{X}$: maximizing the marginal probability of $\boldsymbol{X}$:

$$L(\boldsymbol{\theta}; \boldsymbol{X}) = p(\boldsymbol{X}|\boldsymbol{\theta}) = \int p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta}) \mathrm{d}\boldsymbol{Z}$$

A Generalized Formulation of The EM Algorithm

- Essentially, the EM algorithm is used to find **local** maximum likelihood parameters of a statistical model that involve **unobserved latent variables**
- In GMMs, the unobserved latent variable, its the cluster ID of each sample.
- Denote $\boldsymbol{X}$ as observed data, and $\boldsymbol{Z}$ as latent variables of the data, $\boldsymbol{\theta}$ as the model parameters.
- $L(\boldsymbol{\theta}; \boldsymbol{X}, \boldsymbol{Z}) = p(\boldsymbol{X}, \boldsymbol{Z} | \boldsymbol{\theta})$: a likelihood function.
- The MLE of $\boldsymbol{\theta}$ given observed $\boldsymbol{X}$: maximizing the marginal probability of $\boldsymbol{X}$:

$$
\begin{aligned}
L(\boldsymbol{\theta}; \boldsymbol{X}) = p(\boldsymbol{X} | \boldsymbol{\theta}) &= \int p(\boldsymbol{X}, \boldsymbol{Z} | \boldsymbol{\theta}) \mathrm{d}\boldsymbol{Z} \\
&= \int \underbrace{p(\boldsymbol{X} | \boldsymbol{Z}, \boldsymbol{\theta})}_{L(\boldsymbol{\theta}; \boldsymbol{X}, \boldsymbol{Z})} p(\boldsymbol{Z} | \boldsymbol{\theta}) \mathrm{d}\boldsymbol{Z}
\end{aligned}
$$

A Generalized Formulation of The EM Algorithm

- Essentially, the EM algorithm is used to find **local** maximum likelihood parameters of a statistical model that involve **unobserved latent variables**
- In GMMs, the unobserved latent variable, its the cluster ID of each sample.
- Denote $\boldsymbol{X}$ as observed data, and $\boldsymbol{Z}$ as latent variables of the data, $\boldsymbol{\theta}$ as the model parameters.
- $L(\boldsymbol{\theta}; \boldsymbol{X}, \boldsymbol{Z}) = p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})$: a likelihood function.
- The MLE of $\boldsymbol{\theta}$ given observed $\boldsymbol{X}$: maximizing the marginal probability of $\boldsymbol{X}$:

$$\begin{aligned} L(\boldsymbol{\theta}; \boldsymbol{X}) = p(\boldsymbol{X}|\boldsymbol{\theta}) &= \int p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta}) \mathrm{d}\boldsymbol{Z} \\ &= \int \underbrace{p(\boldsymbol{X}|\boldsymbol{Z}, \boldsymbol{\theta})}_{L(\boldsymbol{\theta}; \boldsymbol{X}, \boldsymbol{Z})} p(\boldsymbol{Z}|\boldsymbol{\theta}) \mathrm{d}\boldsymbol{Z} = \mathbb{E}_{\boldsymbol{Z} \sim p(\boldsymbol{Z}|\boldsymbol{\theta})} [L(\boldsymbol{\theta}; \boldsymbol{X}, \boldsymbol{Z})] \end{aligned} \tag{16}$$

A Generalized Formulation of The EM Algorithm

The objective function of EM:

$$\min_{\boldsymbol{\theta}} -\mathbb{E}_{\boldsymbol{Z}\sim p(\boldsymbol{Z}|\boldsymbol{\theta})}[L(\boldsymbol{\theta};\boldsymbol{X},\boldsymbol{Z})]$$

A Generalized Formulation of The EM Algorithm

The objective function of EM:

$$\min_{\boldsymbol{\theta}} -\mathbb{E}_{\boldsymbol{Z}\sim p(\boldsymbol{Z}|\boldsymbol{\theta})}[L(\boldsymbol{\theta};\boldsymbol{X},\boldsymbol{Z})] \tag{17}$$

E-step: Construct the expected value of the log-likelihood function of $\boldsymbol{\theta}$ given current conditional distribution of $\boldsymbol{Z}$ given $\boldsymbol{X}$ and current $\boldsymbol{\theta}^{(t)}$:

$$Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(t)}) := -\mathbb{E}_{\boldsymbol{Z}\sim p(\boldsymbol{Z}|\boldsymbol{\theta}^{(t)})}[L(\boldsymbol{\theta};\boldsymbol{X},\boldsymbol{Z})].$$

A Generalized Formulation of The EM Algorithm

The objective function of EM:

$$\min_{\boldsymbol{\theta}} -\mathbb{E}_{\boldsymbol{Z}\sim p(\boldsymbol{Z}|\boldsymbol{\theta})}[L(\boldsymbol{\theta};\boldsymbol{X},\boldsymbol{Z})] \tag{17}$$

E-step: Construct the expected value of the log-likelihood function of $\boldsymbol{\theta}$ given current conditional distribution of $\boldsymbol{Z}$ given $\boldsymbol{X}$ and current $\boldsymbol{\theta}^{(t)}$:

$$Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(t)}) := -\mathbb{E}_{\boldsymbol{Z}\sim p(\boldsymbol{Z}|\boldsymbol{\theta}^{(t)})}[L(\boldsymbol{\theta};\boldsymbol{X},\boldsymbol{Z})]. \tag{18}$$

M-step: Maximize the expectation:

$$\boldsymbol{\theta}^{(t+1)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(t)}).$$

A Generalized Formulation of The EM Algorithm

The objective function of EM:

$$\min_{\boldsymbol{\theta}} -\mathbb{E}_{\boldsymbol{Z}\sim p(\boldsymbol{Z}|\boldsymbol{\theta})}[L(\boldsymbol{\theta};\boldsymbol{X},\boldsymbol{Z})] \tag{17}$$

E-step: Construct the expected value of the log-likelihood function of $\boldsymbol{\theta}$ given current conditional distribution of $\boldsymbol{Z}$ given $\boldsymbol{X}$ and current $\boldsymbol{\theta}^{(t)}$:

$$Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(t)}) := -\mathbb{E}_{\boldsymbol{Z}\sim p(\boldsymbol{Z}|\boldsymbol{\theta}^{(t)})}[L(\boldsymbol{\theta};\boldsymbol{X},\boldsymbol{Z})]. \tag{18}$$

M-step: Maximize the expectation:

$$\boldsymbol{\theta}^{(t+1)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(t)}). \tag{19}$$

Question: In GMM, what is Z? and what is $p(Z|\theta^{(t)})$?

In Summary

- Generative modeling and Gaussian mixture models
- EM algorithm
- Revisit K-means from a statistical viewpoint

Next...

- A Bayesian viewpoint of GMMs
- Kernel density estimation
- Mean-shift algorithm