Probabilistic clustering and the EM algorithm

Guillaume Obozinski

Ecole des Ponts - ParisTech

INIT/AERFAI Summer school on Machine Learning
Benicàssim, June 26th 2017

Outline

1. The EM algorithm for the Gaussian mixture model

2. More examples of graphical models

K-means

Key assumption: Data composed of K "roundish" clusters of similar sizes with centroids $(\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_K)$.

K-means

Key assumption: Data composed of K "roundish" clusters of similar sizes with centroids $(\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_K)$.

Problem can be formulated as: $$\min_{\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_K} \frac{1}{n} \sum_{i=1}^{n} \min_k \|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2.$$

K-means

Key assumption: Data composed of K "roundish" clusters of similar sizes with centroids $(\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_K)$.

Problem can be formulated as: $$\min_{\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_K} \frac{1}{n} \sum_{i=1}^{n} \min_k \|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2.$$

Difficult (NP-hard) nonconvex problem.

K-means

Key assumption: Data composed of K "roundish" clusters of similar sizes with centroids $(\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_K)$.

Problem can be formulated as: $\displaystyle \min_{\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_K} \frac{1}{n} \sum_{i=1}^{n} \min_k \|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2.$

Difficult (NP-hard) nonconvex problem.

K-means algorithm

1. Draw centroids at random
2. Assign each point to the closest centroid

$$C_k \leftarrow \left\{ i \mid \|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2 = \min_j \|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2 \right\}$$

3. Recompute centroid as center of mass of the cluster

$$\boldsymbol{\mu}_k \leftarrow \frac{1}{\mid C_k \mid} \sum_{i \in C_k} \mathbf{x}_i$$

4. Go to 2

K-means properties

Three remarks:

- K-means is greedy algorithm

K-means properties

Three remarks:

- K-means is greedy algorithm
- It can be shown that K-means converges in a finite number of steps.

K-means properties

Three remarks:

- K-means is greedy algorithm
- It can be shown that K-means converges in a finite number of steps.
- The algorithm however typically get stuck in local minima and it practice it is necessary to try several restarts of the algorithm with a random initialization to have chances to obtain a better solution.

K-means properties

Three remarks:

- K-means is greedy algorithm
- It can be shown that K-means converges in a finite number of steps.
- The algorithm however typically get stuck in local minima and it practice it is necessary to try several restarts of the algorithm with a random initialization to have chances to obtain a better solution.
- Will fail if the clusters are not round
- A good initialization for K-means is K-means++, (Arthur and Vassilvitskii, 2007), (included in all good libraries).

See Arthur, D. and Vassilvitskii, S. (2007). k-means++: the advantages of careful seeding. Proceedings of the 18th annual ACM-SIAM symposium on Discrete algorithms.

Outline

The Gaussian mixture model
and
the EM algorithm

Gaussian mixture model

- K components
- $\boldsymbol{z}$ component indicator
- $\boldsymbol{z} = (z_1, \ldots, z_K)^\top \in \{0, 1\}^K$
- $\boldsymbol{z} \sim \mathcal{M}(1, (\pi_1, \ldots, \pi_K))$
- $p(\boldsymbol{z}) = \prod_{k=1}^{K} \pi_k^{z_k}$

Gaussian mixture model

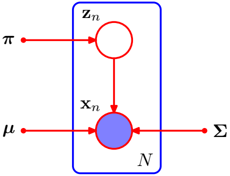
- K components
- $\boldsymbol{z}$ component indicator
- $\boldsymbol{z} = (z_1, \ldots, z_K)^\top \in \{0, 1\}^K$
- $\boldsymbol{z} \sim \mathcal{M}(1, (\pi_1, \ldots, \pi_K))$
- $p(\boldsymbol{z}) = \prod_{k=1}^{K} \pi_k^{z_k}$
- $p(\mathbf{x}|\boldsymbol{z}; (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_k) = \sum_{k=1}^{K} z_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$

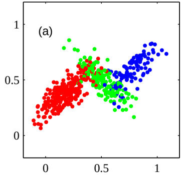
Gaussian mixture model

- K components
- $\boldsymbol{z}$ component indicator
- $\boldsymbol{z} = (z_1, \ldots, z_K)^\top \in \{0,1\}^K$
- $\boldsymbol{z} \sim \mathcal{M}(1, (\pi_1, \ldots, \pi_K))$
- $p(\boldsymbol{z}) = \prod_{k=1}^{K} \pi_k^{z_k}$
- $p(\mathbf{x}|\boldsymbol{z}; (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_k) = \sum_{k=1}^{K} z_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$
- $p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$

Gaussian mixture model

- K components
- $\boldsymbol{z}$ component indicator
- $\boldsymbol{z} = (z_1, \ldots, z_K)^\top \in \{0,1\}^K$
- $\boldsymbol{z} \sim \mathcal{M}(1, (\pi_1, \ldots, \pi_K))$
- $p(\boldsymbol{z}) = \prod_{k=1}^{K} \pi_k^{z_k}$
- $p(\mathbf{x}|\boldsymbol{z}; (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_k) = \sum_{k=1}^{K} z_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$
- $p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$
- Estimation: $\underset{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k}{\operatorname{argmax}} \log \left[\sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right]$





Applying maximum likelihood to the Gaussian mixture

Let $\mathcal{Z} = \{z \in \{0,1\}^K \mid \sum_{k=1}^{K} z_k = 1\}$

Applying maximum likelihood to the Gaussian mixture

Let $\mathcal{Z} = \{z \in \{0,1\}^K \mid \sum_{k=1}^{K} z_k = 1\}$

$p(\mathbf{x}) =$

Applying maximum likelihood to the Gaussian mixture

Let $\mathcal{Z} = \{z \in \{0,1\}^K \mid \sum_{k=1}^{K} z_k = 1\}$

$$p(\mathbf{x}) = \sum_{\boldsymbol{z} \in \mathcal{Z}} p(\mathbf{x}, \boldsymbol{z})$$

Applying maximum likelihood to the Gaussian mixture

Let $\mathcal{Z} = \{z \in \{0,1\}^K \mid \sum_{k=1}^K z_k = 1\}$

$$p(\mathbf{x}) = \sum_{\boldsymbol{z} \in \mathcal{Z}} p(\mathbf{x}, \boldsymbol{z}) = \sum_{\boldsymbol{z} \in \mathcal{Z}} \prod_{k=1}^{K} \left[\pi_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right]^{z_k} =$$

Applying maximum likelihood to the Gaussian mixture

Let $\mathcal{Z} = \{z \in \{0,1\}^K \mid \sum_{k=1}^K z_k = 1\}$

$$p(\mathbf{x}) = \sum_{\boldsymbol{z} \in \mathcal{Z}} p(\mathbf{x}, \boldsymbol{z}) = \sum_{\boldsymbol{z} \in \mathcal{Z}} \prod_{k=1}^{K} \left[\pi_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right]^{z_k} = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Applying maximum likelihood to the Gaussian mixture

Let $\mathcal{Z} = \{z \in \{0,1\}^K \mid \sum_{k=1}^K z_k = 1\}$

$$p(\mathbf{x}) = \sum_{\boldsymbol{z} \in \mathcal{Z}} p(\mathbf{x}, \boldsymbol{z}) = \sum_{\boldsymbol{z} \in \mathcal{Z}} \prod_{k=1}^{K} \Big[\pi_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\Big]^{z_k} = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Issue

- The marginal log-likelihood $\tilde{\ell}(\theta) = \sum_i \log(p(\mathbf{x}^{(i)}))$ with $\theta = \big(\boldsymbol{\pi}, (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_{1 \leq k \leq K}\big)$ is now complicated

Applying maximum likelihood to the Gaussian mixture

Let $\mathcal{Z} = \{z \in \{0,1\}^K \mid \sum_{k=1}^K z_k = 1\}$

$$p(\mathbf{x}) = \sum_{\boldsymbol{z} \in \mathcal{Z}} p(\mathbf{x}, \boldsymbol{z}) = \sum_{\boldsymbol{z} \in \mathcal{Z}} \prod_{k=1}^{K} \left[\pi_k \, \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right]^{z_k} = \sum_{k=1}^{K} \pi_k \, \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Issue

- The marginal log-likelihood $\tilde{\ell}(\theta) = \sum_i \log(p(\mathbf{x}^{(i)}))$ with $\theta = \left(\boldsymbol{\pi}, (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_{1 \leq k \leq K}\right)$ is now complicated
- No hope to find a simple solution to the maximum likelihood problem

Applying maximum likelihood to the Gaussian mixture

Let $\mathcal{Z} = \{z \in \{0,1\}^K \mid \sum_{k=1}^K z_k = 1\}$

$$p(\mathbf{x}) = \sum_{\boldsymbol{z} \in \mathcal{Z}} p(\mathbf{x}, \boldsymbol{z}) = \sum_{\boldsymbol{z} \in \mathcal{Z}} \prod_{k=1}^{K} \Big[\pi_k \, \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\Big]^{z_k} = \sum_{k=1}^{K} \pi_k \, \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Issue

- The marginal log-likelihood $\tilde{\ell}(\theta) = \sum_i \log(p(\mathbf{x}^{(i)}))$ with $\theta = \big(\boldsymbol{\pi}, (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_{1 \le k \le K}\big)$ is now complicated
- No hope to find a simple solution to the maximum likelihood problem
- By contrast the complete log-likelihood has a rather simple form:

$$\tilde{\ell}(\theta) =$$

Applying maximum likelihood to the Gaussian mixture

Let $\mathcal{Z} = \{z \in \{0,1\}^K \mid \sum_{k=1}^K z_k = 1\}$

$$p(\mathbf{x}) = \sum_{\boldsymbol{z}\in\mathcal{Z}} p(\mathbf{x},\boldsymbol{z}) = \sum_{\boldsymbol{z}\in\mathcal{Z}} \prod_{k=1}^{K} \Big[\pi_k \mathcal{N}(\mathbf{x};\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)\Big]^{z_k} = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x};\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)$$

Issue

- The marginal log-likelihood $\tilde{\ell}(\theta) = \sum_i \log(p(\mathbf{x}^{(i)}))$ with $\theta = \big(\boldsymbol{\pi}, (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_{1\le k\le K}\big)$ is now complicated
- No hope to find a simple solution to the maximum likelihood problem
- By contrast the complete log-likelihood has a rather simple form:

$$\tilde{\ell}(\theta) = \sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)})$$

Applying maximum likelihood to the Gaussian mixture

Let $\mathcal{Z} = \{z \in \{0,1\}^K \mid \sum_{k=1}^K z_k = 1\}$

$$p(\mathbf{x}) = \sum_{\boldsymbol{z} \in \mathcal{Z}} p(\mathbf{x}, \boldsymbol{z}) = \sum_{\boldsymbol{z} \in \mathcal{Z}} \prod_{k=1}^{K} \Big[\pi_k \, \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\Big]^{z_k} = \sum_{k=1}^{K} \pi_k \, \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Issue

- The marginal log-likelihood $\tilde{\ell}(\theta) = \sum_i \log(p(\mathbf{x}^{(i)}))$ with $\theta = \big(\boldsymbol{\pi}, (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_{1 \leq k \leq K}\big)$ is now complicated
- No hope to find a simple solution to the maximum likelihood problem
- By contrast the complete log-likelihood has a rather simple form:

$$\tilde{\ell}(\theta) = \sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}) = \sum_{i,\,k} z_k^{(i)} \log \mathcal{N}(x^{(i)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} z_k^{(i)} \log(\pi_k),$$

Applying ML to the multinomial mixture

$\tilde{\ell}(\theta) =$

Applying ML to the multinomial mixture

$$\tilde{\ell}(\theta) = \sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)})$$

Applying ML to the multinomial mixture

$$\tilde{\ell}(\theta) = \sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}) = \sum_{i,k} z_k^{(i)} \log \mathcal{N}(\mathbf{x}^{(i)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} z_k^{(i)} \log(\pi_k),$$

Applying ML to the multinomial mixture

$$\tilde{\ell}(\theta) = \sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}) = \sum_{i,k} z_k^{(i)} \log \mathcal{N}(\mathbf{x}^{(i)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} z_k^{(i)} \log(\pi_k),$$

- If we knew $\boldsymbol{z}^{(i)}$ we could maximize $\tilde{\ell}(\theta)$.

Applying ML to the multinomial mixture

$$\tilde{\ell}(\theta) = \sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}) = \sum_{i,k} z_k^{(i)} \log \mathcal{N}(\mathbf{x}^{(i)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} z_k^{(i)} \log(\pi_k),$$

- If we knew $\boldsymbol{z}^{(i)}$ we could maximize $\tilde{\ell}(\theta)$.
- If we knew $\theta = \big(\boldsymbol{\pi}, (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_{1 \leq k \leq K}\big)$, we could find the best $\boldsymbol{z}^{(i)}$ since we could compute the true a posteriori on $\boldsymbol{z}^{(i)}$ given $\mathbf{x}^{(i)}$:

Applying ML to the multinomial mixture

$$\tilde{\ell}(\theta) = \sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}) = \sum_{i,k} z_k^{(i)} \log \mathcal{N}(\mathbf{x}^{(i)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} z_k^{(i)} \log(\pi_k),$$

- If we knew $\boldsymbol{z}^{(i)}$ we could maximize $\tilde{\ell}(\theta)$.
- If we knew $\theta = \big(\boldsymbol{\pi}, (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_{1 \leq k \leq K}\big)$, we could find the best $\boldsymbol{z}^{(i)}$ since we could compute the true a posteriori on $\boldsymbol{z}^{(i)}$ given $\mathbf{x}^{(i)}$:

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

Applying ML to the multinomial mixture

$$\tilde{\ell}(\theta) = \sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}) = \sum_{i,k} z_k^{(i)} \log \mathcal{N}(\mathbf{x}^{(i)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} z_k^{(i)} \log(\pi_k),$$

- If we knew $\boldsymbol{z}^{(i)}$ we could maximize $\tilde{\ell}(\theta)$.
- If we knew $\theta = \big(\boldsymbol{\pi}, (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_{1 \leq k \leq K}\big)$, we could find the best $\boldsymbol{z}^{(i)}$ since we could compute the true a posteriori on $\boldsymbol{z}^{(i)}$ given $\mathbf{x}^{(i)}$:

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

$\rightarrow$ Seems a chicken and egg problem...

Applying ML to the multinomial mixture

$$\tilde{\ell}(\theta) = \sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}) = \sum_{i,k} z_k^{(i)} \log \mathcal{N}(\mathbf{x}^{(i)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} z_k^{(i)} \log(\pi_k),$$

- If we knew $\boldsymbol{z}^{(i)}$ we could maximize $\tilde{\ell}(\theta)$.
- If we knew $\theta = \big(\boldsymbol{\pi}, (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_{1\le k\le K}\big)$, we could find the best $\boldsymbol{z}^{(i)}$ since we could compute the true a posteriori on $\boldsymbol{z}^{(i)}$ given $\mathbf{x}^{(i)}$:

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

$\rightarrow$ Seems a chicken and egg problem...

- In addition, we want to solve

$$\max_\theta \sum_i \log \left(\sum_{\boldsymbol{z}^{(i)}} p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}) \right)$$

Applying ML to the multinomial mixture

$$\tilde{\ell}(\theta) = \sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}) = \sum_{i,k} z_k^{(i)} \log \mathcal{N}(\mathbf{x}^{(i)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} z_k^{(i)} \log(\pi_k),$$

- If we knew $\boldsymbol{z}^{(i)}$ we could maximize $\tilde{\ell}(\theta)$.
- If we knew $\theta = \big(\boldsymbol{\pi}, (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_{1\leq k\leq K}\big)$, we could find the best $\boldsymbol{z}^{(i)}$ since we could compute the true a posteriori on $\boldsymbol{z}^{(i)}$ given $\mathbf{x}^{(i)}$:

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

$\rightarrow$ Seems a chicken and egg problem...

- In addition, we want to solve

$$\max_{\theta} \sum_i \log \left(\sum_{\boldsymbol{z}^{(i)}} p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}) \right) \quad \text{and not} \quad \max_{\substack{\theta, \\ \boldsymbol{z}^{(1)},\ldots,\boldsymbol{z}^{(M)}}} \sum_i \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)})$$

Applying ML to the multinomial mixture

$$\tilde{\ell}(\theta) = \sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}) = \sum_{i,k} z_k^{(i)} \log \mathcal{N}(\mathbf{x}^{(i)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} z_k^{(i)} \log(\pi_k),$$

- If we knew $\boldsymbol{z}^{(i)}$ we could maximize $\tilde{\ell}(\theta)$.
- If we knew $\theta = \big(\boldsymbol{\pi}, (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)_{1\le k\le K}\big)$, we could find the best $\boldsymbol{z}^{(i)}$ since we could compute the true a posteriori on $\boldsymbol{z}^{(i)}$ given $\mathbf{x}^{(i)}$:

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

$\rightarrow$ Seems a chicken and egg problem...

- In addition, we want to solve

$$\max_{\theta} \sum_i \log \Big(\sum_{\boldsymbol{z}^{(i)}} p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}) \Big) \quad \text{and not} \quad \max_{\substack{\theta, \\ \boldsymbol{z}^{(1)},\dots,\boldsymbol{z}^{(M)}}} \sum_i \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)})$$

- Can we still use the intuitions above to construct an algorithm maximizing the marginal likelihood?

Principle of the Expectation-Maximization Algorithm

$$\log p(\mathbf{x};\boldsymbol{\theta}) \quad =$$

Principle of the Expectation-Maximization Algorithm

$$\log p(\mathbf{x};\boldsymbol{\theta}) = \log \sum_{\boldsymbol{z}} p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})$$

Principle of the Expectation-Maximization Algorithm

$$\log p(\mathbf{x};\boldsymbol{\theta}) = \log \sum_{\boldsymbol{z}} p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta}) = \log \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \frac{p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})}$$

Principle of the Expectation-Maximization Algorithm

$$\begin{aligned}
\log p(\mathbf{x};\boldsymbol{\theta}) &= \log \sum_{\boldsymbol{z}} p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta}) = \log \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \frac{p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})} \\
&\geq \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \log \frac{p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})}
\end{aligned}$$

Principle of the Expectation-Maximization Algorithm

$$
\begin{aligned}
\log p(\mathbf{x}; \boldsymbol{\theta}) &= \log \sum_{\boldsymbol{z}} p(\mathbf{x}, \boldsymbol{z}; \boldsymbol{\theta}) = \log \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \frac{p(\mathbf{x}, \boldsymbol{z}; \boldsymbol{\theta})}{q(\boldsymbol{z})} \\
&\geq \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \log \frac{p(\mathbf{x}, \boldsymbol{z}; \boldsymbol{\theta})}{q(\boldsymbol{z})} \\
&= \mathbb{E}_q[\log p(\mathbf{x}, \boldsymbol{z}; \boldsymbol{\theta})] + H(q)
\end{aligned}
$$

Principle of the Expectation-Maximization Algorithm

$$
\begin{aligned}
\log p(\mathbf{x}; \boldsymbol{\theta}) &= \log \sum_{\boldsymbol{z}} p(\mathbf{x}, \boldsymbol{z}; \boldsymbol{\theta}) = \log \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \frac{p(\mathbf{x}, \boldsymbol{z}; \boldsymbol{\theta})}{q(\boldsymbol{z})} \\
&\geq \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \log \frac{p(\mathbf{x}, \boldsymbol{z}; \boldsymbol{\theta})}{q(\boldsymbol{z})} \\
&= \mathbb{E}_q[\log p(\mathbf{x}, \boldsymbol{z}; \boldsymbol{\theta})] + H(q) =: \mathcal{L}(q, \boldsymbol{\theta})
\end{aligned}
$$

Principle of the Expectation-Maximization Algorithm

$$
\begin{aligned}
\log p(\mathbf{x};\boldsymbol{\theta}) &= \log \sum_{\boldsymbol{z}} p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta}) = \log \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \frac{p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})} \\
&\geq \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \log \frac{p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})} \\
&= \mathbb{E}_q[\log p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})] + H(q) =: \mathcal{L}(q,\boldsymbol{\theta})
\end{aligned}
$$

- This shows that $\mathcal{L}(q,\boldsymbol{\theta}) \leq \log p(\mathbf{x};\boldsymbol{\theta})$

Principle of the Expectation-Maximization Algorithm

$$
\begin{aligned}
\log p(\mathbf{x};\boldsymbol{\theta}) &= \log \sum_{\boldsymbol{z}} p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta}) = \log \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \frac{p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})} \\
&\geq \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \log \frac{p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})} \\
&= \mathbb{E}_q[\log p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})] + H(q) =: \mathcal{L}(q,\boldsymbol{\theta})
\end{aligned}
$$

- This shows that $\mathcal{L}(q,\boldsymbol{\theta}) \leq \log p(\mathbf{x};\boldsymbol{\theta})$
- $\boldsymbol{\theta} \mapsto \mathcal{L}(q,\boldsymbol{\theta})$ is typically a **concave** function[a].

Principle of the Expectation-Maximization Algorithm

$$
\begin{aligned}
\log p(\mathbf{x};\boldsymbol{\theta}) &= \log \sum_{\boldsymbol{z}} p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta}) = \log \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \frac{p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})} \\
&\geq \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \log \frac{p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})} \\
&= \mathbb{E}_q[\log p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})] + H(q) =: \mathcal{L}(q,\boldsymbol{\theta})
\end{aligned}
$$

- This shows that $\mathcal{L}(q,\boldsymbol{\theta}) \leq \log p(\mathbf{x};\boldsymbol{\theta})$
- $\boldsymbol{\theta} \mapsto \mathcal{L}(q,\boldsymbol{\theta})$ is typically a **concave** function[a].
- Finally it is possible to show that

$$\mathcal{L}(q,\boldsymbol{\theta}) = \log p(\mathbf{x};\boldsymbol{\theta}) - KL(q||p(\cdot|\mathbf{x};\boldsymbol{\theta}))$$

Principle of the Expectation-Maximization Algorithm

$$\begin{aligned}\log p(\mathbf{x};\boldsymbol{\theta}) &= \log \sum_{\boldsymbol{z}} p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta}) = \log \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \frac{p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})} \\ &\geq \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \log \frac{p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})} \\ &= \mathbb{E}_q[\log p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})] + H(q) =: \mathcal{L}(q,\boldsymbol{\theta})\end{aligned}$$

- This shows that $\mathcal{L}(q,\boldsymbol{\theta}) \leq \log p(\mathbf{x};\boldsymbol{\theta})$
- $\boldsymbol{\theta} \mapsto \mathcal{L}(q,\boldsymbol{\theta})$ is typically a **concave** function[a].
- Finally it is possible to show that

$$\mathcal{L}(q,\boldsymbol{\theta}) = \log p(\mathbf{x};\boldsymbol{\theta}) - KL(q||p(\cdot|\mathbf{x};\boldsymbol{\theta}))$$

So that if we set $q(\boldsymbol{z}) = p(\boldsymbol{z} \mid \mathbf{x};\boldsymbol{\theta}^{(t)})$ then

$$L(q,\boldsymbol{\theta}^{(t)}) = p(\mathbf{x};\theta^{(t)}).$$

Principle of the Expectation-Maximization Algorithm

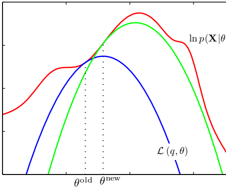
$$
\begin{aligned}
\log p(\mathbf{x};\boldsymbol{\theta}) &= \log \sum_{\boldsymbol{z}} p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta}) = \log \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \frac{p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})} \\
&\geq \sum_{\boldsymbol{z}} q(\boldsymbol{z}) \log \frac{p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})}{q(\boldsymbol{z})} \\
&= \mathbb{E}_q[\log p(\mathbf{x},\boldsymbol{z};\boldsymbol{\theta})] + H(q) =: \mathcal{L}(q,\boldsymbol{\theta})
\end{aligned}
$$

- This shows that $\mathcal{L}(q,\boldsymbol{\theta}) \leq \log p(\mathbf{x};\boldsymbol{\theta})$
- $\boldsymbol{\theta} \mapsto \mathcal{L}(q,\boldsymbol{\theta})$ is typically a **concave** function[a].
- Finally it is possible to show that

$$\mathcal{L}(q,\boldsymbol{\theta}) = \log p(\mathbf{x};\boldsymbol{\theta}) - KL(q||p(\cdot|\mathbf{x};\boldsymbol{\theta}))$$

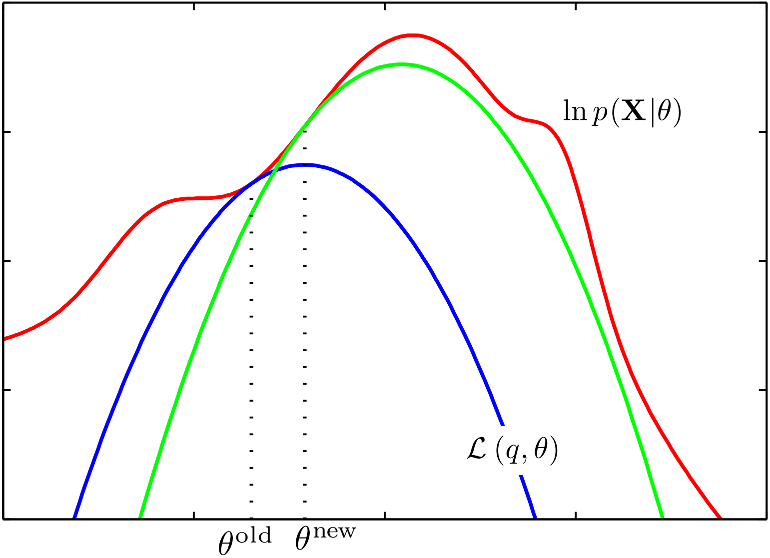
So that if we set $q(\boldsymbol{z}) = p(\boldsymbol{z} \mid \mathbf{x};\boldsymbol{\theta}^{(t)})$ then

$$L(q,\boldsymbol{\theta}^{(t)}) = p(\mathbf{x};\theta^{(t)}).$$



[a]If the complete log-likelihood is a canonical exponential family.

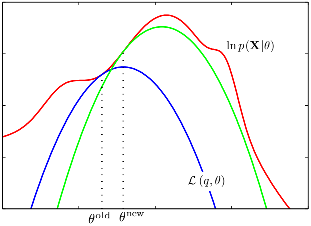
A graphical idea of the EM algorithm



Expectation Maximization algorithm

Expectation step

Maximization step



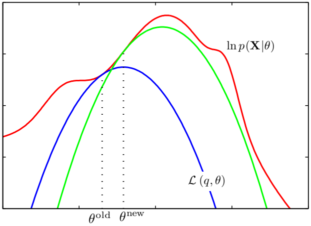
$$\begin{aligned} \boldsymbol{\theta}^{\text{old}} &= \boldsymbol{\theta}^{(t-1)} \\ \boldsymbol{\theta}^{\text{new}} &= \boldsymbol{\theta}^{(t)} \end{aligned}$$

Expectation Maximization algorithm

Expectation step

1. $q(\boldsymbol{z}) = p(\boldsymbol{z} \mid \mathbf{x}; \boldsymbol{\theta}^{(t-1)})$

Maximization step



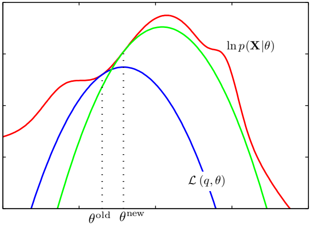
$$
\begin{aligned}
\boldsymbol{\theta}^{\text{old}} &= \boldsymbol{\theta}^{(t-1)} \\
\boldsymbol{\theta}^{\text{new}} &= \boldsymbol{\theta}^{(t)}
\end{aligned}
$$

Expectation Maximization algorithm

Expectation step

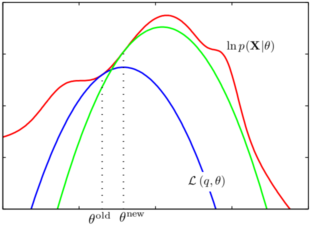
1. $q(\boldsymbol{z}) = p(\boldsymbol{z} \mid \mathbf{x}; \boldsymbol{\theta}^{(t-1)})$
2. $\mathcal{L}(q, \boldsymbol{\theta}) = \mathbb{E}_q\big[\log p(\mathbf{x}, \boldsymbol{z}; \boldsymbol{\theta})\big] + H(q)$

Maximization step



$$\begin{aligned} \boldsymbol{\theta}^{\text{old}} &= \boldsymbol{\theta}^{(t-1)} \\ \boldsymbol{\theta}^{\text{new}} &= \boldsymbol{\theta}^{(t)} \end{aligned}$$

Expectation Maximization algorithm



Expectation step

1. $q(\boldsymbol{z}) = p(\boldsymbol{z} \mid \mathbf{x}; \boldsymbol{\theta}^{(t-1)})$
2. $\mathcal{L}(q, \boldsymbol{\theta}) = \mathbb{E}_q \left[\log p(\mathbf{x}, \boldsymbol{z}; \boldsymbol{\theta}) \right] + H(q)$

Maximization step

1. $\boldsymbol{\theta}^{(t)} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \, \mathbb{E}_q \left[\log p(\mathbf{x}, \boldsymbol{z}; \boldsymbol{\theta}) \right]$

$$
\begin{aligned}
\boldsymbol{\theta}^{\text{old}} &= \boldsymbol{\theta}^{(t-1)} \\
\boldsymbol{\theta}^{\text{new}} &= \boldsymbol{\theta}^{(t)}
\end{aligned}
$$

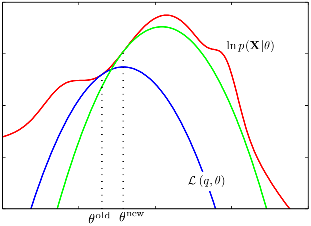
Expectation Maximization algorithm

Initialize $\boldsymbol{\theta} = \boldsymbol{\theta}_0$

WHILE (Not converged)

Expectation step

1. $q(\boldsymbol{z}) = p(\boldsymbol{z} \mid \mathbf{x}; \boldsymbol{\theta}^{(t-1)})$
2. $\mathcal{L}(q, \boldsymbol{\theta}) = \mathbb{E}_q\left[\log p(\mathbf{x}, \boldsymbol{z}; \boldsymbol{\theta})\right] + H(q)$



Maximization step

1. $\boldsymbol{\theta}^{(t)} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, \mathbb{E}_q\left[\log p(\mathbf{x}, \boldsymbol{z}; \boldsymbol{\theta})\right]$

$$\begin{aligned} \boldsymbol{\theta}^{\text{old}} &= \boldsymbol{\theta}^{(t-1)} \\ \boldsymbol{\theta}^{\text{new}} &= \boldsymbol{\theta}^{(t)} \end{aligned}$$

ENDWHILE

Expected complete log-likelihood

With the notation: $q_{ik}^{(t)} = \mathbb{P}_{q_i^{(t)}}(z_k^{(i)} = 1) = \mathbb{E}_{q_i^{(t)}}\left[z_k^{(i)}\right]$, we have

Expected complete log-likelihood

With the notation: $q_{ik}^{(t)} = \mathbb{P}_{q_i^{(t)}}(z_k^{(i)} = 1) = \mathbb{E}_{q_i^{(t)}}\left[z_k^{(i)}\right]$, we have

$$\mathbb{E}_{q^{(t)}}\left[\tilde{\ell}(\boldsymbol{\theta})\right] \quad =$$

Expected complete log-likelihood

With the notation: $q_{ik}^{(t)} = \mathbb{P}_{q_i^{(t)}}(z_k^{(i)} = 1) = \mathbb{E}_{q_i^{(t)}}\left[z_k^{(i)}\right]$, we have

$$\mathbb{E}_{q^{(t)}}\left[\tilde{\ell}(\boldsymbol{\theta})\right] = \mathbb{E}_{q^{(t)}}\left[\log p(\boldsymbol{X}, \boldsymbol{Z}; \boldsymbol{\theta})\right]$$

Expected complete log-likelihood

With the notation: $q_{ik}^{(t)} = \mathbb{P}_{q_i^{(t)}}(z_k^{(i)} = 1) = \mathbb{E}_{q_i^{(t)}}\left[z_k^{(i)}\right]$, we have

$$
\begin{aligned}
\mathbb{E}_{q^{(t)}}\left[\tilde{\ell}(\boldsymbol{\theta})\right] &= \mathbb{E}_{q^{(t)}}\left[\log p(\boldsymbol{X}, \boldsymbol{Z}; \boldsymbol{\theta})\right] \\
&= \mathbb{E}_{q^{(t)}}\left[\sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}; \boldsymbol{\theta})\right]
\end{aligned}
$$

Expected complete log-likelihood

With the notation: $q_{ik}^{(t)} = \mathbb{P}_{q_i^{(t)}}(z_k^{(i)} = 1) = \mathbb{E}_{q_i^{(t)}}\left[z_k^{(i)}\right]$, we have

$$
\begin{aligned}
\mathbb{E}_{q^{(t)}}\left[\tilde{\ell}(\boldsymbol{\theta})\right] &= \mathbb{E}_{q^{(t)}}\left[\log p(\boldsymbol{X}, \boldsymbol{Z}; \boldsymbol{\theta})\right] \\
&= \mathbb{E}_{q^{(t)}}\left[\sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}; \boldsymbol{\theta})\right] \\
&= \mathbb{E}_{q^{(t)}}\left[\sum_{i,k} z_k^{(i)} \log \mathcal{N}(\mathbf{x}^{(i)}, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} z_k^{(i)} \log(\pi_k)\right]
\end{aligned}
$$

Expected complete log-likelihood

With the notation: $q_{ik}^{(t)} = \mathbb{P}_{q_i^{(t)}}(z_k^{(i)} = 1) = \mathbb{E}_{q_i^{(t)}}\left[z_k^{(i)}\right]$, we have

$$
\begin{aligned}
\mathbb{E}_{q^{(t)}}\left[\tilde{\ell}(\boldsymbol{\theta})\right] &= \mathbb{E}_{q^{(t)}}\left[\log p(\boldsymbol{X}, \boldsymbol{Z}; \boldsymbol{\theta})\right] \\
&= \mathbb{E}_{q^{(t)}}\left[\sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}; \boldsymbol{\theta})\right] \\
&= \mathbb{E}_{q^{(t)}}\left[\sum_{i,k} z_k^{(i)} \log \mathcal{N}(\mathbf{x}^{(i)}, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} z_k^{(i)} \log(\pi_k)\right] \\
&= \sum_{i,k} \mathbb{E}_{q_i^{(t)}}\left[z_k^{(i)}\right] \log \mathcal{N}(\mathbf{x}^{(i)}, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} \mathbb{E}_{q_i^{(t)}}\left[z_k^{(i)}\right] \log(\pi_k)
\end{aligned}
$$

Expected complete log-likelihood

With the notation: $q_{ik}^{(t)} = \mathbb{P}_{q_i^{(t)}}(z_k^{(i)} = 1) = \mathbb{E}_{q_i^{(t)}}\left[z_k^{(i)}\right]$, we have

$$
\begin{aligned}
\mathbb{E}_{q^{(t)}}\left[\tilde{\ell}(\boldsymbol{\theta})\right] &= \mathbb{E}_{q^{(t)}}\left[\log p(\boldsymbol{X}, \boldsymbol{Z}; \boldsymbol{\theta})\right] \\
&= \mathbb{E}_{q^{(t)}}\left[\sum_{i=1}^{M} \log p(\mathbf{x}^{(i)}, \boldsymbol{z}^{(i)}; \boldsymbol{\theta})\right] \\
&= \mathbb{E}_{q^{(t)}}\left[\sum_{i,k} z_k^{(i)} \log \mathcal{N}(\mathbf{x}^{(i)}, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} z_k^{(i)} \log(\pi_k)\right] \\
&= \sum_{i,\,k} \mathbb{E}_{q_i^{(t)}}\left[z_k^{(i)}\right] \log \mathcal{N}(\mathbf{x}^{(i)}, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} \mathbb{E}_{q_i^{(t)}}\left[z_k^{(i)}\right] \log(\pi_k) \\
&= \sum_{i,\,k} q_{ik}^{(t)} \log \mathcal{N}(\mathbf{x}^{(i)}, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \sum_{i,k} q_{ik}^{(t)} \log(\pi_k)
\end{aligned}
$$

Expectation step for the Gaussian mixture

We computed previously $q_i^{(t)}(\boldsymbol{z}^{(i)})$, which is a multinomial distribution defined by

$$q_i^{(t)}(\boldsymbol{z}^{(i)}) = p(\boldsymbol{z}^{(i)}|\mathbf{x}^{(i)};\boldsymbol{\theta}^{(t-1)})$$

Expectation step for the Gaussian mixture

We computed previously $q_i^{(t)}(\boldsymbol{z}^{(i)})$, which is a multinomial distribution defined by

$$q_i^{(t)}(\boldsymbol{z}^{(i)}) = p(\boldsymbol{z}^{(i)}|\mathbf{x}^{(i)};\boldsymbol{\theta}^{(t-1)})$$

Abusing notation we will denote $(q_{i1}^{(t)}, \ldots, q_{iK}^{(t)})$ the corresponding vector of probabilities defined by

$$q_{ik}^{(t)} = \mathbb{P}_{q_i^{(t)}}(z_k^{(i)} = 1) = \mathbb{E}_{q_i^{(t)}}\left[z_k^{(i)}\right]$$

Expectation step for the Gaussian mixture

We computed previously $q_i^{(t)}(\boldsymbol{z}^{(i)})$, which is a multinomial distribution defined by

$$q_i^{(t)}(\boldsymbol{z}^{(i)}) = p(\boldsymbol{z}^{(i)}|\mathbf{x}^{(i)};\boldsymbol{\theta}^{(t-1)})$$

Abusing notation we will denote $(q_{i1}^{(t)}, \dots, q_{iK}^{(t)})$ the corresponding vector of probabilities defined by

$$q_{ik}^{(t)} = \mathbb{P}_{q_i^{(t)}}(z_k^{(i)} = 1) = \mathbb{E}_{q_i^{(t)}}\left[z_k^{(i)}\right]$$

$$q_{ik}^{(t)} = p(z_k^{(i)} = 1 \mid \mathbf{x}^{(i)};\boldsymbol{\theta}^{(t-1)}) = \frac{\pi_k^{(t-1)}\mathcal{N}(\mathbf{x}^{(i)},\boldsymbol{\mu}_k^{(t-1)},\boldsymbol{\Sigma}_k^{(t-1)})}{\sum_{j=1}^K \pi_j^{(t-1)}\mathcal{N}(\mathbf{x}^{(i)},\boldsymbol{\mu}_j^{(t-1)},\boldsymbol{\Sigma}_j^{(t-1)})}$$

Maximization step for the Gaussian mixture

$$\left(\boldsymbol{\pi}^t, (\boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)})_{1\leq k\leq K}\right) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, \mathbb{E}_{q^{(t)}}\left[\tilde{\ell}(\boldsymbol{\theta})\right]$$

Maximization step for the Gaussian mixture

$$\left(\boldsymbol{\pi}^t, (\boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)})_{1\leq k\leq K}\right) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, \mathbb{E}_{q^{(t)}}\left[\tilde{\ell}(\boldsymbol{\theta})\right]$$

This yields the updates:

$$\boxed{\boldsymbol{\mu}_k^{(t)} = \frac{\sum_i \mathbf{x}^{(i)}\, q_{ik}^{(t)}}{\sum_i q_{ik}^{(t)}}}, \quad \boxed{\boldsymbol{\Sigma}_k^{(t)} = \frac{\sum_i \left(\mathbf{x}^{(i)} - \boldsymbol{\mu}_k^{(t)}\right)\left(\mathbf{x}^{(i)} - \boldsymbol{\mu}_k^{(t)}\right)^\top q_{ik}^{(t)}}{\sum_i q_{ik}^{(t)}}}$$

and $$\boxed{\pi_k^{(t)} = \frac{\sum_i q_{ik}^{(t)}}{\sum_{i,k'} q_{ik'}^{(t)}}}$$

Final EM algorithm for the Multinomial mixture model

Initialize $\boldsymbol{\theta} = \boldsymbol{\theta}_0$

WHILE (Not converged)

Expectation step

$$q_{ik}^{(t)} \leftarrow \frac{\pi_k^{(t-1)} \mathcal{N}(\mathbf{x}^{(i)}, \boldsymbol{\mu}_k^{(t-1)}, \boldsymbol{\Sigma}_k^{(t-1)})}{\sum_{j=1}^{K} \pi_j^{(t-1)} \mathcal{N}(\mathbf{x}^{(i)}, \boldsymbol{\mu}_j^{(t-1)}, \boldsymbol{\Sigma}_j^{(t-1)})}$$

Maximization step

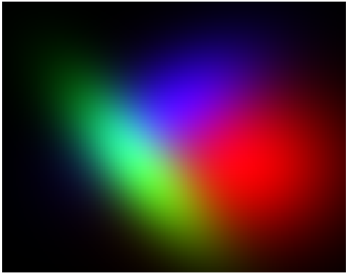
$$\boldsymbol{\mu}_k^{(t)} = \frac{\sum_i \mathbf{x}^{(i)} q_{ik}^{(t)}}{\sum_i q_{ik}^{(t)}}, \quad \boldsymbol{\Sigma}_k^{(t)} = \frac{\sum_i \left(\mathbf{x}^{(i)} - \boldsymbol{\mu}_k^{(t)}\right)\left(\mathbf{x}^{(i)} - \boldsymbol{\mu}_k^{(t)}\right)^\top q_{ik}^{(t)}}{\sum_i q_{ik}^{(t)}}$$

$$\text{and} \qquad \pi_k^{(t)} = \frac{\sum_i q_{ik}^{(t)}}{\sum_{i,k'} q_{ik'}^{(t)}}$$

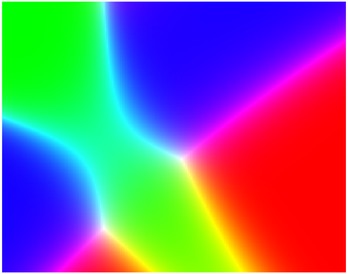
ENDWHILE

EM Algorithm for the Gaussian mixture model III

$p(\mathbf{x}|\boldsymbol{z})$

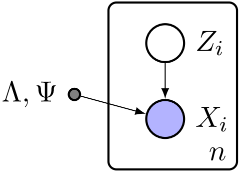


$p(\boldsymbol{z}|\mathbf{x})$



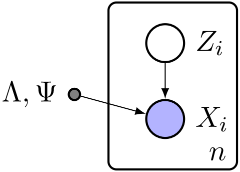
Outline

Factorial Analysis



- $\Lambda \in \mathbb{R}^{d \times k}$ is the matrix of *factors* or *principal directions*

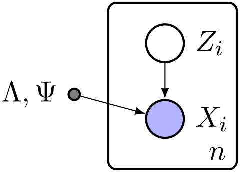
Factorial Analysis



- $\Lambda \in \mathbb{R}^{d \times k}$ is the matrix of *factors* or *principal directions*
- $Z_i \in \mathbb{R}^k$ are the *loadings* or *principal components*

$$Z_i \sim \mathcal{N}(0, I_k)$$

Factorial Analysis



- $\Lambda \in \mathbb{R}^{d \times k}$ is the matrix of *factors* or *principal directions*
- $Z_i \in \mathbb{R}^k$ are the *loadings* or *principal components*

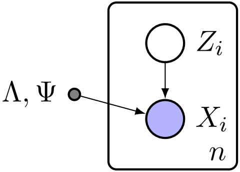
$$Z_i \sim \mathcal{N}(0, I_k)$$

- $X_i \in \mathbb{R}^d$ is the observed data modeled as

$$X_i = \Lambda Z_i + \varepsilon_i \quad \text{with} \quad \varepsilon_i \sim \mathcal{N}(0, \Psi).$$

with $\Psi \in \mathbb{R}^{d \times d}$, constrained to be diagonal.

Factorial Analysis



- $\Lambda \in \mathbb{R}^{d \times k}$ is the matrix of *factors* or *principal directions*
- $Z_i \in \mathbb{R}^k$ are the *loadings* or *principal components*

$$Z_i \sim \mathcal{N}(0, I_k)$$

- $X_i \in \mathbb{R}^d$ is the observed data modeled as

$$X_i = \Lambda Z_i + \varepsilon_i \quad \text{with} \quad \varepsilon_i \sim \mathcal{N}(0, \Psi).$$

  with $\Psi \in \mathbb{R}^{d \times d}$, constrained to be diagonal.

The model essentially retrieves Principal Component Analysis for $\Psi = \sigma^2 I_d$.

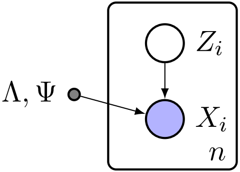
Factorial Analysis

$$Z_i \sim \mathcal{N}(0, I_k)$$

Factorial Analysis

$$Z_i \sim \mathcal{N}(0, I_k)$$

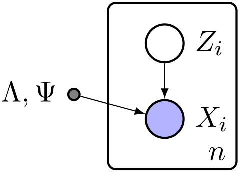
$$X_i = \Lambda Z_i + \varepsilon_i \quad \text{with} \quad \varepsilon_i \sim \mathcal{N}(0, \Psi).$$



Λ can be learned (up to a rotation on the right) together with Ψ using an EM algorithm, where Z is treated as a latent variable.

Factorial Analysis

$$Z_i \sim \mathcal{N}(0, I_k)$$



$$X_i = \Lambda Z_i + \varepsilon_i \quad \text{with} \quad \varepsilon_i \sim \mathcal{N}(0, \Psi).$$

Λ can be learned (up to a rotation on the right) together with Ψ using an EM algorithm, where Z is treated as a latent variable.

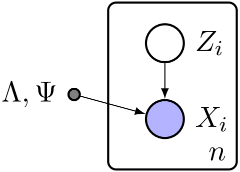
Advantages of the probabilistic formulation over vanilla PCA

- Possible to model non-isotropic noise

Factorial Analysis

$$Z_i \sim \mathcal{N}(0, I_k)$$

$$X_i = \Lambda Z_i + \varepsilon_i \quad \text{with} \quad \varepsilon_i \sim \mathcal{N}(0, \Psi).$$

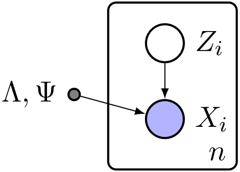


Λ can be learned (up to a rotation on the right) together with Ψ using an EM algorithm, where Z is treated as a latent variable.

Advantages of the probabilistic formulation over vanilla PCA

- Possible to model non-isotropic noise
- X can have missing entries
  (then treated as latent variables in EM)

Factorial Analysis



$$Z_i \sim \mathcal{N}(0, I_k)$$

$$X_i = \Lambda Z_i + \varepsilon_i \quad \text{with} \quad \varepsilon_i \sim \mathcal{N}(0, \Psi).$$

Λ can be learned (up to a rotation on the right) together with Ψ using an EM algorithm, where Z is treated as a latent variable.

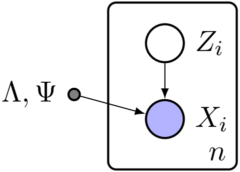
Advantages of the probabilistic formulation over vanilla PCA

- Possible to model non-isotropic noise
- X can have missing entries
  (then treated as latent variables in EM)
- By changing the distributions on Z_i and X_i, we can design variant of PCA more suitable for different type of data: Multinomial PCA, Poisson PCA, etc.

Factorial Analysis

$$Z_i \sim \mathcal{N}(0, I_k)$$

$$X_i = \Lambda Z_i + \varepsilon_i \quad \text{with} \quad \varepsilon_i \sim \mathcal{N}(0, \Psi).$$



Λ can be learned (up to a rotation on the right) together with Ψ using an EM algorithm, where Z is treated as a latent variable.

Advantages of the probabilistic formulation over vanilla PCA

- Possible to model non-isotropic noise
- X can have missing entries
  (then treated as latent variables in EM)
- By changing the distributions on Z_i and X_i, we can design variant of PCA more suitable for different type of data: Multinomial PCA, Poisson PCA, etc.
- Can be inserted in a mixture of Gaussians model to help model Gaussians in high dimension.

Latent Dirichlet Allocation as Multinomial PCA

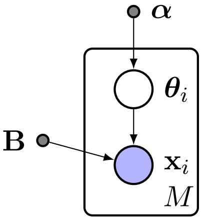
Replacing

- the distribution on Z_i by a Dirichlet distribution
- the distribution of X_i by a Multinomial

Latent Dirichlet Allocation as Multinomial PCA

Replacing

- the distribution on Z_i by a Dirichlet distribution
- the distribution of X_i by a Multinomial



- Topic proportions for document i: $\boldsymbol{\theta}_i \in \mathbb{R}^K$

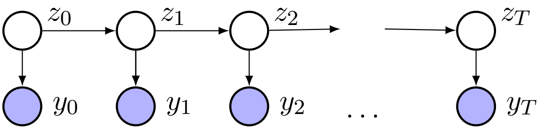
$$\boldsymbol{\theta}_i \sim \mathrm{Dir}(\boldsymbol{\alpha})$$

- Empirical words counts for document i: $\mathbf{x}_i \in \mathbb{R}^d$

$$\mathbf{x}_i \sim \mathcal{M}(N_i, \mathbf{B}\boldsymbol{\theta}_i)$$

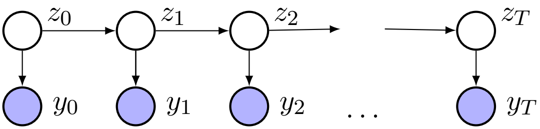
Temporal models

Hidden Markov Model and Kalman Filter

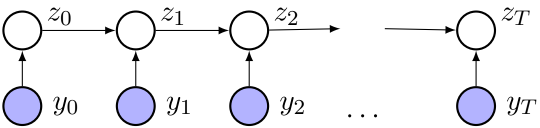


Temporal models

Hidden Markov Model and Kalman Filter



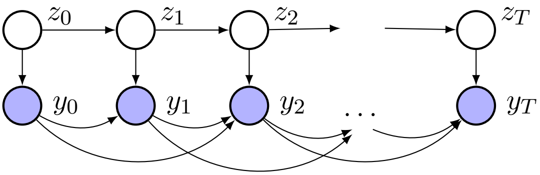
Conditional Random Field (chain case)



- A structured version of *logistic regression* where the output is a sequence.

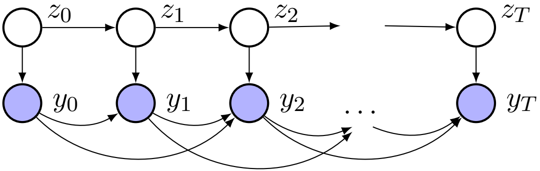
More temporal models

Second order auto-regressive model with latent switching state

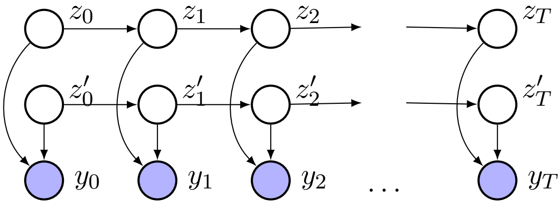


More temporal models

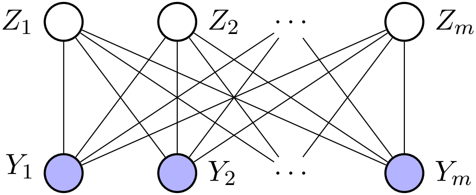
Second order auto-regressive model with latent switching state



Factorial Hidden Markov models (Ghahramani and Jordan, 1996)



Restricted Boltzman Machines (Smolensky, 1986)



$$P(Y,Z) = \exp\left(\langle Y,\theta\rangle + Z^\top WY + \langle Z,\eta\rangle - A(\theta,W,\eta)\right)$$

- $p(Z|Y) = \prod_{i=1}^d p(Z_i|Y)$ are independent Bernoulli r.v.
- $p(Y|Z) = \prod_{i=1}^d p(Y_i|Z)$ are independent Bernoulli r.v.

However the model encodes non-trivial dependences between the variables $(Y_1,\ldots,Y_n)$

Ising model

Reminder: $X = (X_i)_{i \in V}$ is a vector of random variables, taking value in $\{0,1\}^{|V|}$, whose distribution has the following exponential form:

$$p(x) = e^{-A(\eta)} \prod_{i \in V} e^{\eta_i x_i} \prod_{(i,j) \in E} e^{\eta_{i,j} x_i x_j}$$

Ising model

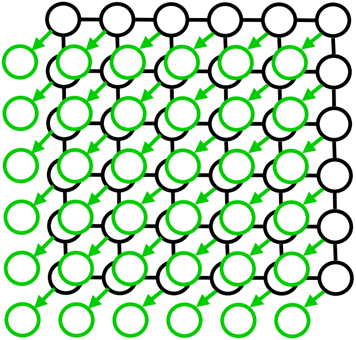
Reminder: $X = (X_i)_{i \in V}$ is a vector of random variables, taking value in $\{0,1\}^{|V|}$, whose distribution has the following exponential form:

$$p(x) = e^{-A(\eta)} \prod_{i \in V} e^{\eta_i x_i} \prod_{(i,j) \in E} e^{\eta_{i,j} x_i x_j}$$

The associated log-likelihood is this:

$$\ell(\eta) = \sum_{i \in V} \eta_i x_i + \sum_{(i,j) \in E} \eta_{i,j} x_i x_j - A(\eta)$$

Hidden Markov Random Field



Original image

Segmentation

Hidden Markov random Field

$$p(y|x) = e^{-A(\eta)} \prod_{i \in V} e^{\langle w, x_i \rangle y_i} \prod_{(i,j) \in E} e^{\eta_{i,j} y_i y_j}$$

Hidden Markov random Field

$$p(y|x) = e^{-A(\eta)} \prod_{i \in V} e^{\langle w, x_i \rangle y_i} \prod_{(i,j) \in E} e^{\eta_{i,j} y_i y_j}$$

The associated log-likelihood is this:

$$\ell(\eta) = \sum_{i \in V} \langle w, x_i \rangle y_i + \sum_{(i,j) \in E} \eta_{i,j} y_i y_j - A(w)$$

Hidden Markov random Field

$$p(y|x) = e^{-A(\eta)} \prod_{i \in V} e^{\langle w, x_i \rangle y_i} \prod_{(i,j) \in E} e^{\eta_{i,j} y_i y_j}$$

The associated log-likelihood is this:

$$\ell(\eta) = \sum_{i \in V} \langle w, x_i \rangle y_i + \sum_{(i,j) \in E} \eta_{i,j} y_i y_j - A(w)$$

References I

Ghahramani, Z. and Jordan, M. I. (1996). Factorial hidden markov models. In *Advances in Neural Information Processing Systems*, pages 472–478.

Smolensky, P. (1986). Information processing in dynamical systems: foundations of harmony theory. In *Parallel distributed processing: explorations in the microstructure of cognition, vol. 1*, pages 194–281. MIT Press.