The Expectation-Maximization Algorithm

Elliot Creager

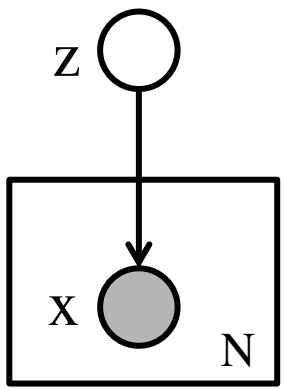
CSC 412 Tutorial
slides due to Yujia Li

March 22, 2018

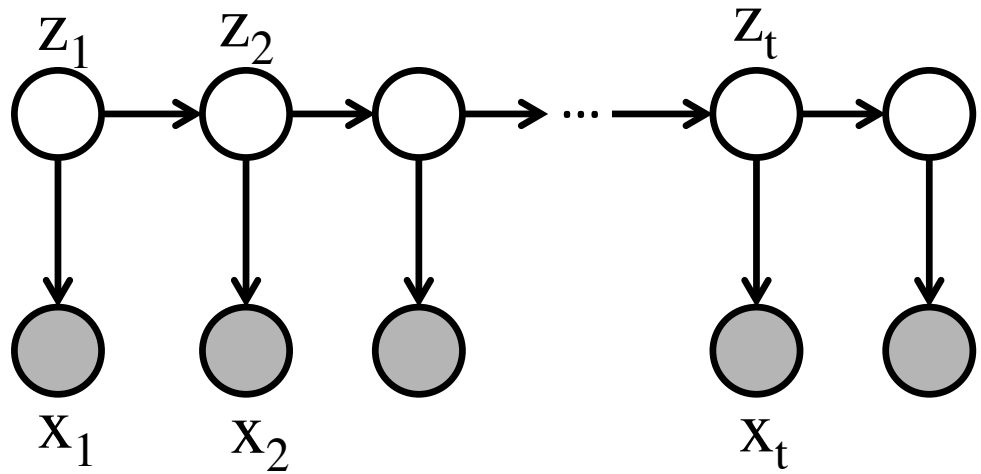
Latent Variable Model

- Some of the variables in the model are not observed.
- Examples: mixture model, HMM, LDA, many more
- We consider the learning problem of latent variable models.

Mixture Model



Hidden Markov Model



Marginal Likelihood

- Joint model $p(\mathbf{x}, \mathbf{z}|\theta)$, θ is model parameter
- With $\mathbf{z}$ unobserved, we marginalize out $\mathbf{z}$ and use the marginal log-likelihood for learning

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

- Example: mixture model. $\mathbf{z} \in \{1, 2, ..., K\}$

$$\begin{aligned} \log p(\mathbf{x}|\theta) &= \log \sum_{k} p(\mathbf{z} = k|\theta_k) p(\mathbf{x}|\mathbf{z} = k, \theta_k) \\ &= \log \sum_{k} \pi_k p(\mathbf{x}|\mathbf{z} = k, \theta_k) \end{aligned}$$

π_k mixing proportions.

Examples

- Mixture of Bernoulli

$$p(\mathbf{x}|\mathbf{z}=k,\theta_k) = p(\mathbf{x}|\mu_k) = \prod_i \mu_k^{x_i}(1-\mu_k)^{1-x_i}$$

- Mixture of Gaussians

$$\begin{aligned} p(\mathbf{x}|\mathbf{z}=k,\theta_k) &= p(\mathbf{x}|\mu_k,\Sigma_k) \\ &= \frac{1}{|2\pi\Sigma_k|^{\frac{D}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu_k)^\top \Sigma_k^{-1}(\mathbf{x}-\mu_k)\right) \end{aligned}$$

- Hidden Markov Model

$$p(\mathbf{Z}) = p(\mathbf{z}_1)\prod_t p(\mathbf{z}_t|\mathbf{z}_{t-1})$$

$$p(\mathbf{X}|\mathbf{Z}) = \prod_t p(\mathbf{x}_t|\mathbf{z}_t)$$

Learning

- If all $\mathbf{z}$ observed, the likelihood factorizes, and learning is relatively easy

$$\ell(\mathbf{x}, \mathbf{z}|\theta) = \log p(\mathbf{x}, \mathbf{z}|\theta) = \log p(\mathbf{z}|\theta) + \log p(\mathbf{x}|\mathbf{z}, \theta)$$

- If $\mathbf{z}$ not observed, we have to handle the sum inside log.
- Idea 1: ignore this problem and simply take derivative and follow the gradient.
- Idea 2: use the current θ to estimate $\mathbf{z}$, fill them in and do fully-observed learning.

EM Algorithm

Intuition: iterate two steps

- Based on the current $\theta^{(t)}$, fill in unobserved $\mathbf{z}$ to get $\mathbf{z}'$
- Update θ to optimize $\ell(\mathbf{x}, \mathbf{z}'|\theta)$

How to choose $\mathbf{z}'$?

- Use $p(\mathbf{z}|\mathbf{x}, \theta^{(t)})$.
- We don't want to ignore any possible $\mathbf{z}$ so optimize the expectation instead

$$\sum_{\mathbf{z}} p(\mathbf{z}|\mathbf{x}, \theta^{(t)}) \ell(\mathbf{x}, \mathbf{z}|\theta)$$

EM Algorithm

More precisely, we start from some initial $\theta^{(0)}$. Then iterate the following two steps until convergence

E(xpectation)-Step Compute $p(\mathbf{z}|\mathbf{x}, \theta^{(t)})$ and form the expectation using the current $\theta^{(t)}$.

M(aximization)-Step Find θ that maximizes the expected complete-data likelihood

$$\theta^{(t+1)} \leftarrow \operatorname*{argmax}_{\theta} \sum_{\mathbf{z}} p(\mathbf{z}|\mathbf{x}, \theta^{(t)}) \ell(\mathbf{x}, \mathbf{z}|\theta)$$

In many cases the expectation is easier to handle than marginal log-likelihood - no sum in the log.

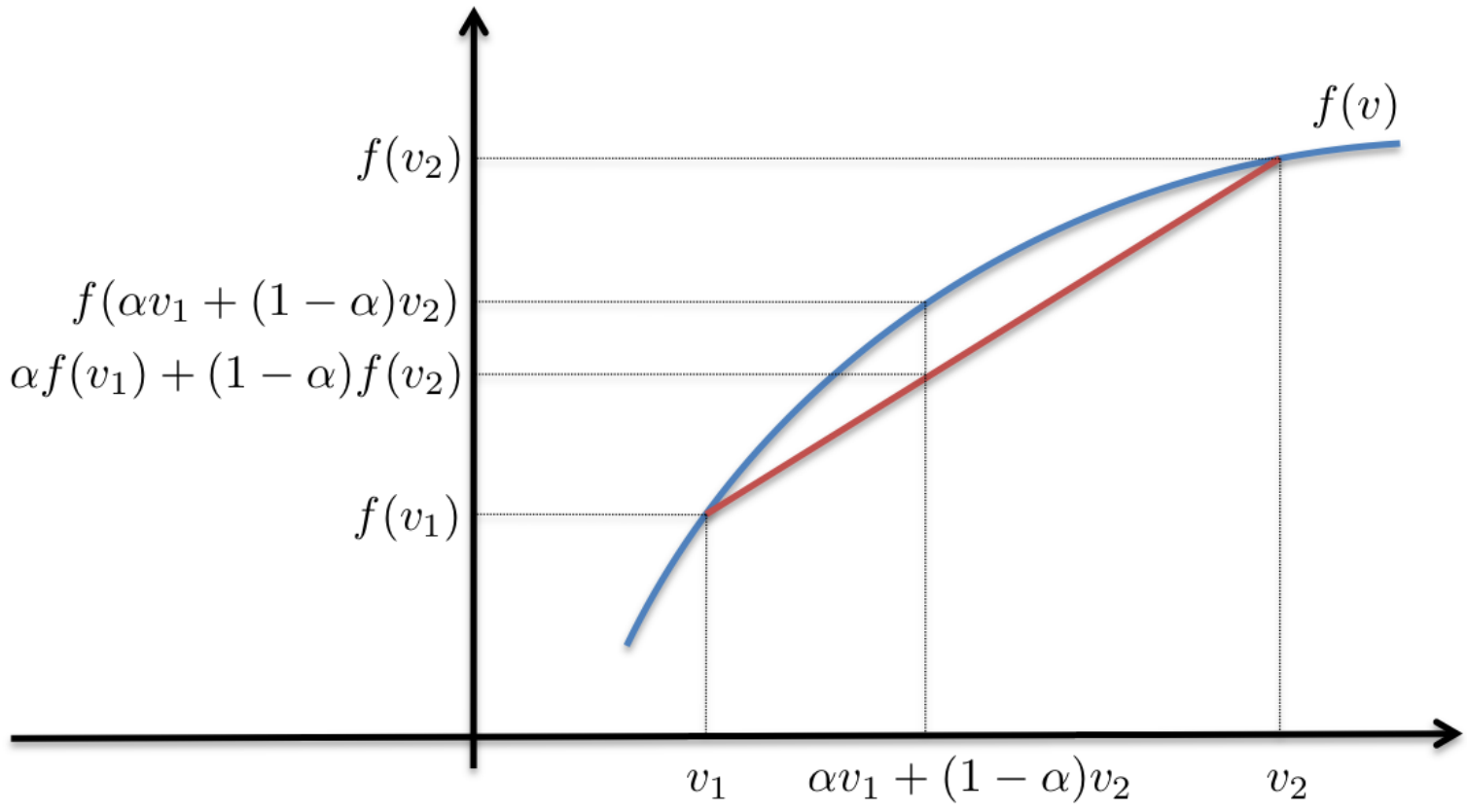
Justification of EM

EM algorithm can be derived as optimizing a lower bound on $\mathcal{L}(\theta) = \ell(\mathbf{x}|\theta)$. As $\log$ is concave, for any distribution $q(\mathbf{z})$

$$\begin{aligned}
\mathcal{L} &= \log \sum_{\mathbf{z}} p(\mathbf{x}, \mathbf{z}) = \log \sum_{\mathbf{z}} q(\mathbf{z}) \frac{p(\mathbf{x}, \mathbf{z})}{q(\mathbf{z})} \\
&\geq \sum_{\mathbf{z}} q(\mathbf{z}) \log \frac{p(\mathbf{x}, \mathbf{z})}{q(\mathbf{z})} \quad (\text{Jensen}'\text{s Inequality}) \\
&= \sum_{\mathbf{z}} q(\mathbf{z}) \log p(\mathbf{x}, \mathbf{z}) - \sum_{\mathbf{z}} q(\mathbf{z}) \log q(\mathbf{z})
\end{aligned}$$

- $\sum_{\mathbf{z}} q(\mathbf{z}) \log p(\mathbf{x}, \mathbf{z})$: expected complete-data log-likelihood
- $\mathcal{H}(q) \triangleq -\sum_{\mathbf{z}} q(\mathbf{z}) \log q(\mathbf{z})$: entropy
- $\mathcal{F}(q, \theta) \triangleq \sum_{\mathbf{z}} q(\mathbf{z}) \log p(\mathbf{x}, \mathbf{z}|\theta) - \sum_{\mathbf{z}} q(\mathbf{z}) \log q(\mathbf{z})$: free-energy

Jensen's inequality



$$\log \mathbb{E}[f] \geq \mathbb{E}[\log f]$$

Maximize Lower Bound on Log-Likelihood

- Free-energy is a lower bound of the true log-likelihood

$$\mathcal{L}(\theta) \geq \mathcal{F}(q, \theta)$$

- EM is simply doing coordinate ascent on $\mathcal{F}(q, \theta)$.

  E-Step $q^{(t)} \leftarrow \operatorname{argmax}_q \mathcal{F}(q, \theta^{(t)})$

  M-Step $\theta^{(t+1)} \leftarrow \operatorname{argmax}_\theta \mathcal{F}(q^{(t)}, \theta)$

- Put another way, EM is a special case of variational inference with q chosen iteratively to maximize $\mathcal{F}$ with for a fixed $\theta^{(t)}$
- Properties:
  - Each iteration improves $\mathcal{F}$
  - Each iteration improves $\mathcal{L}$ as well, will show later

E-Step

Find q that maximizes $\mathcal{F}(q, \theta^{(t)})$.

$$
\begin{aligned}
\mathcal{F}(q) &= \sum_{\mathbf{z}} q(\mathbf{z}) \log p(\mathbf{x}, \mathbf{z}) - \sum_{\mathbf{z}} q(\mathbf{z}) \log q(\mathbf{z}) \\
&= \sum_{\mathbf{z}} q(\mathbf{z}) \log \frac{p(\mathbf{z}|\mathbf{x}) p(\mathbf{x})}{q(\mathbf{z})} \\
&= \sum_{\mathbf{z}} q(\mathbf{z}) \log \frac{p(\mathbf{z}|\mathbf{x})}{q(\mathbf{z})} + \sum_{\mathbf{z}} q(\mathbf{z}) \log p(\mathbf{x}) \\
&= \sum_{\mathbf{z}} q(\mathbf{z}) \log \frac{p(\mathbf{z}|\mathbf{x})}{q(\mathbf{z})} + \log p(\mathbf{x}) \\
&= -\mathrm{KL}(q(\mathbf{z}) || p(\mathbf{z}|\mathbf{x})) + \mathcal{L}
\end{aligned}
$$

E-Step

$$\mathcal{F}(q, \theta^{(t)}) = -\mathrm{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}, \theta^{(t)})) + \mathcal{L}(\theta^{(t)})$$

For any p and q, $\mathrm{KL}(q||p) = \sum_{\mathbf{z}} q(\mathbf{z}) \log \frac{q(\mathbf{z})}{p(\mathbf{z})}$ is the Kullback-Leibler (KL) divergence, a measure of distance between distributions.

- Always non-negative
- $\mathrm{KL} = 0$ iff $p = q$.

Therefore $\mathcal{F}$ is maximized when $q^{(t)}(\mathbf{z}) = p(\mathbf{z}|\mathbf{x}, \theta^{(t)})$, which implies

$$\mathcal{F}(q^{(t)}, \theta^{(t)}) = \mathcal{L}(\theta^{(t)})$$

So when we compute $p(\mathbf{z}|\mathbf{x}, \theta^{(t)})$ we are actually computing $\mathrm{argmax}_q \mathcal{F}(q, \theta^{(t)})$.

M-Step

Find $\theta^{(t+1)}$ that maximizes $\mathcal{F}(q^{(t)}, \theta)$

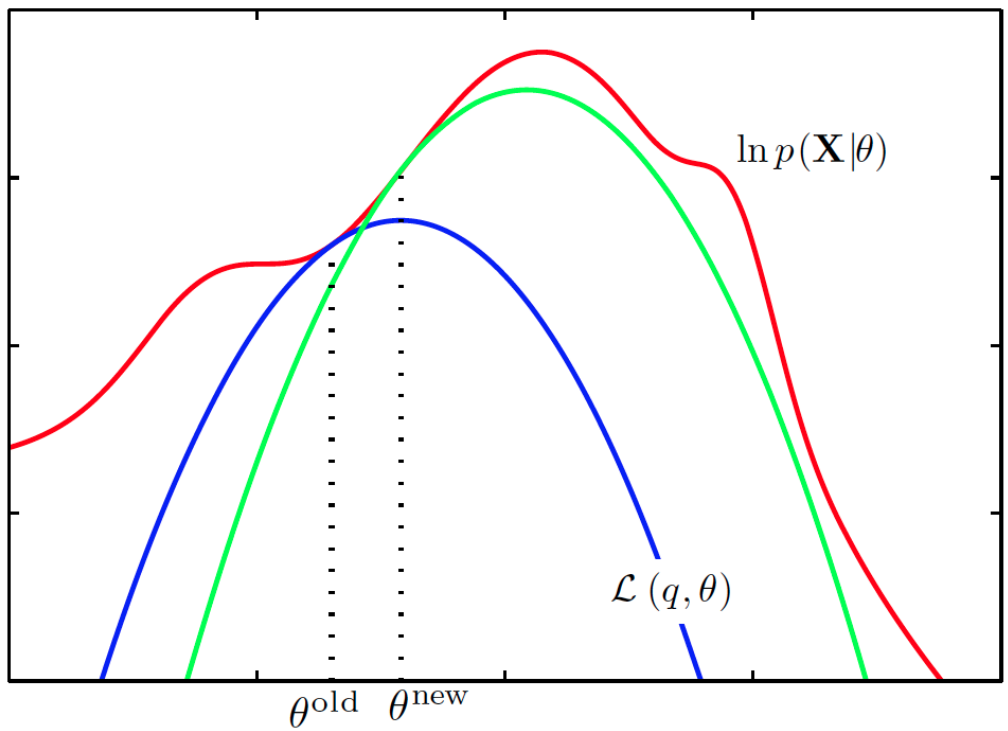
$$\mathcal{F}(q^{(t)}, \theta) = \sum_{\mathbf{z}} q^{(t)}(\mathbf{z}) \log p(\mathbf{x}, \mathbf{z}|\theta) + \mathcal{H}(q^{(t)})$$

$\mathcal{H}(q^{(t)})$ is constant in this step, we only care about the expected complete-data log-likelihood $\sum_{\mathbf{z}} q^{(t)}(\mathbf{z}) \log p(\mathbf{x}, \mathbf{z}|\theta)$.

- Usually can be solved in the same way as the full-observed model.

Improving $\mathcal{L}$ in Every Iteration

$$\begin{aligned}\mathcal{L}(\theta^{(t+1)}) &\geq \mathcal{F}(q^{(t)}, \theta^{(t+1)}) \\ &= \max_{\theta} \mathcal{F}(q^{(t)}, \theta) \\ &\geq \mathcal{F}(q^{(t)}, \theta^{(t)}) = \mathcal{L}(\theta^{(t)})\end{aligned}$$



Summary

EM algorithm is a way to find maximum likelihood estimates of the parameters of a latent variable model. Applicable when the original MLE problem can be broken into two pieces that are easy to solve

- Estimate missing/unobserved data from observed data using current parameters
- Find maximum likelihood parameters using the complete data

Good

- No need for gradients/learning rates/etc.
- Fast convergence
- Guaranteed to improve $\mathcal{L}$ at every iteration
- Modular design: reuse complete-data MLE code

Bad

- Can get stuck at local optima
- Requires "nice" distributions; need to compute $p(z|x)$

Example: Mixture of Gaussians

- $\mathbf{x} \in \mathbb{R}^D$, $\mathbf{z} = (z_1, ..., z_k) \in \{0, 1\}^K$, $\sum_k z_k = 1$
- $\{\pi_k, \mu_k, \Sigma_k\}$ are model parameters
- Complete-data log-likelihood

$$
\begin{aligned}
p(\mathbf{x}, \mathbf{z}) &= \log \pi_{\mathbf{z}} \mathcal{N}(\mathbf{x}|\mu_{\mathbf{z}}, \Sigma_{\mathbf{z}}) \\
&= \log \prod_k \left[\pi_k \mathcal{N}(\mathbf{x}|\mu_k, \Sigma_k)\right]^{z_k} \\
&= \sum_k z_k \left[\log \pi_k + \log \mathcal{N}(\mathbf{x}|\mu_k, \Sigma_k)\right]
\end{aligned}
$$

E-Step

- Assume we have a data set $\{\mathbf{x}_1, ..., \mathbf{x}_N\}$, each $\mathbf{x}_n \in \mathbb{R}^D$.
- E-Step: compute posterior using the current parameters, Bayes theorem

$$
\begin{aligned}
p(\mathbf{z}_n|\mathbf{x}_n) &= \frac{p(\mathbf{x}_n, \mathbf{z}_n)}{\sum_{\mathbf{z}} p(\mathbf{x}_n, \mathbf{z})} \\
p(z_{nk} = 1|\mathbf{x}_n) &= \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\mu_k, \Sigma_k)}{\sum_k \mathcal{N}(\mathbf{x}_n|\mu_k, \Sigma_k)}
\end{aligned}
$$

- Denote $\gamma_{nk} \triangleq p(z_{nk} = 1|\mathbf{x}_n)$, called responsibilities, $\sum_k \gamma_{nk} = 1$

M-Step

- Expected complete-data log-likelihood

$$
\begin{aligned}
L &= \sum_n \sum_{\mathbf{z}_n} p(\mathbf{z}_n|\mathbf{x}_n) \log p(\mathbf{x}_n, \mathbf{z}_n) \\
&= \sum_n \sum_{\mathbf{z}_n} p(\mathbf{z}_n|\mathbf{x}_n) \sum_k z_{nk} \left[\log \pi_k + \log \mathcal{N}(\mathbf{x}_n|\mu_k, \Sigma_k)\right] \\
&= \sum_n \sum_k \left(\sum_{\mathbf{z}_n} p(\mathbf{z}_n|\mathbf{x}_n) z_{nk} \right) \left[\log \pi_k + \log \mathcal{N}(\mathbf{x}_n|\mu_k, \Sigma_k)\right] \\
&= \sum_n \sum_k \gamma_{nk} \left[\log \pi_k + \log \mathcal{N}(\mathbf{x}_n|\mu_k, \Sigma_k)\right]
\end{aligned}
$$

M-Step

Substitute $\mathcal{N}(\mathbf{x}_n|\mu_k, \Sigma_k)$ in

$$L = \sum_n \sum_k \gamma_{nk} \left[\log \pi_k - \frac{D}{2} \log |2\pi\Sigma_k| - \frac{1}{2}(\mathbf{x}_n - \mu_k)^\top \Sigma^{-1}(\mathbf{x}_n - \mu_k) \right]$$

Maximize L subject to $\sum_k \pi_k = 1$

- Add a Lagrange multiplier for this constraint

$$L' = L + \lambda \left[1 - \sum_k \pi_k \right]$$

M-Step

Set derivative to 0

$$\frac{\partial L'}{\partial \pi_k} = \frac{\sum_n \gamma_{nk}}{\pi_k} - \lambda = 0$$

Therefore

$$\pi_k^* \propto \sum_n \gamma_{nk}$$

Renormalize

$$\pi_k^* = \frac{\sum_n \gamma_{nk}}{\sum_k \sum_n \gamma_{nk}} = \frac{\sum_n \gamma_{nk}}{\sum_n (\sum_k \gamma_{nk})} = \frac{\sum_n \gamma_{nk}}{N}$$

π_k^* is the fraction of data that belongs to mixture component k.

Define $N_k \triangleq \sum_n \gamma_{nk}$, weighted number of data points in mixture component k. Then $\pi_k^* = \frac{N_k}{N}$

M-Step

$$\frac{\partial L'}{\partial \mu_k} = \sum_n \gamma_{nk} \Sigma_k^{-1} (\mathbf{x}_n - \mu_k) = \Sigma_k^{-1} \sum_n \gamma_{nk} (\mathbf{x}_n - \mu_k) = 0$$

Therefore

$$\mu_k^* = \frac{\sum_n \gamma_{nk} \mathbf{x}_n}{\sum_n \gamma_{nk}} = \frac{1}{N_k} \sum_n \gamma_{nk} \mathbf{x}_n$$

μ_k^* is the weighted mean of data points assigned to mixture component k.

Similarly (and slightly more involved), you can get

$$\Sigma_k^* = \frac{1}{N_k} \sum_n \gamma_{nk} (\mathbf{x}_n - \mu_k^*)(\mathbf{x}_n - \mu_k^*)^\top$$

Summary

EM algorithm for mixture of Gaussians

E-Step Compute responsibilities

$$\gamma_{nk} = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\mu_k, \Sigma_k)}{\sum_k \pi_k \mathcal{N}(\mathbf{x}_n|\mu_k, \Sigma_k)}$$

M-Step Update parameters

$$\begin{aligned}
\pi_k^* &= \frac{N_k}{N} \\
\mu_k^* &= \frac{1}{N_k} \sum_n \gamma_{nk} \mathbf{x}_n \\
\Sigma_k^* &= \frac{1}{N_k} \sum_n \gamma_{nk} (\mathbf{x}_n - \mu_k^*)(\mathbf{x}_n - \mu_k^*)^\top
\end{aligned}$$

Special case: k-means

- Suppose we fix $\pi_k = \frac{1}{K}$ and $\Sigma_k = \sigma \mathbf{I}$ for all k
- $\{\mu_k\}$ are the only parameters.
- E-Step

$$\gamma_{nk} = \frac{\exp\left(-\frac{1}{2\sigma^2}||\mathbf{x}_n - \mu_k||^2\right)}{\sum_{k'} \exp\left(-\frac{1}{2\sigma^2}||\mathbf{x}_n - \mu_{k'}||^2\right)}$$

- Now let $\sigma \to 0$,

$$\gamma_{nk} \to \begin{cases} 1, & ||\mathbf{x}_n - \mu_k|| = \min_{k'}\{||\mathbf{x}_n - \mu_{k'}||\} \\ 0, & \text{otherwise} \end{cases}$$

- Assign $\mathbf{x}_n$ to the closest μ_k

Special case: k-means

- M-Step

$$\mu_k = \frac{1}{N_k} \sum_n \gamma_{nk} \mathbf{x}_n = \frac{1}{N_k} \sum_{n:\gamma_{nk}=1} \mathbf{x}_n$$

- Move μ_k to the mean of data points assigned to it.

- We recovered the k-means algorithm!

Example: Hidden Markov Model

- Sequence data $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T$, each $\mathbf{x}_t \in \mathbb{R}^D$
- Hidden variables $\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_T$, each $\mathbf{z}_t \in \{0, 1\}^K$, $\sum_k z_{tk} = 1$
- Joint distribution

$$p(\mathbf{X}, \mathbf{Z}) = p(\mathbf{z}_1)p(\mathbf{x}_1|\mathbf{z}_1) \prod_{t=2}^{T} p(\mathbf{z}_t|\mathbf{z}_{t-1})p(\mathbf{x}_t|\mathbf{z}_t)$$

- $p(\mathbf{x}_t|\mathbf{z}_t)$ is the emission probability, e.g. a Gaussian

$$p(\mathbf{x}_t|z_{tk} = 1) = \mathcal{N}(\mathbf{x}_t|\mu_k, \Sigma_k)$$

- $p(\mathbf{z}_t|\mathbf{z}_{t-1})$ is the transition probability, a $K \times K$ matrix $A_{ij} = p(z_{tj} = 1|z_{t-1,i} = 1)$, $\sum_j A_{ij} = 1$.
- $p(\mathbf{z}_1)$ is the prior for the first hidden state, e.g. a multinomial

$$p(\mathbf{z}_1) = \prod_k \pi_k^{z_{1k}}, \qquad \sum_k \pi_k = 1$$

M-Step

We take a look at M-Step first. The expected complete-data log-likelihood

$$
\begin{aligned}
L &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}) \log p(\mathbf{X}, \mathbf{Z}) \\
&= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}) \left[\log p(\mathbf{z}_1) + \sum_{t=1}^{T} \log p(\mathbf{x}_t|\mathbf{z}_t) + \sum_{t=2}^{T} \log p(\mathbf{z}_t|\mathbf{z}_{t-1}) \right] \\
&= \sum_{\mathbf{z}_1} p(\mathbf{z}_1|\mathbf{X}) \log p(\mathbf{z}_1) + \sum_{t=1}^{T} \sum_{\mathbf{z}_t} p(\mathbf{z}_t|\mathbf{X}) \log p(\mathbf{x}_t|\mathbf{z}_t) \\
&\quad + \sum_{t=2}^{T} \sum_{\mathbf{z}_{t-1}, \mathbf{z}_t} p(\mathbf{z}_{t-1}, \mathbf{z}_t|\mathbf{X}) \log p(\mathbf{z}_t|\mathbf{z}_{t-1})
\end{aligned}
$$

M-Step

- We only need to have access to unary and pairwise marginal posteriors $p(\mathbf{z}_t|\mathbf{X})$ and $p(\mathbf{z}_{t-1}, \mathbf{z}_t|\mathbf{X})$.
- These quantities will be estimated in E-Step.
- Now assume we already have $\gamma_{tk} \triangleq p(z_{tk} = 1|\mathbf{X})$, and $\delta_{ij} \triangleq p(z_{t-1,i} = 1, z_{t,j} = 1|\mathbf{X})$.
- The objective is then

$$L = \sum_k \gamma_{1k} \log \pi_k + \sum_{t=1}^{T} \sum_k \gamma_{tk} \log \mathcal{N}(\mathbf{x}_t|\mu_k, \Sigma_k)$$
$$+ \sum_{t=2}^{T} \sum_{k,k'} \delta_{k,k'} \log A_{k,k'}$$

- Finding the optimal $\pi_k, \mu_k, \Sigma_k, A_{ij}$ is easy - just like in mixture of Gaussians

E-Step

- We need to compute $p(\mathbf{z}_{t-1}, \mathbf{z}_t|\mathbf{X})$, then $p(\mathbf{z}_t|\mathbf{X}) = \sum_{\mathbf{z}_{t-1}} p(\mathbf{z}_{t-1}, \mathbf{z}_t|\mathbf{X})$

$$p(\mathbf{z}_{t-1}, \mathbf{z}_t|\mathbf{X}) = \sum_{\mathbf{Z}\setminus\{\mathbf{z}_{t-1},\mathbf{z}_t\}} \frac{p(\mathbf{X},\mathbf{Z})}{p(\mathbf{X})} \propto \sum_{\mathbf{Z}\setminus\{\mathbf{z}_{t-1},\mathbf{z}_t\}} p(\mathbf{X},\mathbf{Z})$$
$$= \sum_{\mathbf{Z}\setminus\{\mathbf{z}_{t-1},\mathbf{z}_t\}} p(\mathbf{z}_1)p(\mathbf{x}_1|\mathbf{x}_1)\prod_{t'} p(\mathbf{z}_{t'}|\mathbf{z}_{t'-1})p(\mathbf{x}_{t'}|\mathbf{z}_{t'})$$

- This is a sum over exponentially many terms, but we can use the model structure to distribute the sum.

E-Step

$$
\begin{aligned}
& p(\mathbf{z}_{t-1}, \mathbf{z}_t | \mathbf{X}) \\
\propto & \sum_{\mathbf{z}_1} p(\mathbf{z}_1) p(\mathbf{x}_1 | \mathbf{z}_1) \cdots \sum_{\mathbf{z}_{t-2}} p(\mathbf{z}_{t-2} | \mathbf{z}_{t-3}) p(\mathbf{x}_{t-2} | \mathbf{z}_{t-2}) \\
& p(\mathbf{z}_{t-1} | \mathbf{z}_{t-2}) p(\mathbf{x}_{t-1} | \mathbf{z}_{t-1}) p(\mathbf{z}_t | \mathbf{z}_{t-1}) p(\mathbf{x}_t | \mathbf{z}_t) \\
& \sum_{\mathbf{z}_{t+1}} p(\mathbf{z}_{t+1} | \mathbf{z}_t) p(\mathbf{x}_{t+1} | \mathbf{z}_{t+1}) \cdots \sum_{\mathbf{z}_T} p(\mathbf{z}_T | \mathbf{z}_{T-1}) p(\mathbf{x}_T | \mathbf{z}_T)
\end{aligned}
$$

- Compute the sum from both ends and propagate to $\mathbf{z}_{t-1}, \mathbf{z}_t$
- Total time linear in T
- Partial sum can be reused for all $\mathbf{z}_{t-1}, \mathbf{z}_t$ pairs
- An instance of the belief propagation/message passing/forward-backward/... algorithm for exact inference