

# Expectation Maximization (EM) Algorithm and Generative Models for Dim. Red.

Piyush Rai

Machine Learning (CS771A)

Sept 28, 2016

# Recap: GMM

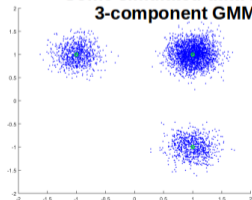
- The generative story for each  $\mathbf{x}_n$ ,  $n = 1, 2, \dots, N$ 
  - First choose one of the  $K$  mixture components as

$$\mathbf{z}_n \sim \text{Multinomial}(\mathbf{z}_n | \boldsymbol{\pi}) \quad (\text{from the prior } p(\mathbf{z}) \text{ over } \mathbf{z})$$

- Suppose  $\mathbf{z}_n = k$ . Now generate  $\mathbf{x}_n$  from the  $k$ -th Gaussian as

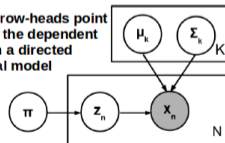
$$\mathbf{x}_n \sim \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \quad (\text{from the data distr. } p(\mathbf{x} | \mathbf{z}))$$

Some simulated data from a  
3-component GMM



Directed Graphical Model  
for a K-component GMM

Note: Arrow-heads point  
towards the dependent  
nodes in a directed  
graphical model



Shaded nodes: Observed

White nodes: Unknowns

# Recap: Learning GMM

- Initialize the parameters  $\Theta = \{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$  randomly, or using  $K$ -means

# Recap: Learning GMM

- Initialize the parameters  $\Theta = \{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$  randomly, or using  $K$ -means
- Iterate until convergence (e.g., when  $\log p(\mathbf{x}|\Theta)$  ceases to increase)

# Recap: Learning GMM

- Initialize the parameters  $\Theta = \{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$  randomly, or using  $K$ -means
- Iterate until convergence (e.g., when  $\log p(\mathbf{x}|\Theta)$  ceases to increase)
  - Given  $\Theta$ , compute each expectation  $z_{nk}$  (post. prob. of  $z_{nk} = 1$ ),  $\forall n, k$

$$\gamma_{nk} = \mathbb{E}[z_{nk}] \propto \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \quad (\text{and re-normalize s.t. } \sum_{k=1}^K \gamma_{nk} = 1)$$

# Recap: Learning GMM

- Initialize the parameters  $\Theta = \{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$  randomly, or using  $K$ -means
- Iterate until convergence (e.g., when  $\log p(\mathbf{x}|\Theta)$  ceases to increase)
  - Given  $\Theta$ , compute each expectation  $z_{nk}$  (post. prob. of  $z_{nk} = 1$ ),  $\forall n, k$

$$\gamma_{nk} = \mathbb{E}[z_{nk}] \propto \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \quad (\text{and re-normalize s.t. } \sum_{k=1}^K \gamma_{nk} = 1)$$

- Given  $\gamma_{nk} = \mathbb{E}[z_{nk}]$ , and  $N_k = \sum_{n=1}^N \gamma_{nk}$ , update  $\Theta$  as

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{n=1}^N \gamma_{nk} \mathbf{x}_n$$

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^N \gamma_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^\top$$

$$\pi_k = \frac{N_k}{N}$$

# Recap: Learning GMM

- Initialize the parameters  $\Theta = \{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$  randomly, or using  $K$ -means
- Iterate until convergence (e.g., when  $\log p(\mathbf{x}|\Theta)$  ceases to increase)
  - Given  $\Theta$ , compute each expectation  $z_{nk}$  (post. prob. of  $z_{nk} = 1$ ),  $\forall n, k$

$$\gamma_{nk} = \mathbb{E}[z_{nk}] \propto \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \quad (\text{and re-normalize s.t. } \sum_{k=1}^K \gamma_{nk} = 1)$$

- Given  $\gamma_{nk} = \mathbb{E}[z_{nk}]$ , and  $N_k = \sum_{n=1}^N \gamma_{nk}$ , update  $\Theta$  as

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{n=1}^N \gamma_{nk} \mathbf{x}_n$$

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^N \gamma_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^\top$$

$$\pi_k = \frac{N_k}{N}$$

(This algorithm is an instance of the more general Expectation Maximization (EM) algorithm which we will look at today )

# Expectation Maximization (EM)



# Parameter Estimation with Latent Variables

- Consider a generative model with joint distr.  $p(\mathbf{X}, \mathbf{Z}|\Theta) = \prod_{n=1}^N p(\mathbf{x}_n, \mathbf{z}_n)$ 
  - Observed data:  $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$
  - Latent variables:  $\mathbf{Z} = \{\mathbf{z}_n\}_{n=1}^N$ . All the model parameters:  $\Theta$

# Parameter Estimation with Latent Variables

- Consider a generative model with joint distr.  $p(\mathbf{X}, \mathbf{Z}|\Theta) = \prod_{n=1}^N p(\mathbf{x}_n, \mathbf{z}_n)$ 
  - Observed data:  $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$
  - Latent variables:  $\mathbf{Z} = \{\mathbf{z}_n\}_{n=1}^N$ . All the model parameters:  $\Theta$

# Parameter Estimation with Latent Variables

- Consider a generative model with joint distr.  $p(\mathbf{X}, \mathbf{Z}|\Theta) = \prod_{n=1}^N p(\mathbf{x}_n, \mathbf{z}_n)$ 
  - Observed data:  $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$
  - Latent variables:  $\mathbf{Z} = \{\mathbf{z}_n\}_{n=1}^N$ . All the model parameters:  $\Theta$
- Goal: Estimate the model parameters  $\Theta$  via MLE (or MAP)

# Parameter Estimation with Latent Variables

- Consider a generative model with joint distr.  $p(\mathbf{X}, \mathbf{Z}|\Theta) = \prod_{n=1}^N p(\mathbf{x}_n, \mathbf{z}_n)$ 
  - Observed data:  $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$
  - Latent variables:  $\mathbf{Z} = \{\mathbf{z}_n\}_{n=1}^N$ . All the model parameters:  $\Theta$
- Goal: Estimate the model parameters  $\Theta$  via MLE (or MAP)

$$\hat{\Theta} = \arg \max_{\Theta} \log p(\mathbf{X}|\Theta) = \arg \max_{\Theta} \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) \quad (\text{when } \mathbf{Z} \text{ is discrete})$$

# Parameter Estimation with Latent Variables

- Consider a generative model with joint distr.  $p(\mathbf{X}, \mathbf{Z}|\Theta) = \prod_{n=1}^N p(\mathbf{x}_n, \mathbf{z}_n)$ 
  - Observed data:  $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$
  - Latent variables:  $\mathbf{Z} = \{\mathbf{z}_n\}_{n=1}^N$ . All the model parameters:  $\Theta$
- Goal: Estimate the model parameters  $\Theta$  via MLE (or MAP)

$$\begin{aligned}\hat{\Theta} = \arg \max_{\Theta} \log p(\mathbf{X}|\Theta) &= \arg \max_{\Theta} \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) \quad (\text{when } \mathbf{Z} \text{ is discrete}) \\ &= \arg \max_{\Theta} \log \int_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) d\mathbf{Z} \quad (\text{when } \mathbf{Z} \text{ is continuous})\end{aligned}$$

# Parameter Estimation with Latent Variables

- Consider a generative model with joint distr.  $p(\mathbf{X}, \mathbf{Z}|\Theta) = \prod_{n=1}^N p(\mathbf{x}_n, \mathbf{z}_n)$

- Observed data:  $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$

- Latent variables:  $\mathbf{Z} = \{\mathbf{z}_n\}_{n=1}^N$ . All the model parameters:  $\Theta$

- Goal: Estimate the model parameters  $\Theta$  via MLE (or MAP)

$$\begin{aligned}\hat{\Theta} = \arg \max_{\Theta} \log p(\mathbf{X}|\Theta) &= \arg \max_{\Theta} \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) \quad (\text{when } \mathbf{Z} \text{ is discrete}) \\ &= \arg \max_{\Theta} \log \int_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) d\mathbf{Z} \quad (\text{when } \mathbf{Z} \text{ is continuous})\end{aligned}$$

- Doing MLE in such models can be difficult because of the **log-sum/integral**

# Parameter Estimation with Latent Variables

- Consider a generative model with joint distr.  $p(\mathbf{X}, \mathbf{Z}|\Theta) = \prod_{n=1}^N p(\mathbf{x}_n, \mathbf{z}_n)$

- Observed data:  $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$

- Latent variables:  $\mathbf{Z} = \{\mathbf{z}_n\}_{n=1}^N$ . All the model parameters:  $\Theta$

- Goal: Estimate the model parameters  $\Theta$  via MLE (or MAP)

$$\begin{aligned}\hat{\Theta} = \arg \max_{\Theta} \log p(\mathbf{X}|\Theta) &= \arg \max_{\Theta} \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) \quad (\text{when } \mathbf{Z} \text{ is discrete}) \\ &= \arg \max_{\Theta} \log \int_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) d\mathbf{Z} \quad (\text{when } \mathbf{Z} \text{ is continuous})\end{aligned}$$

- Doing MLE in such models can be difficult because of the **log-sum/integral**
  - In general, can't do usual MLE/MAP to get closed form solution for  $\Theta$

# Parameter Estimation with Latent Variables

- Consider a generative model with joint distr.  $p(\mathbf{X}, \mathbf{Z}|\Theta) = \prod_{n=1}^N p(\mathbf{x}_n, \mathbf{z}_n)$

- Observed data:  $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$

- Latent variables:  $\mathbf{Z} = \{\mathbf{z}_n\}_{n=1}^N$ . All the model parameters:  $\Theta$

- Goal: Estimate the model parameters  $\Theta$  via MLE (or MAP)

$$\begin{aligned}\hat{\Theta} = \arg \max_{\Theta} \log p(\mathbf{X}|\Theta) &= \arg \max_{\Theta} \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) \quad (\text{when } \mathbf{Z} \text{ is discrete}) \\ &= \arg \max_{\Theta} \log \int_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) d\mathbf{Z} \quad (\text{when } \mathbf{Z} \text{ is continuous})\end{aligned}$$

- Doing MLE in such models can be difficult because of the **log-sum/integral**
  - In general, can't do usual MLE/MAP to get closed form solution for  $\Theta$
  - A reason: Even if  $p(\mathbf{X}, \mathbf{Z}|\Theta)$  is in **exponential family**,  $p(\mathbf{X}|\Theta)$  in general isn't



# Parameter Estimation with Latent Variables

- Consider a generative model with joint distr.  $p(\mathbf{X}, \mathbf{Z}|\Theta) = \prod_{n=1}^N p(\mathbf{x}_n, \mathbf{z}_n)$

- Observed data:  $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$

- Latent variables:  $\mathbf{Z} = \{\mathbf{z}_n\}_{n=1}^N$ . All the model parameters:  $\Theta$

- Goal: Estimate the model parameters  $\Theta$  via MLE (or MAP)

$$\begin{aligned}\hat{\Theta} = \arg \max_{\Theta} \log p(\mathbf{X}|\Theta) &= \arg \max_{\Theta} \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) \quad (\text{when } \mathbf{Z} \text{ is discrete}) \\ &= \arg \max_{\Theta} \log \int_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) d\mathbf{Z} \quad (\text{when } \mathbf{Z} \text{ is continuous})\end{aligned}$$

- Doing MLE in such models can be difficult because of the **log-sum/integral**
  - In general, can't do usual MLE/MAP to get closed form solution for  $\Theta$
  - A reason: Even if  $p(\mathbf{X}, \mathbf{Z}|\Theta)$  is in **exponential family**,  $p(\mathbf{X}|\Theta)$  in general isn't
  - Note: Exp. famil dist. are easy to work with when doing MLE/MAP on them (note that **log exp()** would give simple expressions; easy to work with)

# Exponential Family

- An exponential family distribution is defined as

$$p(x; \theta) = h(x)e^{\eta(\theta)T(x) - A(\theta)}$$

- $\theta$  is called the parameter of the family
- $h(x)$ ,  $\eta(\theta)$ ,  $T(x)$ , and  $A(\theta)$  are known functions
- $p(\cdot)$  depends on  $x$  only through  $T(x)$
- $T(x)$  is called the **sufficient statistics**: summarizes the entire  $p(x; \theta)$
- Exponential family is the only family for which **conjugate priors** exist (often also in the exponential family)
- Many other nice properties (especially useful in Bayesian inference)

Many well-known distribution (Bernoulli, Binomial, categorical, beta, gamma, Gaussian, etc.) are exponential family distributions

[https://en.wikipedia.org/wiki/Exponential\\_family](https://en.wikipedia.org/wiki/Exponential_family)

# Parameter Estimation with Latent Variables

- Assume  $\mathbf{Z}$  is known to us (somehow)

# Parameter Estimation with Latent Variables

- Assume  $\mathbf{Z}$  is known to us (somehow)
- Now do MLE on the joint p.d.f.  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$  instead of  $\log p(\mathbf{X}|\Theta)$

# Parameter Estimation with Latent Variables

- Assume  $\mathbf{Z}$  is known to us (somehow)
- Now do MLE on the joint p.d.f.  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$  instead of  $\log p(\mathbf{X}|\Theta)$

# Parameter Estimation with Latent Variables

- Assume  $\mathbf{Z}$  is known to us (somehow)
- Now do MLE on the joint p.d.f.  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$  instead of  $\log p(\mathbf{X}|\Theta)$ 
  - .. actually MLE on the **expected**  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$ , since  $\mathbf{Z}$  is random

# Parameter Estimation with Latent Variables

- Assume  $\mathbf{Z}$  is known to us (somehow)
- Now do MLE on the joint p.d.f.  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$  instead of  $\log p(\mathbf{X}|\Theta)$ 
  - .. actually MLE on the **expected**  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$ , since  $\mathbf{Z}$  is random
  - Assume that MLE of  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$  is easy to solve (e.g., will be the case if  $p(\mathbf{Z})$  and  $p(\mathbf{X}|\mathbf{Z})$  are in **exponential family**) than solving MLE of  $\log p(\mathbf{X}|\Theta)$

# Parameter Estimation with Latent Variables

- Assume  $\mathbf{Z}$  is known to us (somehow)
- Now do MLE on the joint p.d.f.  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$  instead of  $\log p(\mathbf{X}|\Theta)$ 
  - .. actually MLE on the **expected**  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$ , since  $\mathbf{Z}$  is random
  - Assume that MLE of  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$  is easy to solve (e.g., will be the case if  $p(\mathbf{Z})$  and  $p(\mathbf{X}|\mathbf{Z})$  are in **exponential family**) than solving MLE of  $\log p(\mathbf{X}|\Theta)$
- **Two questions** to consider here:
  - How do we come up with our “guess” of  $\mathbf{Z}$ ?



# Parameter Estimation with Latent Variables

- Assume  $\mathbf{Z}$  is known to us (somehow)
- Now do MLE on the joint p.d.f.  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$  instead of  $\log p(\mathbf{X}|\Theta)$ 
  - .. actually MLE on the **expected**  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$ , since  $\mathbf{Z}$  is random
  - Assume that MLE of  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$  is easy to solve (e.g., will be the case if  $p(\mathbf{Z})$  and  $p(\mathbf{X}|\mathbf{Z})$  are in **exponential family**) than solving MLE of  $\log p(\mathbf{X}|\Theta)$
- **Two questions** to consider here:
  - How do we come up with our “guess” of  $\mathbf{Z}$ ?
  - Is MLE on  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$  equivalent to MLE on  $\log p(\mathbf{X}|\Theta)$ ?

# Parameter Estimation with Latent Variables

- Assume  $\mathbf{Z}$  is known to us (somehow)
- Now do MLE on the joint p.d.f.  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$  instead of  $\log p(\mathbf{X}|\Theta)$ 
  - .. actually MLE on the **expected**  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$ , since  $\mathbf{Z}$  is random
  - Assume that MLE of  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$  is easy to solve (e.g., will be the case if  $p(\mathbf{Z})$  and  $p(\mathbf{X}|\mathbf{Z})$  are in **exponential family**) than solving MLE of  $\log p(\mathbf{X}|\Theta)$
- **Two questions** to consider here:
  - How do we come up with our “guess” of  $\mathbf{Z}$ ?
    - Given current estimate of  $\Theta = \Theta^{old}$ , guess  $\mathbf{Z}$  using the posterior dist. of  $\mathbf{Z}$ 
$$p(\mathbf{Z}|\Theta^{old}, \mathbf{X}) \propto p(\mathbf{Z})p(\mathbf{X}|\mathbf{Z}) \quad (\text{but why this dist.? we will see shortly})$$
  - Is MLE on  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$  equivalent to MLE on  $\log p(\mathbf{X}|\Theta)$ ?

# Parameter Estimation with Latent Variables

- Assume  $\mathbf{Z}$  is known to us (somehow)
- Now do MLE on the joint p.d.f.  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$  instead of  $\log p(\mathbf{X}|\Theta)$ 
  - .. actually MLE on the **expected**  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$ , since  $\mathbf{Z}$  is random
  - Assume that MLE of  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$  is easy to solve (e.g., will be the case if  $p(\mathbf{Z})$  and  $p(\mathbf{X}|\mathbf{Z})$  are in **exponential family**) than solving MLE of  $\log p(\mathbf{X}|\Theta)$
- **Two questions** to consider here:
  - How do we come up with our “guess” of  $\mathbf{Z}$ ?
    - Given current estimate of  $\Theta = \Theta^{old}$ , guess  $\mathbf{Z}$  using the posterior dist. of  $\mathbf{Z}$ 
$$p(\mathbf{Z}|\Theta^{old}, \mathbf{X}) \propto p(\mathbf{Z})p(\mathbf{X}|\mathbf{Z}) \quad (\text{but why this dist.? we will see shortly})$$
  - Is MLE on  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$  equivalent to MLE on  $\log p(\mathbf{X}|\Theta)$ ?

# Parameter Estimation with Latent Variables

- Assume  $\mathbf{Z}$  is known to us (somehow)
- Now do MLE on the joint p.d.f.  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$  instead of  $\log p(\mathbf{X}|\Theta)$ 
  - .. actually MLE on the **expected**  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$ , since  $\mathbf{Z}$  is random
  - Assume that MLE of  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$  is easy to solve (e.g., will be the case if  $p(\mathbf{Z})$  and  $p(\mathbf{X}|\mathbf{Z})$  are in **exponential family**) than solving MLE of  $\log p(\mathbf{X}|\Theta)$

- **Two questions** to consider here:

- How do we come up with our “guess” of  $\mathbf{Z}$ ?
  - Given current estimate of  $\Theta = \Theta^{old}$ , guess  $\mathbf{Z}$  using the posterior dist. of  $\mathbf{Z}$

$$p(\mathbf{Z}|\Theta^{old}, \mathbf{X}) \propto p(\mathbf{Z})p(\mathbf{X}|\mathbf{Z}) \quad (\text{but why this dist.? we will see shortly})$$

- Is MLE on  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$  equivalent to MLE on  $\log p(\mathbf{X}|\Theta)$ ?
  - (We will see that)  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$  is a tight **lower-bound** on  $\log p(\mathbf{X}|\Theta)$

# Parameter Estimation with Latent Variables

- Assume  $\mathbf{Z}$  is known to us (somehow)
- Now do MLE on the joint p.d.f.  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$  instead of  $\log p(\mathbf{X}|\Theta)$ 
  - .. actually MLE on the **expected**  $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$ , since  $\mathbf{Z}$  is random
  - Assume that MLE of  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$  is easy to solve (e.g., will be the case if  $p(\mathbf{Z})$  and  $p(\mathbf{X}|\mathbf{Z})$  are in **exponential family**) than solving MLE of  $\log p(\mathbf{X}|\Theta)$

- **Two questions** to consider here:

- How do we come up with our “guess” of  $\mathbf{Z}$ ?
  - Given current estimate of  $\Theta = \Theta^{old}$ , guess  $\mathbf{Z}$  using the posterior dist. of  $\mathbf{Z}$

$$p(\mathbf{Z}|\Theta^{old}, \mathbf{X}) \propto p(\mathbf{Z})p(\mathbf{X}|\mathbf{Z}) \quad (\text{but why this dist.? we will see shortly})$$

- Is MLE on  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$  equivalent to MLE on  $\log p(\mathbf{X}|\Theta)$ ?
  - (We will see that)  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$  is a tight **lower-bound** on  $\log p(\mathbf{X}|\Theta)$
  - Maximizing this lower-bound iteratively will also improve  $\log p(\mathbf{X}|\Theta)$

# Justification

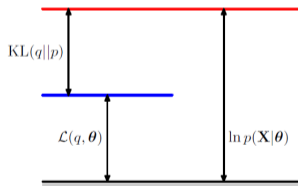
- The incomplete data log lik. can be written as a **sum of two terms**

$$\log p(\mathbf{X}|\Theta) = \mathcal{L}(q, \Theta) + \text{KL}(q||p_z)$$

where  $q$  is some distr. on  $\mathbf{Z}$ ,  $p_z = p(\mathbf{Z}|\mathbf{X}, \Theta)$  is the posterior over  $\mathbf{Z}$ , and

$$\mathcal{L}(q, \Theta) = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \left\{ \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \right\}$$

$$\text{KL}(q||p_z) = - \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \left\{ \frac{p(\mathbf{Z}|\mathbf{X}, \Theta)}{q(\mathbf{Z})} \right\}$$



(to verify, use  $\log p(\mathbf{X}, \mathbf{Z}|\Theta) = \log p(\mathbf{Z}|\mathbf{X}, \Theta) + \log p(\mathbf{X}|\Theta)$  in the expression of  $\mathcal{L}(q, \Theta)$ )

# Justification

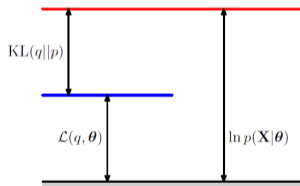
- The incomplete data log lik. can be written as a **sum of two terms**

$$\log p(\mathbf{X}|\Theta) = \mathcal{L}(q, \Theta) + \text{KL}(q||p_z)$$

where  $q$  is some distr. on  $\mathbf{Z}$ ,  $p_z = p(\mathbf{Z}|\mathbf{X}, \Theta)$  is the posterior over  $\mathbf{Z}$ , and

$$\mathcal{L}(q, \Theta) = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \left\{ \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \right\}$$

$$\text{KL}(q||p_z) = - \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \left\{ \frac{p(\mathbf{Z}|\mathbf{X}, \Theta)}{q(\mathbf{Z})} \right\}$$



(to verify, use  $\log p(\mathbf{X}, \mathbf{Z}|\Theta) = \log p(\mathbf{Z}|\mathbf{X}, \Theta) + \log p(\mathbf{X}|\Theta)$  in the expression of  $\mathcal{L}(q, \Theta)$ )

- Since  $\text{KL}(q||p_z) \geq 0$ ,  $\mathcal{L}(q, \Theta)$  is a lower-bound on  $\log p(\mathbf{X}|\Theta)$  for any  $q$

## Justification (contd.)

Recall  $\log p(\mathbf{X}|\Theta) = \mathcal{L}(q, \Theta) + \text{KL}(q||p_z)$ . Consider the following scheme:



## Justification (contd.)

Recall  $\log p(\mathbf{X}|\Theta) = \mathcal{L}(q, \Theta) + \text{KL}(q||p_z)$ . Consider the following scheme:

- With  $\Theta$  fixed to  $\Theta^{old}$ , maximize the “functional”  $\mathcal{L}(q, \Theta^{old})$  w.r.t.  $q$

$$\hat{q} = \arg \max_q \mathcal{L}(q, \Theta^{old})$$

## Justification (contd.)

Recall  $\log p(\mathbf{X}|\Theta) = \mathcal{L}(q, \Theta) + \text{KL}(q||p_z)$ . Consider the following scheme:

- With  $\Theta$  fixed to  $\Theta^{old}$ , maximize the “functional”  $\mathcal{L}(q, \Theta^{old})$  w.r.t.  $q$

$$\hat{q} = \arg \max_q \mathcal{L}(q, \Theta^{old})$$

which is equivalent to making  $\text{KL}(q||p_z) = 0$  or setting  $\hat{q} = p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$

## Justification (contd.)

Recall  $\log p(\mathbf{X}|\Theta) = \mathcal{L}(q, \Theta) + \text{KL}(q||p_z)$ . Consider the following scheme:

- With  $\Theta$  fixed to  $\Theta^{old}$ , maximize the “functional”  $\mathcal{L}(q, \Theta^{old})$  w.r.t.  $q$

$$\hat{q} = \arg \max_q \mathcal{L}(q, \Theta^{old})$$

which is equivalent to making  $\text{KL}(q||p_z) = 0$  or setting  $\hat{q} = p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$

(This step makes  $\mathcal{L}(\hat{q}, \Theta^{old}) = \log p(\mathbf{X}|\Theta^{old})$ ; see next slide)

## Justification (contd.)

Recall  $\log p(\mathbf{X}|\Theta) = \mathcal{L}(q, \Theta) + \text{KL}(q||p_z)$ . Consider the following scheme:

- With  $\Theta$  fixed to  $\Theta^{old}$ , maximize the “functional”  $\mathcal{L}(q, \Theta^{old})$  w.r.t.  $q$

$$\hat{q} = \arg \max_q \mathcal{L}(q, \Theta^{old})$$

which is equivalent to making  $\text{KL}(q||p_z) = 0$  or setting  $\hat{q} = p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$

(This step makes  $\mathcal{L}(\hat{q}, \Theta^{old}) = \log p(\mathbf{X}|\Theta^{old})$ ; see next slide)

- With  $\hat{q}$  fixed at  $p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$ , maximize  $\mathcal{L}(\hat{q}, \Theta)$  w.r.t.  $\Theta$ , where

$$\mathcal{L}(\hat{q}, \Theta) = \sum_{\mathbf{z}} p(\mathbf{z}|\mathbf{x}, \Theta^{old}) \log p(\mathbf{x}, \mathbf{z}|\Theta) - \underbrace{\sum_{\mathbf{z}} p(\mathbf{z}|\mathbf{x}, \Theta^{old}) \log p(\mathbf{z}|\mathbf{x}, \Theta^{old})}_{\text{constant w.r.t. } \Theta}$$

## Justification (contd.)

Recall  $\log p(\mathbf{X}|\Theta) = \mathcal{L}(q, \Theta) + \text{KL}(q||p_z)$ . Consider the following scheme:

- With  $\Theta$  fixed to  $\Theta^{old}$ , maximize the “functional”  $\mathcal{L}(q, \Theta^{old})$  w.r.t.  $q$

$$\hat{q} = \arg \max_q \mathcal{L}(q, \Theta^{old})$$

which is equivalent to making  $\text{KL}(q||p_z) = 0$  or setting  $\hat{q} = p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$

(This step makes  $\mathcal{L}(\hat{q}, \Theta^{old}) = \log p(\mathbf{X}|\Theta^{old})$ ; see next slide)

- With  $\hat{q}$  fixed at  $p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$ , maximize  $\mathcal{L}(\hat{q}, \Theta)$  w.r.t.  $\Theta$ , where

$$\begin{aligned} \mathcal{L}(\hat{q}, \Theta) &= \sum_{\mathbf{z}} p(\mathbf{z}|\mathbf{x}, \Theta^{old}) \log p(\mathbf{x}, \mathbf{z}|\Theta) - \underbrace{\sum_{\mathbf{z}} p(\mathbf{z}|\mathbf{x}, \Theta^{old}) \log p(\mathbf{z}|\mathbf{x}, \Theta^{old})}_{\text{constant w.r.t. } \Theta} \\ &= Q(\Theta, \Theta^{old}) + \text{const} \end{aligned}$$

## Justification (contd.)

Recall  $\log p(\mathbf{X}|\Theta) = \mathcal{L}(q, \Theta) + \text{KL}(q||p_z)$ . Consider the following scheme:

- With  $\Theta$  fixed to  $\Theta^{old}$ , maximize the “functional”  $\mathcal{L}(q, \Theta^{old})$  w.r.t.  $q$

$$\hat{q} = \arg \max_q \mathcal{L}(q, \Theta^{old})$$

which is equivalent to making  $\text{KL}(q||p_z) = 0$  or setting  $\hat{q} = p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$

(This step makes  $\mathcal{L}(\hat{q}, \Theta^{old}) = \log p(\mathbf{X}|\Theta^{old})$ ; see next slide)

- With  $\hat{q}$  fixed at  $p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$ , maximize  $\mathcal{L}(\hat{q}, \Theta)$  w.r.t.  $\Theta$ , where

$$\begin{aligned} \mathcal{L}(\hat{q}, \Theta) &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta^{old}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \underbrace{\sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta^{old}) \log p(\mathbf{Z}|\mathbf{X}, \Theta^{old})}_{\text{constant w.r.t. } \Theta} \\ &= \mathcal{Q}(\Theta, \Theta^{old}) + \text{const} \end{aligned}$$

$$\Theta^{new} = \arg \max_{\Theta} \mathcal{Q}(\Theta, \Theta^{old})$$

(where  $\mathcal{Q}(\Theta, \Theta^{old}) = \mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$ )

## Justification (contd.)

Recall  $\log p(\mathbf{X}|\Theta) = \mathcal{L}(q, \Theta) + \text{KL}(q||p_z)$ . Consider the following scheme:

- With  $\Theta$  fixed to  $\Theta^{old}$ , maximize the “functional”  $\mathcal{L}(q, \Theta^{old})$  w.r.t.  $q$

$$\hat{q} = \arg \max_q \mathcal{L}(q, \Theta^{old})$$

which is equivalent to making  $\text{KL}(q||p_z) = 0$  or **setting  $\hat{q} = p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$**

(This step makes  $\mathcal{L}(\hat{q}, \Theta^{old}) = \log p(\mathbf{X}|\Theta^{old})$ ; see next slide)

- With  $\hat{q}$  fixed at  $p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$ , maximize  $\mathcal{L}(\hat{q}, \Theta)$  w.r.t.  $\Theta$ , where

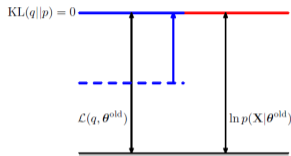
$$\begin{aligned} \mathcal{L}(\hat{q}, \Theta) &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta^{old}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \underbrace{\sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta^{old}) \log p(\mathbf{Z}|\mathbf{X}, \Theta^{old})}_{\text{constant w.r.t. } \Theta} \\ &= \mathcal{Q}(\Theta, \Theta^{old}) + \text{const} \end{aligned}$$

$$\Theta^{new} = \arg \max_{\Theta} \mathcal{Q}(\Theta, \Theta^{old}) \quad (\text{where } \mathcal{Q}(\Theta, \Theta^{old}) = \mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)])$$

(This step ensures that  $\log p(\mathbf{X}|\Theta^{new}) \geq \log p(\mathbf{X}|\Theta^{old})$ ; see next slide)

## Justification (contd.)

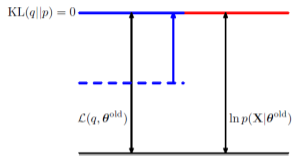
**Step 1:** Set  $q = p(\mathbf{Z}|\mathbf{X}, \Theta)$ ,  $\text{KL}(q||p_z)$  becomes 0,  $\mathcal{L}(q, \Theta^{old})$  increases and becomes equal to  $\log p(\mathbf{X}|\Theta^{old})$



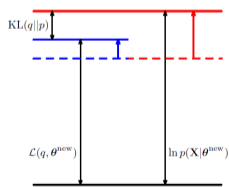


# Justification (contd.)

**Step 1:** Set  $q = p(\mathbf{Z}|\mathbf{X}, \Theta)$ ,  $\text{KL}(q||p_z)$  becomes 0,  $\mathcal{L}(q, \Theta^{old})$  increases and becomes equal to  $\log p(\mathbf{X}|\Theta^{old})$

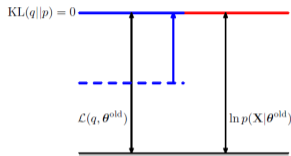


**Step 2:**  $\Theta^{new}$  makes  $\mathcal{L}(q, \Theta^{new})$  go further up, makes  $\text{KL}(q||p_z) > 0$  again because  $q \neq p(\mathbf{Z}|\mathbf{X}, \Theta^{new})$  and thus ensures that  $\log p(\mathbf{X}|\Theta^{new}) \geq \log p(\mathbf{X}|\Theta^{old})$

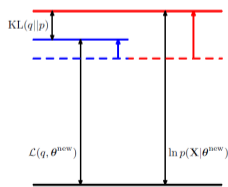


# Justification (contd.)

**Step 1:** Set  $q = p(\mathbf{Z}|\mathbf{X}, \Theta)$ ,  $\text{KL}(q||p_z)$  becomes 0,  $\mathcal{L}(q, \Theta^{old})$  increases and becomes equal to  $\log p(\mathbf{X}|\Theta^{old})$



**Step 2:**  $\Theta^{new}$  makes  $\mathcal{L}(q, \Theta^{new})$  go further up, makes  $\text{KL}(q||p_z) > 0$  again because  $q \neq p(\mathbf{Z}|\mathbf{X}, \Theta^{new})$  and thus ensures that  $\log p(\mathbf{X}|\Theta^{new}) \geq \log p(\mathbf{X}|\Theta^{old})$



These two steps never decrease  $\log p(\mathbf{X}|\Theta)$ . Thus it's a good way of doing MLE

# An Alternate Justification

- Consider the “incomplete” data log likelihood

$$\log p(\mathbf{X}|\Theta) = \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta)$$

# An Alternate Justification

- Consider the “incomplete” data log likelihood

$$\log p(\mathbf{X}|\Theta) = \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.})$$

# An Alternate Justification

- Consider the “incomplete” data log likelihood

$$\begin{aligned}\log p(\mathbf{X}|\Theta) &= \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.}) \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})}\end{aligned}$$

# An Alternate Justification

- Consider the “incomplete” data log likelihood

$$\begin{aligned}\log p(\mathbf{X}|\Theta) &= \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.}) \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{concave } f, \text{ Jensen's Ineq.: } f(\sum \lambda_i x_i) \geq \sum \lambda_i f(x_i))\end{aligned}$$

# An Alternate Justification

- Consider the “incomplete” data log likelihood

$$\begin{aligned}\log p(\mathbf{X}|\Theta) &= \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.}) \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{concave } f, \text{ Jensen's Ineq.: } f(\sum \lambda_i x_i) \geq \sum \lambda_i f(x_i)) \\ \log p(\mathbf{X}|\Theta) &\geq \underbrace{\sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \sum_{\mathbf{Z}} q(\mathbf{Z}) \log q(\mathbf{Z})}_{\text{doesn't depend on } \Theta}\end{aligned}$$

# An Alternate Justification

- Consider the “incomplete” data log likelihood

$$\begin{aligned}\log p(\mathbf{X}|\Theta) &= \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.}) \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{concave } f, \text{ Jensen's Ineq.: } f(\sum \lambda_i x_i) \geq \sum \lambda_i f(x_i)) \\ \log p(\mathbf{X}|\Theta) &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \underbrace{\sum_{\mathbf{Z}} q(\mathbf{Z}) \log q(\mathbf{Z})}_{\text{doesn't depend on } \Theta} = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) + \text{const.}\end{aligned}$$



# An Alternate Justification

- Consider the “incomplete” data log likelihood

$$\begin{aligned}\log p(\mathbf{X}|\Theta) &= \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.}) \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{concave } f, \text{ Jensen's Ineq.: } f(\sum \lambda_i x_i) \geq \sum \lambda_i f(x_i)) \\ \log p(\mathbf{X}|\Theta) &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \underbrace{\sum_{\mathbf{Z}} q(\mathbf{Z}) \log q(\mathbf{Z})}_{\text{doesn't depend on } \Theta} = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) + \text{const.}\end{aligned}$$

- If we set  $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \Theta)$ , the above inequality becomes equality

# An Alternate Justification

- Consider the “incomplete” data log likelihood

$$\begin{aligned}\log p(\mathbf{X}|\Theta) &= \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.}) \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{concave } f, \text{ Jensen's Ineq.: } f(\sum \lambda_i x_i) \geq \sum \lambda_i f(x_i)) \\ \log p(\mathbf{X}|\Theta) &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \underbrace{\sum_{\mathbf{Z}} q(\mathbf{Z}) \log q(\mathbf{Z})}_{\text{doesn't depend on } \Theta} = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) + \text{const.}\end{aligned}$$

- If we set  $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \Theta)$ , the above inequality becomes equality

$$\sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})}$$

# An Alternate Justification

- Consider the ‘incomplete’ data log likelihood

$$\begin{aligned}\log p(\mathbf{X}|\Theta) &= \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.}) \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{concave } f, \text{ Jensen's Ineq.: } f(\sum \lambda_i x_i) \geq \sum \lambda_i f(x_i)) \\ \log p(\mathbf{X}|\Theta) &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \underbrace{\sum_{\mathbf{Z}} q(\mathbf{Z}) \log q(\mathbf{Z})}_{\text{doesn't depend on } \Theta} = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) + \text{const.}\end{aligned}$$

- If we set  $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \Theta)$ , the above inequality becomes equality

$$\sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log \frac{p(\mathbf{Z}|\mathbf{X}, \Theta) p(\mathbf{X}|\Theta)}{p(\mathbf{Z}|\mathbf{X}, \Theta)}$$

# An Alternate Justification

- Consider the ‘incomplete’ data log likelihood

$$\begin{aligned}\log p(\mathbf{X}|\Theta) &= \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.}) \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{concave } f, \text{ Jensen's Ineq.: } f(\sum \lambda_i x_i) \geq \sum \lambda_i f(x_i)) \\ \log p(\mathbf{X}|\Theta) &\geq \underbrace{\sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \sum_{\mathbf{Z}} q(\mathbf{Z}) \log q(\mathbf{Z})}_{\text{doesn't depend on } \Theta} = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) + \text{const.}\end{aligned}$$

- If we set  $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \Theta)$ , the above inequality becomes equality

$$\sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log \frac{\cancel{p(\mathbf{Z}|\mathbf{X}, \Theta)} p(\mathbf{X}|\Theta)}{\cancel{p(\mathbf{Z}|\mathbf{X}, \Theta)}} = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log p(\mathbf{X}|\Theta)$$

# An Alternate Justification

- Consider the ‘incomplete’ data log likelihood

$$\begin{aligned}\log p(\mathbf{X}|\Theta) &= \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.}) \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{concave } f, \text{ Jensen's Ineq.: } f(\sum \lambda_i x_i) \geq \sum \lambda_i f(x_i)) \\ \log p(\mathbf{X}|\Theta) &\geq \underbrace{\sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \sum_{\mathbf{Z}} q(\mathbf{Z}) \log q(\mathbf{Z})}_{\text{doesn't depend on } \Theta} = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) + \text{const.}\end{aligned}$$

- If we set  $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \Theta)$ , the above inequality becomes equality

$$\begin{aligned}\sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log \frac{p(\mathbf{Z}|\mathbf{X}, \Theta) p(\mathbf{X}|\Theta)}{p(\mathbf{Z}|\mathbf{X}, \Theta)} = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log p(\mathbf{X}|\Theta) \\ &= \log p(\mathbf{X}|\Theta) \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta)\end{aligned}$$

# An Alternate Justification

- Consider the ‘incomplete’ data log likelihood

$$\begin{aligned}\log p(\mathbf{X}|\Theta) &= \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.}) \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{concave } f, \text{ Jensen's Ineq.: } f(\sum \lambda_i x_i) \geq \sum \lambda_i f(x_i)) \\ \log p(\mathbf{X}|\Theta) &\geq \underbrace{\sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \sum_{\mathbf{Z}} q(\mathbf{Z}) \log q(\mathbf{Z})}_{\text{doesn't depend on } \Theta} = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) + \text{const.}\end{aligned}$$

- If we set  $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \Theta)$ , the above inequality becomes equality

$$\begin{aligned}\sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log \frac{p(\mathbf{Z}|\mathbf{X}, \Theta) p(\mathbf{X}|\Theta)}{p(\mathbf{Z}|\mathbf{X}, \Theta)} = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log p(\mathbf{X}|\Theta) \\ &= \log p(\mathbf{X}|\Theta) \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) = \log p(\mathbf{X}|\Theta)\end{aligned}$$

# An Alternate Justification

- Consider the ‘incomplete’ data log likelihood

$$\begin{aligned}\log p(\mathbf{X}|\Theta) &= \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.}) \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{concave } f, \text{ Jensen's Ineq.: } f(\sum \lambda_i x_i) \geq \sum \lambda_i f(x_i)) \\ \log p(\mathbf{X}|\Theta) &\geq \underbrace{\sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \sum_{\mathbf{Z}} q(\mathbf{Z}) \log q(\mathbf{Z})}_{\text{doesn't depend on } \Theta} = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) + \text{const.}\end{aligned}$$

- If we set  $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \Theta)$ , the above inequality becomes equality

$$\begin{aligned}\sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log \frac{p(\mathbf{Z}|\mathbf{X}, \Theta) p(\mathbf{X}|\Theta)}{p(\mathbf{Z}|\mathbf{X}, \Theta)} = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log p(\mathbf{X}|\Theta) \\ &= \log p(\mathbf{X}|\Theta) \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) = \log p(\mathbf{X}|\Theta)\end{aligned}$$

- Thus for  $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \Theta)$ , we have

# An Alternate Justification

- Consider the ‘incomplete’ data log likelihood

$$\begin{aligned}\log p(\mathbf{X}|\Theta) &= \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.}) \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{concave } f, \text{ Jensen's Ineq.: } f(\sum \lambda_i x_i) \geq \sum \lambda_i f(x_i)) \\ \log p(\mathbf{X}|\Theta) &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \underbrace{\sum_{\mathbf{Z}} q(\mathbf{Z}) \log q(\mathbf{Z})}_{\text{doesn't depend on } \Theta} = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) + \text{const.}\end{aligned}$$

- If we set  $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \Theta)$ , the above inequality becomes equality

$$\begin{aligned}\sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log \frac{p(\mathbf{Z}|\mathbf{X}, \Theta) p(\mathbf{X}|\Theta)}{p(\mathbf{Z}|\mathbf{X}, \Theta)} = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log p(\mathbf{X}|\Theta) \\ &= \log p(\mathbf{X}|\Theta) \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) = \log p(\mathbf{X}|\Theta)\end{aligned}$$

- Thus for  $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \Theta)$ , we have

$$\log p(\mathbf{X}|\Theta) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log p(\mathbf{X}, \mathbf{Z}|\Theta) + \text{const.}$$



# An Alternate Justification

- Consider the ‘incomplete’ data log likelihood

$$\begin{aligned}\log p(\mathbf{X}|\Theta) &= \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.}) \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{concave } f, \text{ Jensen's Ineq.: } f(\sum \lambda_i x_i) \geq \sum \lambda_i f(x_i)) \\ \log p(\mathbf{X}|\Theta) &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \underbrace{\sum_{\mathbf{Z}} q(\mathbf{Z}) \log q(\mathbf{Z})}_{\text{doesn't depend on } \Theta} = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) + \text{const.}\end{aligned}$$

- If we set  $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \Theta)$ , the above inequality becomes equality

$$\begin{aligned}\sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log \frac{p(\mathbf{Z}|\mathbf{X}, \Theta) p(\mathbf{X}|\Theta)}{p(\mathbf{Z}|\mathbf{X}, \Theta)} = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log p(\mathbf{X}|\Theta) \\ &= \log p(\mathbf{X}|\Theta) \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) = \log p(\mathbf{X}|\Theta)\end{aligned}$$

- Thus for  $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \Theta)$ , we have

$$\log p(\mathbf{X}|\Theta) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log p(\mathbf{X}, \mathbf{Z}|\Theta) + \text{const.} = \mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)] + \text{const.}$$

# An Alternate Justification

- Consider the ‘incomplete’ data log likelihood

$$\begin{aligned}\log p(\mathbf{X}|\Theta) &= \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\Theta) = \log \sum_{\mathbf{Z}} q(\mathbf{Z}) \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{where } q(\mathbf{Z}) \text{ is some dist.}) \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \quad (\text{concave } f, \text{ Jensen's Ineq.: } f(\sum \lambda_i x_i) \geq \sum \lambda_i f(x_i)) \\ \log p(\mathbf{X}|\Theta) &\geq \underbrace{\sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) - \sum_{\mathbf{Z}} q(\mathbf{Z}) \log q(\mathbf{Z})}_{\text{doesn't depend on } \Theta} = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) + \text{const.}\end{aligned}$$

- If we set  $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \Theta)$ , the above inequality becomes equality

$$\begin{aligned}\sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log \frac{p(\mathbf{Z}|\mathbf{X}, \Theta) p(\mathbf{X}|\Theta)}{p(\mathbf{Z}|\mathbf{X}, \Theta)} = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log p(\mathbf{X}|\Theta) \\ &= \log p(\mathbf{X}|\Theta) \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) = \log p(\mathbf{X}|\Theta)\end{aligned}$$

- Thus for  $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \Theta)$ , we have

$$\log p(\mathbf{X}|\Theta) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta) \log p(\mathbf{X}, \mathbf{Z}|\Theta) + \text{const.} = \mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)] + \text{const.}$$

- Thus  $\log p(\mathbf{X}|\Theta)$  is tightly lower-bounded by  $\mathbb{E}[\log p(\mathbf{X}, \mathbf{z}|\Theta)]$  which EM maximizes

# The Expectation Maximization (EM) Algorithm

Initialize the parameters:  $\Theta^{old}$ . Then alternate between these steps:

# The Expectation Maximization (EM) Algorithm

Initialize the parameters:  $\Theta^{old}$ . Then alternate between these steps:

- **E (Expectation) step:**

# The Expectation Maximization (EM) Algorithm

Initialize the parameters:  $\Theta^{old}$ . Then alternate between these steps:

- **E (Expectation) step:**

- Compute the posterior  $p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$  over latent variables  $\mathbf{Z}$  using  $\Theta^{old}$

# The Expectation Maximization (EM) Algorithm

Initialize the parameters:  $\Theta^{old}$ . Then alternate between these steps:

- **E (Expectation) step:**

- Compute the posterior  $p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$  over latent variables  $\mathbf{Z}$  using  $\Theta^{old}$
- Compute the expected complete data log-likelihood w.r.t. *this* posterior

$$Q(\Theta, \Theta^{old}) = \mathbb{E}_{p(\mathbf{Z}|\mathbf{X}, \Theta^{old})}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)] = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta^{old}) \log p(\mathbf{X}, \mathbf{Z}|\Theta)$$

# The Expectation Maximization (EM) Algorithm

Initialize the parameters:  $\Theta^{old}$ . Then alternate between these steps:

- **E (Expectation) step:**

- Compute the posterior  $p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$  over latent variables  $\mathbf{Z}$  using  $\Theta^{old}$
- Compute the expected complete data log-likelihood w.r.t. *this* posterior

$$Q(\Theta, \Theta^{old}) = \mathbb{E}_{p(\mathbf{Z}|\mathbf{X}, \Theta^{old})}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)] = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta^{old}) \log p(\mathbf{X}, \mathbf{Z}|\Theta)$$

- **M (Maximization) step:**

# The Expectation Maximization (EM) Algorithm

Initialize the parameters:  $\Theta^{old}$ . Then alternate between these steps:

- **E (Expectation) step:**

- Compute the posterior  $p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$  over latent variables  $\mathbf{Z}$  using  $\Theta^{old}$
- Compute the expected complete data log-likelihood w.r.t. *this* posterior

$$Q(\Theta, \Theta^{old}) = \mathbb{E}_{p(\mathbf{Z}|\mathbf{X}, \Theta^{old})}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)] = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta^{old}) \log p(\mathbf{X}, \mathbf{Z}|\Theta)$$

- **M (Maximization) step:**

- Maximize the expected complete data log-likelihood w.r.t.  $\Theta$

$$\Theta^{new} = \arg \max_{\Theta} Q(\Theta, \Theta^{old}) \quad (\text{if doing MLE})$$

$$\Theta^{new} = \arg \max_{\Theta} \{Q(\Theta, \Theta^{old}) + \log p(\Theta)\} \quad (\text{if doing MAP})$$



# The Expectation Maximization (EM) Algorithm

Initialize the parameters:  $\Theta^{old}$ . Then alternate between these steps:

- **E (Expectation) step:**

- Compute the posterior  $p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$  over latent variables  $\mathbf{Z}$  using  $\Theta^{old}$
- Compute the expected complete data log-likelihood w.r.t. *this* posterior

$$Q(\Theta, \Theta^{old}) = \mathbb{E}_{p(\mathbf{Z}|\mathbf{X}, \Theta^{old})}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)] = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta^{old}) \log p(\mathbf{X}, \mathbf{Z}|\Theta)$$

- **M (Maximization) step:**

- Maximize the expected complete data log-likelihood w.r.t.  $\Theta$

$$\Theta^{new} = \arg \max_{\Theta} Q(\Theta, \Theta^{old}) \quad (\text{if doing MLE})$$

$$\Theta^{new} = \arg \max_{\Theta} \{Q(\Theta, \Theta^{old}) + \log p(\Theta)\} \quad (\text{if doing MAP})$$

- If the log-likelihood or the parameter values not converged then set  $\Theta^{old} = \Theta^{new}$  and go to the E step.

# The Expectation Maximization (EM) Algorithm

Initialize the parameters:  $\Theta^{old}$ . Then alternate between these steps:

- **E (Expectation) step:**

- Compute the posterior  $p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$  over latent variables  $\mathbf{Z}$  using  $\Theta^{old}$
- Compute the expected complete data log-likelihood w.r.t. *this* posterior

$$Q(\Theta, \Theta^{old}) = \mathbb{E}_{p(\mathbf{Z}|\mathbf{X}, \Theta^{old})}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)] = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta^{old}) \log p(\mathbf{X}, \mathbf{Z}|\Theta)$$

- **M (Maximization) step:**

- Maximize the expected complete data log-likelihood w.r.t.  $\Theta$

$$\Theta^{new} = \arg \max_{\Theta} Q(\Theta, \Theta^{old}) \quad (\text{if doing MLE})$$

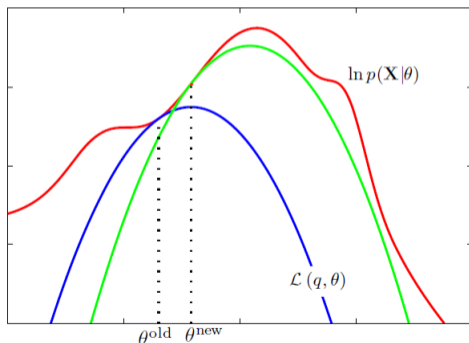
$$\Theta^{new} = \arg \max_{\Theta} \{Q(\Theta, \Theta^{old}) + \log p(\Theta)\} \quad (\text{if doing MAP})$$

- If the log-likelihood or the parameter values not converged then set  $\Theta^{old} = \Theta^{new}$  and go to the E step.

The algorithm converges to a local maxima of  $p(\mathbf{X}|\Theta)$  (as we saw)

# EM: A View in the Parameter Space

- E-step: Update of  $q$  makes the  $\mathcal{L}(q, \Theta)$  curve touch the  $\log p(\mathbf{X}|\Theta)$  curve
- M-step gives the maxima  $\Theta^{new}$  of  $\mathcal{L}(q, \Theta)$
- Next E-step readjusts  $\mathcal{L}(q, \Theta)$  curve (green) to meet  $\log p(\mathbf{X}|\Theta)$  curve again
- This continues until a local maxima of  $\log p(\mathbf{X}|\Theta)$  is reached



# EM: Some Comments

- A general framework for parameter estimation in latent variable models

# EM: Some Comments

- A general framework for parameter estimation in latent variable models
- Very widely used in problems with “missing data”, e.g., missing features, or missing labels (semi-supervised learning)

# EM: Some Comments

- A general framework for parameter estimation in latent variable models
- Very widely used in problems with “missing data”, e.g., missing features, or missing labels (semi-supervised learning)
  - “Missing” parts can be treated as latent variables  $z$  and estimated using EM
- More advanced probabilistic inference algorithms are based on similar ideas

# EM: Some Comments

- A general framework for parameter estimation in latent variable models
- Very widely used in problems with “missing data”, e.g., missing features, or missing labels (semi-supervised learning)
  - “Missing” parts can be treated as latent variables  $z$  and estimated using EM
- More advanced probabilistic inference algorithms are based on similar ideas
  - E.g., variational Bayesian inference

# EM: Some Comments

- A general framework for parameter estimation in latent variable models
- Very widely used in problems with “missing data”, e.g., missing features, or missing labels (semi-supervised learning)
  - “Missing” parts can be treated as latent variables  $z$  and estimated using EM
- More advanced probabilistic inference algorithms are based on similar ideas
  - E.g., variational Bayesian inference
- Very easy to extend to online learning setting and gives SGD like algorithms (will post a reading on “Online EM” on the class webpage)



# EM: Some Comments

- A general framework for parameter estimation in latent variable models
- Very widely used in problems with “missing data”, e.g., missing features, or missing labels (semi-supervised learning)
  - “Missing” parts can be treated as latent variables  $z$  and estimated using EM
- More advanced probabilistic inference algorithms are based on similar ideas
  - E.g., variational Bayesian inference
- Very easy to extend to online learning setting and gives SGD like algorithms (will post a reading on “Online EM” on the class webpage)
- Note: The E and M steps may not always be possible to perform exactly (approximate inference methods may be needed in such cases)

# Generative Models for Dimensionality Reduction

# Generative Model for Dimensionality Reduction

- Assume the following generative model for each  $\mathbf{x}_n \in \mathbb{R}^D$

# Generative Model for Dimensionality Reduction

- Assume the following generative model for each  $\mathbf{x}_n \in \mathbb{R}^D$ 
  - First draw a latent variable (**latent factors** or **latent features**)  $\mathbf{z}_n \in \mathbb{R}^K$  as
$$\mathbf{z}_n \sim \mathcal{N}(\mathbf{z} | 0, \mathbf{I}_K)$$

# Generative Model for Dimensionality Reduction

- Assume the following generative model for each  $\mathbf{x}_n \in \mathbb{R}^D$ 
  - First draw a latent variable (**latent factors** or **latent features**)  $\mathbf{z}_n \in \mathbb{R}^K$  as
$$\mathbf{z}_n \sim \mathcal{N}(\mathbf{z} | 0, \mathbf{I}_K)$$
  - Now draw  $\mathbf{x}_n$  by transforming  $\mathbf{z}_n$  as  $\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$

# Generative Model for Dimensionality Reduction

- Assume the following generative model for each  $\mathbf{x}_n \in \mathbb{R}^D$ 
  - First draw a latent variable (**latent factors** or **latent features**)  $\mathbf{z}_n \in \mathbb{R}^K$  as
$$\mathbf{z}_n \sim \mathcal{N}(\mathbf{z} | 0, \mathbf{I}_K)$$
  - Now draw  $\mathbf{x}_n$  by transforming  $\mathbf{z}_n$  as  $\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$ , where  $\mathbf{W}$  is a  $D \times K$  matrix,  $K \ll D$

# Generative Model for Dimensionality Reduction

- Assume the following generative model for each  $\mathbf{x}_n \in \mathbb{R}^D$ 
  - First draw a latent variable (**latent factors** or **latent features**)  $\mathbf{z}_n \in \mathbb{R}^K$  as
$$\mathbf{z}_n \sim \mathcal{N}(\mathbf{z}|0, \mathbf{I}_K)$$
  - Now draw  $\mathbf{x}_n$  by transforming  $\mathbf{z}_n$  as  $\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$ , where  $\mathbf{W}$  is a  $D \times K$  matrix,  $K \ll D$  and Gaussian noise  $\epsilon_n \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_D)$

# Generative Model for Dimensionality Reduction

- Assume the following generative model for each  $\mathbf{x}_n \in \mathbb{R}^D$ 
  - First draw a latent variable (**latent factors** or **latent features**)  $\mathbf{z}_n \in \mathbb{R}^K$  as
- Now draw  $\mathbf{x}_n$  by transforming  $\mathbf{z}_n$  as  $\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$ , where  $\mathbf{W}$  is a  $D \times K$  matrix,  $K \ll D$  and Gaussian noise  $\epsilon_n \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_D)$ . Equivalent to saying

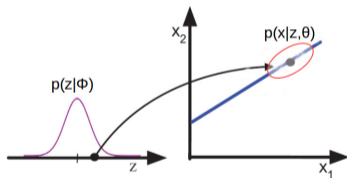
$$\mathbf{x}_n \sim \mathcal{N}(\mathbf{x} | \mathbf{W}\mathbf{z}_n, \sigma^2 \mathbf{I}_D)$$



# Generative Model for Dimensionality Reduction

- Assume the following generative model for each  $\mathbf{x}_n \in \mathbb{R}^D$ 
  - First draw a latent variable (latent factors or latent features)  $\mathbf{z}_n \in \mathbb{R}^K$  as
$$\mathbf{z}_n \sim \mathcal{N}(\mathbf{z}|0, \mathbf{I}_K)$$
  - Now draw  $\mathbf{x}_n$  by transforming  $\mathbf{z}_n$  as  $\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$ , where  $\mathbf{W}$  is a  $D \times K$  matrix,  $K \ll D$  and Gaussian noise  $\epsilon_n \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_D)$ . Equivalent to saying

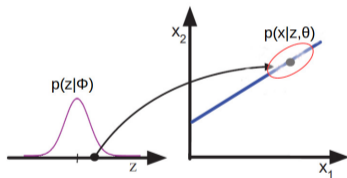
$$\mathbf{x}_n \sim \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z}_n, \sigma^2 \mathbf{I}_D)$$



# Generative Model for Dimensionality Reduction

- Assume the following generative model for each  $\mathbf{x}_n \in \mathbb{R}^D$ 
  - First draw a latent variable (**latent factors** or **latent features**)  $\mathbf{z}_n \in \mathbb{R}^K$  as
$$\mathbf{z}_n \sim \mathcal{N}(\mathbf{z}|0, \mathbf{I}_K)$$
  - Now draw  $\mathbf{x}_n$  by transforming  $\mathbf{z}_n$  as  $\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$ , where  $\mathbf{W}$  is a  $D \times K$  matrix,  $K \ll D$  and Gaussian noise  $\epsilon_n \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_D)$ . Equivalent to saying

$$\mathbf{x}_n \sim \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z}_n, \sigma^2 \mathbf{I}_D)$$

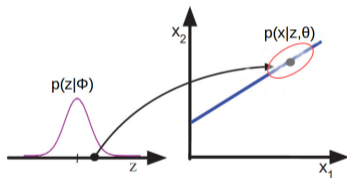


- This defines a **probabilistic PCA** (PPCA) generative model

# Generative Model for Dimensionality Reduction

- Assume the following generative model for each  $\mathbf{x}_n \in \mathbb{R}^D$ 
  - First draw a latent variable (**latent factors** or **latent features**)  $\mathbf{z}_n \in \mathbb{R}^K$  as
$$\mathbf{z}_n \sim \mathcal{N}(\mathbf{z}|0, \mathbf{I}_K)$$
  - Now draw  $\mathbf{x}_n$  by transforming  $\mathbf{z}_n$  as  $\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$ , where  $\mathbf{W}$  is a  $D \times K$  matrix,  $K \ll D$  and Gaussian noise  $\epsilon_n \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_D)$ . Equivalent to saying

$$\mathbf{x}_n \sim \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z}_n, \sigma^2 \mathbf{I}_D)$$

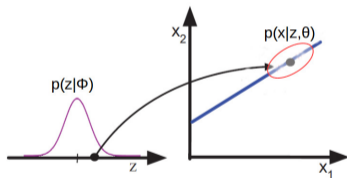


- This defines a **probabilistic PCA** (PPCA) generative model
- When Gaussian noise has diag. instead of spherical covar: **Factor Analysis**

# Generative Model for Dimensionality Reduction

- Assume the following generative model for each  $\mathbf{x}_n \in \mathbb{R}^D$ 
  - First draw a latent variable (latent factors or latent features)  $\mathbf{z}_n \in \mathbb{R}^K$  as
$$\mathbf{z}_n \sim \mathcal{N}(\mathbf{z}|0, \mathbf{I}_K)$$
  - Now draw  $\mathbf{x}_n$  by transforming  $\mathbf{z}_n$  as  $\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$ , where  $\mathbf{W}$  is a  $D \times K$  matrix,  $K \ll D$  and Gaussian noise  $\epsilon_n \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_D)$ . Equivalent to saying

$$\mathbf{x}_n \sim \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z}_n, \sigma^2 \mathbf{I}_D)$$



- This defines a probabilistic PCA (PPCA) generative model
- When Gaussian noise has diag. instead of spherical covar: Factor Analysis
- Given observations  $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$ , we want to learn params  $\Theta = \{\mathbf{W}, \sigma^2\}$  and latent variables  $\mathbf{Z} = \{\mathbf{z}_n\}_{n=1}^N$ . EM gives a nice and efficient way of doing this.

# Generative Model for Dimensionality Reduction

- The model for each observation  $\mathbf{x}_n$

$$\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$$

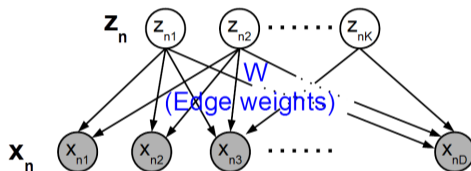
- Note: We'll assume data to be centered, otherwise  $\mathbf{x}_n = \boldsymbol{\mu} + \mathbf{W}\mathbf{z}_n + \epsilon_n$

# Generative Model for Dimensionality Reduction

- The model for each observation  $\mathbf{x}_n$

$$\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$$

- Note: We'll assume data to be centered, otherwise  $\mathbf{x}_n = \boldsymbol{\mu} + \mathbf{W}\mathbf{z}_n + \epsilon_n$
- Zooming in at the relationship between each  $\mathbf{x}_n$  and each  $\mathbf{z}_n$

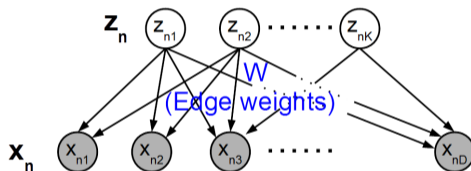


# Generative Model for Dimensionality Reduction

- The model for each observation  $\mathbf{x}_n$

$$\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$$

- Note: We'll assume data to be centered, otherwise  $\mathbf{x}_n = \boldsymbol{\mu} + \mathbf{W}\mathbf{z}_n + \epsilon_n$
- Zooming in at the relationship between each  $\mathbf{x}_n$  and each  $\mathbf{z}_n$



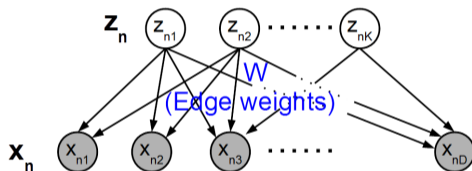
- A **directed graphical model** linking  $\mathbf{z}_n$  and  $\mathbf{x}_n$  via “edge weights”  $\mathbf{W}$

# Generative Model for Dimensionality Reduction

- The model for each observation  $\mathbf{x}_n$

$$\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$$

- Note: We'll assume data to be centered, otherwise  $\mathbf{x}_n = \boldsymbol{\mu} + \mathbf{W}\mathbf{z}_n + \epsilon_n$
- Zooming in at the relationship between each  $\mathbf{x}_n$  and each  $\mathbf{z}_n$



- A **directed graphical model** linking  $\mathbf{z}_n$  and  $\mathbf{x}_n$  via “edge weights”  $\mathbf{W}$
- The  $D \times K$  matrix  $\mathbf{W}$  is also called the **factor loading matrix**

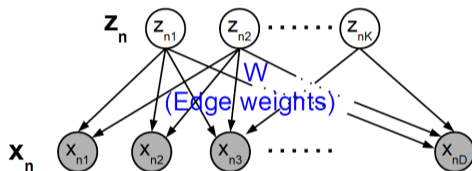


# Generative Model for Dimensionality Reduction

- The model for each observation  $\mathbf{x}_n$

$$\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$$

- Note: We'll assume data to be centered, otherwise  $\mathbf{x}_n = \boldsymbol{\mu} + \mathbf{W}\mathbf{z}_n + \epsilon_n$
- Zooming in at the relationship between each  $\mathbf{x}_n$  and each  $\mathbf{z}_n$



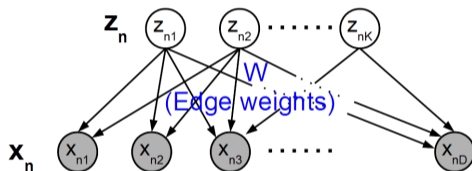
- A **directed graphical model** linking  $\mathbf{z}_n$  and  $\mathbf{x}_n$  via “edge weights”  $\mathbf{W}$
- The  $D \times K$  matrix  $\mathbf{W}$  is also called the **factor loading matrix**
  - Can think of each column of  $\mathbf{W}$  as a basis (but not mutually orthogonal)

# Generative Model for Dimensionality Reduction

- The model for each observation  $\mathbf{x}_n$

$$\mathbf{x}_n = \mathbf{W}\mathbf{z}_n + \epsilon_n$$

- Note: We'll assume data to be centered, otherwise  $\mathbf{x}_n = \boldsymbol{\mu} + \mathbf{W}\mathbf{z}_n + \epsilon_n$
- Zooming in at the relationship between each  $\mathbf{x}_n$  and each  $\mathbf{z}_n$



- A **directed graphical model** linking  $\mathbf{z}_n$  and  $\mathbf{x}_n$  via “edge weights”  $\mathbf{W}$
- The  $D \times K$  matrix  $\mathbf{W}$  is also called the **factor loading matrix**
  - Can think of each column of  $\mathbf{W}$  as a basis (but not mutually orthogonal)
  - $\mathbf{W}$  can be used to interpret the relationship of b/w the  $K$  latent features and  $D$  observed features of each observation  $\mathbf{x}_n$

# Some Nice Aspects about PPCA/FA

- Can also be seen as modeling data using a **low-rank Gaussian**

$$p(\mathbf{x}_n) = \mathcal{N}(\mathbf{x}_n | 0, \mathbf{W}\mathbf{W}^\top + \sigma^2 \mathbf{I}_D)$$

# Some Nice Aspects about PPCA/FA

- Can also be seen as modeling data using a **low-rank Gaussian**

$$p(\mathbf{x}_n) = \mathcal{N}(\mathbf{x}_n | 0, \mathbf{W}\mathbf{W}^\top + \sigma^2 \mathbf{I}_D)$$

- PPCA reduces to PCA as the noise variance  $\sigma^2$  tends to zero

# Some Nice Aspects about PPCA/FA

- Can also be seen as modeling data using a **low-rank Gaussian**

$$p(\mathbf{x}_n) = \mathcal{N}(\mathbf{x}_n | 0, \mathbf{W}\mathbf{W}^\top + \sigma^2 \mathbf{I}_D)$$

- PPCA reduces to PCA as the noise variance  $\sigma^2$  tends to zero
- Can use EM to estimate the model parameters (which can be more efficient than standard PCA based on eigen-decomposition)

# Some Nice Aspects about PPCA/FA

- Can also be seen as modeling data using a **low-rank Gaussian**

$$p(\mathbf{x}_n) = \mathcal{N}(\mathbf{x}_n | 0, \mathbf{W}\mathbf{W}^\top + \sigma^2 \mathbf{I}_D)$$

- PPCA reduces to PCA as the noise variance  $\sigma^2$  tends to zero
- Can use EM to estimate the model parameters (which can be more efficient than standard PCA based on eigen-decomposition)
- Gaussian assumption of  $\mathbf{x}_n$  and  $\mathbf{z}_n$  can be removed to model other data types

# Some Nice Aspects about PPCA/FA

- Can also be seen as modeling data using a **low-rank Gaussian**

$$p(\mathbf{x}_n) = \mathcal{N}(\mathbf{x}_n | 0, \mathbf{W}\mathbf{W}^\top + \sigma^2 \mathbf{I}_D)$$

- PPCA reduces to PCA as the noise variance  $\sigma^2$  tends to zero
- Can use EM to estimate the model parameters (which can be more efficient than standard PCA based on eigen-decomposition)
- Gaussian assumption of  $\mathbf{x}_n$  and  $\mathbf{z}_n$  can be removed to model other data types
- Can extend this basic model to dynamic settings, e.g., by changing the prior

$$p(\mathbf{z}_n) = \mathcal{N}(\mathbf{z}_n | \mathbf{z}_{n-1}, \mathbf{I}_K)$$

# Some Nice Aspects about PPCA/FA

- Can also be seen as modeling data using a **low-rank Gaussian**

$$p(\mathbf{x}_n) = \mathcal{N}(\mathbf{x}_n | 0, \mathbf{W}\mathbf{W}^\top + \sigma^2 \mathbf{I}_D)$$

- PPCA reduces to PCA as the noise variance  $\sigma^2$  tends to zero
- Can use EM to estimate the model parameters (which can be more efficient than standard PCA based on eigen-decomposition)
- Gaussian assumption of  $\mathbf{x}_n$  and  $\mathbf{z}_n$  can be removed to model other data types
- Can extend this basic model to dynamic settings, e.g., by changing the prior

$$p(\mathbf{z}_n) = \mathcal{N}(\mathbf{z}_n | \mathbf{z}_{n-1}, \mathbf{I}_K)$$

- Can model data using a mixture of PPCA or mixture of FA models



# Next Class

- Talk in more detail about PPCA, Factor Analysis, and extensions
- EM algorithm for parameter estimation in these models
- Finish off the discussion of generative models and unsupervised learning and move on to “Assorted Topics”