

# CS5340

## Uncertainty Modeling in AI

### Lecture 8: Mixture Models and the EM Algorithm

Asst. Prof. Lee Gim Hee

AY 2018/19

Semester 1

# Course Schedule

Week	Date	Topic	Remarks
1	15 Aug	Introduction to probabilities and probability distributions	
2	22 Aug	Fitting probability models	Hari Raya Haji*
3	29 Aug	Bayesian networks (Directed graphical models)	
4	05 Sep	Markov random Fields (Undirected graphical models)	
5	12 Sep	I will be traveling	No Lecture
6	19 Sep	Variable elimination and belief propagation	
-	26 Sep	Recess week	No lecture
7	03 Oct	Factor graph and the junction tree algorithm	
8	10 Oct	Parameter learning with complete data	
9	17 Oct	Mixture models and the EM algorithm	
10	24 Oct	Hidden Markov Models (HMM)	
11	31 Oct	Monte Carlo inference (Sampling)	
12	07 Nov	Variational inference	
13	14 Nov	Graph-cut and alpha expansion	

\* **Make-up lecture:** 25 Aug (Sat), 9.30am-12.30pm, LT 15

# Acknowledgements

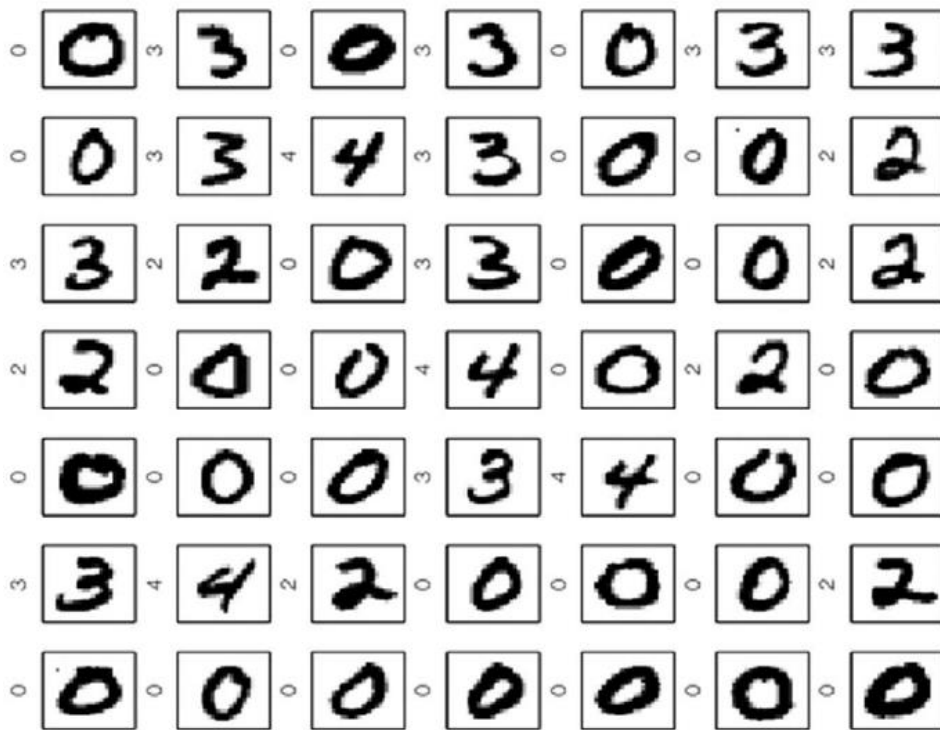
- A lot of slides and content of this lecture are adopted from:
  1. “Pattern Recognition and Machine Learning”, Christopher Bishop, Chapter 8.
  2. “Machine Learning – A Probabilistic Perspective”, Kevin Murphy, Chapter 11.
  3. “An Introduction to Probabilistic Graphical Models”, Michael I. Jordan, Chapters 10 and 11.  
<http://people.eecs.berkeley.edu/~jordan/prelims/chapter10.pdf>  
<http://people.eecs.berkeley.edu/~jordan/prelims/chapter11.pdf>
  4. “Probabilistic Graphical Models”, Daphne Koller and Nir Friedman, chapter 19.2.2.
  5. “Computer Vision: Models, Learning and Inference”, Simon Prince, Chapters 7.1-7.4 and 7.8.

# Learning Outcomes

- Students should be able to:
  1. Use the non-probabilistic **k-mean algorithm** to solve the clustering problem.
  2. Describe the **Gaussian-mixture model**.
  3. Apply the **Expectation-Maximization algorithm** for estimation of both the unknown parameters and latent variables.

# Motivation

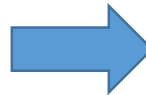
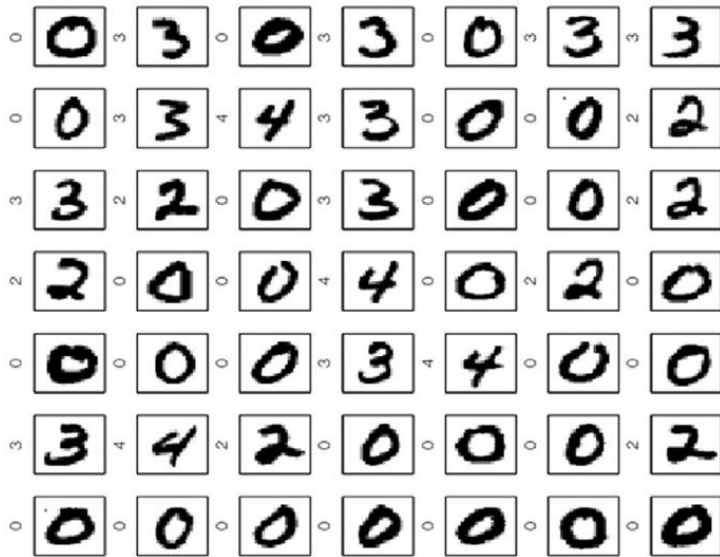
- Suppose that we are given **many images of handwritten digits**, and we can extract a 2D representation of each image  $x_n \in \mathbb{R}^2$ .



$x_1, x_2, x_3, \dots, x_N$

# Motivation

- We can observe **clusters** from the 2D plot of  $x_1, \dots, x_N$ , where  $x_n \in \mathbb{R}^2$ .



$x_1, x_2, x_3, \dots, x_N$

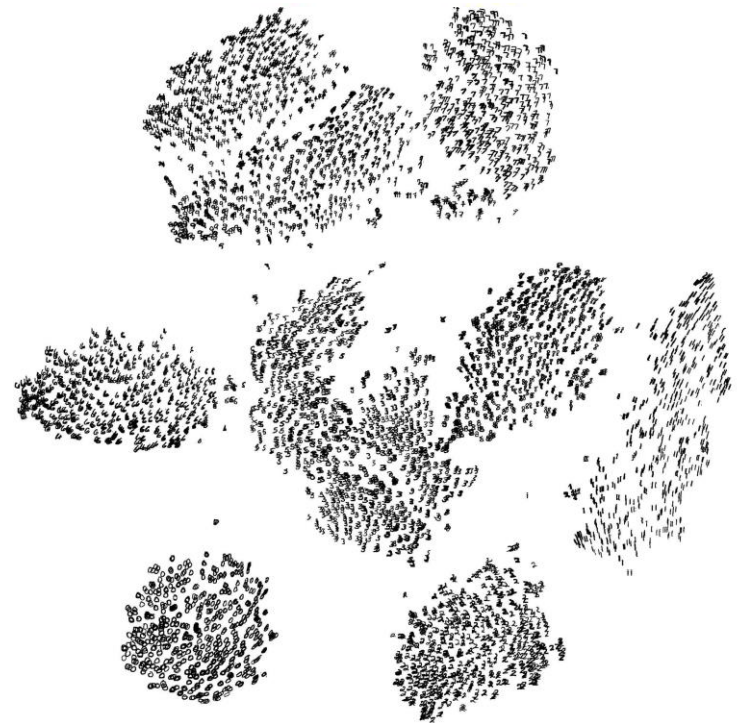


Image source: <https://stab-iitb.org/electronics-club/blog/2016/05/deep-learning-based-image-classification/>

# Motivation

- We can observe **clusters** from the 2D plot of  $x_1, \dots, x_N$ , where  $x_n \in \mathbb{R}^2$ .

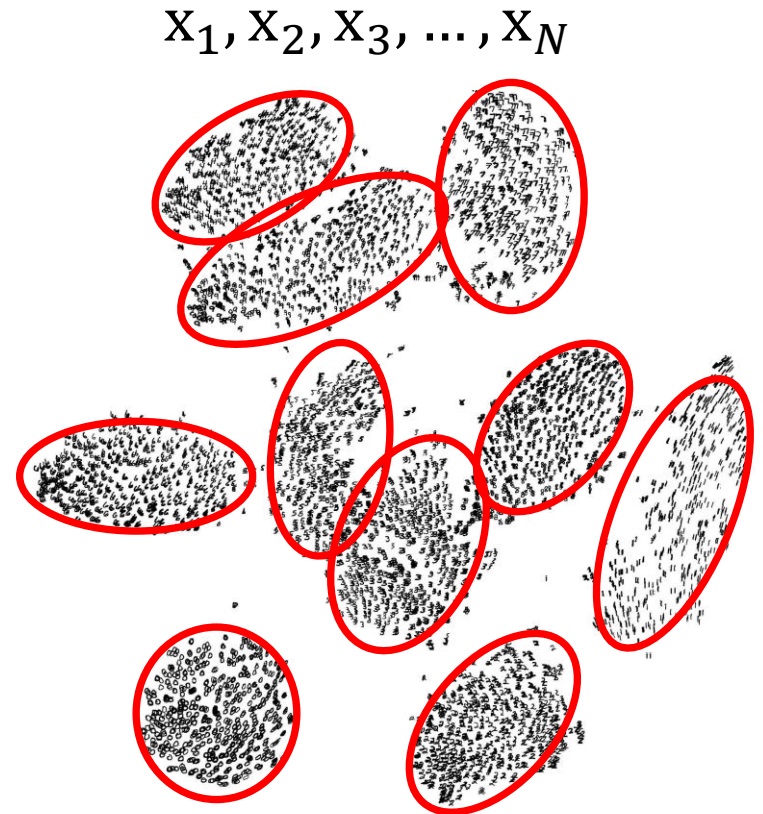
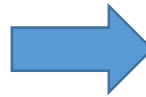
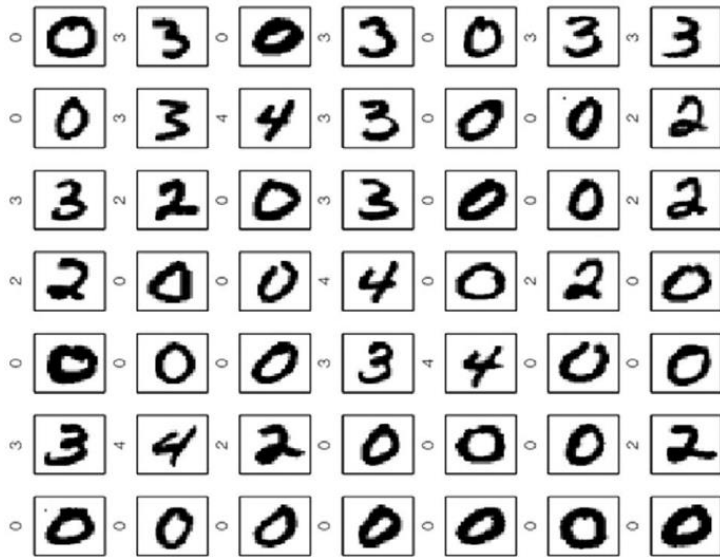


Image source: <https://stab-iitb.org/electronics-club/blog/2016/05/deep-learning-based-image-classification/>



# Motivation

- It turns out that the data points  $x_n$  with the same digit tend to be **associated with the same cluster**!
- This suggests that clusters can be **used to represent complex distributions**.

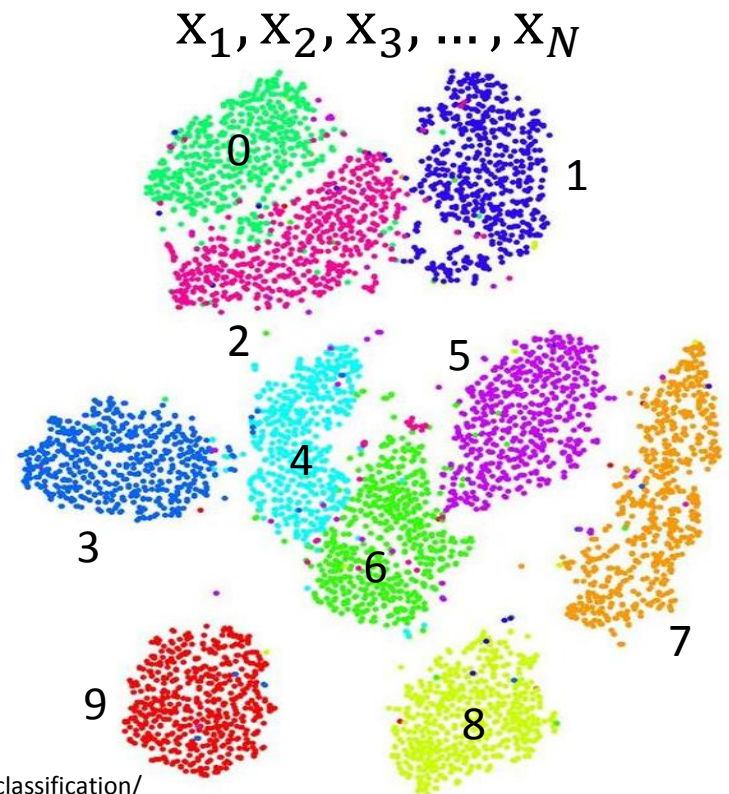
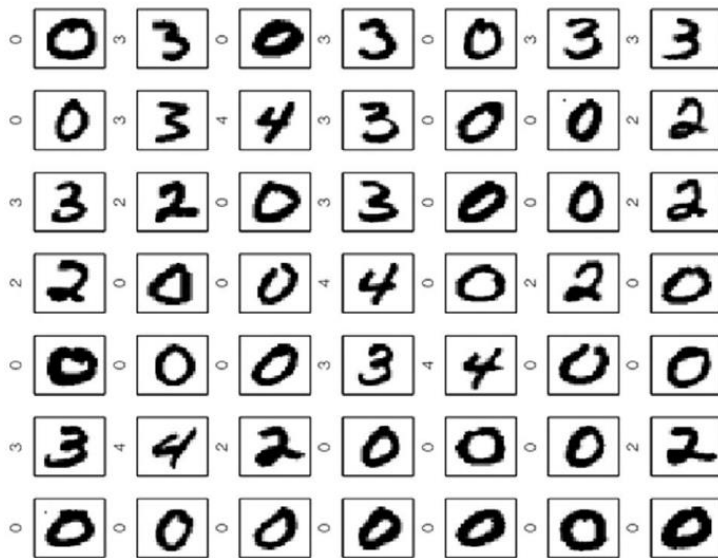


Image source: <https://stab-iitb.org/electronics-club/blog/2016/05/deep-learning-based-image-classification/>



# Motivation

## Chicken-and-Egg Problem:

1. How to find **unknown parameters** of the clusters?
2. Which cluster does each data belong to, i.e. **data association**?

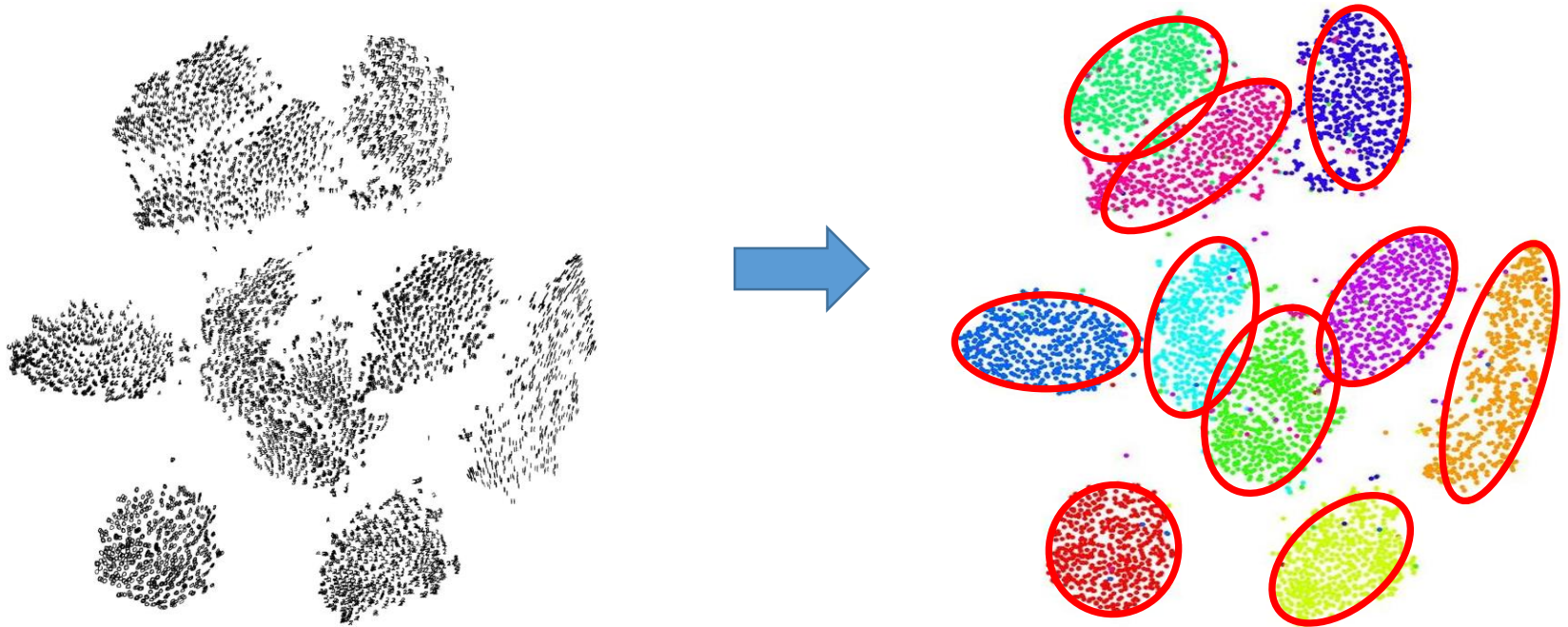


Image source: <https://stab-iitb.org/electronics-club/blog/2016/05/deep-learning-based-image-classification/>

# Non-Probabilistic Approach: K-Means

## Given:

1. **Data set**  $\{x_1, \dots, x_N\}$  of observations
2. **Number of clusters**  $K$

## Find:

1. The  $K$  **cluster centers**  $\{\mu_1, \dots, \mu_K\}$ , i.e. unknown parameters (assume each cluster is a circle)
2. **Assignment of each point**  $x_n$  to a cluster center  $k$ , i.e. data association

# Non-Probabilistic Approach: K-Means

## 1-of- $K$ coding:

- For each data point  $x_n$ , we introduce a corresponding set of **binary indicator variables**:

$$r_{nk} \in \{0,1\} \quad \forall k = 1, \dots, K, \quad \text{s.t.} \quad \underbrace{\sum_k r_{nk}}_{\text{1-of-}K \text{ constraint}} = 1.$$

- This binary indicator variable describes which of the  $K$  clusters the data point  $x_n$  is assigned to.
- 1-of- $K$  constraint ensures that each data point  $x_n$  gets **assigned to only ONE cluster  $k$** .

# Non-Probabilistic Approach: K-Means

- Formally, the **goal** of k-means is to find values for  $\{r_{nk}\}$  and  $\{\mu_k\}$  so as to **minimize the “distortion measure”**:

$$J = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \|\mathbf{x}_n - \mu_k\|^2, \quad \text{s.t.} \quad \sum_k r_{nk} = 1.$$

- Represents the **sum-of-squares** of the distances of each point  $\mathbf{x}_n$  to its assigned vector  $\mu_k$ .

# Non-Probabilistic Approach: K-Means

## K-Means (a.k.a. Lloyd) Algorithm:

1. **Initialization**: Randomly choose some initial values for  $\{\mu_k\}$ .
2. **Assignment step**: Minimize  $J$  w.r.t.  $\{r_{nk}\}$ , while keeping  $\{\mu_k\}$  fixed.
3. **Update step**: Minimize  $J$  w.r.t.  $\{\mu_k\}$ , while keeping  $\{r_{nk}\}$  fixed.

Iterate until  
convergence

# Non-Probabilistic Approach: K-Means

## Assignment Step:

- We can **optimize each  $r_n$  separately** since they are independent:

$$\operatorname{argmin}_{r_n} \sum_k r_{nk} \|\mathbf{x}_n - \mu_k\|^2, \quad \text{s.t.} \quad \sum_k r_{nk} = 1.$$

- By inspection, the minimum occurs when we assign  $\mathbf{x}_n$  to the **current closest cluster center**:

$$r_{nk} = \begin{cases} 1 & \text{if } k = \arg \min_j \|\mathbf{x}_n - \mu_j\|^2 \\ 0 & \text{otherwise.} \end{cases}$$

# Non-Probabilistic Approach: K-Means

## Update Step:

- We can **optimize each  $\mu_k$  separately** since they are independent:

$$\operatorname{argmin}_{\mu_k} \underbrace{\sum_n r_{nk} \|\mathbf{x}_n - \mu_k\|^2}_L$$

- Differentiation of  $L$  w.r.t.  $\mu_k$ , and equate to zero gives:

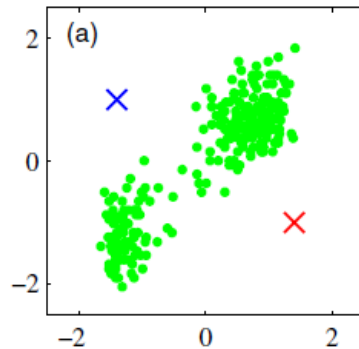
$$2 \sum_{n=1}^N r_{nk} (\mathbf{x}_n - \mu_k) = 0 \quad \Rightarrow \quad \boxed{\mu_k = \frac{\sum_n r_{nk} \mathbf{x}_n}{\sum_n r_{nk}}}$$

Mean of all points assigned to cluster  $k$ , hence “k-means”!

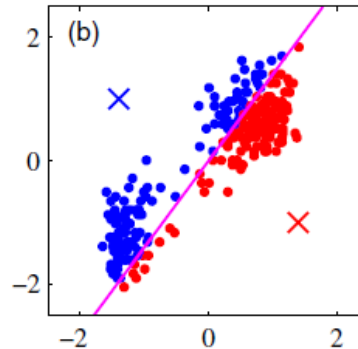


# Non-Probabilistic Approach: K-Means

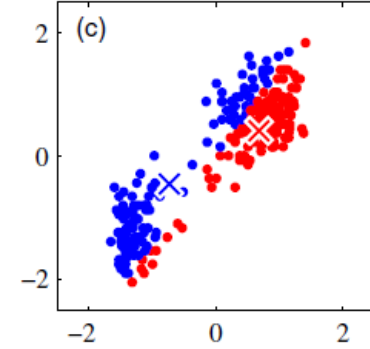
**Initialization ( $K = 2$ ):**  
Choose  $\mu_k$



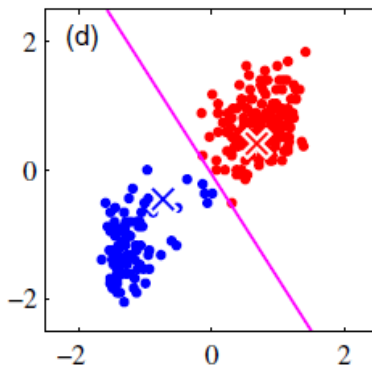
**Iteration (1):**  
Assignment of  $r_{nk}$



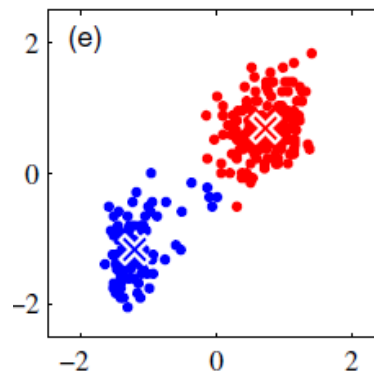
**Iteration (1):**  
Update of  $\mu_k$



**Iteration (2):**  
Assignment of  $r_{nk}$



**Iteration (2):**  
Update of  $\mu_k$



**Iteration (3):**  
Assignment of  $r_{nk}$

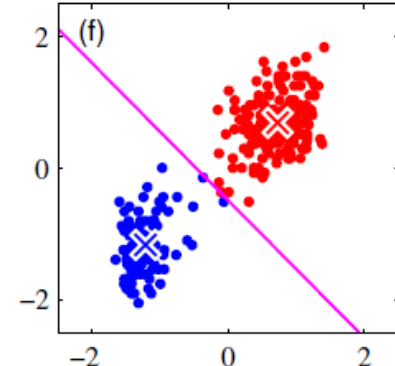
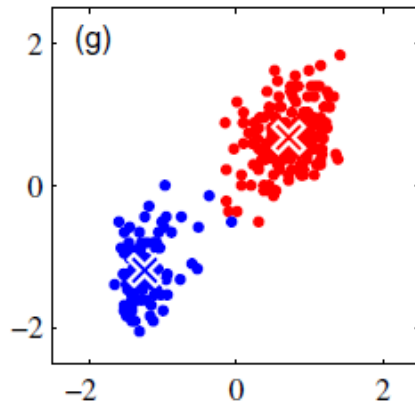


Image source: "Pattern recognition and machine learning", Christopher Bishop

# Non-Probabilistic Approach: K-Means

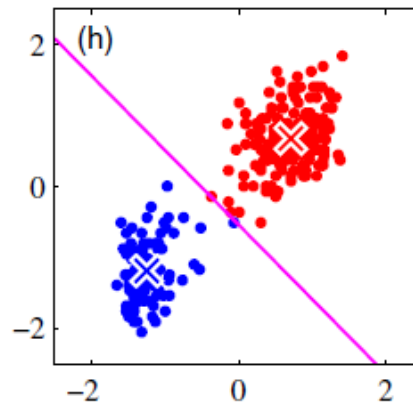
**Iteration (3):**

Update of  $\mu_k$



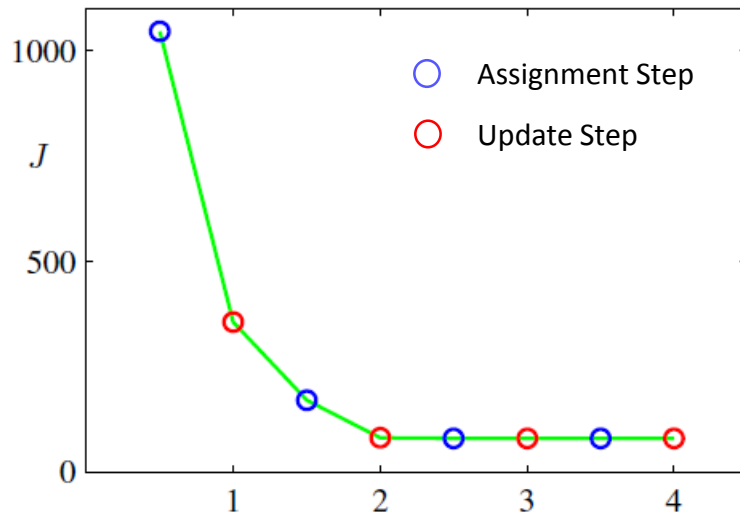
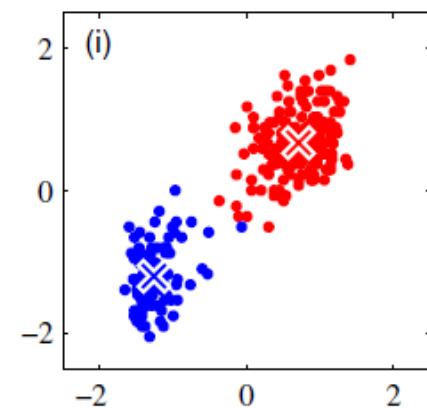
**Iteration (4):**

Assignment of  $r_{nk}$



**Iteration (4):**

Update of  $\mu_k$



- Plot of the **cost function  $J$**  after each iteration.
- The algorithm has **converged** after the third update step.

Image source: "Pattern recognition and machine learning", Christopher Bishop

# Application: Image Segmentation



Image source: "Pattern recognition and machine learning", Christopher Bishop

# Probabilistic Approach

Can we model the problem with a probabilistic approach?

# Gaussian Mixture Models

- “Old Faithful” dataset: 272 measurements of the eruption of the Old Faithful geyser at Yellowstone National Park, USA.
- Data set forms two dominant clumps, a simple Gaussian distribution is unable to capture this structure.
- A linear superposition of two Gaussians gives a better characterization of the data set.



Photo source: “Old Faithful”, courtesy of Chen Li, June’18

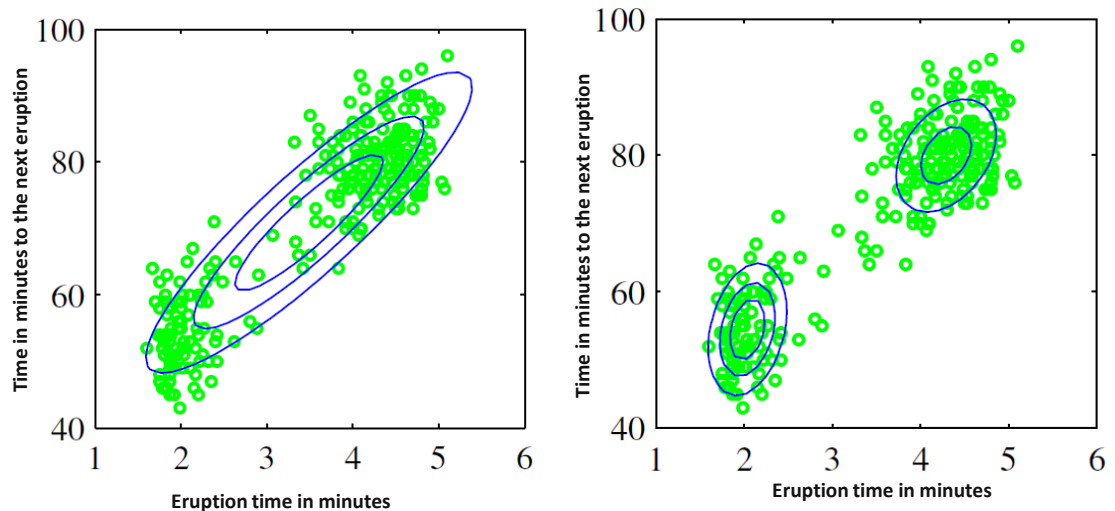
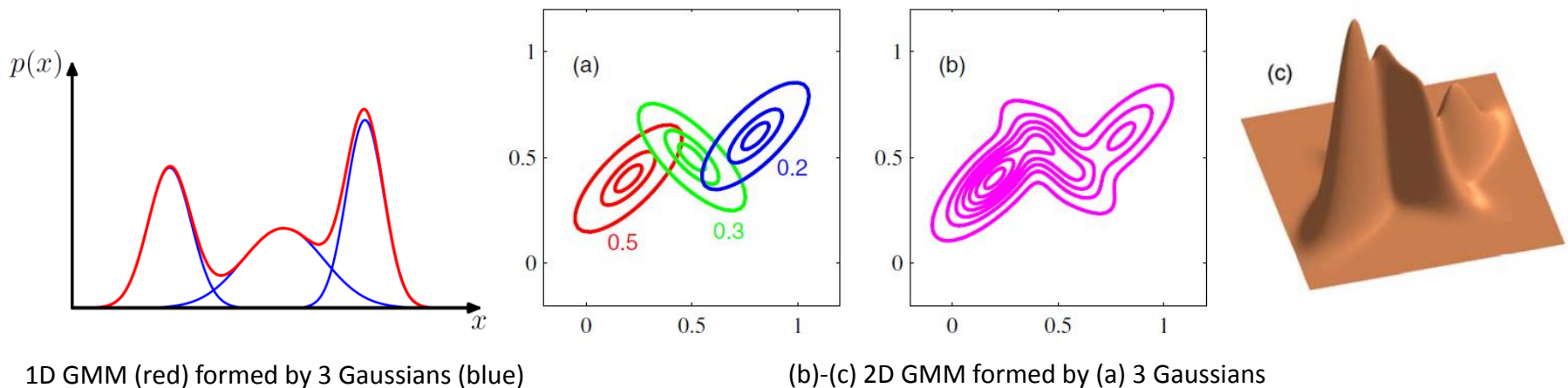


Image source: “Pattern recognition and machine learning”, Christopher Bishop

# Gaussian Mixture Models

- **Mixture distributions**: Probabilistic models formed by taking linear combinations of more basic distributions such as Gaussian a.k.a **Gaussian Mixture Model (GMM)**.
- Linear combination of sufficient number of Gaussians give rise to very complex densities that can be used to **approximate almost any continuous density** with arbitrary accuracy.

## Example:



# Gaussian Mixture Models

- The **probability distribution** of a mixture of Gaussians is given by the superposition of  $K$  Gaussian densities:

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

- Each Gaussian density  $\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$  is called a **component of the mixture**, and has its own mean  $\boldsymbol{\mu}_k$  and covariance  $\boldsymbol{\Sigma}_k$ .
- The parameters  $0 \leq \pi_k \leq 1$  is the **mixing coefficients**, and must sum to one:

$$\sum_{k=1}^K \pi_k = 1$$



# Gaussian Mixture Models

- Let us introduce a  $K$ -dimensional **binary random variable  $Z$**  having a **1-of- $K$  representation**.
- $z_k = 1 \Rightarrow z_{j \neq k} = 0$  indicates the **assignment of the random variable  $x$**  to the  $k^{th}$  Gaussian density.
- The values of  $Z_k$  must satisfy:

$$z_k \in \{0,1\} \quad \text{and} \quad \sum_k z_k = 1$$

- **$K$  possible states** for the vector  $Z$  according to which element is non-zero.

# Gaussian Mixture Models

- The marginal distribution of  $Z$  is a **categorical distribution** specified in terms of the **mixing coefficients**  $\pi_k$ :

$$p(z) = \prod_{k=1}^K \pi_k^{z_k} = \text{cat}_z[\pi]$$

where the parameter  $\pi = [\pi_1, \dots, \pi_K]$  must be:

$$0 \leq \pi_k \leq 1 \quad \text{and} \quad \sum_{k=1}^K \pi_k = 1$$

# Gaussian Mixture Models

- **Conditional distribution** of  $X$  given a particular value for  $Z$  is a Gaussian:

$$p(\mathbf{x}|z_k = 1) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

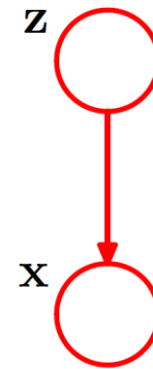
- Which can also be written as:

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

# Gaussian Mixture Models

- The **joint distribution**  $p(\mathbf{x}, \mathbf{z})$  is given by the following DGM:

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



- The **marginal distribution of  $\mathbf{X}$**  is then obtained by summing the joint distribution over all possible states of the **latent variable  $\mathbf{Z}$**  to give:

$$p(\mathbf{x}) = \sum_{\mathbf{z}} p(\mathbf{z})p(\mathbf{x}|\mathbf{z}) = \sum_{\mathbf{z}} \prod_{k=1}^K \pi_k^{z_k} \mathcal{N}(\mathbf{x} | \mu_k, \Sigma_k)^{z_k}$$

# Gaussian Mixture Models

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

$$= \sum_{\mathbf{z}} \prod_{k=1}^K \pi_k^{z_k} \mathcal{N}(\mathbf{x} | \mu_k, \Sigma_k)^{z_k}$$

$$= \underbrace{\left( \prod_{k=1}^K \pi_k^{z_k} \mathcal{N}(\mathbf{x} | \mu_k, \Sigma_k)^{z_k} \right)_{z_{k=1}=1}}_{\pi_{k=1} \mathcal{N}(\mathbf{x} | \mu_{k=1}, \Sigma_{k=1})} + \dots \dots + \underbrace{\left( \prod_{k=1}^K \pi_k^{z_k} \mathcal{N}(\mathbf{x} | \mu_k, \Sigma_k)^{z_k} \right)_{z_{k=K}=1}}_{\pi_{k=K} \mathcal{N}(\mathbf{x} | \mu_{k=K}, \Sigma_{k=K})}$$


$$= \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x} | \mu_k, \Sigma_k)$$

# Gaussian Mixture Models

- Another quantity that will play an important role is  $p(z_k = 1 | \mathbf{x})$  denoted as  $\gamma(z_k)$ , whose value can be found using Bayes' theorem:

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

Prior probability  
of  $z_k = 1$



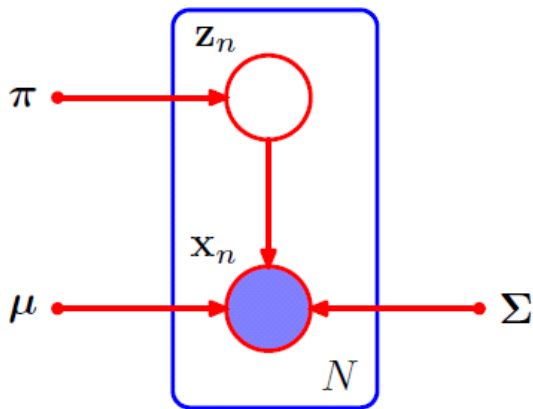
- As we shall see later,  $\gamma(z_k)$  can also be viewed as the *responsibility* that component  $k$  takes for 'explaining' the observation  $\mathbf{x}$ .

# Maximum Log-Likelihood

- Suppose we have a data set of  **$N$  i.i.d. observations**  $\{x_1, \dots, x_N\}: x_n \in \mathbb{R}^D$ , we model the **log-likelihood** as:

$$\ln p(x_1, \dots, x_N | \theta) = \sum_{n=1}^N \ln \underbrace{\sum_{z_n} p(x_n, z_n | \theta)}_{\text{Unknown parameter } \theta}$$

Incomplete data because  $Z = \{z_1, \dots, z_N\} \in \mathbb{R}^{N \times K}$  is a latent variable



$$= \sum_{n=1}^N \ln \sum_{z_n} p(z_n | \pi) p(x_n | z_n, \mu, \Sigma)$$

$$= \sum_{n=1}^N \ln \sum_{k=1}^K \pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k)$$

where

$$\theta = \{\pi_1 \dots \pi_K, \mu_1 \dots \mu_K, \Sigma_1 \dots \Sigma_K\}$$



# Maximum Log-Likelihood

$$\operatorname{argmax}_{\theta} \ln p(\mathbf{x}_1, \dots, \mathbf{x}_N | \theta) = \operatorname{argmax}_{\pi, \mu, \Sigma} \sum_{n=1}^N \ln \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_n | \mu_k, \Sigma_k)$$

- The **summation term inside the logarithm** prevents the logarithm function from acting directly on the Gaussian.
- We shall see that we will **no longer obtain a closed-form solution** of the unknown parameters by setting the derivatives to zero!

# Maximum Log-Likelihood

- Setting the derivatives of  $\ln p(\mathbf{x}_1, \dots, \mathbf{x}_N | \theta)$  **w.r.t.  $\mu_k$**  of the Gaussian components to zero, we obtain:

$$0 = - \sum_{n=1}^N \underbrace{\frac{\pi_k \mathcal{N}(\mathbf{x}_n | \mu_k, \Sigma_k)}{\sum_j \pi_j \mathcal{N}(\mathbf{x}_n | \mu_j, \Sigma_j)}}_{\gamma(z_{nk})} \Sigma_k (\mathbf{x}_n - \mu_k)$$

$\gamma(z_{nk})$  : Responsibility

- Multiplying by  $\Sigma_k^{-1}$  (which we assume to be non-singular) and rearranging, we obtain:

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) \mathbf{x}_n, \quad \text{where} \quad N_k = \sum_{n=1}^N \gamma(z_{nk})$$

**Not closed-form** since responsibility is a function of  $\pi_k, \mu_k, \Sigma_k$ !

# Maximum Log-Likelihood

- If we set the derivative of  $\ln p(\mathbf{x}_1, \dots, \mathbf{x}_N | \theta)$  w.r.t.  $\Sigma_k$  to zero, we get:

$$\Sigma_k = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T$$

**Not closed-form** since responsibility is a function of  $\pi_k, \mu_k, \Sigma_k$ !

# Maximum Log-Likelihood

- Finally, we maximize  $\ln p(\mathbf{x}_1, \dots, \mathbf{x}_N | \theta)$  w.r.t.  $\pi_k$  subjected to  $\sum_k \pi_k = 1$  by maximizing the following **auxiliary equation**:

$$\ln p(\mathbf{X} | \boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) + \lambda \left( \sum_{k=1}^K \pi_k - 1 \right)$$

Lagrange multiplier

- Which gives:

$$0 = \sum_{n=1}^N \underbrace{\frac{\mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_j \pi_j \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}}_{= N_k / \pi_k} + \lambda$$

- Multiply both sides by  $\pi_k$  and sum over  $k$  making use of the constraint  $\sum_k \pi_k = 1$ , we get:

$$\sum_{n=1}^N \frac{\sum_k \pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_j \pi_j \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} = -\lambda \sum_k \pi_k \quad \Rightarrow \quad \lambda = -N$$

# Maximum Log-Likelihood

- Using  $\lambda = -N$  to eliminate  $\lambda$  and rearranging, we get:

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

**Not closed-form** since responsibility is a function of  $\pi_k, \mu_k, \Sigma_k$ !

- This is the **average responsibility** which the  $k^{th}$  component takes for explaining the data points.

# Maximum Log-Likelihood

- The maximum log-likelihood estimates of the unknown parameter **do not constitute a closed-form solution** because of the responsibilities  $\gamma(z_{nk})$ .
- However, these results do suggest a **simple iterative scheme** for finding a solution to the maximum likelihood problem!

# EM for Gaussian Mixtures

Given a Gaussian mixture model, the goal is to **maximize the likelihood function** w.r.t. the parameters  $\theta = \{\pi_k, \mu_k, \Sigma_k\}$ .

1. **Initialize** the means  $\mu_k$ , covariances  $\Sigma_k$  and mixing coefficients  $\pi_k$ , and **evaluate** the initial value of the log likelihood.
2. **Expectation Step**: Evaluate the **responsibilities**  $\gamma(Z)$  using the current parameter values

$\gamma(Z)$  is a  $N \times K$  table  
where each entry is  $\gamma(z_{nk})$

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



# EM for Gaussian Mixtures

Given a Gaussian mixture model, the goal is to **maximize the likelihood function** w.r.t. the parameters  $\theta = \{\pi_k, \mu_k, \Sigma_k\}$ .

3. **Maximization Step**: Re-estimate the **parameters** using the current responsibilities

$$\begin{aligned}\mu_k^{\text{new}} &= \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) \mathbf{x}_n \\ \Sigma_k^{\text{new}} &= \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) (\mathbf{x}_n - \mu_k^{\text{new}}) (\mathbf{x}_n - \mu_k^{\text{new}})^T \\ \pi_k^{\text{new}} &= \frac{N_k}{N}\end{aligned}$$

where

$$N_k = \sum_{n=1}^N \gamma(z_{nk}).$$

# EM for Gaussian Mixtures

Given a Gaussian mixture model, the goal is to **maximize the likelihood function** w.r.t. the parameters  $\theta = \{\pi_k, \mu_k, \Sigma_k\}$ .

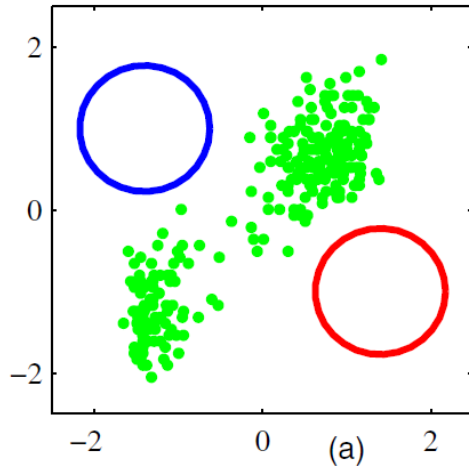
4. Evaluate the **log likelihood**:

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

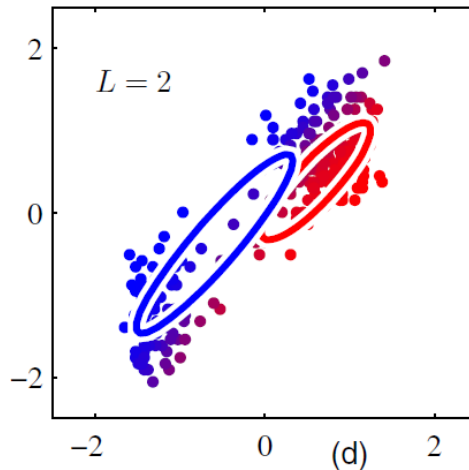
and **check for convergence** of either the parameters or the log likelihood. If the convergence criterion is NOT satisfied return to step 2.

# Illustration of the EM Algorithm

**Initialization:** random  $\mu_k$ ,  
identity  $\Sigma_k$  and  $\pi_k = 0.5$

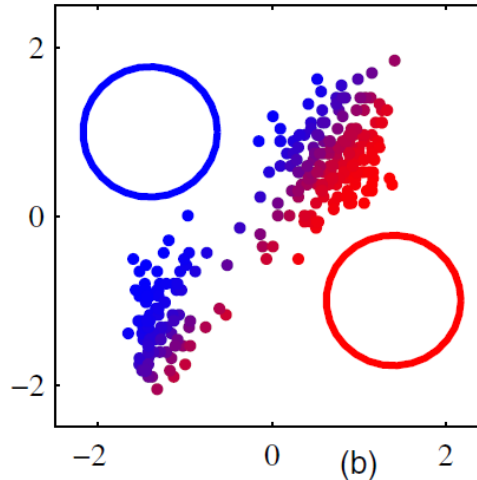


**M Step: 2<sup>nd</sup> iteration**

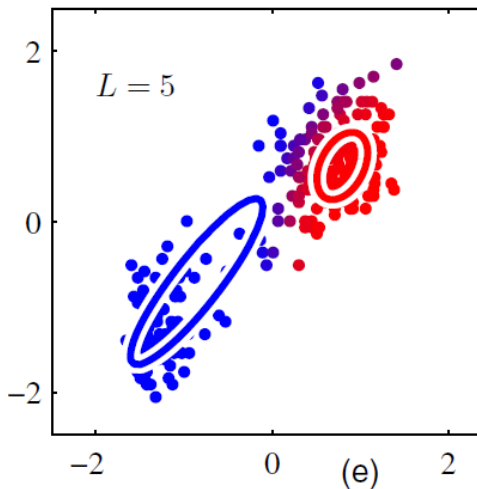


**E Step: 1<sup>st</sup> iteration**

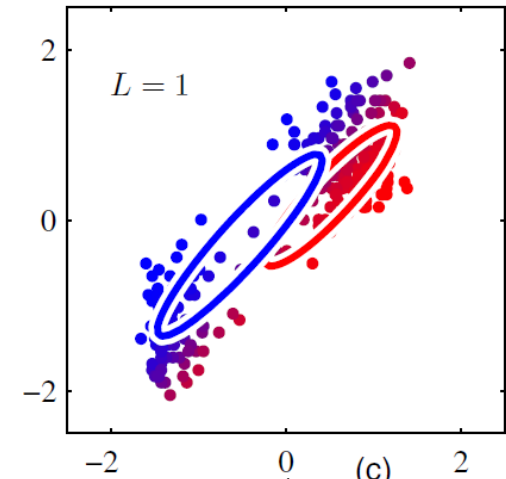
(color of dots used to illustrate  
strength of mixing coefficients)



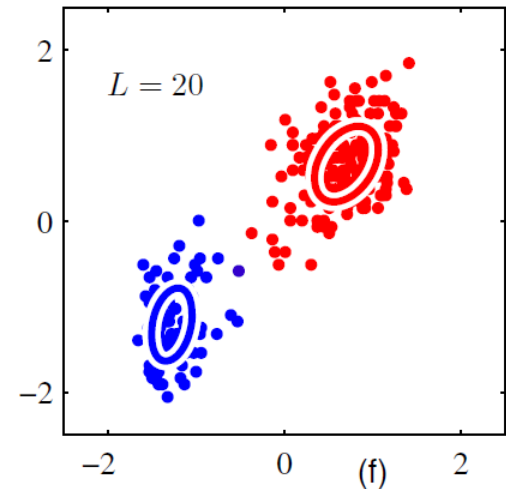
**M Step: 5<sup>th</sup> iteration**



**M Step: 1<sup>st</sup> iteration**



**M Step: 20<sup>th</sup> iteration**



# EM for GMM: Initialization

- EM algorithm takes **many more iterations** to reach (approximate) convergence compared with the *K*-means algorithm.
- And each cycle requires **significantly more computation**.
- Run the *K*-means algorithm first to **find a suitable initialization** for a GMM that is subsequently adapted using EM.

# The General EM Algorithm

- The goal of the EM algorithm is to find **maximum likelihood solutions** for models having **latent variables**.
- The log-likelihood function is given by:

$$\ln p(\mathbf{X}|\theta) = \ln \left\{ \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\theta) \right\} \quad \text{or} \quad \ln p(X|\theta) = \ln \left\{ \int_{\mathbf{Z}} p(X, \mathbf{Z}|\theta) \right\}$$

where

Discrete latent variable

Continuous latent variable

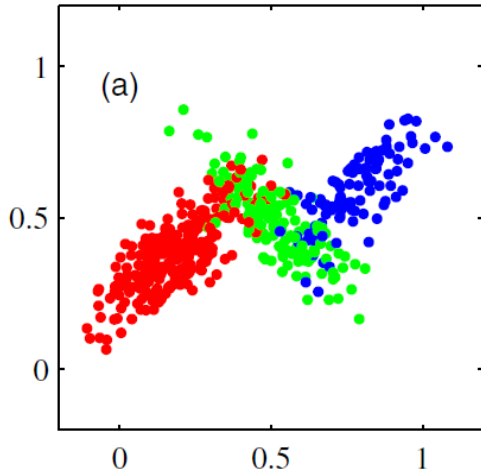
- $\mathbf{X}$ : set of all **observed data** with  $n^{th}$  row represents  $\mathbf{x}_n^T$
- $\mathbf{Z}$ : set of all **latent variables** with corresponding row  $\mathbf{z}_n^T$
- $\theta$ : set of all **model parameters**

# The General EM Algorithm

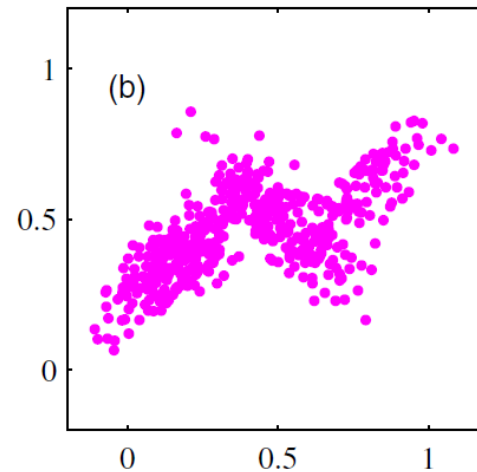
$$\ln p(\mathbf{X}|\boldsymbol{\theta}) = \ln \left\{ \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta}) \right\} \quad \text{or} \quad \ln p(X|\theta) = \ln \left\{ \int_{\mathbf{Z}} p(X, \mathbf{Z}|\theta) \right\}$$

- A key observation is that the summation/integration over the latent variables **appears inside the logarithm**.
- Marginal distribution  $p(X|\theta)$  **does not simplify** even if the joint distribution  $p(X, \mathbf{Z}|\theta)$  belongs to the exponential family, e.g. Gaussian.
- Resulting in **complicated (non closed-form) expressions** for the maximum likelihood solution.

# The General EM Algorithm



Assignment of  $x_n$  into clusters is known: Complete data



Assignment of  $x_n$  into clusters is unknown: Incomplete data

- **Complete data:** Both observation  $X$  and latent variable  $Z$  are known.
- **Incomplete data:** Observation  $X$  is known, but latent variable  $Z$  is unknown.

# The General EM Algorithm

- It is straightforward to maximize the likelihood function for the **complete data set**  $\ln p(X, Z|\theta)$ .
- In practice, however, we are **not given** the complete data set  $\{X, Z\}$ , but only the incomplete data  $X$ .
- Instead, we consider **maximization** of the **expected value** of the complete data log-likelihood  $\ln p(X, Z|\theta)$  w.r.t.  $p(Z|X, \theta)$ .
- Do the Expectation and Maximization steps **iteratively** until convergence.



# The General EM Algorithm

## Expectation Step:

- Use the **current parameter values**  $\theta^{old}$  to find the posterior distribution of the latent variables given by  $p(\mathbf{Z}|\mathbf{X}, \theta^{old})$ .
- We then use  $p(\mathbf{Z}|\mathbf{X}, \theta^{old})$  to find the **expectation of the complete-data log likelihood** evaluated for some general parameter value  $\theta$ .
- This expectation, denoted  $Q(\theta, \theta^{old})$ , is given by:

$$\begin{aligned} Q(\theta, \theta^{old}) &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \theta^{old}) \ln p(\mathbf{X}, \mathbf{Z}|\theta) \\ &= \mathbb{E}_{\mathbf{Z}|\mathbf{X}, \theta^{old}} [\ln p(\mathbf{X}, \mathbf{Z}|\theta)] \end{aligned}$$

# The General EM Algorithm

## Maximization Step:

- Determine the **revised parameter estimate**  $\theta^{new}$  by maximizing this function:

$$\begin{aligned}\theta^{new} &= \arg \max_{\theta} Q(\theta, \theta^{old}) \\ &= \arg \max_{\theta} \underbrace{\sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \theta^{old}) \ln p(\mathbf{X}, \mathbf{Z}|\theta)}\end{aligned}$$

**Log is now inside the summation!**

- Since the logarithm now acts directly on the joint distribution  $p(\mathbf{X}, \mathbf{Z}|\theta)$ , the corresponding M-step will be **tractable**.

# The General EM Algorithm

- **Given:** a **joint distribution**  $p(X, Z|\theta)$  over observed variables  $X$  and latent variables  $Z$ , governed by parameters  $\theta$ .
- **Goal:** is to **maximize the likelihood function**  $p(X|\theta)$  with respect to  $\theta$ .

# The General EM Algorithm

1. Choose an **initial setting** for the parameters  $\theta^{old}$ .

2. **Expectation step**: Evaluate  $p(\mathbf{Z}|\mathbf{X}, \theta^{old})$ .

3. **Maximization step**: Evaluate  $\theta^{new}$  given by:

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

where

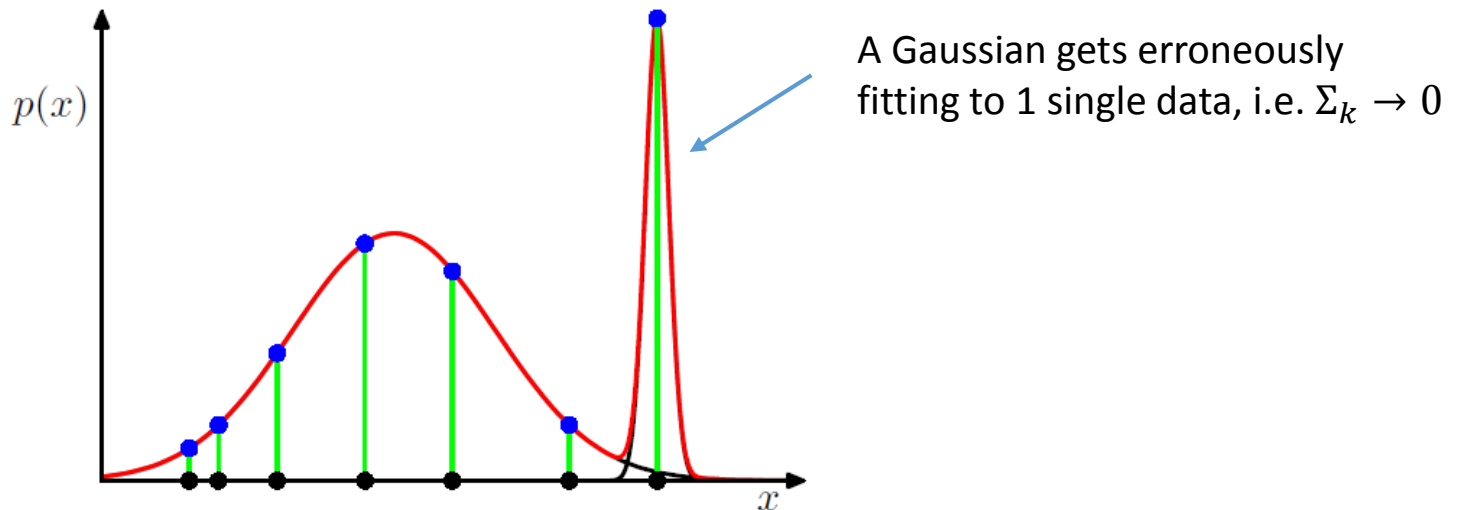
$$Q(\theta, \theta^{old}) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \theta^{old}) \ln p(\mathbf{X}, \mathbf{Z}|\theta)$$

4. Check for convergence of either the log likelihood or the parameter values, **if not converged**:

$$\theta^{old} \leftarrow \theta^{new}$$

# Singularities and MAP

Illustration of the “singularity problem”:



- Problem can be alleviated by applying MAP on  $Q(\theta, \theta^{old})$ .

$$Q(\theta, \theta^{old}) + \ln p(\theta) \leftarrow \text{Dirichlet prior on } \pi_k, \text{ and normalized inversed Gaussian on } (\mu_k, \sigma_k).$$

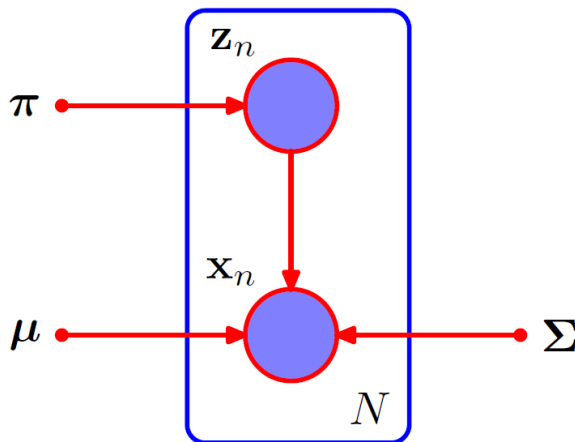
# Gaussian Mixture Revisited

- The **complete data log-likelihood** is given by:

$$\ln p(\mathbf{X}, \mathbf{Z} | \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi}) = \ln \prod_{n=1}^N \prod_{k=1}^K \pi_k^{z_{nk}} \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{z_{nk}}$$

$$= \sum_{n=1}^N \sum_{k=1}^K z_{nk} \underbrace{\{ \ln \pi_k + \ln \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \}}$$

Now the log is inside the summation!



# Gaussian Mixture Revisited

## Complete data log-likelihood:

$$\ln p(X, Z \mid \pi, \mu, \Sigma) = \sum_{k=1}^K \sum_{n=1}^N z_{nk} (\ln \pi_k + \ln \mathcal{N}(\mathbf{x}_n \mid \mu_k, \Sigma_k))$$

- We can **switch the sums** and only look at the points associated with the respective mixture component  $k$ .
- Maximization of each  $(\mu_k, \Sigma_k)$  can be **done separately**, similarly we can solve for  $\pi_k$  by enforcing the sum to one constraint using Lagrange multiplier.
- **Problem:** the latent variables  $Z$  are **not observed!!!**

# Gaussian Mixture Revisited

- Instead of maximizing the joint log-likelihood, we maximize its **expectation** under the latent variable distribution:

$$\begin{aligned}\mathbb{E}_Z[\ln p(X, Z|\theta)] &= \sum_Z p(Z|X, \theta^{old}) \ln p(X, Z|\theta) \\&= \sum_n \sum_k p(z_{nk}|x_n, \theta^{old}) \ln p(x_n, z_{nk}|\theta) \\&= \sum_n \sum_k \frac{p(x_n|z_{nk}, \theta^{old})p(z_{nk}|\theta^{old})}{p(x_n|\theta^{old})} \ln p(x_n, z_{nk}|\theta) && \text{(Bayes Rule)} \\&= \sum_n \sum_k \underbrace{\frac{\mathcal{N}(x_n|\mu_k^{old}, \Sigma_k^{old})\pi_k^{old}}{\sum_j \mathcal{N}(x_n|\mu_j^{old}, \Sigma_j^{old})\pi_j^{old}}}_{\gamma(z_{nk})} \ln p(x_n, z_{nk}|\theta) \\&\quad \gamma(z_{nk}) \quad \text{Responsibility that stays fixed} \\&= \sum_n \sum_k \gamma(z_{nk}) [\ln \pi_k + \ln \mathcal{N}(x_n|\mu_k, \Sigma_k)]\end{aligned}$$



# Gaussian Mixture Revisited

$$\mathbb{E}_Z[\ln p(X, Z|\theta)] = \sum_k \sum_n \gamma(z_{nk}) [\ln \pi_k + \ln \mathcal{N}(x_n|\mu_k, \Sigma_k)]$$

- Similar to the complete log-likelihood case, we can **switch the sums** and only look at the points associated with the respective mixture component  $k$ .
- Since  $\gamma(z_{nk})$  stays fixed, maximization of each  $(\mu_k, \Sigma_k)$  can be **done separately**, similarly we can solve for  $\pi_k$  by enforcing the sum to one constraint using Lagrange multiplier.

# The Theory Behind EM Algorithm

- Maximizing the log-likelihood  $\ln p(X|\theta)$  is difficult due to the **need of marginalizing over the latent variables** inside the logarithm:

$$\ln p(X|\theta) = \ln \sum_Z p(X, Z|\theta)$$

- It turns out that  $\ln p(X|\theta)$  can be maximized by **maximizing its lower bound**  $\mathcal{L}(q, \theta)$  within the EM algorithm.

# The Theory Behind EM Algorithm

- Let us rewrite the log-likelihood into:

$$\ln p(\mathbf{X}|\boldsymbol{\theta}) = \underbrace{\mathcal{L}(q, \boldsymbol{\theta})}_{\text{Lower bound}} + \underbrace{\text{KL}(q||p)}_{\text{KL Divergence}}$$

where

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

$$\text{KL}(q||p) = - \sum_{\mathbf{Z}} q(\mathbf{Z}) \ln \left\{ \underbrace{\frac{p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta})}{q(\mathbf{Z})}} \right\} \geq 0$$

$q(\mathbf{Z})$  is a distribution we defined over the latent variable

# The Theory Behind EM Algorithm

## Proof:

$$\begin{aligned}\mathcal{L}(q, \theta) + \text{KL}(q||p) &= \sum_Z q(Z) \ln \left\{ \frac{p(X, Z|\theta)}{q(Z)} \right\} - \sum_Z q(Z) \ln \left\{ \frac{p(Z|X, \theta)}{q(Z)} \right\} \\&= \sum_Z q(Z) \{ \underbrace{\ln p(X, Z|\theta)}_{\ln p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}) + \ln p(\mathbf{X}|\boldsymbol{\theta})} - \cancel{\ln q(Z)} - \ln p(Z|X, \theta) + \cancel{\ln q(Z)} \} \\&= \sum_Z q(Z) \ln p(X|\theta) \\&= \ln p(X|\theta)\end{aligned}$$

# The Theory Behind EM Algorithm

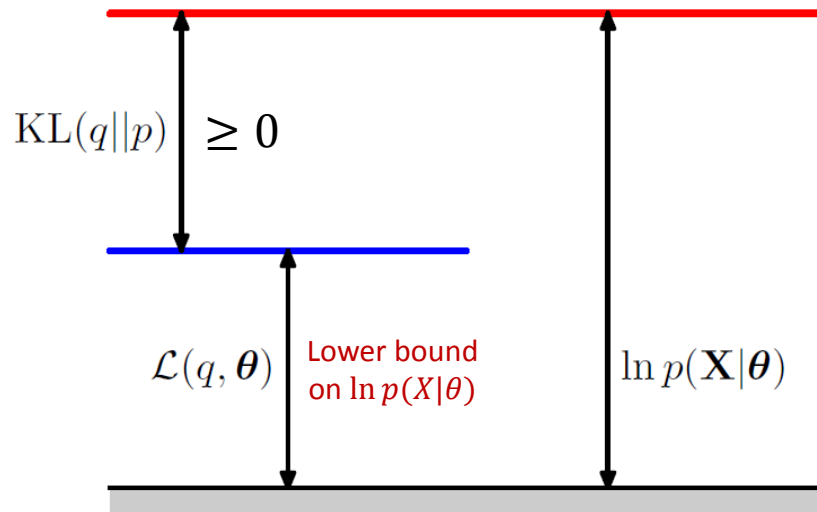


Illustration of the decomposition given by:

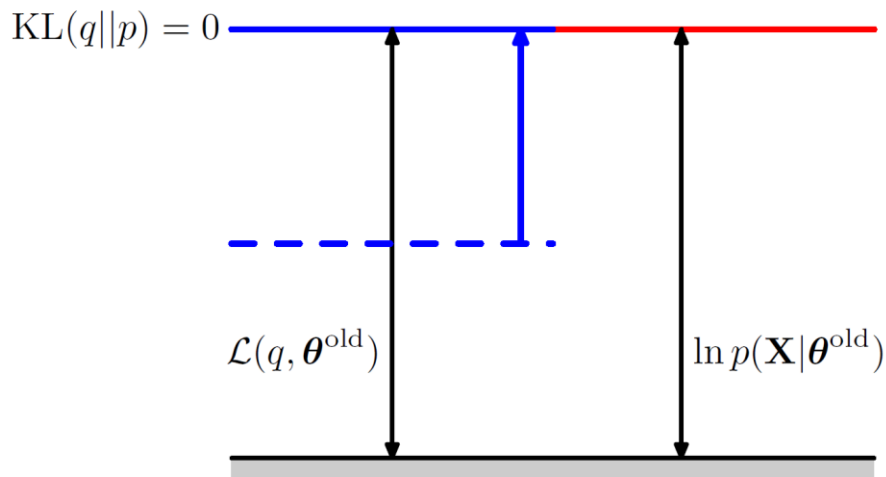
$$\ln p(\mathbf{X}|\theta) = \mathcal{L}(q, \theta) + KL(q||p)$$

which holds for any choice of  $q(\mathbf{Z})$ .

- Because the **Kullback-Leibler divergence** satisfies  $KL(q||p) \geq 0$ , we see that the quantity  $\mathcal{L}(q, \theta)$  is a **lower bound** on the log-likelihood function  $\ln p(X|\theta)$ .

# The Theory Behind EM Algorithm

## Illustration of the E-Step:



Lower bound  $\mathcal{L}(q, \theta)$  is maximized by choosing  $q(Z) = p(Z|X, \theta^{old})$ !

**Proof:**

$$\begin{aligned} \text{KL}(q||p) &= - \sum_{\mathbf{Z}} q(\mathbf{Z}) \ln \left\{ \frac{p(\mathbf{Z}|\mathbf{X}, \theta)}{p(\mathbf{Z}|\mathbf{X}, \theta)} \right\} \\ &= 0 \end{aligned}$$

$\Rightarrow \mathcal{L}(q, \theta)$  must be at its maximum since

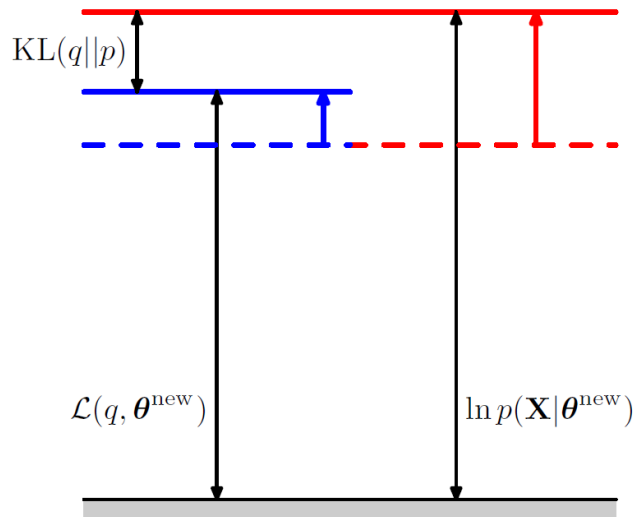
$$\ln p(\mathbf{X}|\theta) = \mathcal{L}(q, \theta) + \text{KL}(q||p)$$

- Equivalence of **expectation** under the latent variable distribution:

$$\begin{aligned} \mathcal{L}(q, \theta) &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \theta^{old}) \ln p(\mathbf{X}, \mathbf{Z}|\theta) - \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \theta^{old}) \ln p(\mathbf{Z}|\mathbf{X}, \theta^{old}) \\ &= \mathcal{Q}(\theta, \theta^{old}) + \text{const} \end{aligned}$$

# The Theory Behind EM Algorithm

## Illustration of the M-Step:

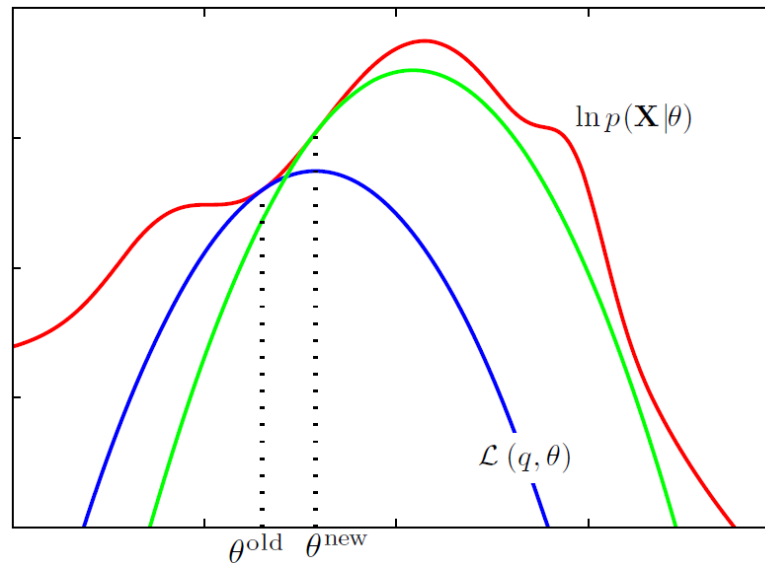


$$\mathcal{L}(q, \theta^{new}) = \sum_Z p(Z|X, \theta^{old}) \ln p(X, Z|\theta^{new}) + \text{const}$$

$$\text{KL}(q||p) = - \sum_Z p(Z|X, \theta^{old}) \ln \left\{ \frac{p(Z|X, \theta^{new})}{p(Z|X, \theta^{old})} \right\}$$

- The distribution  $q(Z)$  is held fixed and the **lower bound  $\mathcal{L}(q, \theta)$  is maximized** w.r.t  $\theta$  to give a revised value  $\theta^{new}$ .
- Because the KL divergence is nonnegative, this causes the log-likelihood  $\ln p(X|\theta)$  **to increase** by at least as much as the lower bound does.

# The Theory Behind EM Algorithm



- **E-step:** we compute the **convex lower bound** given the old parameters  $\theta^{old}$  (blue curve).
- **M-step:** we **maximize this lower bound** to get new parameters  $\theta^{new}$ .
- This is **repeated** (green curve) until convergence.



# Summary

- We have looked at how to:
  1. Use the non-probabilistic **k-mean algorithm** to solve the clustering problem.
  2. Describe the **Gaussian-mixture model**.
  3. Apply the **Expectation-Maximization algorithm** for estimation of both the unknown parameters and latent variables.