

# EECS 545: Machine Learning

## Lecture 15. Unsupervised Learning: Clustering & EM

Honglak Lee

3/10/2025



# Logistics

- HW4 due 03/18/25
- Project progress report due 03/14/25
- See [project information document](#) for detailed info

## **Additional tips: Recommended content in the report**

- Dataset and/or data collection
- Pre-processing
- Feature representation
- Pipeline of the method
- Initial results
  - Tables
  - Plots
  - Figures (e.g., qualitative examples)
  - Visualization of results/models
- Hyperparameters and how they were tuned
  - make it clear how you performed cross-validation (hold-out CV, or K-fold CV, etc.)
  - For hold-out CV, specify whether you separated training, validation, and test set
  - For K-fold CV, specify K value.
- Error and failure case analysis
- Summary and future work

In your project progress report, please include the following:

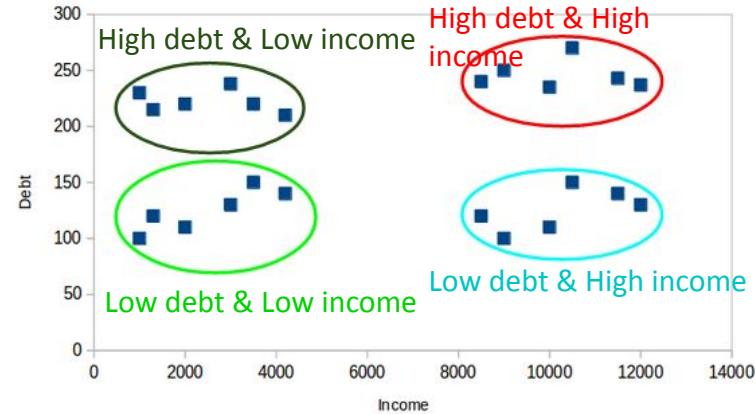
- Project Title
  - Please put a specific and informative title (that describes what your project is about)
  - For example, "545 Project Progress Report" is NOT a specific and informative title
- Team members (\* Please put these at the beginning of your report)
  - Please specify the full name and email addresses in [uniqueusername@umich.edu](mailto:uniqueusername@umich.edu)
  - You can specify the names and emails as the following format
    - author1, author2, author3, ..., authorN
    - {email1, email2, ..., emailN}@umich.edu
- Abstract (1 paragraph)
- Section 1. Introduction:
  - Problem description and motivation.
  - Why do you want to solve this problem?
  - What's the impact if you can solve this problem?
- Section 2. Proposed method:
  - How are you going to solve this problem?
  - Why should the proposed method work?
  - Provide explanations/rationale of why you chose this specific method
  - Provide technical details of your approach if you are proposing a novel method.
  - Description of the pipeline. Including a figure (e.g., block diagram) that explains the pipeline will be very helpful
- Section 3. Related work:
  - What are existing methods?
  - What are the state-of-the-art methods for this problem?
  - How is your approach similar to other existing work?
  - How is your approach different from the related work?
- Section 4. (Preliminary) Experimental results:
  - Milestones achieved so far (add all relevant experimental results).
  - How do these results support your claim?
- Section 5. Future milestones: Dates and sub-goals (please set sub-goals on a weekly basis so that they can be done in a week)
- Section 6. Conclusion: Summary of your progress and your final expected goal (what do you expect to achieve or demonstrate for the final project?)
- Author Contributions
  - Please see the guideline "Contribution of each team member" below
- References

# Outline

- Unsupervised Learning: Clustering
  - K-means clustering
- Expectation Maximization
  - General Recipe of EM
  - Gaussian Mixtures

# Clustering

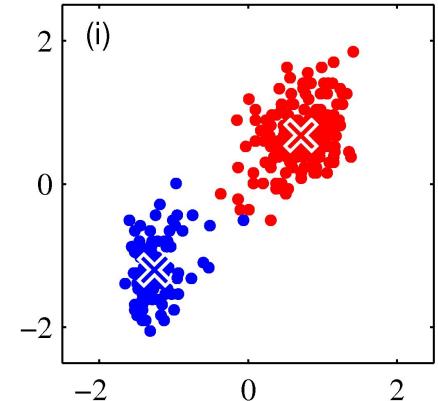
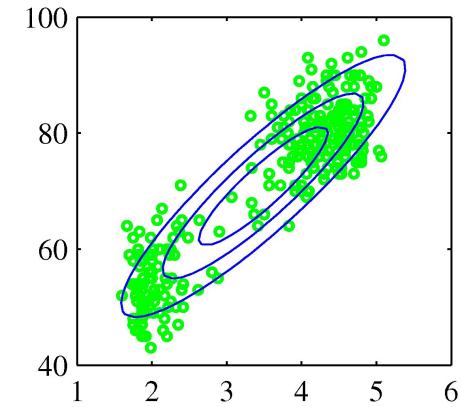
- A motivating example:  
**Customer segmentation**
  - Group customers so that it can be helpful for decision making (e.g., credit card request approval) or marketing (e.g., promotion of products)
  - Customer information (e.g., income, debt, age, etc.) can be used for input features.
- A type of unsupervised learning
  - No label/target needed to learn the grouping!



# K-Means Clustering

# The K-Means Algorithm

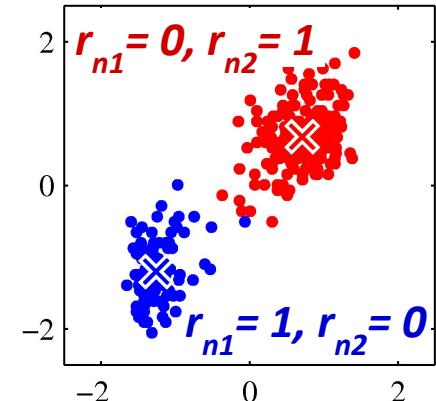
- Given *unlabeled* data  
 $\{\mathbf{x}^{(n)}\}$  ( $n = 1, \dots, N$ ),
- and believing it belongs in  $K$  clusters  
(say  $K = 2$  here),
- How do we find the clusters?
  - What would be the objective function?



# The K-Means Algorithm

- Use indicator variables  $r_{nk} \in \{0, 1\}$ :
  - $r_{nk} = 1$  if  $\mathbf{x}^{(n)}$  is in cluster  $k$
  - and  $r_{nk} = 0$  for all  $j \neq k$
- Find cluster centers  $\mu_k$  and assignments  $r_{nk}$  to minimize the distortion measure  $J$ :
  - Sum of squared distance of points from the center of its own cluster (*Intra-cluster variation*):

$$J = \sum_{k=1}^K \sum_{n=1}^N r_{nk} \|\mathbf{x}^{(n)} - \mu_k\|^2 \quad \mu_k = \frac{1}{N_k} \sum_{n: \mathbf{x}^{(n)} \in \text{cluster } k} \mathbf{x}^{(n)} = \frac{\sum_{n=1}^N r_{nk} \mathbf{x}^{(n)}}{\sum_{n=1}^N r_{nk}}$$



# The K-Means Algorithm

- Initialize the cluster centers (centroids)
- Repeat the following update until convergence:
  1.  $r := \arg \min_r J(r, \mu)$
  2.  $\mu := \arg \min_\mu J(r, \mu)$

where  $J = \sum_{k=1}^K \sum_{n=1}^N r_{nk} \|\mathbf{x}^{(n)} - \mu_k\|^2$

$$\mu_k = \frac{1}{N_k} \sum_{n: \mathbf{x}^{(n)} \in \text{cluster } k} \mathbf{x}^{(n)} = \frac{\sum_{n=1}^N r_{nk} \mathbf{x}^{(n)}}{\sum_{n=1}^N r_{nk}}$$

# The K-Means Algorithm

- Initialize the cluster centers.
- Repeat until convergence:

– **Cluster assignment (“E-Step”):**

assign each point to closest center.

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

– **Parameter update (“M-Step”):** update the centers

$$\mu_k = \frac{\sum_n r_{nk} \mathbf{x}^{(n)}}{\sum_n r_{nk}}$$

Q. Verify this

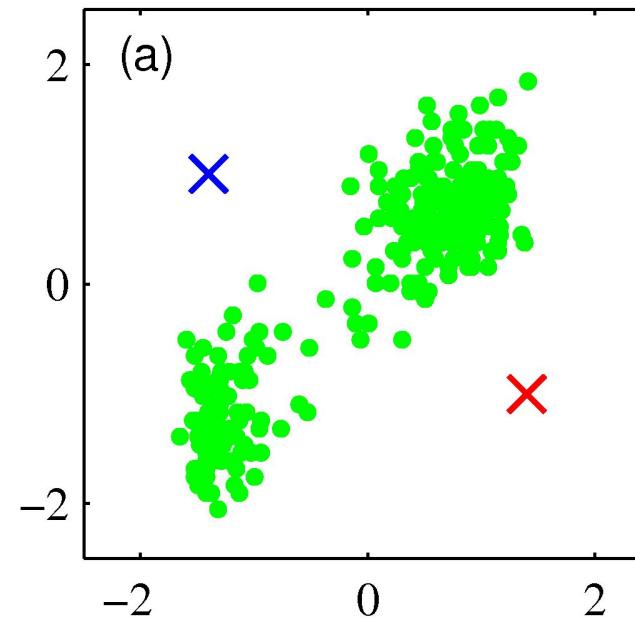
Note: E, M stands for:

- E: Expectation
- M: Maximization

(We will revisit EM later.)

# K-Means Clustering

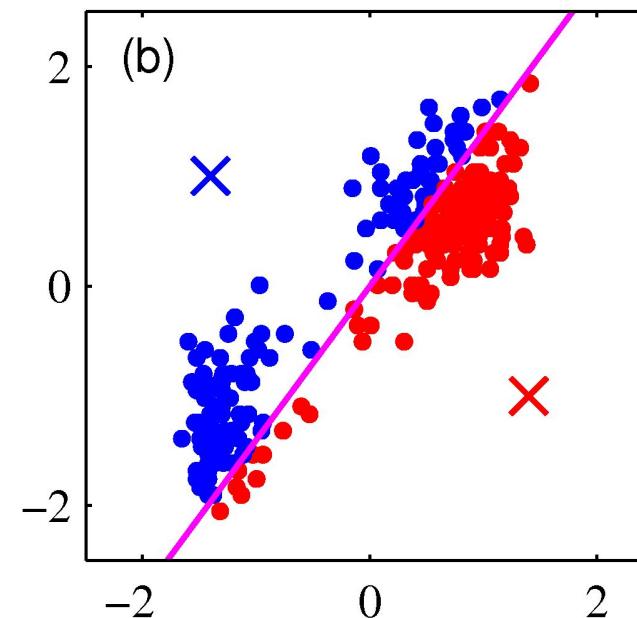
- Select K. Pick random centroids.
  - Here K=2.



# K-Means Clustering

## Cluster assignment Step (“E-Step”)

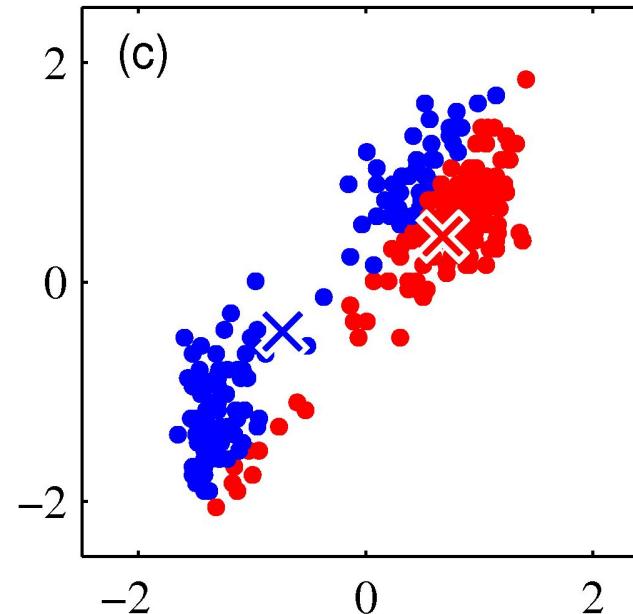
- Assign each point to the nearest center.



# K-Means Clustering

Update parameters (centroids) (“M-Step”)

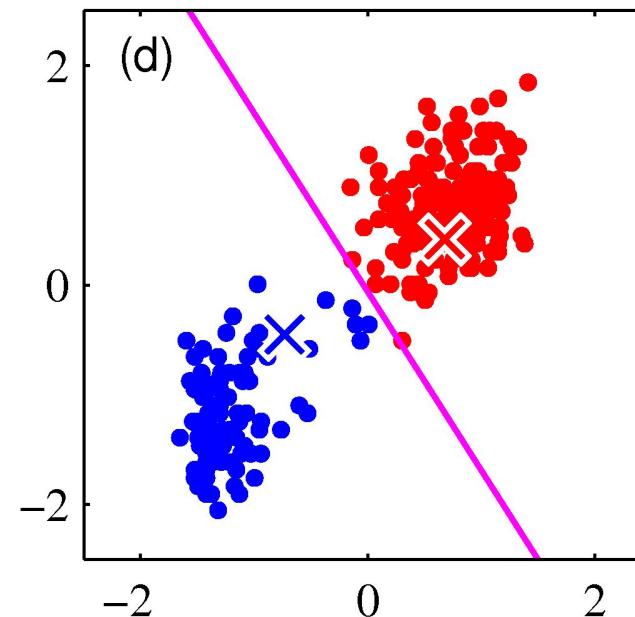
- Compute new centers for each cluster.



# K-Means Clustering

Cluster assignment Step (“E-Step”) again

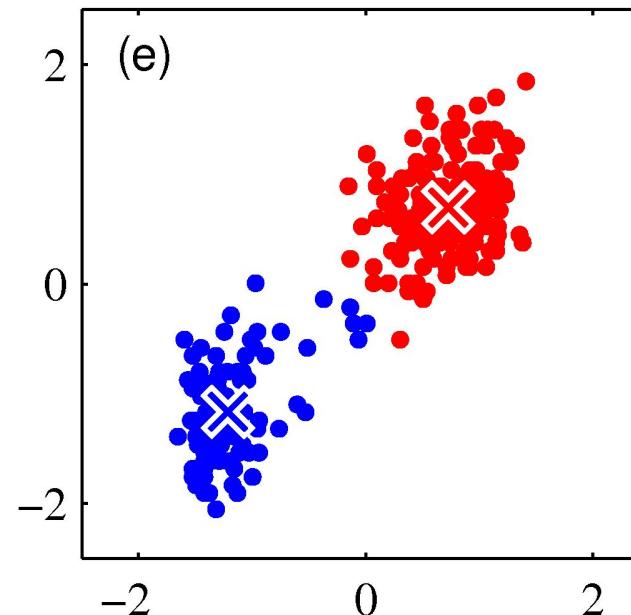
- Re-assign points to the now-nearest center.



# K-Means Clustering

Update parameters (centroids) (“M-Step”) again

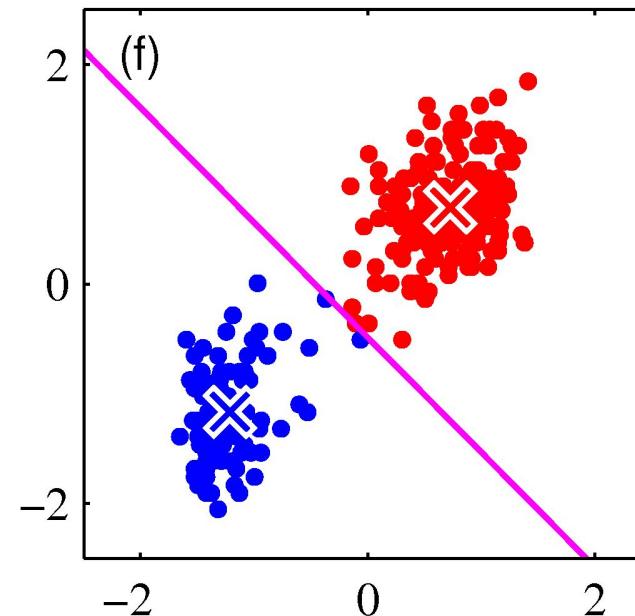
- Compute centers for the new clusters.



# K-Means Clustering

Another Cluster assignment Step (“E-Step”)

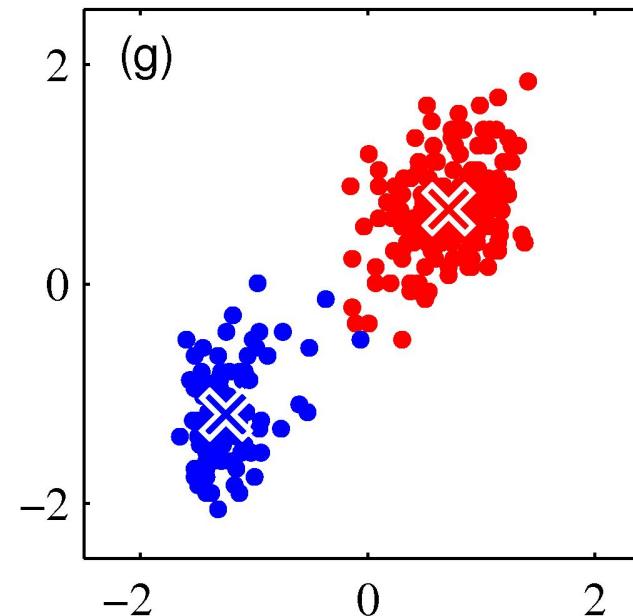
- Reassign the points to centers.



# K-Means Clustering

Update parameters (centroids) (“M-Step”) again

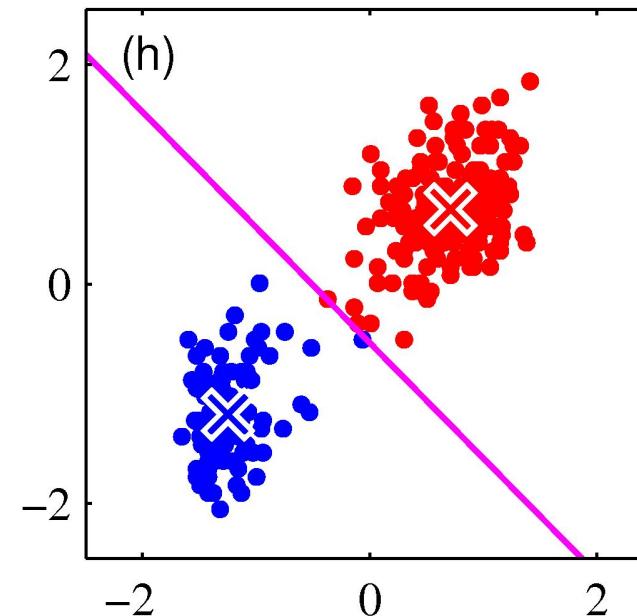
- New centers.



# K-Means Clustering

Another Cluster assignment Step (“E-Step”)

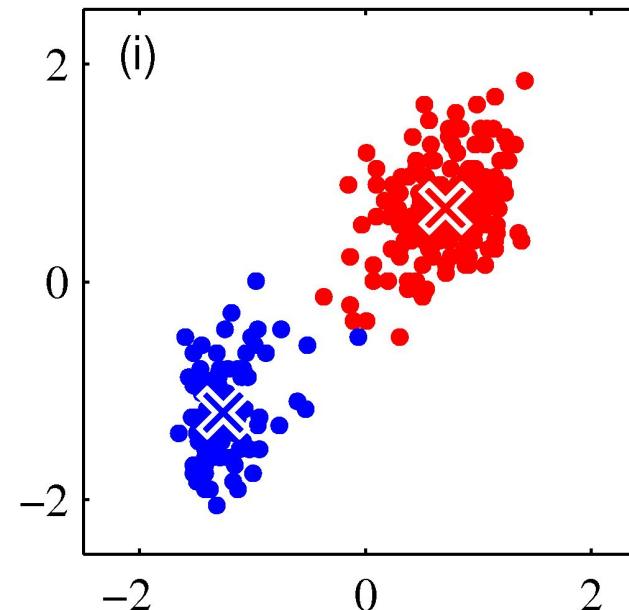
- New cluster assignments.



# K-Means Clustering

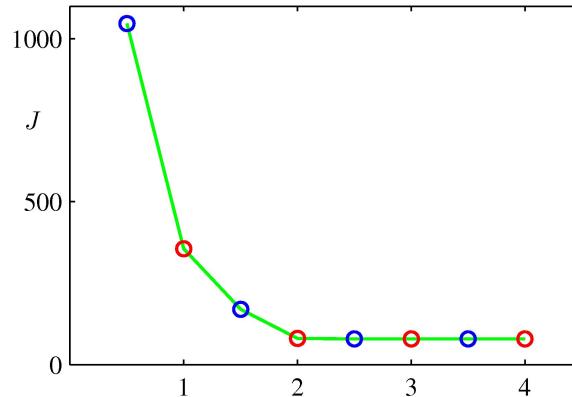
Update parameters (centroids) (“M-Step”) again

- The cluster centers have stopped changing.



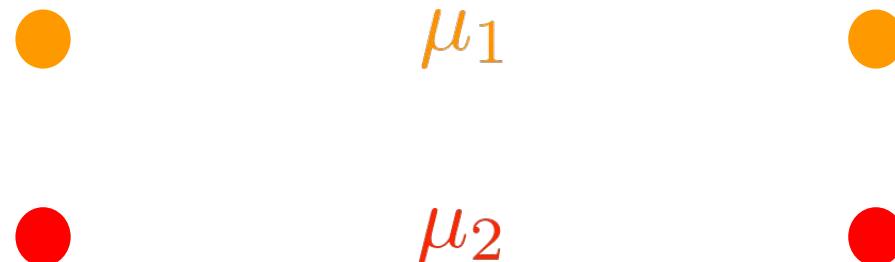
# Convergence

- The objective function of K-means decreases monotonically as the K-means procedure reduces  $J$  in both E-step and M-step.
- Convergence is relatively quick, in steps.
  - blue circles after E-step: assign each point to a cluster
  - red circles after M-step: recompute the cluster centers
  - However, all those distance computations are expensive.



# Convergence

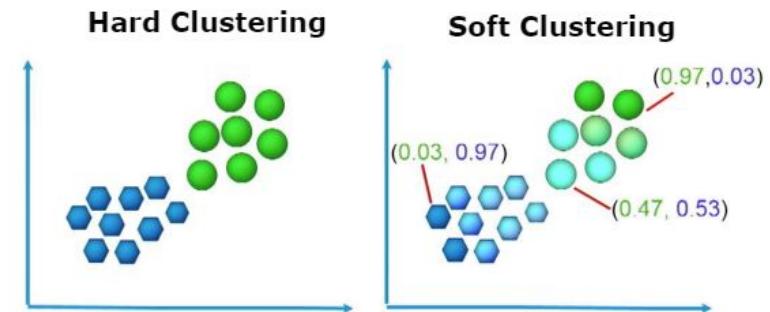
- No guarantee that we found the globally optimal solution. The quality of local optimum depends on the initial values.
- The following clustering is a stable local optima



# Gaussian Mixtures and Expectation-Maximization

# Hard and Soft Clusters

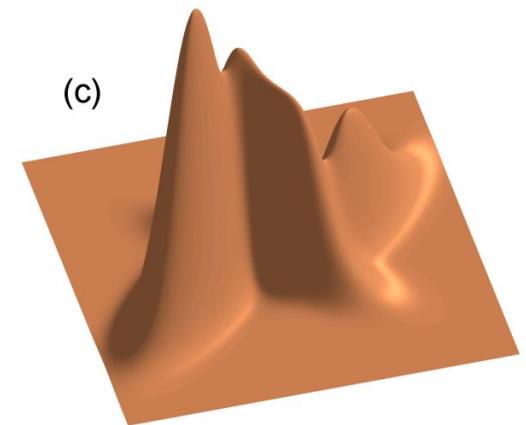
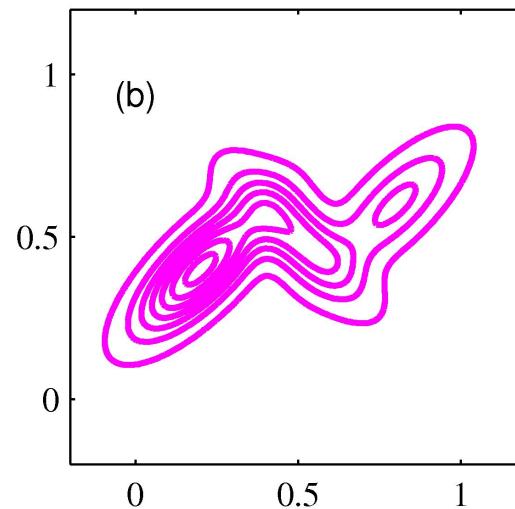
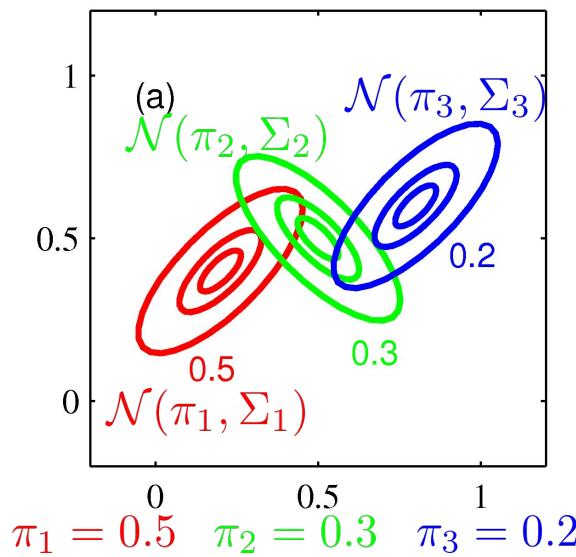
- K-Means uses ***hard clustering assignment***.
  - A point belongs to exactly one cluster.
- Mixture of Gaussians uses ***soft clustering***.
  - A point could be explained by more than one cluster.
  - Different clusters take different levels of “responsibility” (posterior probability) for that point.
  - (It was actually generated by only one cluster, but we don’t know which one; we assign a probability)



# Mixtures of Gaussians

- Mixtures of Gaussians make it possible to describe much richer distributions.

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



# Mixtures of Gaussians

- Note the mixing coefficients in

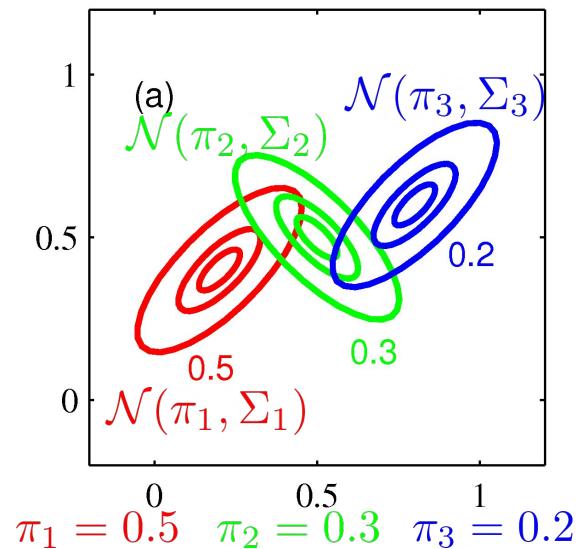
$$p(\mathbf{x}) = \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x} \mid \mu_k, \Sigma_k) \quad \sum_{k=1}^K \pi_k = 1$$

- Let  $\mathbf{z}$  in  $\{0,1\}^K$  be a 1-of- $K$  random variable;

$$p(z_k = 1) = \pi_k \quad p(\mathbf{z}) = \prod_{k=1}^K \pi_k^{z_k}$$

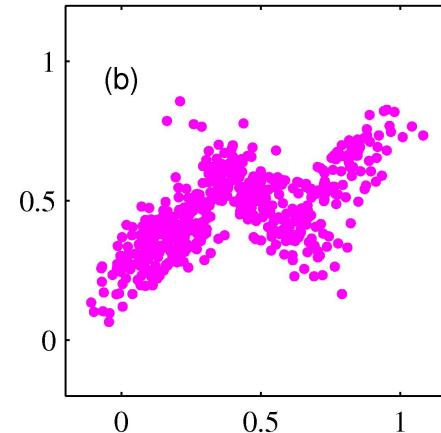
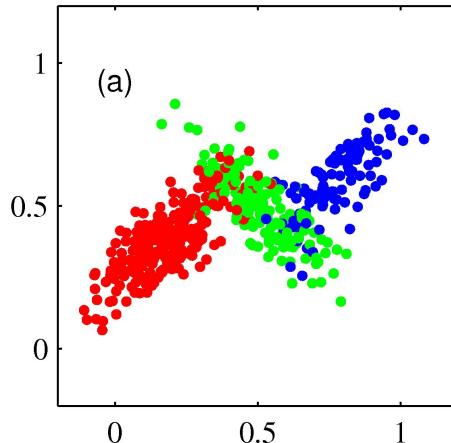
$$p(\mathbf{x} \mid z_k = 1) = \mathcal{N}(\mathbf{x} \mid \mu_k, \Sigma_k)$$

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



# Mixtures of Gaussians

- To generate samples from a Gaussian mixture distribution  $p(\mathbf{x})$ , use  $p(\mathbf{x}, \mathbf{z}) = p(\mathbf{z}) p(\mathbf{x} \mid \mathbf{z})$ :
  - Select a value  $\mathbf{z}$  from the marginal  $p(\mathbf{z})$ ;
  - Then select a value  $\mathbf{x}$  from  $p(\mathbf{x} \mid \mathbf{z})$  for that  $\mathbf{z}$ .



# Latent Variables

- A system with observed data  $\mathbf{X}$ 
  - may be far easier to understand in terms of additional variables  $\mathbf{Z}$  corresponding to  $\mathbf{X}$ ,
  - but they are not observed (**latent**).
- For example, in a mixture of Gaussians,
  - For a single sample  $\mathbf{x}$ , the latent variable  $\mathbf{z}$  specifies which Gaussian generated the sample  $\mathbf{x}$ .
  - The *responsibility* is the **posterior**  $p(\mathbf{z}|\mathbf{x})$ .

# Latent Variables

- A system with observed variables  $\mathbf{X}$ 
  - may be easier to understand with latent variables  $\mathbf{Z}$ , but they are not observed (**latent**).
- Notations:
  - We denote the set of all observed data by  $\mathbf{X}$ , in which the  $n^{\text{th}}$  row represents  $\mathbf{x}_n^T$ .
  - Similarly we denote the set of all latent variables by  $\mathbf{Z}$ , with a corresponding row  $\mathbf{z}_n^T$ .
  - Note: we use lowercase symbol for single sample ( $\mathbf{x}$ ), matrix symbol for all data ( $\mathbf{X}$ ).

# Learning a Latent Variable Model

- We find model parameters by maximizing the log-likelihood of observed data  $\log p(\mathbf{X} \mid \theta)$ .
- If we had complete data  $\{\mathbf{X}, \mathbf{Z}\}$ , we could easily maximize the *complete* data likelihood  $p(\mathbf{X}, \mathbf{Z} \mid \theta)$ .
- Unfortunately, with *incomplete* data ( $\mathbf{X}$  only), we must marginalize over  $\mathbf{Z}$ , so

$$\log p(\mathbf{X} \mid \theta) = \log \left[ \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z} \mid \theta) \right]$$

(the sum inside the log makes it hard.)

# The EM Algorithm in General

- Expectation-Maximization (EM) is a general recipe for finding the parameters that maximize the (log-) likelihood of *latent* variable models
- To find a parameter  $\theta$  that maximizes the likelihood  $p(\mathbf{X} | \theta) = \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z} | \theta)$ , the EM algorithm first introduces a new (variable) distribution  $q(\mathbf{Z})$  over the latent variables.
- A lower bound  $\mathcal{L}(q, \theta)$  for the log-likelihood  $\log p(\mathbf{X} | \theta)$  is established based on  $q$  and  $\theta$ .
- Then,  $q(\mathbf{Z})$  and  $\theta$  are alternatingly updated (keeping the other fixed) so that  $\mathcal{L}(q, \theta)$  is maximized (similar to coordinate ascent) until convergence.

# The EM Algorithm in General

- Our goal is to maximize  $p(\mathbf{X} \mid \theta) = \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z} \mid \theta)$
- For *any distribution*  $q(\mathbf{Z})$  over latent variables:

$$\begin{aligned}\log p(\mathbf{X} \mid \theta) &= \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X} \mid \theta) \\ &= \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z} \mid \theta)}{p(\mathbf{Z} \mid \mathbf{X}, \theta)} \\ &= \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z} \mid \theta)}{q(\mathbf{Z})} \frac{q(\mathbf{Z})}{p(\mathbf{Z} \mid \mathbf{X}, \theta)} \\ &= \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z} \mid \theta)}{q(\mathbf{Z})} + \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{q(\mathbf{Z})}{p(\mathbf{Z} \mid \mathbf{X}, \theta)} \\ &= \mathcal{L}(q, \theta) + KL(q(\mathbf{Z}) \| p(\mathbf{Z} \mid \mathbf{X}, \theta)) \\ &\geq \mathcal{L}(q, \theta)\end{aligned}$$

# Note: KL Divergence

Let  $p$  and  $q$  be probability distributions of a random variable  $Z$ .

$$\begin{aligned} KL(q \parallel p) &= \mathbb{E}_{z \sim q(z)} \left[ \log \frac{q(z)}{p(z)} \right] = \sum_z q(z) \log \frac{q(z)}{p(z)} \\ &= -\sum_z q(z) \log p(z) + \sum_z q(z) \log q(z) \end{aligned}$$

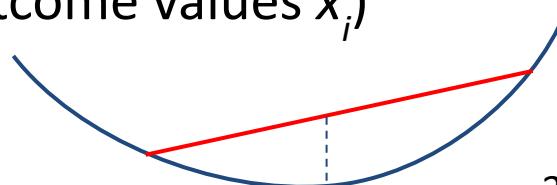
This is one way to measure the **dissimilarity** of two probability distributions.

Remarks: (note: the first can be proved using Jensen's inequality)

- $KL(q \parallel p) \geq 0$ , with equality iff  $p = q$ .
- $KL(q \parallel p) \neq KL(p \parallel q)$  in general

# Background note: Jensen's Inequality

- If  $f$  is convex, then for any  $\theta_i$  s.t.  $0 \leq \theta_i \leq 1$  ( $\forall i$ ),  
$$\theta_1 + \theta_2 + \cdots + \theta_k = 1$$
$$f(\theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_k x_k) \leq \theta_1 f(x_1) + \cdots + \theta_k f(x_k)$$
- It can be seen as a generalization of the definition of convex function:  
$$f \text{ is convex} \iff f(\theta x + (1 - \theta)y) \leq \theta f(x) + (1 - \theta)f(y) \text{ for all } 0 \leq \theta \leq 1$$
- Jensen's inequality can be written in expectation form  
(think of  $\theta_i$  as probability mass for different outcome values  $x_i$ )  
$$f(\mathbb{E}[x]) \leq \mathbb{E}[f(x)]$$



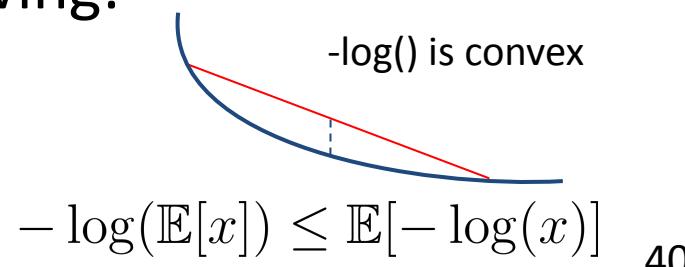
# Background note: Jensen's Inequality

- If  $f$  is convex, then for any  $\theta_i$  s.t.  $0 \leq \theta_i \leq 1$  ( $\forall i$ ),  
$$\theta_1 + \theta_2 + \cdots + \theta_k = 1$$
$$f(\theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_k x_k) \leq \theta_1 f(x_1) + \cdots + \theta_k f(x_k)$$
- Jensen's inequality can be written in expectation form

$$f(\mathbb{E}[x]) \leq \mathbb{E}[f(x)]$$

- To show  $KL(q \| p)$  is non-negative for any  $p, q$ , plug in  $f(\dots) = -\log(\dots)$  and the following:

$$\theta_i = q(z), x_i = \frac{p(z)}{q(z)}$$

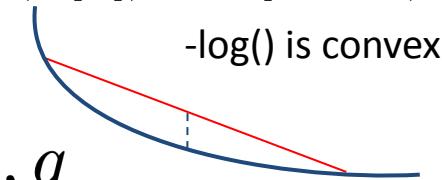


# Non-negativity of KL divergence

- Jensen's inequality can be written in expectation form for a convex function  $f$ 

$$f(\mathbb{E}[x]) \leq \mathbb{E}[f(x)]$$
- To show  $KL(q \| p)$  is non-negative for any  $p, q$ , plug in  $f(\dots) = -\log(\dots)$  and the following:  $\theta_i = q(z), x_i = \frac{p(z)}{q(z)}$

$$-\log(\mathbb{E}[x]) \leq \mathbb{E}[-\log(x)]$$



$$\begin{aligned} KL(q||p) &= \sum_z q(z) \log\left(\frac{q(z)}{p(z)}\right) \\ &= \sum_z q(z) \left( -\log\left(\frac{p(z)}{q(z)}\right) \right) \\ &\geq -\log \left( \underbrace{\sum_z q(z) \frac{p(z)}{q(z)}}_{=\sum_z p(z)=1} \right) \\ &= 0 \end{aligned}$$

Jensen's inequality for  $-\log()$ :

$$-\log(\mathbb{E}[x]) \leq \mathbb{E}[-\log(x)]$$

i.e., plugin

$$-\log\left(\sum_i \theta_i x_i\right) \leq \sum_i \theta_i (-\log(x_i))$$

with  $\theta_i = q(z), x_i = \frac{p(z)}{q(z)}$

# The EM Algorithm in a nutshell

- We have shown that: [variational lower bound]

$$\begin{aligned}\log p(\mathbf{X} \mid \theta) &= \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z} \mid \theta)}{q(\mathbf{Z})} + \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{q(\mathbf{Z})}{p(\mathbf{Z} \mid \mathbf{X}, \theta)} \\ &= \mathcal{L}(q, \theta) + KL(q(\mathbf{Z}) \parallel p(\mathbf{Z} \mid \mathbf{X}, \theta)) \\ &\geq \mathcal{L}(q, \theta) \quad \text{Evidence Lower bound (ELBO) or variational lower bound}\end{aligned}$$

with equality holding if and only if  $q(\mathbf{Z}) = p(\mathbf{Z} \mid \mathbf{X}, \theta)$

- **EM algorithm:**

\* E: expectation  
\* M: maximization

Repeat alternating optimization until convergence:

- E-step: for fixed  $\theta$ , find  $q$  that maximizes  $\mathcal{L}(q, \theta)$
- M-step: for fixed  $q$ , find  $\theta$  that maximizes  $\mathcal{L}(q, \theta)$

# The EM Algorithm: E-step

- We have shown that: [variational lower bound]

$$\begin{aligned}\log p(\mathbf{X} \mid \theta) &= \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z} \mid \theta)}{q(\mathbf{Z})} + \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{q(\mathbf{Z})}{p(\mathbf{Z} \mid \mathbf{X}, \theta)} \\ &= \mathcal{L}(q, \theta) + KL(q(\mathbf{Z}) \parallel p(\mathbf{Z} \mid \mathbf{X}, \theta)) \\ &\geq \mathcal{L}(q, \theta) \quad \text{Evidence Lower bound (ELBO) or variational lower bound}\end{aligned}$$

with equality holding if and only if  $q(\mathbf{Z}) = p(\mathbf{Z} \mid \mathbf{X}, \theta)$

- **(E-step)** For a fixed  $\theta$ , which  $q$  maximizes  $\mathcal{L}(q, \theta)$ ?
- $\Rightarrow p(\mathbf{Z} \mid \mathbf{X}, \theta)$  , because all other  $q$  would make  $\mathcal{L}(q, \theta)$  strictly less than  $\log p(\mathbf{X} \mid \theta)$

# The EM Algorithm: M-step

- We also note that for a fixed  $q$ , the  $\mathcal{L}(q, \theta)$  term can be decomposed into two terms:

$$\begin{aligned}\mathcal{L}(q, \theta) &= \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z} \mid \theta)}{q(\mathbf{Z})} \\ &= \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z} \mid \theta) - \sum_{\mathbf{Z}} q(\mathbf{Z}) \log q(\mathbf{Z})\end{aligned}$$

- (1) A weighted sum of  $\log p(\mathbf{X}, \mathbf{Z} \mid \theta)$ .  
This is tractable and can be optimized w.r.t  $\theta$
- (2) Entropy of  $q(\mathbf{Z})$  which is independent of  $\theta$  since  $q$  is fixed.
- **(M-step)** Thus, when  $q$  is fixed, we can find  $\theta$  that maximizes  $\mathcal{L}(q, \theta)$ .

# The EM Algorithm: summary

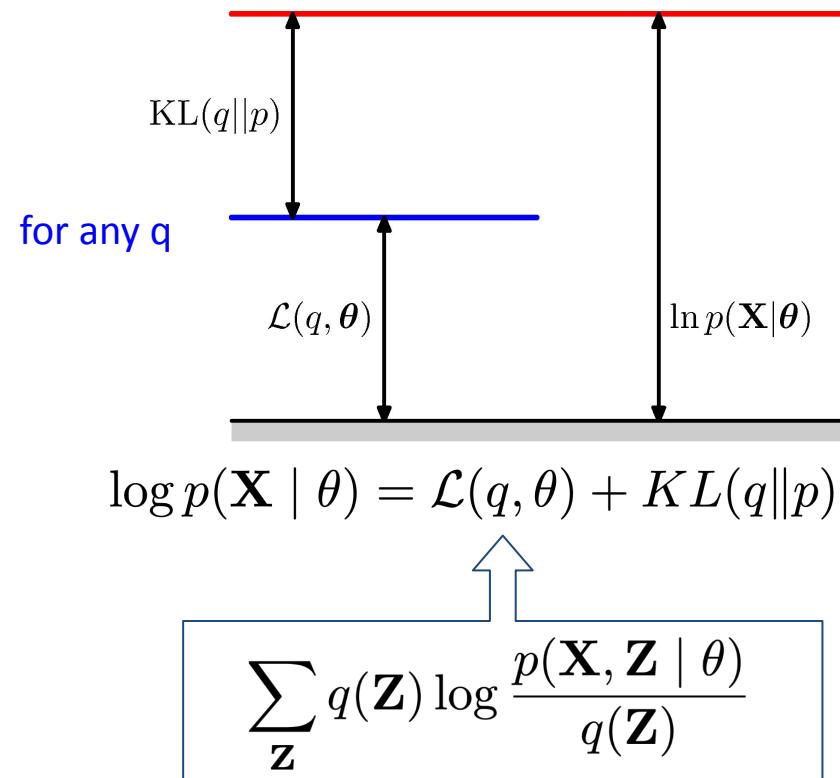
- Initialize parameters  $\theta$  randomly
- Repeat until convergence:  
(optimize  $\mathcal{L}(q, \theta)$  w.r.t.  $q$  and  $\theta$  alternatingly.)
  - “E-step”: Set  $q(\mathbf{Z}) = p(\mathbf{Z} \mid \mathbf{X}, \theta)$  compute posterior → optimal  $q(\mathbf{Z})$ !
  - “M-step”: Update  $\theta$  via the following maximization

$$\operatorname{argmax}_{\theta} \mathcal{L}(q, \theta) = \operatorname{argmax}_{\theta} \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z} | \theta)$$

use  $q(\mathbf{Z})$  as (factional) pseudo-counts and maximize the “data completion” log-likelihood

- Note we have assumed that  $p(\mathbf{Z} \mid \mathbf{X}, \theta)$  is tractable  
(i.e., find exact posterior  $p(\mathbf{Z} \mid \mathbf{X}, \theta)$ ). Q. What if it is not?

# Visualize the Decomposition

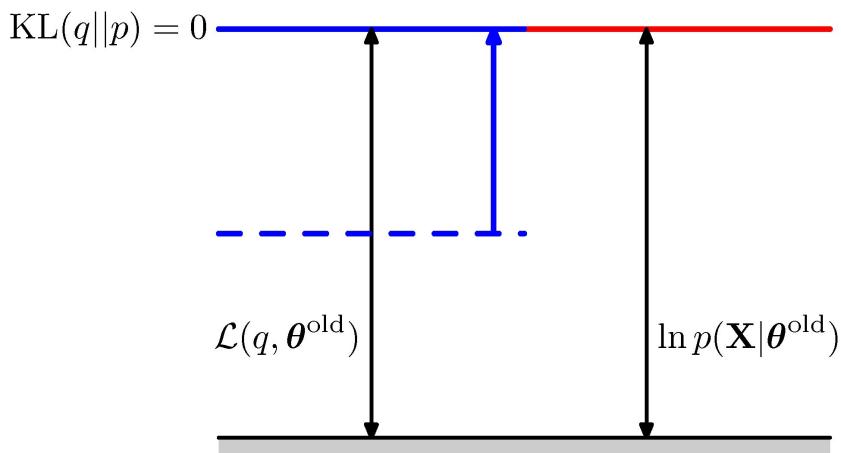


- Note:  $KL(q||p) \geq 0$ 
  - with equality only when  $q=p$ .
- Thus,  $\mathcal{L}(q, \theta)$  is a lower bound on  $\log p(\mathbf{X} | \theta)$  which EM tries to maximize.

# Visualize the E-Step

- E-step: for fixed  $\theta$ , find  $q$  that maximizes  $\mathcal{L}(q, \theta)$

for  $q(Z) = P(Z|X, \theta)$



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



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

- E-Step changes  $q(\mathbf{Z})$  to maximize  $\mathcal{L}(q, \theta)$

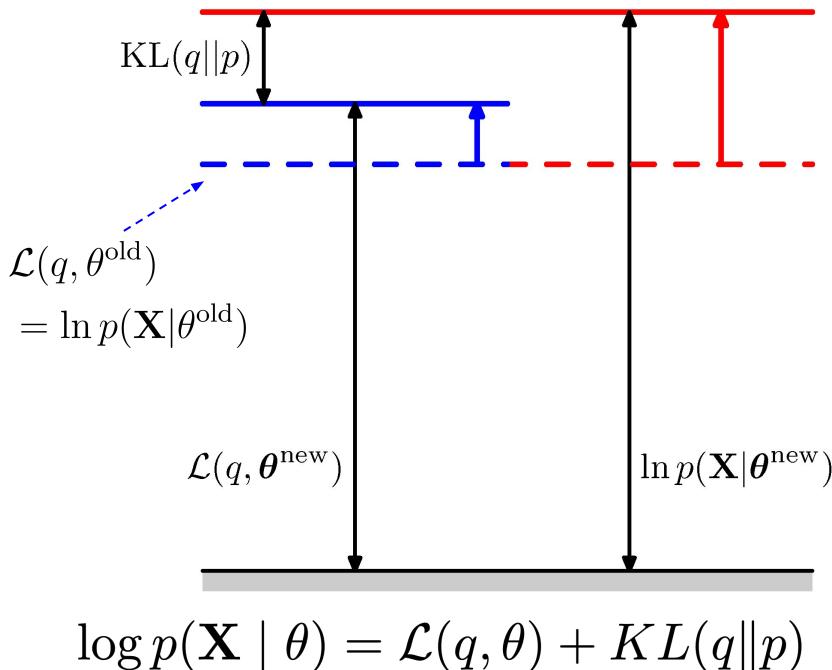
- So maximized when

$$KL(q||p) = 0$$

$$q(\mathbf{Z}) = p(\mathbf{Z} | \mathbf{X}, \theta)$$

# Visualize the M-Step

- M-step: for fixed  $q$ , find  $\theta$  that maximizes  $\mathcal{L}(q, \theta)$



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



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

- Holding  $q(\mathbf{Z})$  constant; increase  $\mathcal{L}(q, \theta)$

- Updating  $\theta$  will make  $\log p(\mathbf{X} | \theta)$  increase!

$$\ln p(\mathbf{X} | \theta^{\text{new}}) \geq \ln p(\mathbf{X} | \theta^{\text{old}})$$

- But now  $p \neq q$
- so  $KL(q \| p) > 0$

# Mixtures of Gaussians (recap)

$$p(z_k = 1) = \pi_k$$

$$p(\mathbf{z}) = \prod_{k=1}^K \pi_k^{z_k}$$

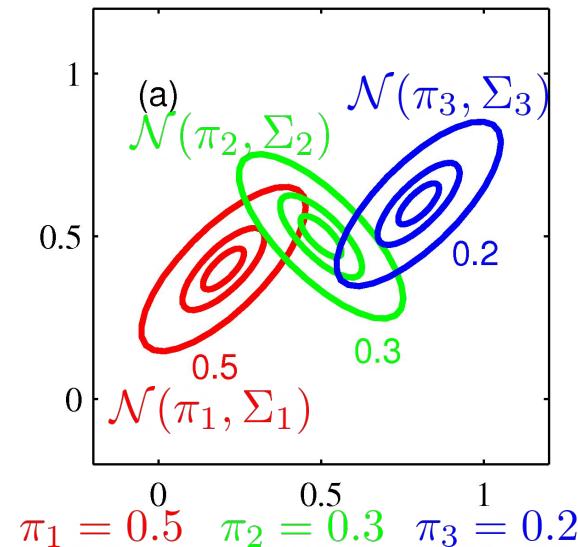
$$p(\mathbf{x} \mid z_k = 1) = \mathcal{N}(\mathbf{x} \mid \mu_k, \Sigma_k)$$

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

- Joint and marginal distributions:

$$p(\mathbf{x}, \mathbf{z}) = \pi_k \mathcal{N}(\mathbf{x} \mid \mu_k, \Sigma_k)$$

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



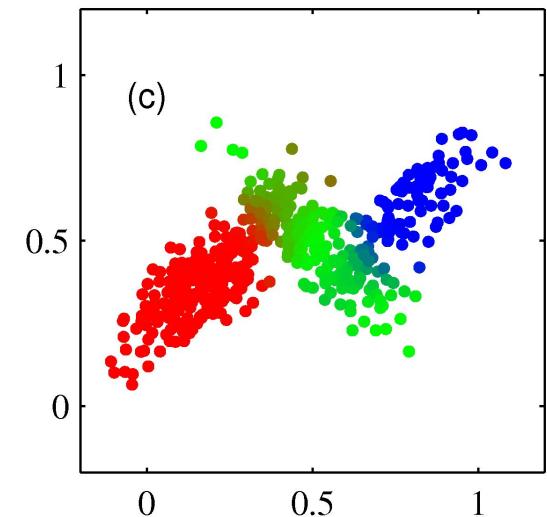
# EM for Gaussian Mixtures: E-step

- Initialize means, covariances, and mixing coefficients for the K Gaussians.
- E Step: Given the parameters, evaluate the responsibilities.

$$q(\mathbf{Z}) = p(\mathbf{Z} \mid \mathbf{X}, \theta)$$

$$\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(\mathbf{x}^{(n)} | \mu_k, \Sigma_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}^{(n)} | \mu_j, \Sigma_j)} = P(z_k = 1 | \mathbf{x}^{(n)})$$

(Recap) E-step: compute posterior  $\rightarrow$  optimal  $q(\mathbf{Z})$ !



(Hint: Use Bayes Rule)

# EM for Gaussian Mixtures: M-step

- M Step: Given the responsibilities, re-evaluate the parameters (note: this is very similar to GDA!).

$$\pi_k^{\text{new}} = \frac{N_k}{N} = \frac{\sum_n \gamma(z_{nk})}{N}$$

$$\operatorname{argmax}_{\theta} \sum_{\mathbf{Z}} q(\mathbf{Z}) \log p(\mathbf{X}, \mathbf{Z} | \theta)$$

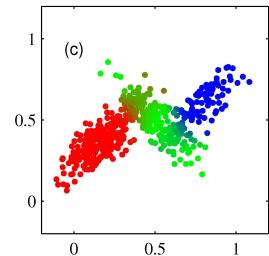
$$\mu_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) \mathbf{x}^{(n)}$$

M-step: use  $q(\mathbf{Z})$  as pseudo-counts and maximize the “data completion” log-likelihood (i.e. GDA with pseudo-counts!)

$$\Sigma_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) (\mathbf{x}^{(n)} - \mu_k^{\text{new}}) (\mathbf{x}^{(n)} - \mu_k^{\text{new}})^{\top}$$

- Stop when either parameters or log-likelihood converges.

# EM for Gaussian Mixtures (1-page summary)



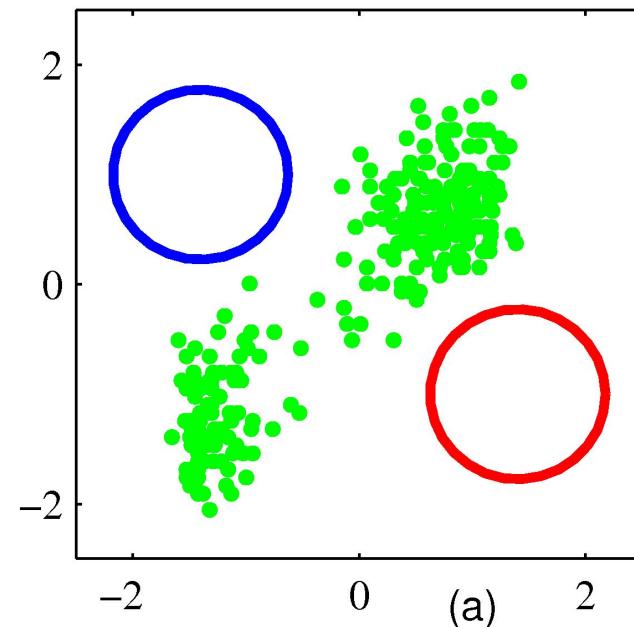
- Initialize parameters randomly  $\theta = \{\pi_k, \mu_k, \Sigma_k\}_{k=1}^K$
- Repeat until convergence (alternating optimization)
  - E Step: Given fixed parameters  $\theta$ , optimize  $q(\mathbf{Z})$ : find posterior probabilities of  $\mathbf{Z}$  given  $\mathbf{X}$ ,  $q(\mathbf{Z}) = p(\mathbf{Z} | \mathbf{X}, \theta)$

$$\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(\mathbf{x}^{(n)} | \mu_k, \Sigma_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}^{(n)} | \mu_j, \Sigma_j)} = P(z_k = 1 | \mathbf{x}^{(n)})$$

- M Step: Given fixed  $q(\mathbf{Z})$  (or  $\gamma(z_{nk})$ ), optimize  $\theta$  (estimate mean, covariance, etc.): equivalent to MLE for Gaussian Discriminant Analysis using  $q(\mathbf{Z})$  as pseudo-counts!

# EM Example

- Initialize parameters: means, covariances, and mixing coefficients.

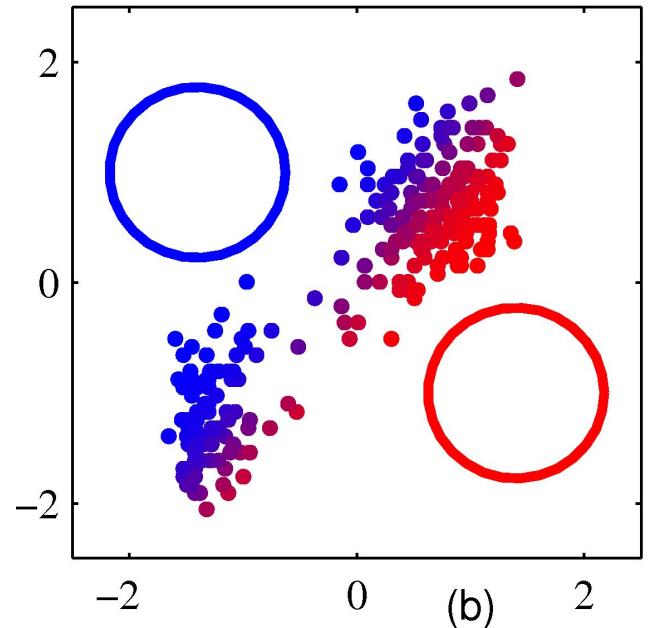


# EM Example

- First E Step

For each sample  $n$ , calculate:

$$\begin{aligned}\gamma(z_{nk}) &= \frac{\pi_k \mathcal{N}(\mathbf{x}^{(n)} | \mu_k, \Sigma_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(\mathbf{x}^{(n)} | \mu_j, \Sigma_j)} \\ &= P(z_k = 1 | \mathbf{x}^{(n)})\end{aligned}$$



# EM Example

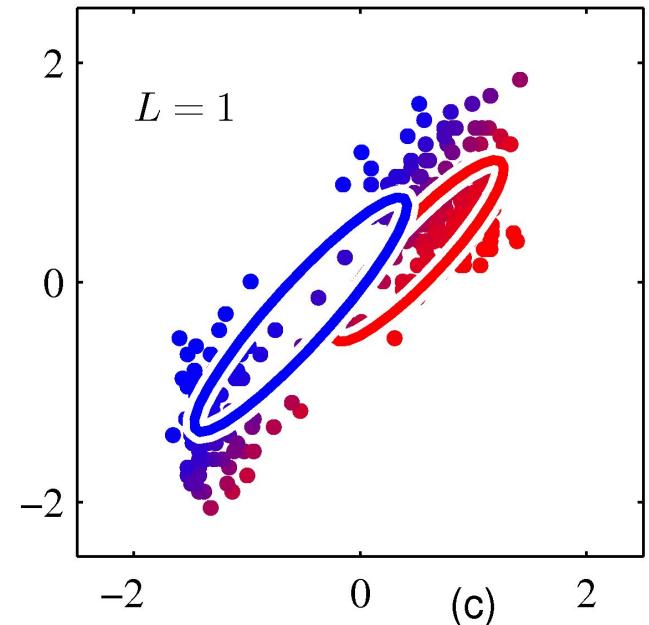
- First M Step

Update Gaussian parameters:

$$\pi_k^{\text{new}} = \frac{N_k}{N} = \frac{\sum_n \gamma(z_{nk})}{N}$$

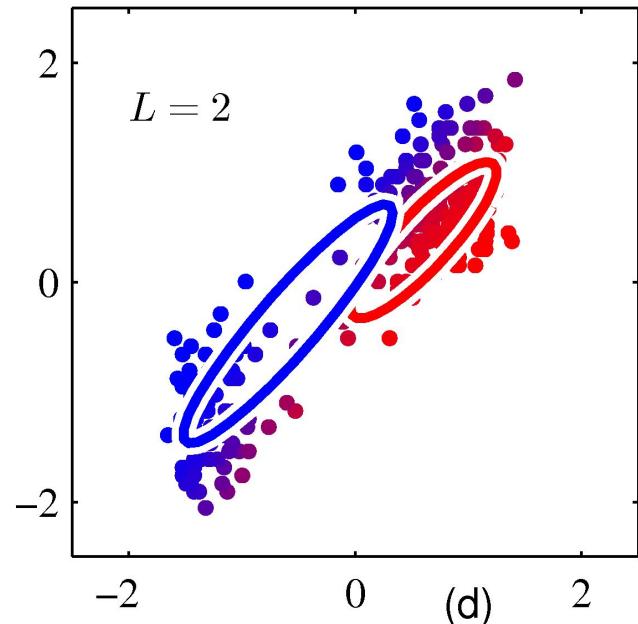
$$\mu_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) \mathbf{x}^{(n)}$$

$$\Sigma_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) (\mathbf{x}^{(n)} - \mu_k^{\text{new}}) (\mathbf{x}^{(n)} - \mu_k^{\text{new}})^{\top}$$



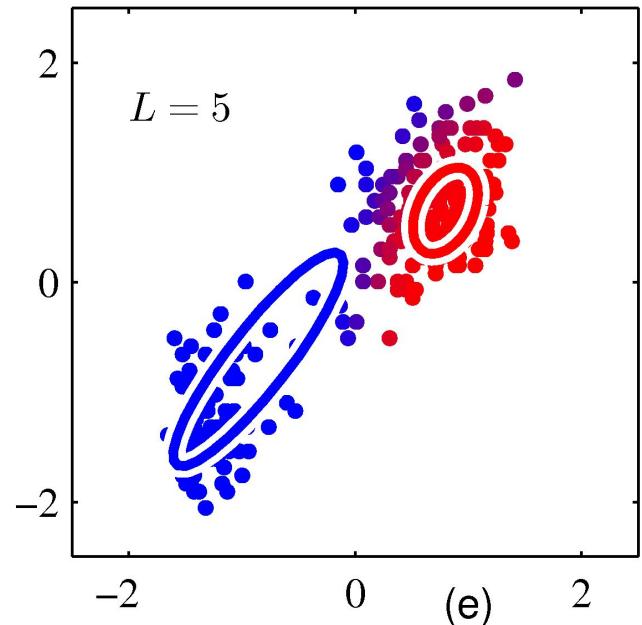
# EM Example

- Second E and M Steps



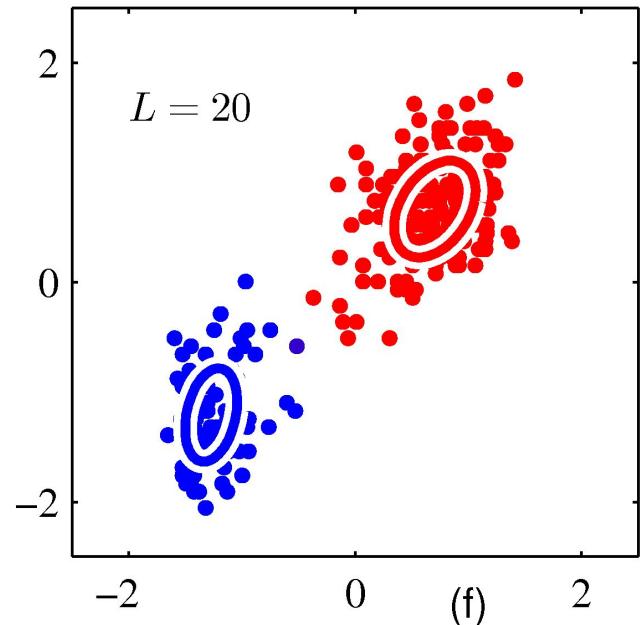
# EM Example

- Three more E-M cycles



# EM Example

- Fifteen E-M cycles later



# Relation to K-means

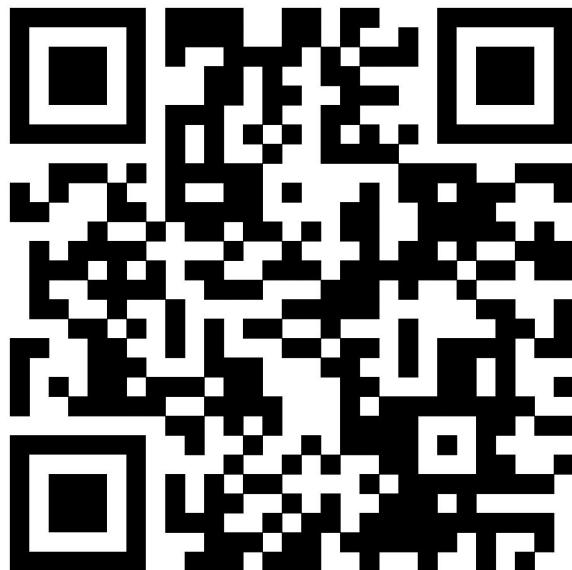
- In Gaussian mixture, we fix the covariance matrix for each cluster as  $\sigma^2 I$
- We take  $\sigma^2 \rightarrow 0$  then
- the update equations converge to doing K-means clustering

# Next Lecture: EM and PCA

- Derive E-step and M-step for Gaussian Mixtures
- PCA (Principle Component Analysis)

# **Any feedback (about lecture, slide, homework, project, etc.)?**

(via anonymous google form: <https://forms.gle/fpYmiBtG9Me5qbP37>)



Change Log of lecture slides:

<https://docs.google.com/document/d/e/2PACX-1vSSIHjklypK7rKFSR1-5GYXyBCEW8UPtpSfCR9AR6M1l7K9ZQEmxfFwaWaW7kLDxusthsF8WICyZJ-/pub>