EECS 545: Machine Learning

Lecture 15. Unsupervised Learning: Clustering & EM

Honglak Lee

3/11/2024

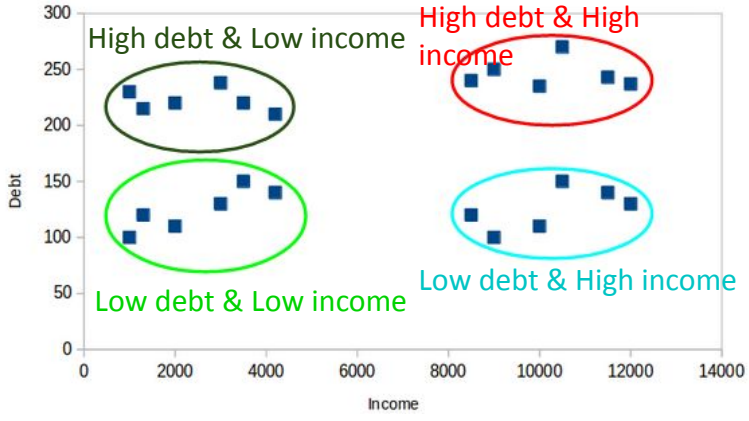
UNIVERSITY OF MICHIGAN

Outline

- Unsupervised Learning: Clustering
  - K-means clustering
- Expectation Maximization
  - General Receipe 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)



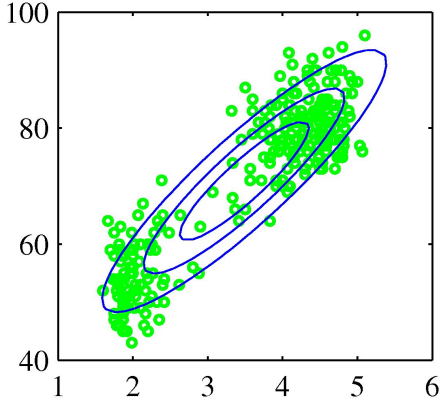
  - Customer information (e.g., income, debt, age, etc.) can be used for input features.

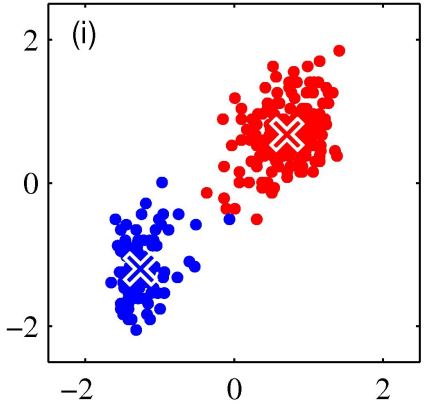
- A type of unsupervised learning
  - No label/target needed to learn the grouping!

K-Means Clustering

The K-Means Algorithm

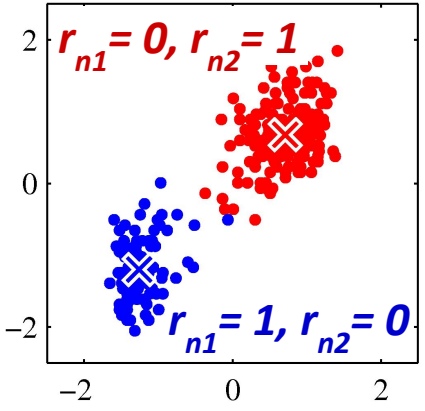
- Given ***unlabeled*** data
  $\{\mathbf{x}^{(n)}\}$ $(n = 1, \ldots, N)$,
- and believing it belongs in K clusters (say $K = 2$ here),
- How do we find the clusters?
  - What would be the objective function?





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 μ_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*):

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

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

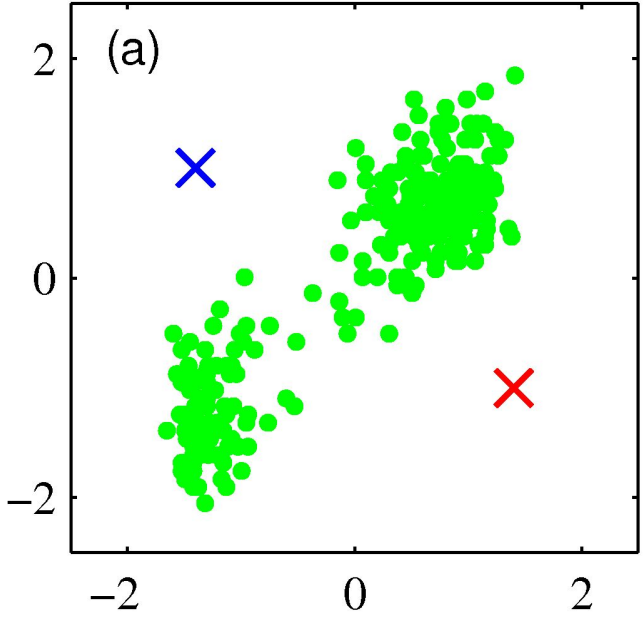
Note: E, M stands for:

- E: Expectation
- M: Maximization

(We will revisit EM later.)

K-Means Clustering

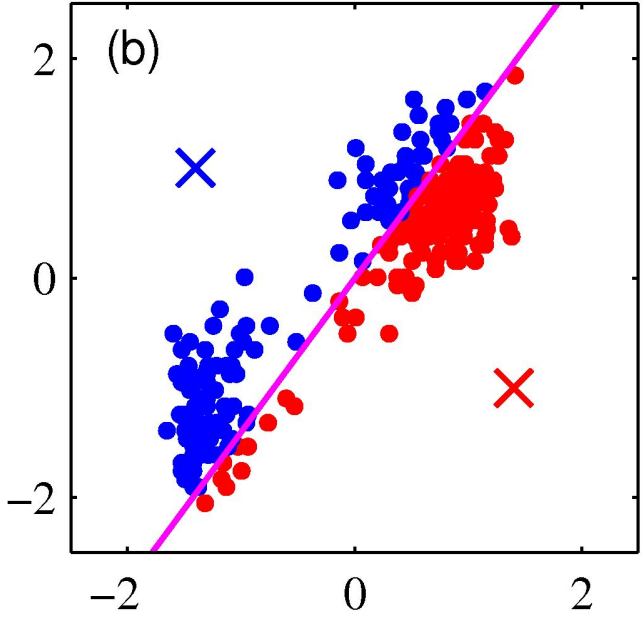
- Select K. Pick random centroids.
  - Here K=2.



K-Means Clustering

Cluster assignment Step ("E-Step")

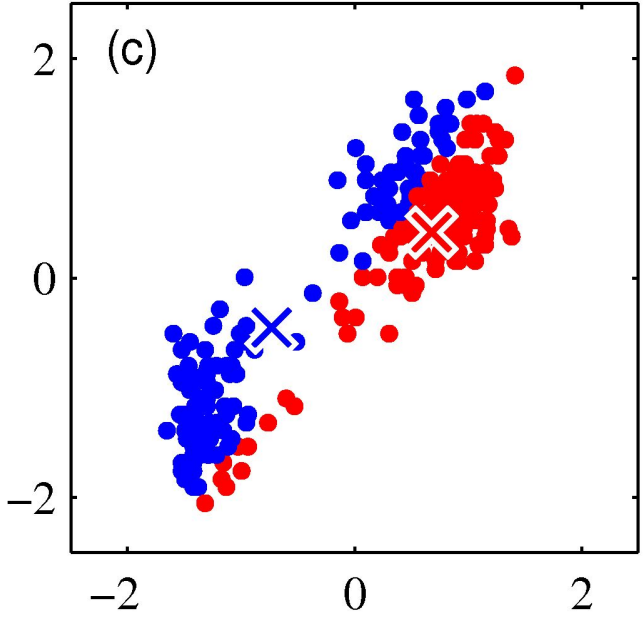
- Assign each point to the nearest center.



K-Means Clustering

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

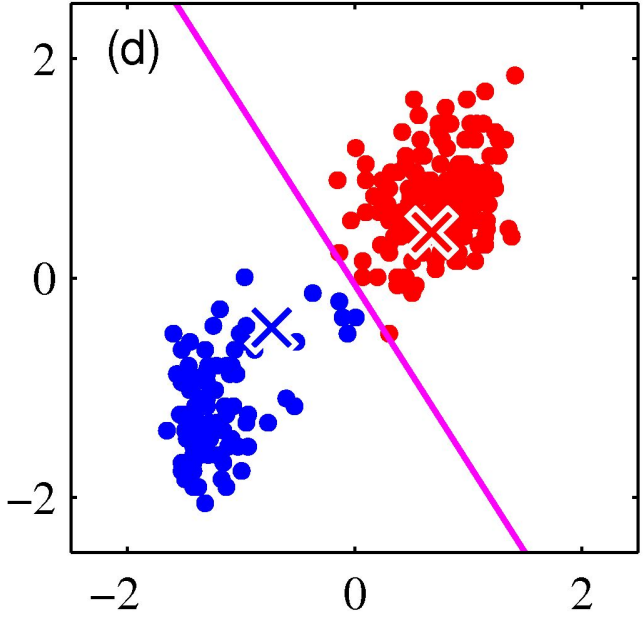
- Compute new centers for each cluster.



K-Means Clustering

Cluster assignment Step ("E-Step") again

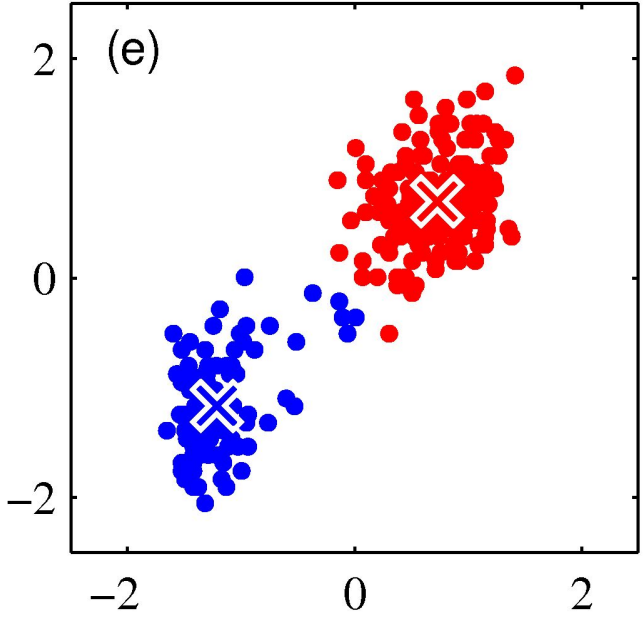
- Re-assign points to the now-nearest center.



K-Means Clustering

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

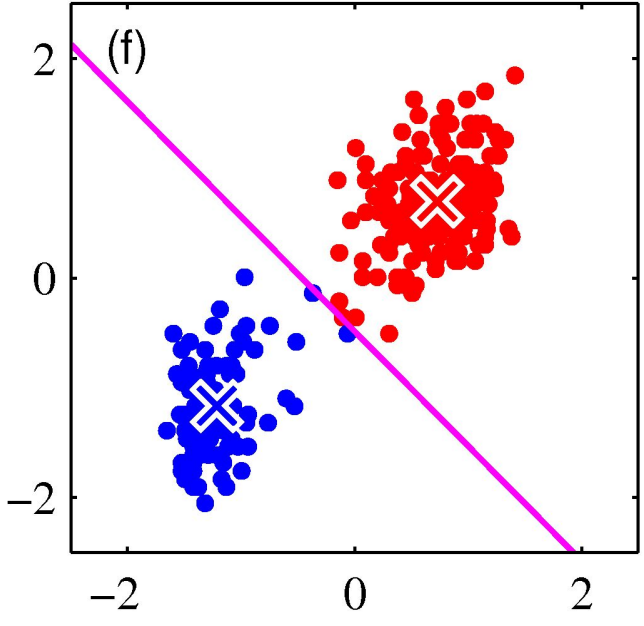
- Compute centers for the new clusters.



K-Means Clustering

Another Cluster assignment Step ("E-Step")

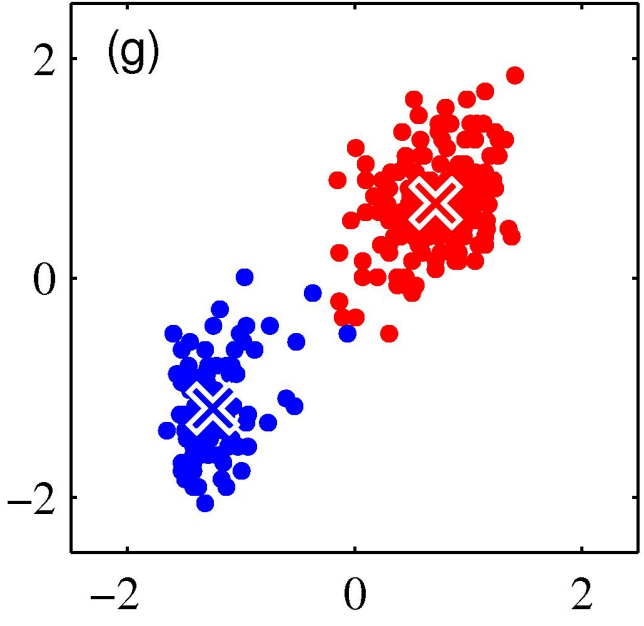
- Reassign the points to centers.



K-Means Clustering

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

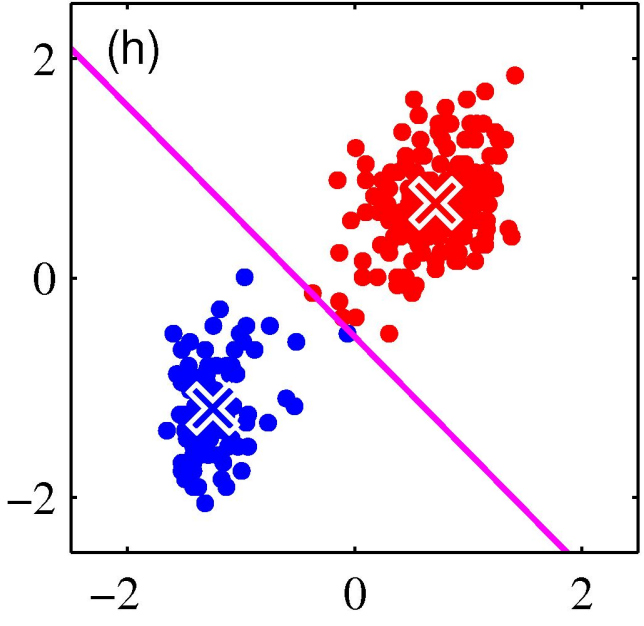
- New centers.



K-Means Clustering

Another Cluster assignment Step ("E-Step")

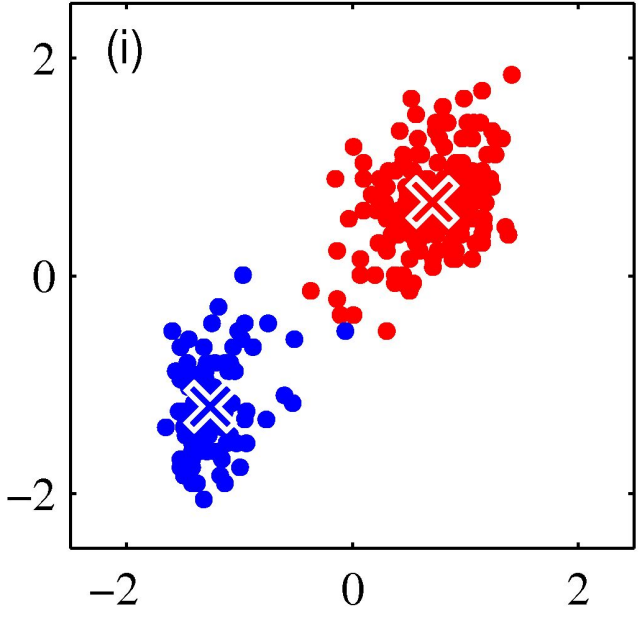
- New cluster assignments.



K-Means Clustering

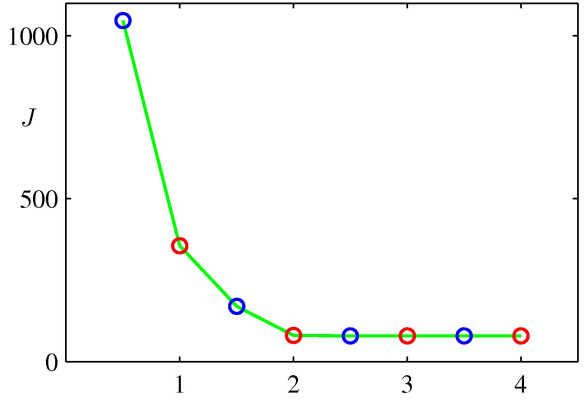
Update parameters (centroids) (“M-Step”) again

- The cluster centers have stopped changing.



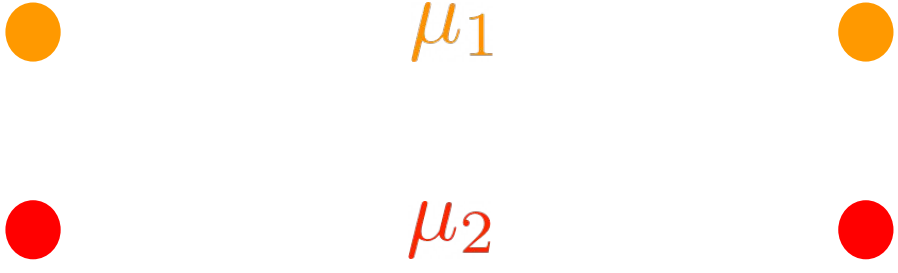
Convergence

- 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

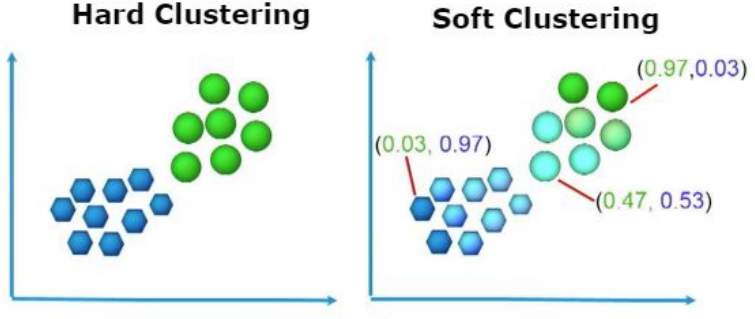


Gaussian Mixtures and Expectation-Maximization

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 takedifferent levels of "responsibility" (posterior probability) for that point.

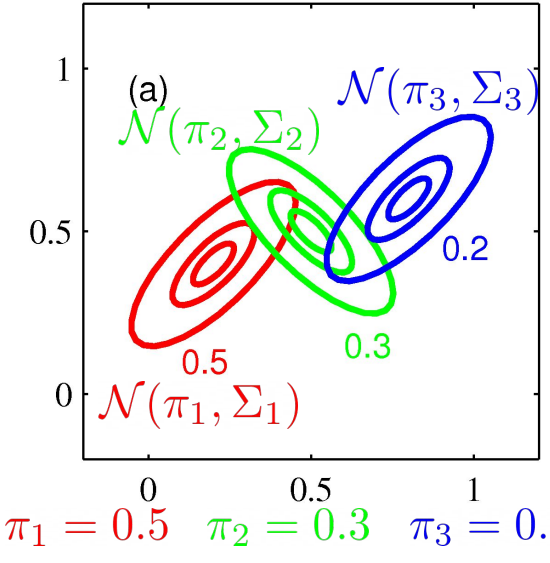


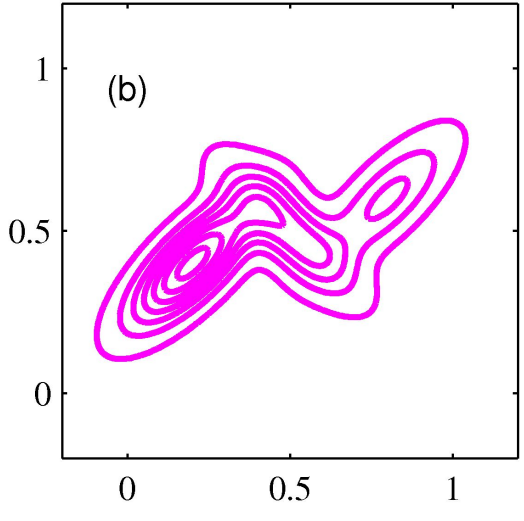
  - (It was actually generated by only one cluster, but we don't know which one; we assign a probability)

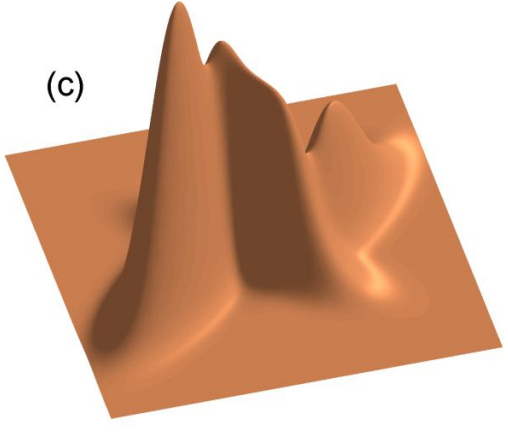
Mixtures of Gaussians

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

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

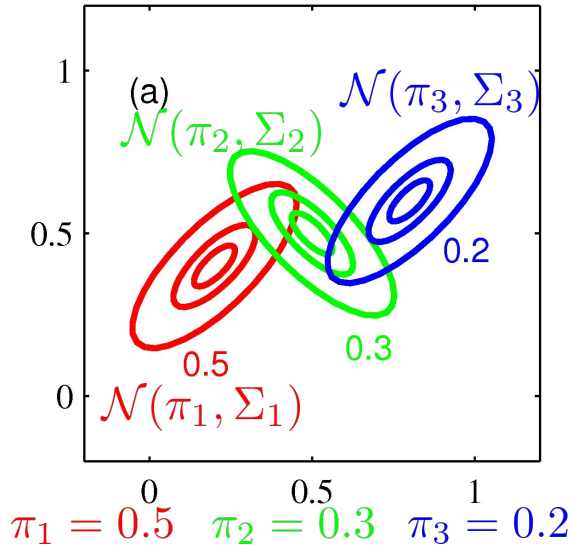






$\pi_1 = 0.5$ $\pi_2 = 0.3$ $\pi_3 = 0.2$

Mixtures of Gaussians



- Note the mixing coefficients in

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

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

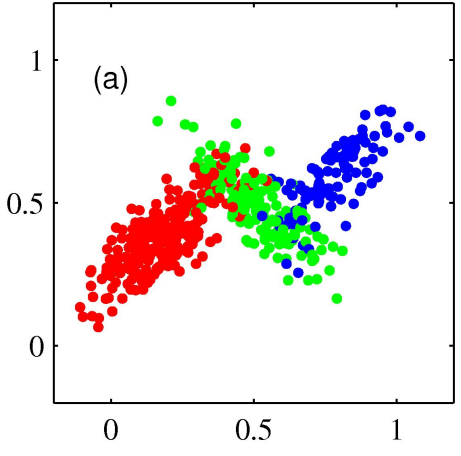
$$p(z_k = 1) = \pi_k \qquad p(\mathbf{z}) = \prod_{k=1}^{K} \pi_k^{z_k}$$

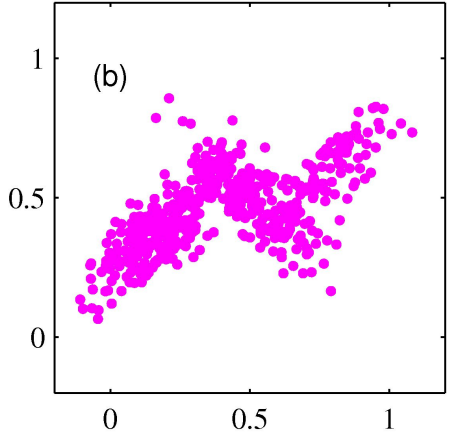
$$p(\mathbf{x} \mid z_k = 1) = \mathcal{N}(\mathbf{x} \mid \mu_k, \Sigma_k)$$

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

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

Latent Variables

- A system with observed variables **X**
  - may be easier to understand with latent variables **Z**, but they are not observed (**latent**).
- Notations:
  - We denote the set of all observed data by **X**, in which the n^{th} row represents $x^T_{n.}$
  - Similarly we denote the set of all latent variables by **Z**, with a corresponding row z^T_n.
  - Note: we use lowercase symbol for single sample (**x**), matrix symbol for all data (**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 {**X**, **Z**}, we could easily maximize the *complete* data likelihood $p(\mathbf{X}, \mathbf{Z} \mid \theta)$.
- Unfortunately, with *incomplete* data (**X** only), we must marginalize over **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 θ that maximizes the likelihood $p(\mathbf{X} \mid \theta) = \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z} \mid \theta)$, 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 $\log p(\mathbf{X} \mid \theta)$ is established based on q and θ.
- Then, $q(\mathbf{Z})$ and θ 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:

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

Note: KL Divergence

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

$$KL(q \,\|\, p) = \mathbb{E}_{z \sim q(z)} \left[\log \frac{q(z)}{p(z)} \right] = \sum_z q(z) \log \frac{q(z)}{p(z)}$$

$$= -\sum_z q(z) \log p(z) + \sum_z q(z) \log q(z)$$

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

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

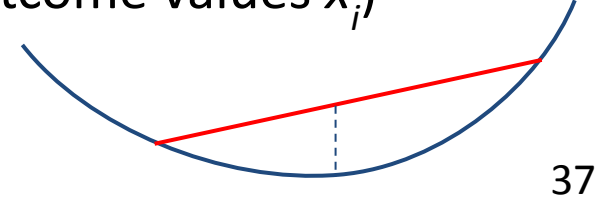
- $KL(q \,\|\, p) \geq 0, \quad \text{with equality iff } p = q.$
- $KL(q \,\|\, p) \neq KL(p \,\|\, q) \quad \text{in general}$

Background note: Jensen's Inequality

- If f is convex, then for any θ_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 θ_i as probability mass for different outcome values x_i)
$$f(\mathbb{E}[x]) \leq \mathbb{E}[f(x)]$$

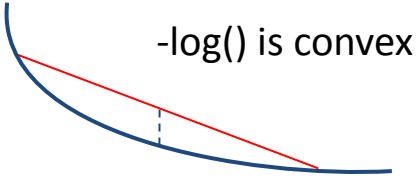


Background note: Jensen's Inequality

- If f is convex, then for any θ_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*(...) = -log (...) and the following:
$$\theta_i = q(z), x_i = \frac{p(z)}{q(z)}$$



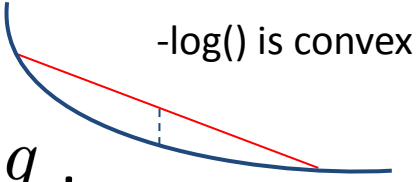
$$-\log(\mathbb{E}[x]) \leq \mathbb{E}[-\log(x)]$$

Non-negativity of KL divergence

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

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

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



- To show $KL(q \,\|\, p)$ is non-negative for any p, q , plug in *f*(...) = -log (...) and the following: $\theta_i = q(z), x_i = \frac{p(z)}{q(z)}$

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

Jensen's inequality for -log():

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

i.e., plugin

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

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]

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

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

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

Evidence Lower bound (ELBO) or variational lower bound

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

- **EM algorithm**:

  Repeat alternating optimization until convergence:
  - E-step: for fixed θ, find q that maximizes $\mathcal{L}(q, \theta)$
  - M-step: for fixed q, find θ that maximizes $\mathcal{L}(q, \theta)$

* E: expectation
* M: maximization

The EM Algorithm: E-step

- We have shown that: [variational lower bound]

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

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

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

Evidence Lower bound (ELBO) or variational lower bound

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

- **(E-step)** For a fixed θ, which q maximizes $\mathcal{L}(q, \theta)$?

$\Rightarrow p(\mathbf{Z} \mid \mathbf{X})$, 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 $p(\mathbf{X}, \mathbf{Z}|\theta)$.
    This is tractable and can be optimized w.r.t θ
  - (2) Entropy of $q(\mathbf{Z})$ which is independent of θ since q is fixed.

- **(M-step)** Thus, when q is fixed, we can find θ that maximizes $\mathcal{L}(q, \theta)$.

The EM Algorithm: summary

- Initialize parameters θ randomly
- Repeat until convergence:

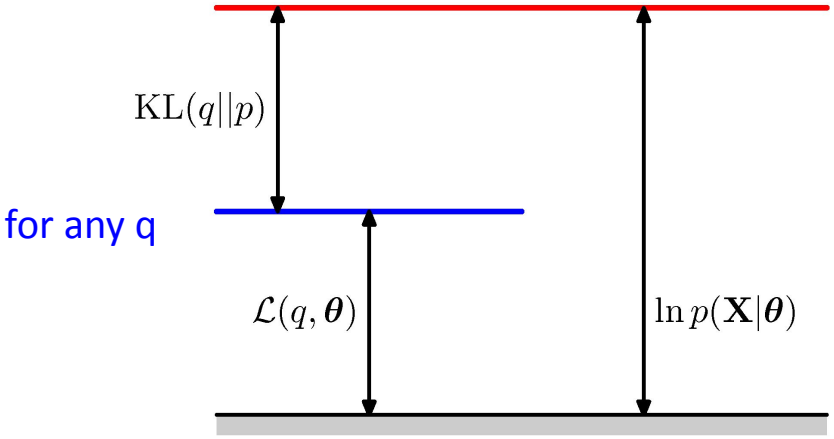
  (optimize $\mathcal{L}(q, \theta)$ w.r.t. q and θ alternatingly.)
  - "E-step": Set $q(\mathbf{Z}) = p(\mathbf{Z} \mid \mathbf{X}, \theta)$ compute posterior → optimal q(Z)!
  - "M-step": Update θ via the following maximization

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

use *q*(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



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

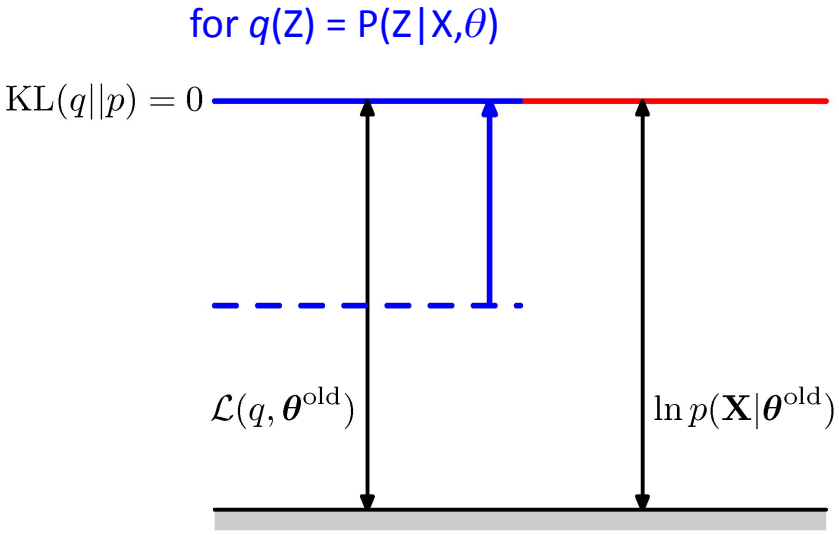
$$\sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z} \mid \theta)}{q(\mathbf{Z})}$$

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

  which EM tries to maximize.

Visualize the E-Step

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



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

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

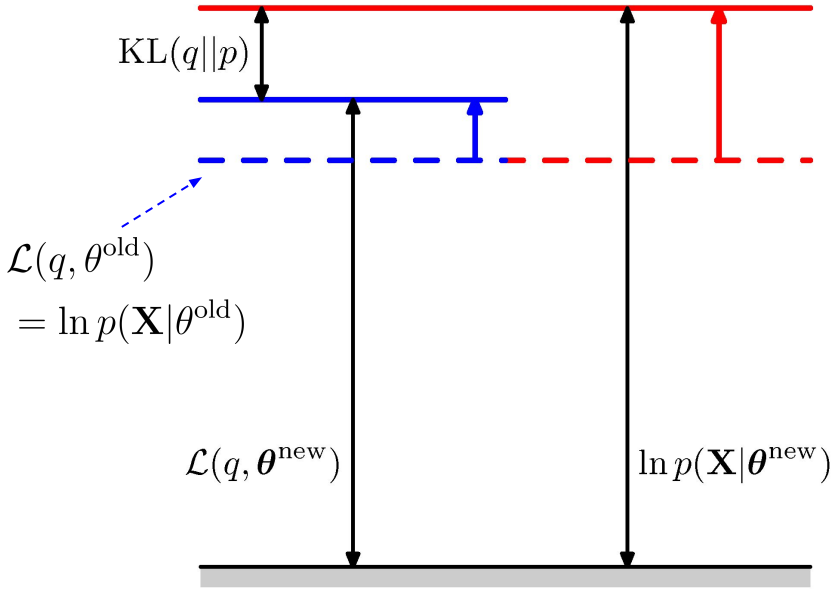
- E-Step changes $q(\mathbf{Z})$ to maximize $\mathcal{L}(q, \theta)$
- So maximized when

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

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

Visualize the M-Step

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



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

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

- Holding $q(\mathbf{Z})$ constant; increase $\mathcal{L}(q, \theta)$
- Updating θ will make $\log p(\mathbf{X} \mid \theta)$ increase!

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

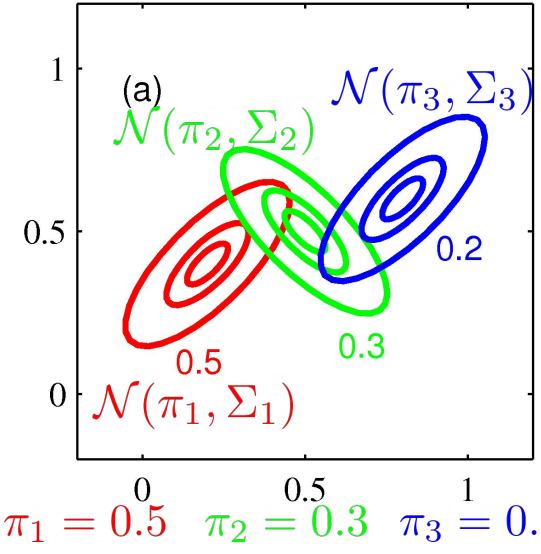
- But now $p \neq q$
- so $KL(q||p) > 0$

Mixtures of Gaussians (recap)

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

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

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



- Joint and marginal distributions:

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

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

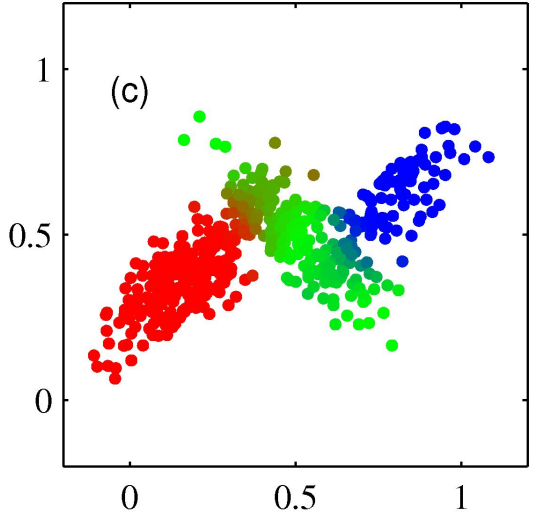
Derivation: in the next lecture!

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 → optimal q(Z)!

(Hint: Use Bayes Rule)

Derivation: in the next lecture!

EM for Gaussian Mixtures: M-step

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

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

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

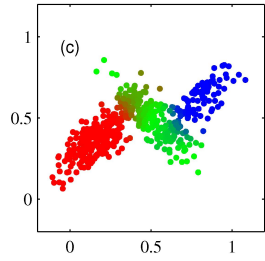
$$\Sigma_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) (\mathbf{x}^{(n)} - \mu_k^{\text{new}})(\mathbf{x}^{(n)} - \mu_k^{\text{new}})^\top$$

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

M-step: use *q*(Z) as pseudo-counts and maximize the "data completion" log-likelihood (i.e. GDA with pseudo-counts!)

- Stop when either parameters or log-likelihood converges.

EM for Gaussian Mixtures
(1-page summary)



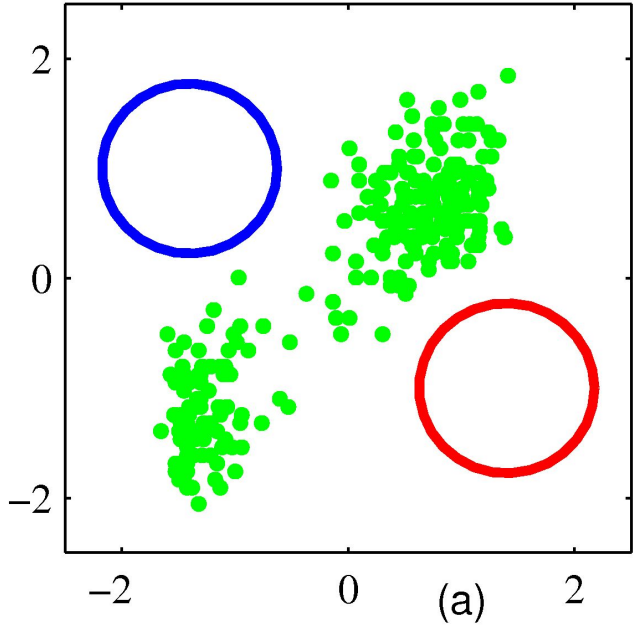
- Initialize parameters randomly $\theta = \{\pi_k, \mu_k, \Sigma_k\}_{k=1}^{K}$
- Repeat until convergence (alternating optimization)
  - E Step: Given fixed parameters θ, optimize $q(\mathbf{Z})$: find posterior probabilities of Z given X, $q(\mathbf{Z}) = p(\mathbf{Z} \mid \mathbf{X}, \theta)$

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

  - M Step: Given fixed $q(\mathbf{Z})$ (or $\gamma(z_{nk})$), optimize θ (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.



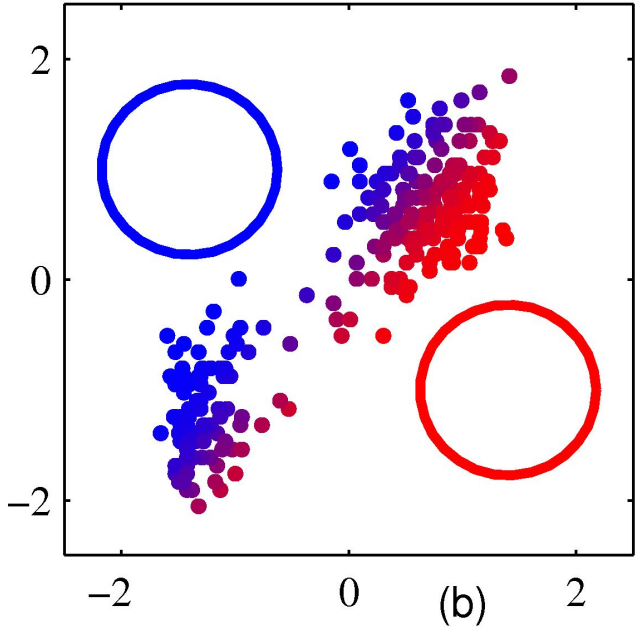
EM Example

- First E Step

For each sample n, calculate:

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

$$= P(z_k = 1|\mathbf{x}^{(n)})$$



EM Example

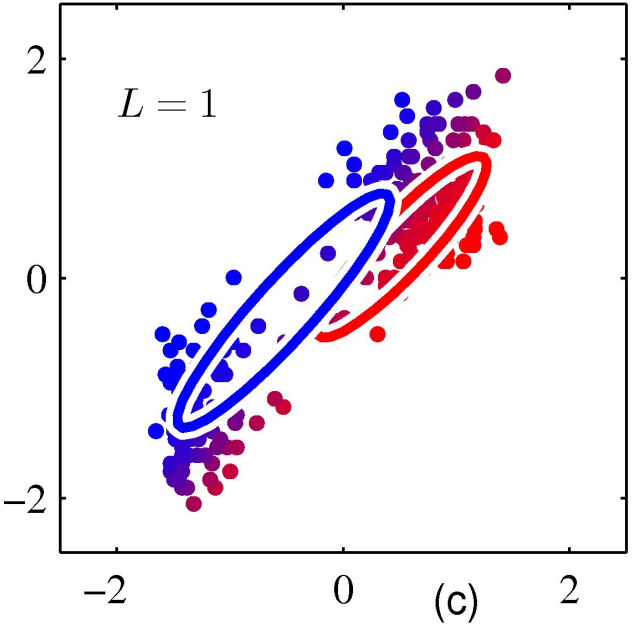
- First M Step

Update Gaussian parameters:

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

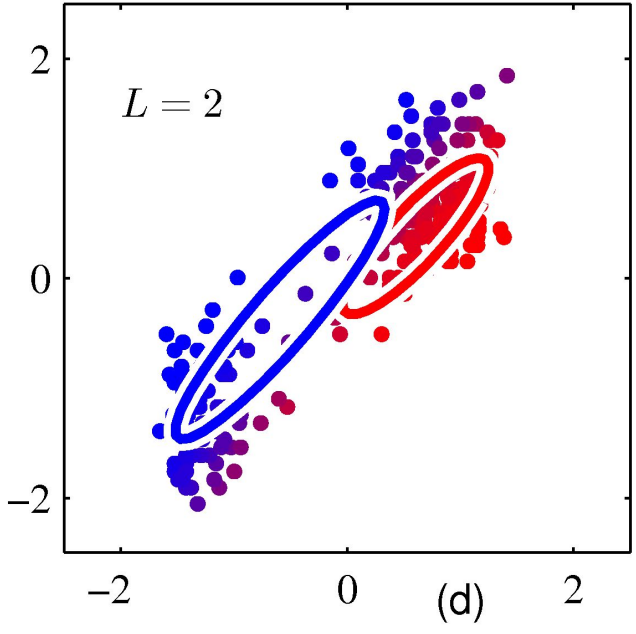
$$\mu_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}^{(n)}$$

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



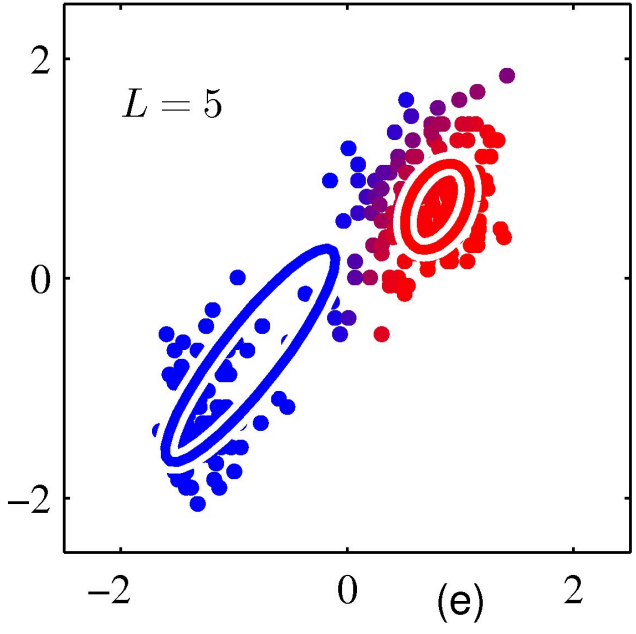
EM Example

- Second E and M Steps



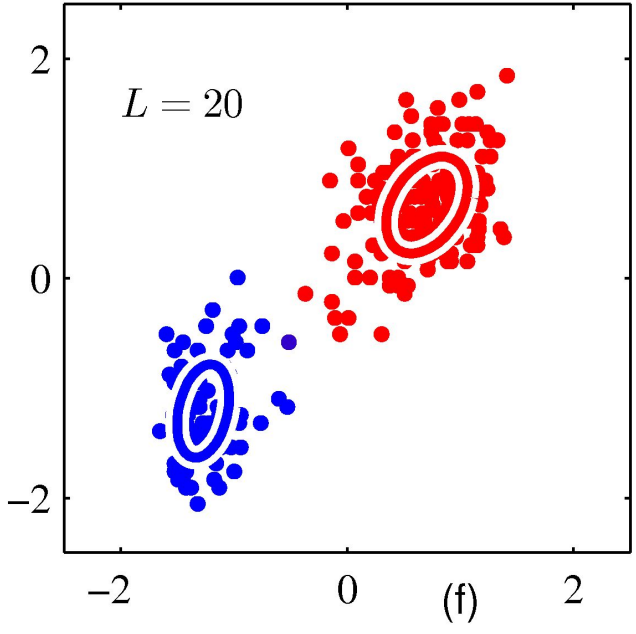
EM Example

- Three more E-M cycles



EM Example

- Fifteen E-M cycles later



Relation to K-means

- In Guassian mixture, we fix the covariance matrix for each cluster as $\sigma^2 I$
- We take $\sigma^2 \to 0$ then
- the update equations coverge 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/99jeftYTaozJvCEF8)

Change Log of lecture slides:
https://docs.google.com/document/d/e/2PACX-1vRKx40eOJKACqrKWraio0AmlFS1_xBMINuWcc-jzpfo-ySj_gBuqTVdfHy8v4HDmqDJ3b3TvAW1FVuH/pub