EECS 545: Machine Learning

Lecture 15. Unsupervised Learning: Clustering

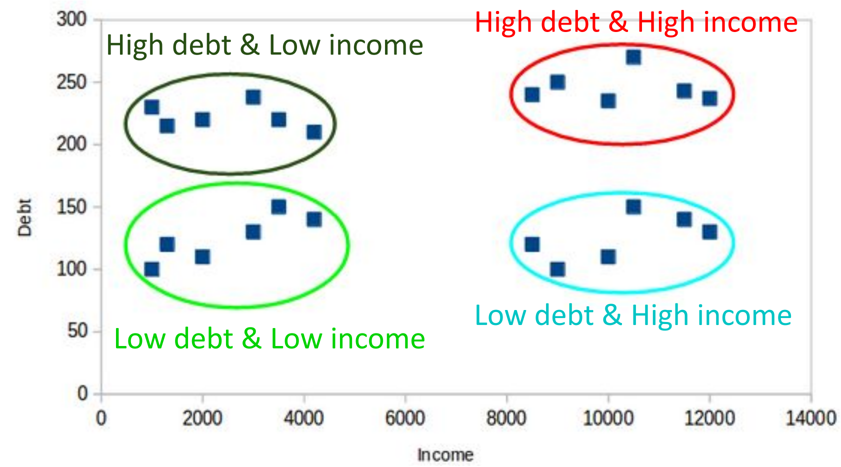
Honglak Lee and Michał Dereziński

03/07/2022

Outline

- Unsupervised Learning: Clustering
  - K-means clustering
- Expectation Maximization
  - Gaussian Mixtures

Clustering

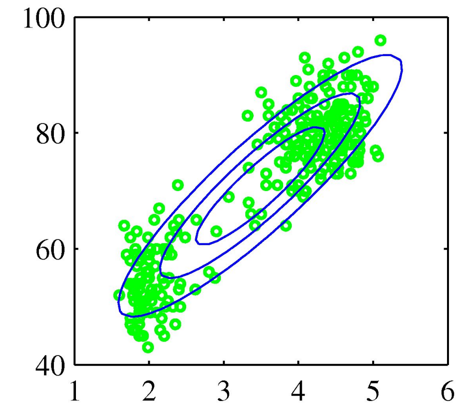


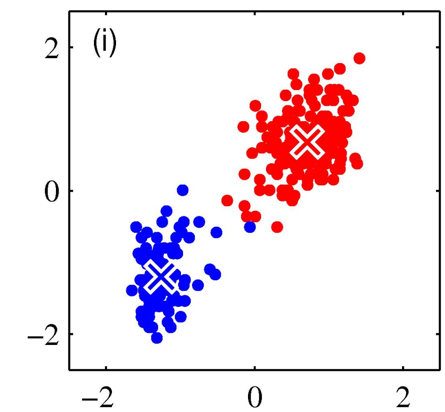
- Motivating example: Customer segmentation
  - Group customers so that it can be helpful for decision making (e.g., credit card request approval) or marketing (e.g., promotion of products)
  - Customer information (e.g., income, debt, age, etc.) can be used for input features.
- A type of unsupervised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

K-Means

The K-Means Algorithm

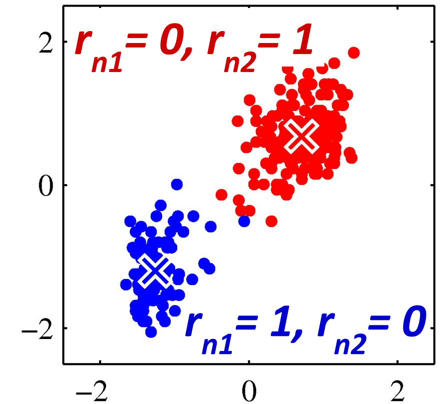
- Given unlabeled data $\{x^{(n)}\}$ $(n=1,\ldots,N)$,
- And believing it belongs in *K* clusters (say K=2 here),
- How do we find the clusters?
  - What would be the objective function?





The K-Means Algorithm

- Use indicator variables $\boldsymbol{r}_{nk}$ in {0,1}.
  - $\boldsymbol{r}_{nk} = 1$ if $\mathbf{x}^{(n)}$ is in cluster k.
  - and $\boldsymbol{r}_{nj} = 0$ for all j other than k.



- Find cluster centers μ_k and assignments $\boldsymbol{r}_{nk}$ to minimize the distortion measure J
  - Sum of squared distance of points from the center of its own cluster (*Intracluster variation*):

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

The K-Means Algorithm

- Initialize the cluster centers (centroids) arbitrarily.

- Repeat the following updates until convergence

  1. $r := \arg\min_r J(r, \mu)$
  2. $\mu := \arg\min_\mu J(r, \mu)$

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

The K-Means Algorithm

Note: E, M stands for:

- E: Expectation
- M: Maximization

(We will revisit this later.)

- Set the cluster centers arbitrarily.
- Repeat until convergence:
  - **Cluster assignment ("E-Step"): assign each point to closest center.**

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

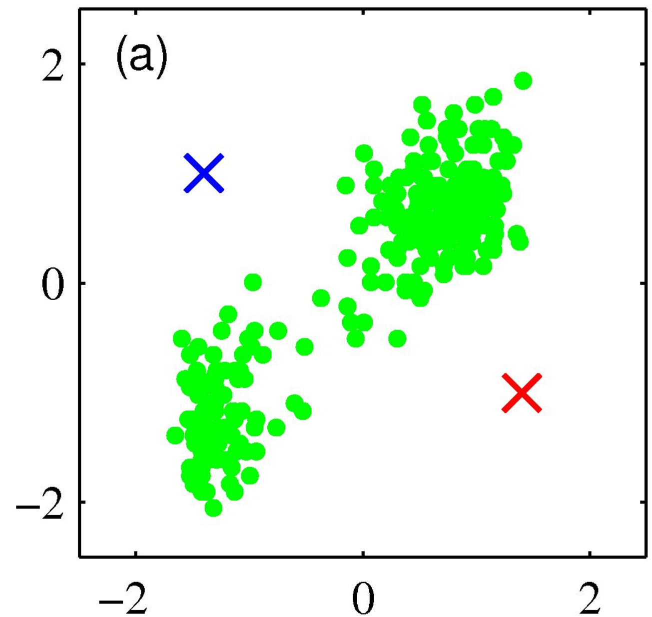
  - **Parameter update ("M-Step"): update the centers**

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

Q. Verify this

K-Means Clustering

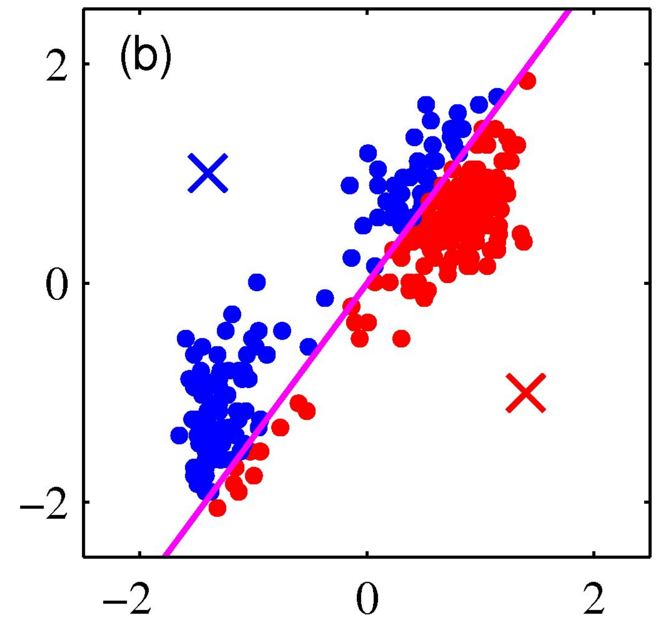
- Select K. Pick random centroids.
  - Here K=2.



K-Means Clustering

Cluster assignment Step (“E-Step”)

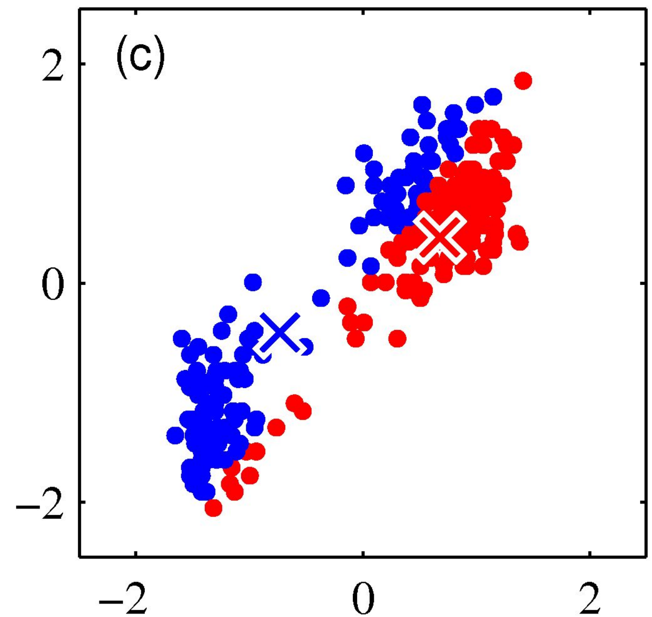
- Assign each point to the nearest center.



K-Means Clustering

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

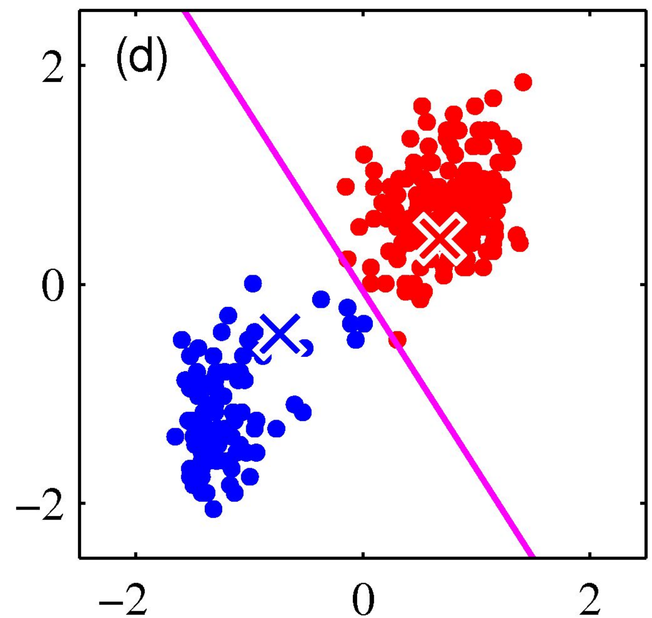
- Compute new centers for each cluster.



K-Means Clustering

Cluster assignment Step ("E-Step") again

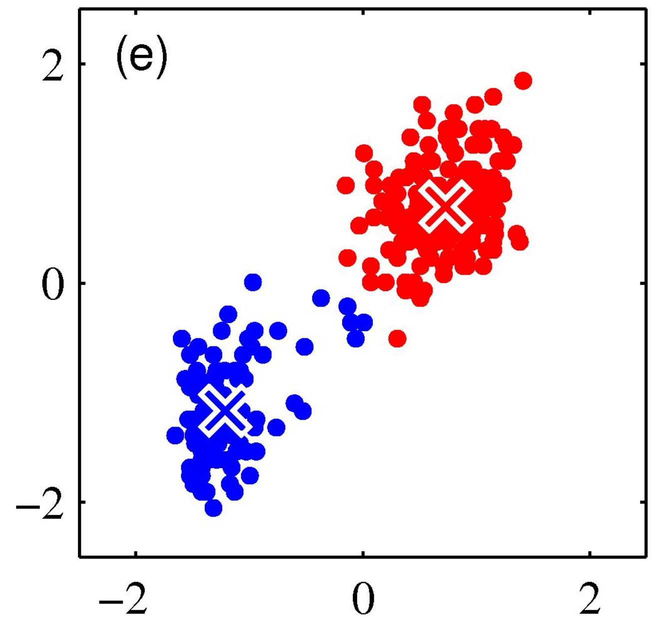
- Re-assign points to the now-nearest center.



K-Means Clustering

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

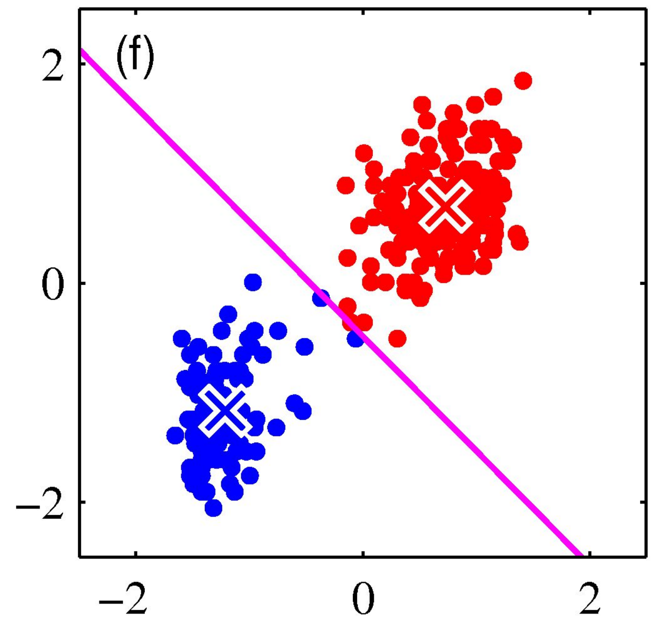
- Compute centers for the new clusters.



K-Means Clustering

Another Cluster assignment Step ("E-Step")

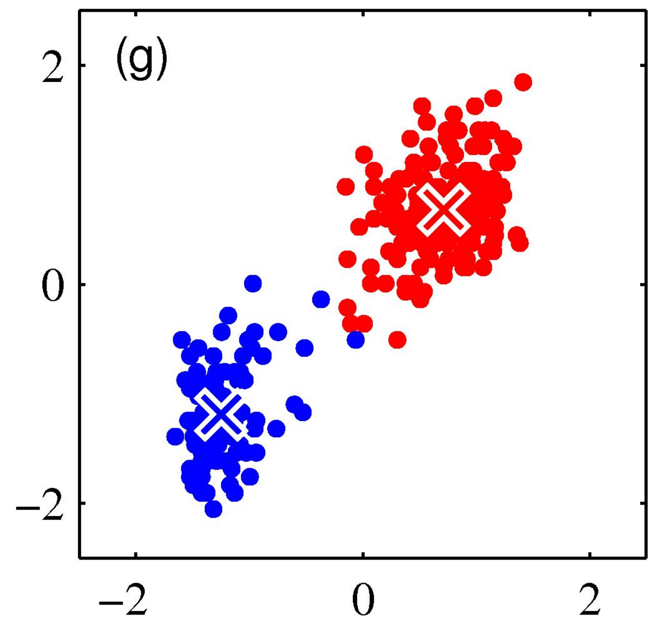
- Reassign the points to centers.



K-Means Clustering

Update parameters (centroids) ("M-Step") again

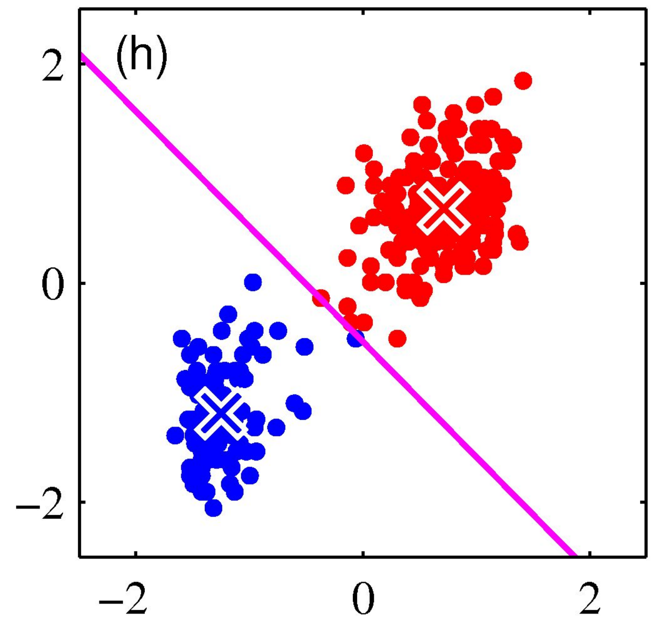
- New centers.



K-Means Clustering

Another Cluster assignment Step (“E-Step”)

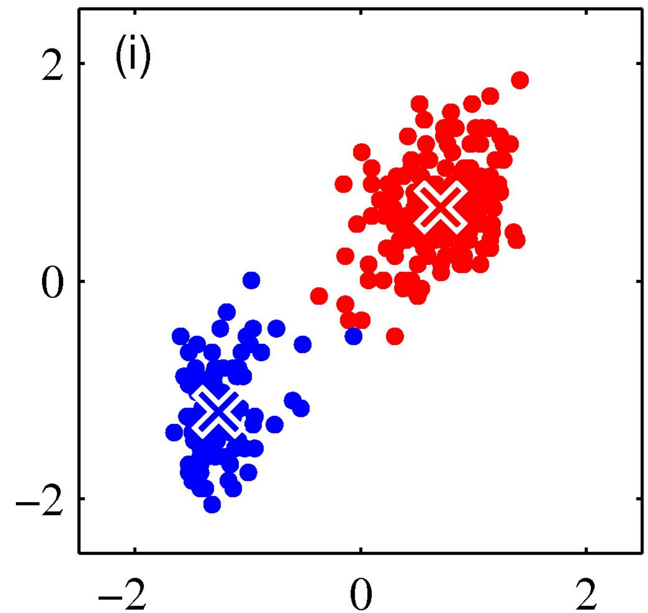
- New cluster assignments.



K-Means Clustering

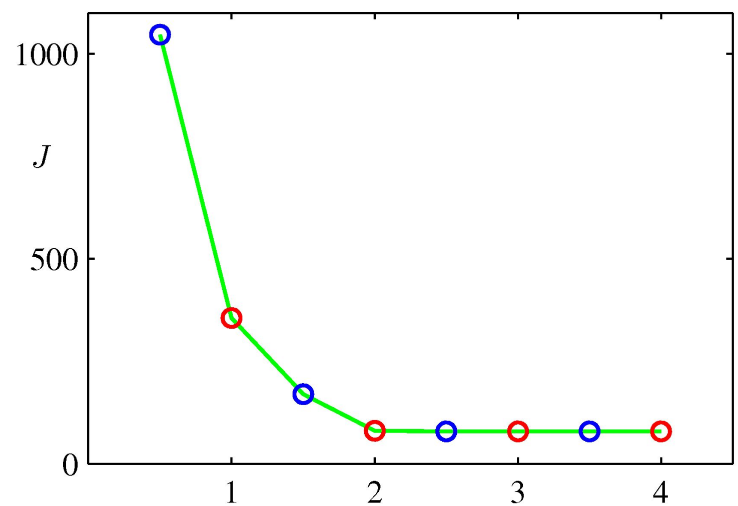
Update parameters (centroids) (“M-Step”) again

- The cluster centers have stopped changing.



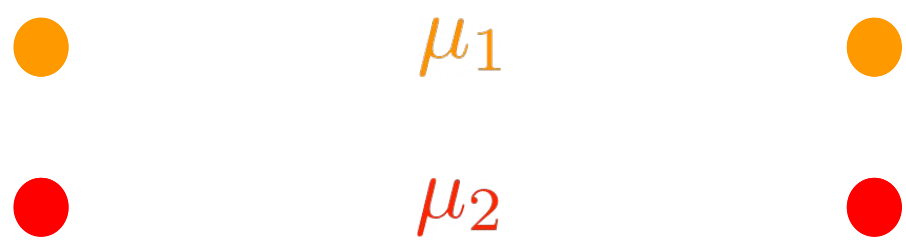
Convergence

- 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

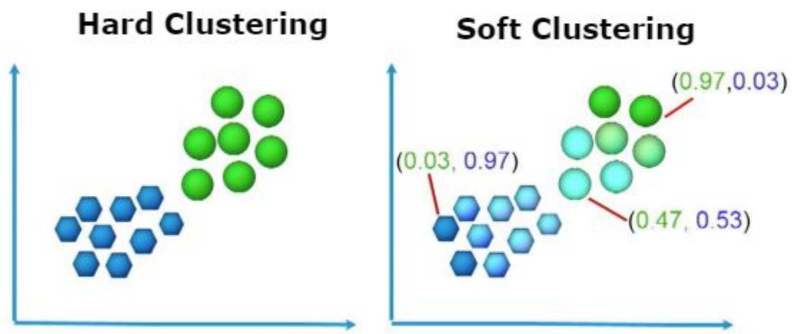


Gaussian Mixtures and Expectation-Maximization

Hard and Soft Clusters

- K-Means uses **hard clustering assignment**.
  - A point belongs to exactly one cluster.

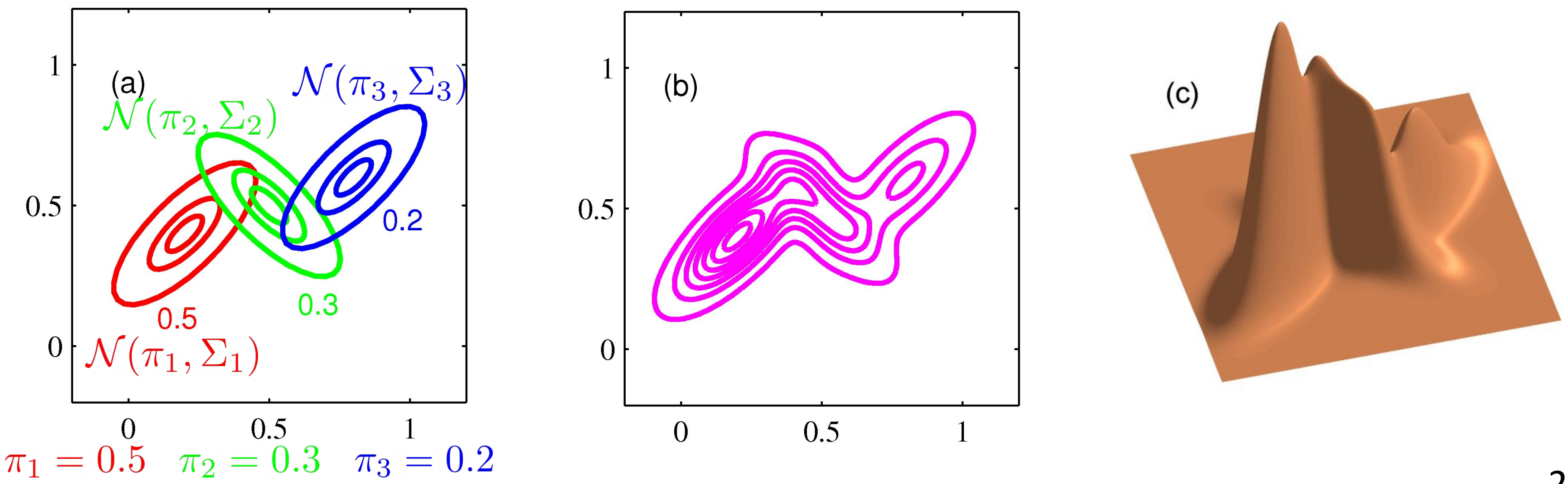
- Mixture of Gaussians uses **soft clustering**.
  - **A point could be explained by more than one cluster.**
  - Different clusters take different levels of "responsibility" (posterior probability) for that point.



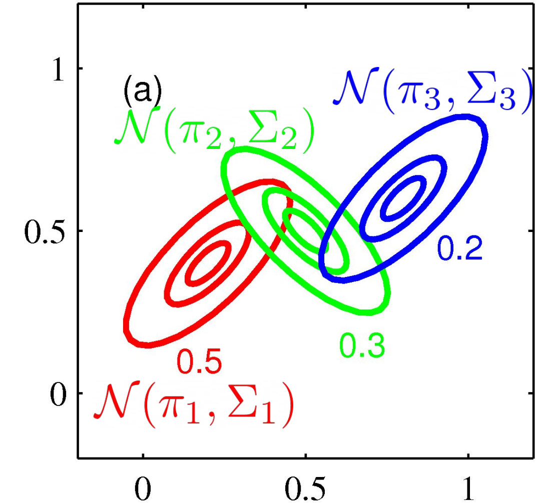
Mixtures of Gaussians

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

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



Mixtures of Gaussians



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

- Note the mixing coefficients in

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

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

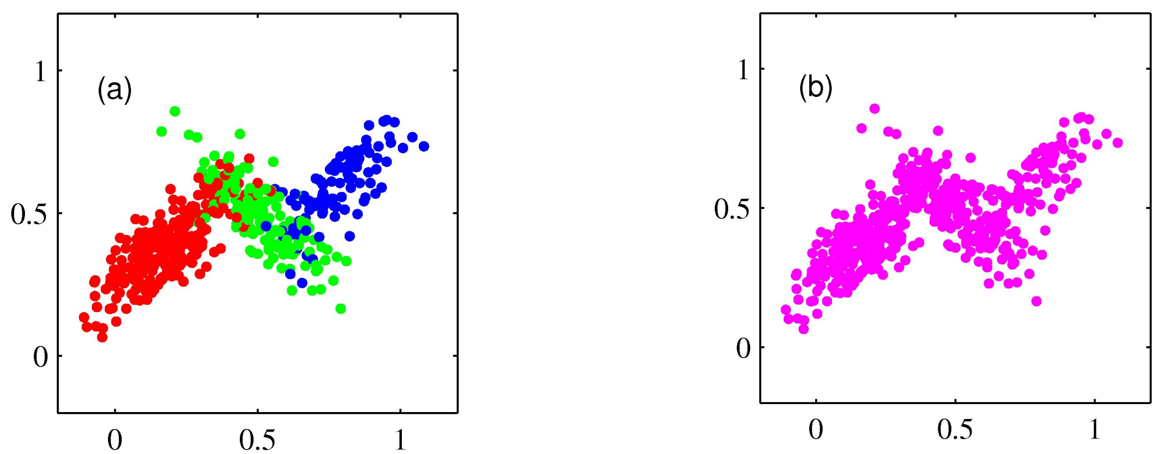
$$p(z_k = 1) = \pi_k \qquad\qquad p(\mathbf{z}) = \prod_{k=1}^{K} \pi_k^{z_k}$$

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

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

Mixtures of Gaussians

- To generate samples from a Gaussian mixture distribution *p*(x), use *p*(x,z):
  - Select a value **z** from the marginal *p*(**z**);
  - Then select a value **x** from *p*(**x** | **z**) for that **z**.



Latent Variables

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

Learning a Latent Variable Model

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

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

(The sum inside the log makes it hard.)

The EM Algorithm in General

- Expectation-Maximization is a general recipe for finding the parameters that maximize the (log) likelihood of latent variable models
- To find θ 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 $L(q, \theta)$ for the log-likelihood $p(\mathbf{X}|\theta)$ is established based on q and θ.
- Then, $q(\mathbf{Z})$ and θ are alternatingly updated (keeping the other fixed) so that $L(q, \theta)$ is maximized (similar to co-ordinate ascent) until convergence.

The EM Algorithm in General

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

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

Note: KL Divergence

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

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

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

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

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

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

Background note: Jensen's Inequality

- If f is convex, then for any θ_i s.t. $0 \leq \theta_i \leq 1 \ (\forall i)$

$$\theta_1 + \theta_2 + \cdots + \theta_k = 1$$

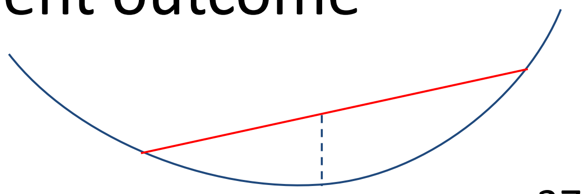
$$f(\theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_k x_k) \leq \theta_1 f(x_1) + \cdots + \theta_k f(x_k)$$

- 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)

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



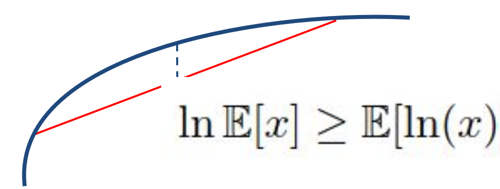
Background note: Jensen's Inequality

- 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 (think of θ_i as probability mass for different outcome values x_i) :

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



- To show KL(q // p) is non-negative for any p, q, plug in f() = -ln () and the following.

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

Note:

- ln() is concave
- -ln() is convex

The EM Algorithm in General

- We have thus shown that:

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

Evidence Lower bound (ELBO) or variational lower bound

with equality holding if and only if

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

- For a fixed θ, what is the q that maximizes $L(q, \theta)$?
- $p(\mathbf{Z}|\mathbf{X})$ because all other q result in strictly less than log $p(\mathbf{X}|\theta)$.

The EM Algorithm in General

- We also note that for a fixed *q*, *L(q, θ)* can be decomposed into two terms:
  - A weighted sum of log $p(\mathbf{X}, \mathbf{Z}|\theta)$. This is tractable and can be optimized wrt θ
  - Entropy of $q(\mathbf{Z})$ which is independent of θ since *q* is fixed.

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

Thus, we can find θ that maximize $L(q, \theta)$ when *q* is fixed.

The EM Algorithm

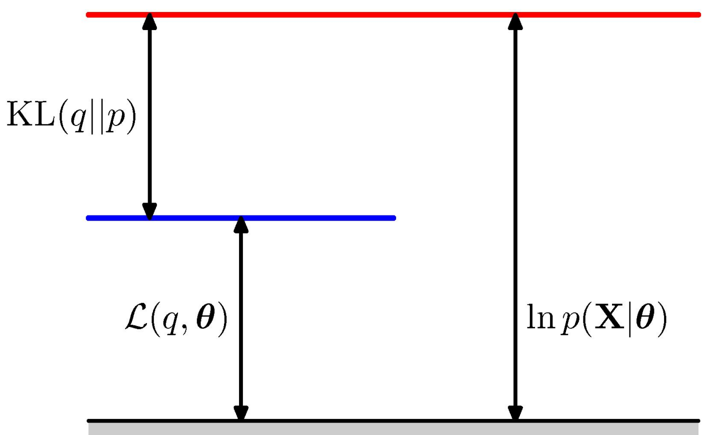
- Initialize random parameters θ
- Repeat until convergence:
  - "E - step": Set $q(\mathbf{Z}) = p(\mathbf{Z}|\mathbf{X}, \theta)$
  - "M - step": Update θ via the following maximization

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

- Note we have assumed that $p(\mathbf{Z}|\mathbf{X}, \theta)$ is tractable (i.e., find exact posterior $p(Z|X, \theta)$) .

Q. What if its not?

Visualize the Decomposition

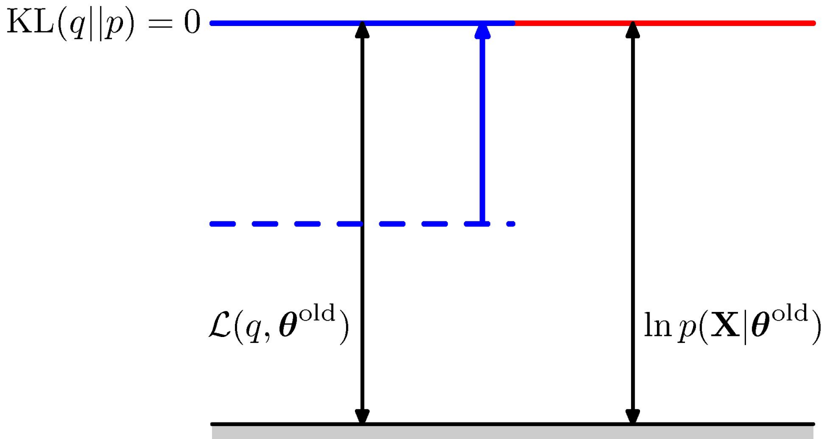


$$\ln p(\mathbf{X}|\theta) = \mathcal{L}(q, \theta) + KL(q||p)$$

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

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

Visualize the E-Step



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

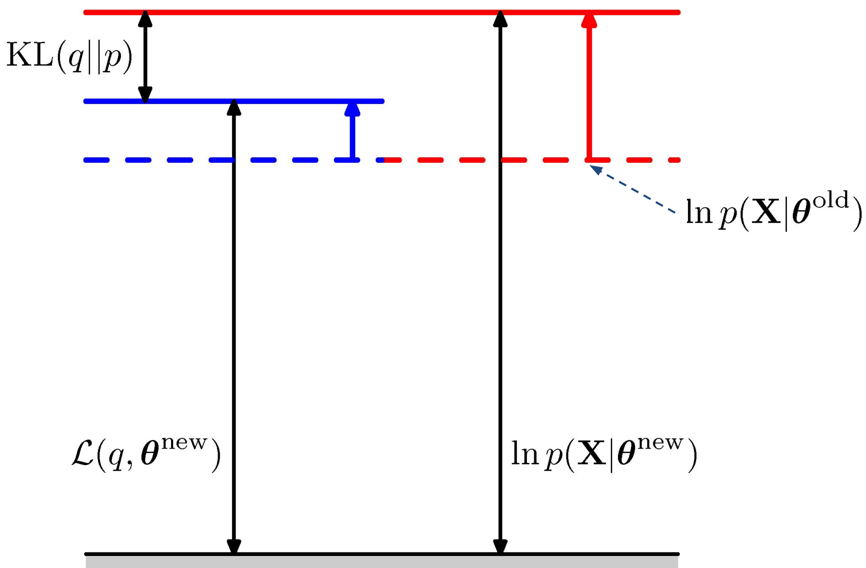
$$\sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\theta)}{q(\mathbf{Z})}$$

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

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

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

Visualize the M-Step



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

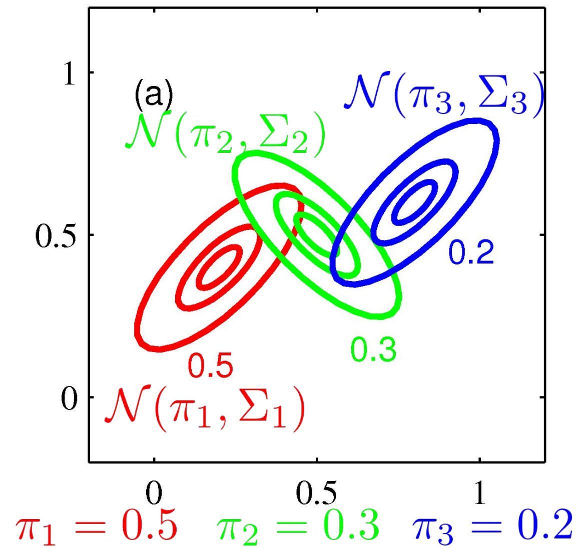
$$\sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X}, \mathbf{Z}|\theta)}{q(\mathbf{Z})}$$

- Holding *q*(Z) constant; increase $\mathcal{L}(q, \theta)$
- This increases $\ln p(\mathbf{X}|\theta)$
- But now $p \neq q$
- so $KL(q||p) > 0$

Mixtures of Gaussians (recap)

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

$$p(z_k = 1) = \pi_k \qquad \sum_{k=1}^{K} \pi_k = 1$$



- Generate x from Gaussian given the selected cluster assignment z

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

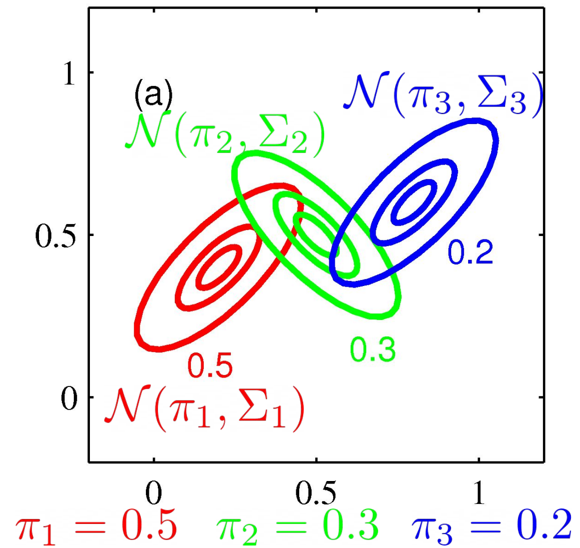
$$p(\mathbf{x}) = \sum_{\mathbf{z}} p(\mathbf{z}) p(\mathbf{x}|\mathbf{z}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\mu_k, \mathbf{\Sigma}_k)$$

Mixtures of Gaussians (recap)

- In other words, generate (sample) z then x:

$$p(z_k = 1) = \pi_k \qquad \sum_{k=1}^{K} \pi_k = 1$$

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



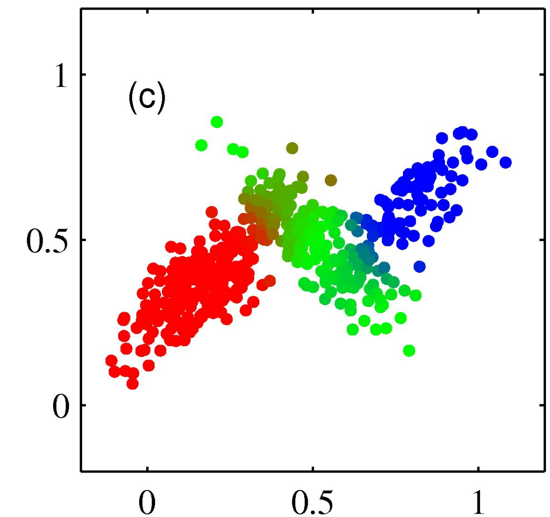
- Joint and marginal distributions:

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

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

EM for Gaussian Mixtures (summary)

- Initialize means, covariances, and mixing coefficients for the K Gaussians.



- E Step: Given the coefficients, evaluate the responsibilities.

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

(Hint: Use Bayes Rule)

EM for Gaussian Mixtures (summary)

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

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

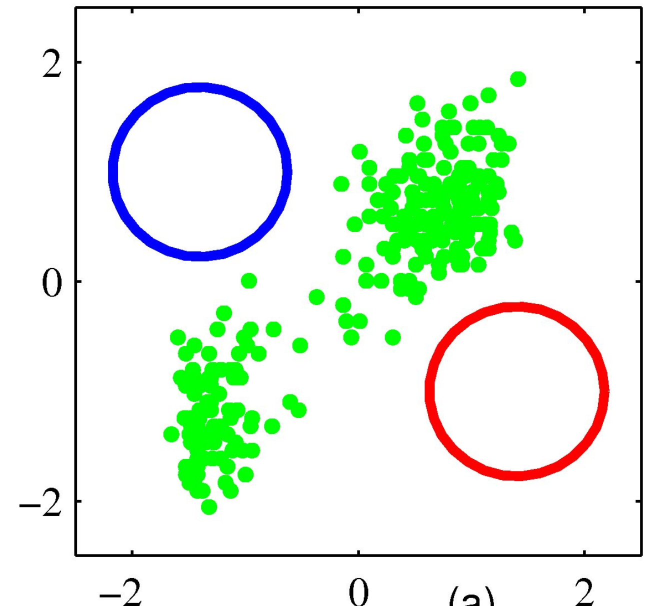
$$\mu_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}^{(n)}$$

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

- Stop when either coefficients or log likelihood converges.

EM Example

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



(a)

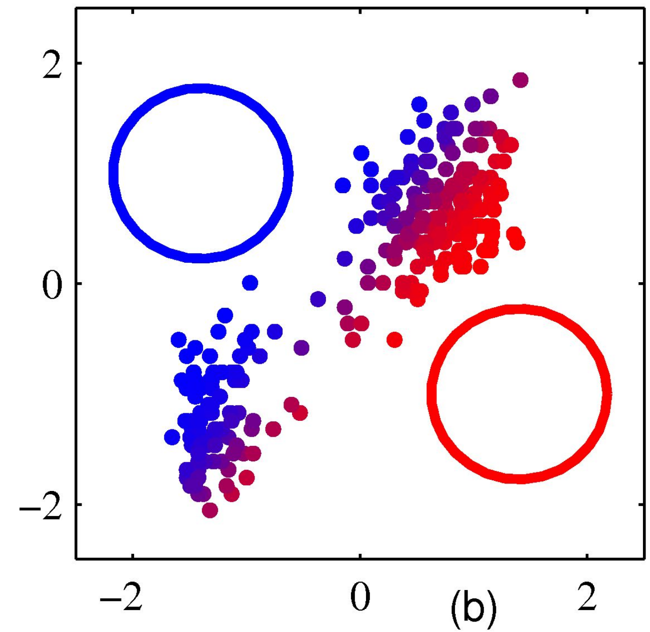
EM Example

- 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

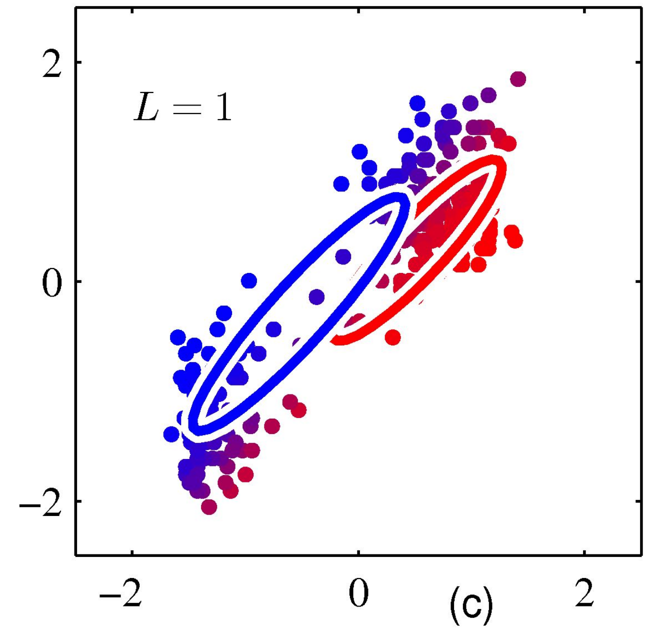
- First M Step

Update Gaussian parameters:

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

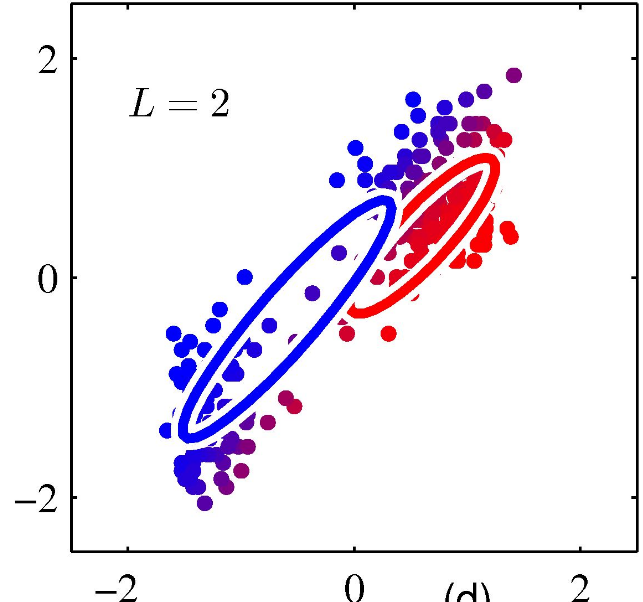
$$\mu_k^{\text{new}} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}^{(n)}$$

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



EM Example

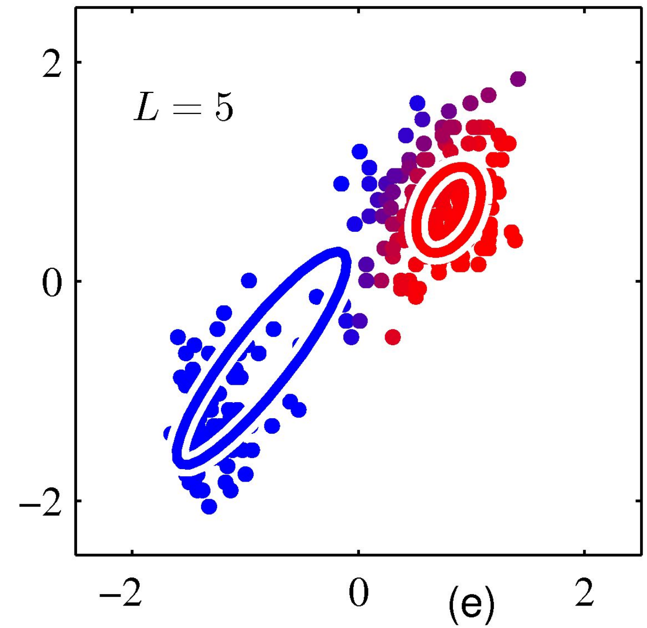
- Second E and M Steps



(d)

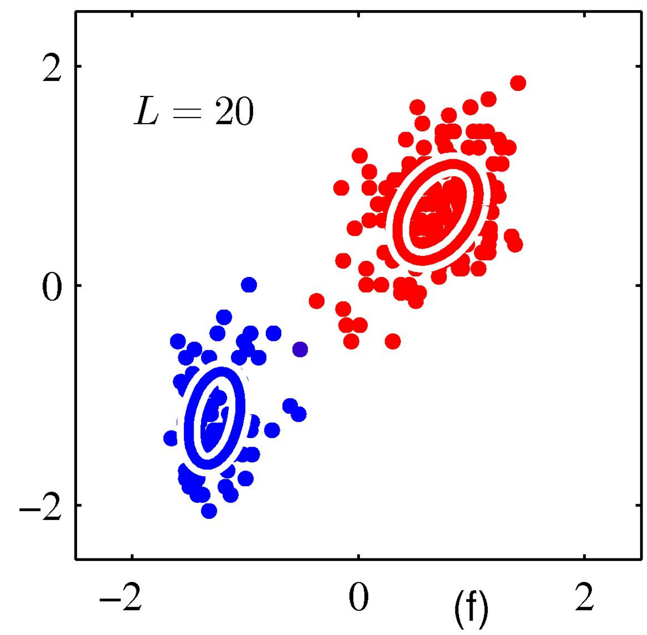
EM Example

- 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 $\sigma^2 \to 0$
- The update equations coverge to doing K-means clustering

Thank you!

Quiz: Scan QR code / click here

Next class: EM for Gaussian Mixtures, Principal Component Analysis