Gaussian Mixture Models & EM

Machine Learning

Dr. Hamid R Rabiee – Zahra Dehghanian
Spring 2025

Sharif University
of Technology

K-means Clustering

- **Input**: a set $\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(N)}$ of data points (in a d-dim feature space) and an integer K
- **Output**: a set of K representatives $\boldsymbol{c}_1, \boldsymbol{c}_2, \dots, \boldsymbol{c}_K \in \mathbb{R}^d$ as the cluster representatives
  - data points are assigned to the clusters according to their distances to $\boldsymbol{c}_1, \boldsymbol{c}_2, \dots, \boldsymbol{c}_K$
    - Each data is assigned to the cluster whose representative is nearest to it
- **Objective**: choose $\boldsymbol{c}_1, \boldsymbol{c}_2, \dots, \boldsymbol{c}_K$ to minimize:

$$\sum_{i=1}^{N} \min_{j \in 1,\dots,K} d^2(\boldsymbol{x}^{(i)}, \boldsymbol{c}_j)$$

Clustering Problem (Fuzzy Clustering)

- $\mathcal{X} = \{\boldsymbol{x}^{(i)}\}_{i=1}^{N}$
- $u_1, u_2, \dots, u_k$ are membership function $u_i: \mathcal{X} \to [0,1]$
  - $\forall j = 1, \dots k, \quad 0 < \sum_{i=1}^{N} u_j\left(\boldsymbol{x}^{(i)}\right) < N$
  - $\forall i = 1, \dots, N, \quad \sum_{j=1}^{k} u_j\left(\boldsymbol{x}^{(i)}\right) = 1$

Fuzzy (soft) clustering: $u_i(\boldsymbol{x})$ to what degree $\boldsymbol{x}$ belongs to cluster $\mathcal{C}_i$

Fuzzy c-means

- Cost function:

$$J(\boldsymbol{U}, \boldsymbol{\mu}) = \sum_{j=1}^{k} \sum_{i \in \mathcal{C}_j} (u_{ij})^q d(\boldsymbol{x}^{(i)}, \boldsymbol{\mu}_j)$$

  - $\forall j = 1, \dots k, \quad 0 < \sum_{i=1}^{N} u_{ij} < N$
  - $\forall i = 1, \dots, N, \ \sum_{j=1}^{k} u_{ij} = 1$
  - $\forall j = 1, \dots k, \forall i = 1, \dots, N, u_{ij} \in [0,1]$

- q is a fuzziness parameter (usually $1 < q < 2$)
  - Fuzzy clustering becomes crisp clustering when $q \to 1$

Fuzzy c-means

- Minimization of the cost function:

$$\frac{\partial J(\boldsymbol{U},\boldsymbol{\mu})}{\partial u_{ij}} = 0 \Rightarrow u_{ij} = \frac{1}{\sum_{l=1}^{k}\left(\frac{d(\boldsymbol{x}^{(i)},\boldsymbol{\mu}_j)}{d(\boldsymbol{x}^{(i)},\boldsymbol{\mu}_l)}\right)^{1/q}}$$

$$\frac{\partial J(\boldsymbol{U},\boldsymbol{\mu})}{\partial \boldsymbol{\mu}_j} = 0 \Rightarrow \sum_{i=1}^{N} u_{ij}^{q} \frac{\partial d(\boldsymbol{x}^{(i)},\boldsymbol{\mu}_j)}{\partial \boldsymbol{\mu}_j} = 0$$

- If $d\left(\boldsymbol{x}^{(i)},\boldsymbol{\mu}_j\right) = \left\|\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_j\right\|^2$

$$\Rightarrow \boldsymbol{\mu}_j = \frac{\sum_{i=1}^{N} u_{ij}^{q}\boldsymbol{x}^{(i)}}{\sum_{i=1}^{N} u_{ij}^{q}}$$

Fuzzy c-means

```
Select $k$ random points $\boldsymbol{\mu}_1(0), \boldsymbol{\mu}_2(0), \ldots \boldsymbol{\mu}_k(0)\}$ as clusters' initial centroids.
$t \leftarrow 0$

Repeat until a stopping criterion is reached:
    for i=1 to n do
     for j=1 to k do
          $u_{ij}(t) = \frac{1}{\sum_{l=1}^{k}\left(\frac{d(\boldsymbol{x}^{(i)},\boldsymbol{\mu}_j(t))}{d(\boldsymbol{x}^{(i)},\boldsymbol{\mu}_l(t))}\right)^{1/q}}$
    for j=1 to k do
        $\boldsymbol{\mu}_j(t+1) = \frac{\sum_{i=1}^{N} u_{ij}^{q}(t)\boldsymbol{x}^{(i)}}{\sum_{i=1}^{N} u_{ij}^{q}(t)}$
    $t \leftarrow t + 1$
```

Gaussian Mixture Models (GMMs)

-

- Gaussian Mixture Models: $p(\boldsymbol{x}|M_j;\boldsymbol{\theta}_j) \sim N(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$

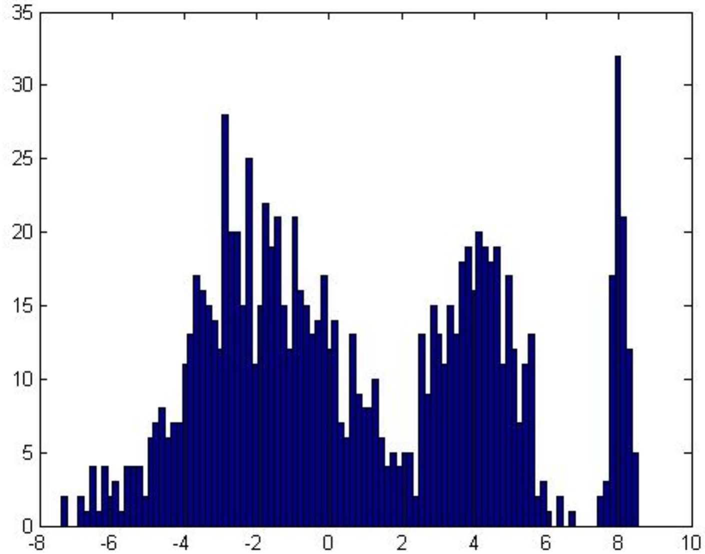
$$p(\boldsymbol{x}) = \sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$$

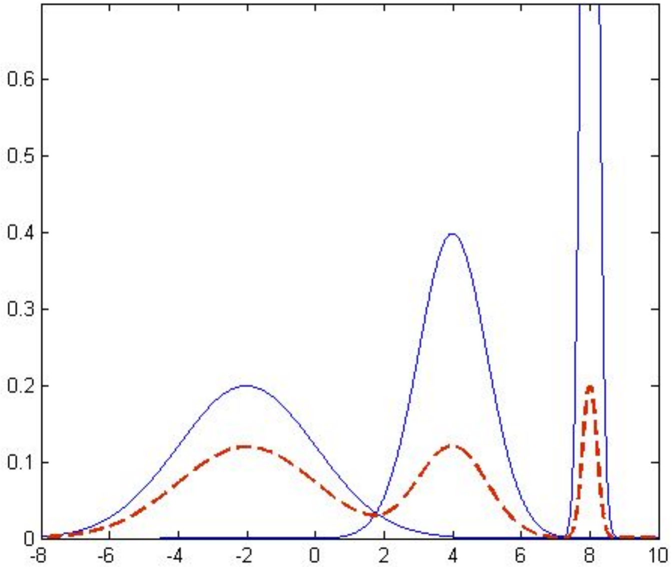
$$0 \leq \pi_j \leq 1$$

$$\sum_{j=1}^{K} \pi_j = 1$$

- Fitting the Gaussian mixture model
  - Input: data points $\{\boldsymbol{x}^{(i)}\}_{i=1}^{N}$
  - Goal: find the parameters of GMM $(\pi_j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j, j = 1, \dots, K)$

GMM: 1-D Example





$\mu_1 = -2$

$\sigma_1 = 2$

$\pi_1 = 0.6$

$\mu_2 = 4$

$\sigma_2 = 1$

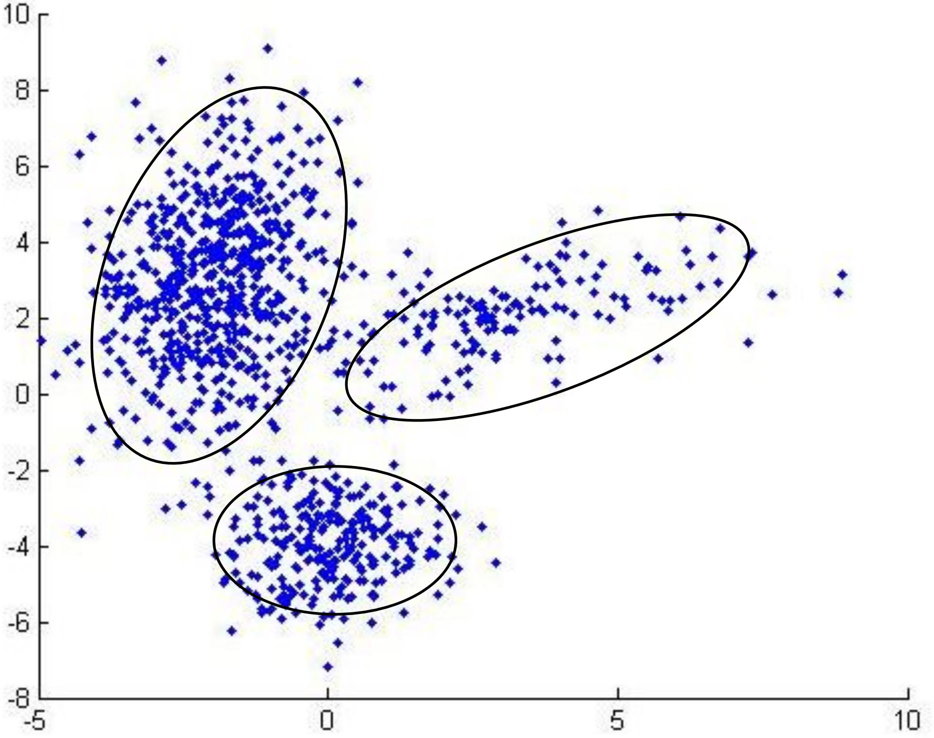
$\pi_2 = 0.3$

$\mu_3 = 8$

$\sigma_3 = 0.2$

$\pi_3 = 0.1$

GMM: 2-D Example



$\boldsymbol{\mu}_1 = [-2 \quad 3]$

$\Sigma_1 = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 4 \end{bmatrix}$

$\pi_1 = 0.6$

$\boldsymbol{\mu}_2 = [0 \quad -4]$

$\Sigma_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

$\pi_2 = 0.25$

$\boldsymbol{\mu}_3 = [3 \quad 2]$

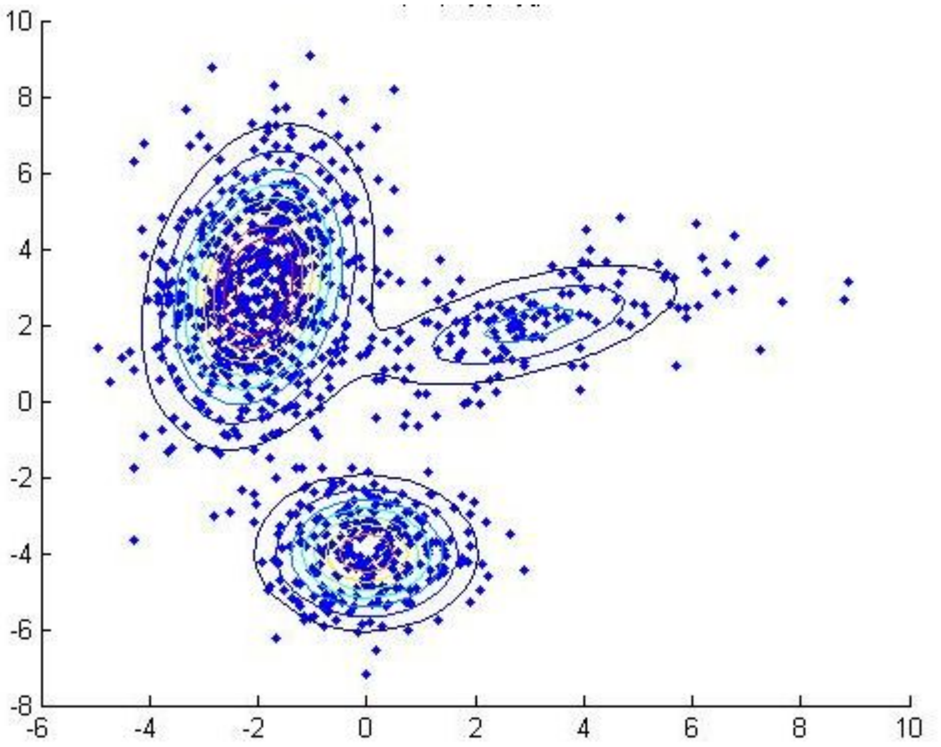
$\Sigma_3 = \begin{bmatrix} 3 & 1 \\ 1 & 1 \end{bmatrix}$

$\pi_3 = 0.15$

k = 3

GMM: 2-D Example

- GMM distribution



k = 3

$$\boldsymbol{\mu}_1 = \begin{bmatrix} -2 & 3 \end{bmatrix}$$
$$\Sigma_1 = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 4 \end{bmatrix}$$
$$\pi_1 = 0.6$$

$$\boldsymbol{\mu}_2 = \begin{bmatrix} 0 & -4 \end{bmatrix}$$
$$\Sigma_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$
$$\pi_2 = 0.25$$

$$\boldsymbol{\mu}_3 = \begin{bmatrix} 3 & 2 \end{bmatrix}$$
$$\Sigma_3 = \begin{bmatrix} 3 & 1 \\ 1 & 1 \end{bmatrix}$$
$$\pi_3 = 0.15$$

Mixture Models: definition

- Mixture models: Linear supper-position of mixtures or components

$$p(\boldsymbol{x}|\boldsymbol{\theta}) = \sum_{j=1}^{K} P(z=j)\, p\left(\boldsymbol{x}\middle|z=j;\boldsymbol{\theta}_j\right)$$

  - $\sum_{j=1}^{K} P(z=j) = 1$
  - $P(z=j)$: the prior probability of j-th mixture
  - $\boldsymbol{\theta}_j$: the parameters of j-th mixture
  - $p\left(\boldsymbol{x}\middle|z=j;\boldsymbol{\theta}_j\right)$: the probability of $\boldsymbol{x}$ according to j-th mixture

- Framework for finding more complex probability distributions
  - Goal: estimate $p(\boldsymbol{x}|\theta)$ E.g., Multi-modal density estimation

How to Fit GMM?

- In order to maximize log likelihood:

$$\boldsymbol{X} = \{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(N)}\}$$

$$\ln p(\boldsymbol{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{i=1}^{N} \ln \left\{ \sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) \right\}$$

  - The sum over components appears inside the log and there is no closed form solution for maximum likelihood.

$$\frac{\partial \ln p(\boldsymbol{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma})}{\partial \boldsymbol{\mu}_k} = \boldsymbol{0}$$

$$\frac{\partial \ln p(\boldsymbol{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma})}{\partial \boldsymbol{\Sigma}_k} = \boldsymbol{0}$$

$$\frac{\partial \ln p(\boldsymbol{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) + \lambda\left(\sum_{j=1}^{K} \pi_j - 1\right)}{\partial \pi_k} = 0$$

$$k = 1, \dots, K$$

ML for GMM: Alternating optimization

- $$\boldsymbol{\mu}_k = \frac{1}{N_k}\sum_{i=1}^{N} \boxed{\frac{\pi_k \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}} \boldsymbol{x}^{(i)}$$

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k}\sum_{i=1}^{N} \boxed{\frac{\pi_k \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}} (\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_k)(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_k)^T$$

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

$$N_k = \sum_{i=1}^{N} \boxed{\frac{\pi_k \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}}$$

$$\frac{\partial \log|\boldsymbol{A}^{-1}|}{\partial \boldsymbol{A}^{-1}} = \boldsymbol{A}^T \qquad \frac{\partial \boldsymbol{x}^T \boldsymbol{A} \boldsymbol{x}}{\partial \boldsymbol{A}} = \boldsymbol{x}\boldsymbol{x}^T$$

Mixture models: discrete latent variables

- $$p(\boldsymbol{x}) = \sum P(z_j = 1) p(\boldsymbol{x}|z_j = 1) = \sum_{j=1}^{K} \pi_j \, p(\boldsymbol{x}|z_j = 1)$$

- z: latent or hidden variable
  - specifies the mixture component

- $P(z_j = 1) = \pi_j$
  - $0 \leq \pi_j \leq 1$
  - $\sum_{j=1}^{K} \pi_j = 1$

EM algorithm

- An iterative algorithm in which each iteration is guaranteed to improve the log-likelihood function

- General algorithm for finding ML estimation when the data is incomplete (missing or unobserved data).
    - EM finds the maximum likelihood parameters in cases where the models involve unobserved variables Z in addition to unknown parameters θ and known data observations X.

EM for GMM

- Initialize $\boldsymbol{\theta}_k \; k = 1, \dots, K$
- **E step**: $i = 1, \dots, N, j = 1, \dots, K$

$$\gamma_j^i = P\left(z_j^{(i)} = 1 | \boldsymbol{x}^{(i)}\right) = \frac{P\left(\boldsymbol{x}^{(i)} | z_j^{(i)} = 1\right) P\left(z_j^{(i)} = 1\right)}{\sum_k P\left(\boldsymbol{x}^{(i)} | z_k^{(i)} = 1\right) P\left(z_k^{(i)} = 1\right)}$$

- **M Step**: $j = 1, \dots, K$
  - Update parameters $\boldsymbol{\theta}_j$

$\boldsymbol{\theta} = [\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}]$

- Repeat E and M steps until convergence

$z^{(i)}$ is a one-hot vector shows the mixture from which $x^{(i)}$ is generated

EM for GMM

- Initialize $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k \quad k = 1, \dots, K$
- **E step**: $i = 1, \dots, N, j = 1, \dots, K$

$$\gamma_j^i = P\left(z_j^{(i)} = 1 | \boldsymbol{x}^{(i)}, \boldsymbol{\theta}^{old}\right) = \frac{\pi_j^{old} \mathcal{N}(\boldsymbol{x}^{(i)} | \boldsymbol{\mu}_j^{old}, \boldsymbol{\Sigma}_j^{old})}{\sum_{k=1}^{K} \pi_k^{old} \mathcal{N}(\boldsymbol{x}^{(i)} | \boldsymbol{\mu}_k^{old}, \boldsymbol{\Sigma}_k^{old})}$$

- **M Step**: $j = 1, \dots, K$

$$\boldsymbol{\mu}_j^{new} = \frac{\sum_{i=1}^{N} \gamma_j^i \boldsymbol{x}^{(i)}}{\sum_{i=1}^{N} \gamma_j^i}$$

$$\boldsymbol{\Sigma}_j^{new} = \frac{1}{\sum_{i=1}^{N} \gamma_j^i} \sum_{i=1}^{N} \gamma_j^i (\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_j^{\text{new}})(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_j^{\text{new}})^T$$

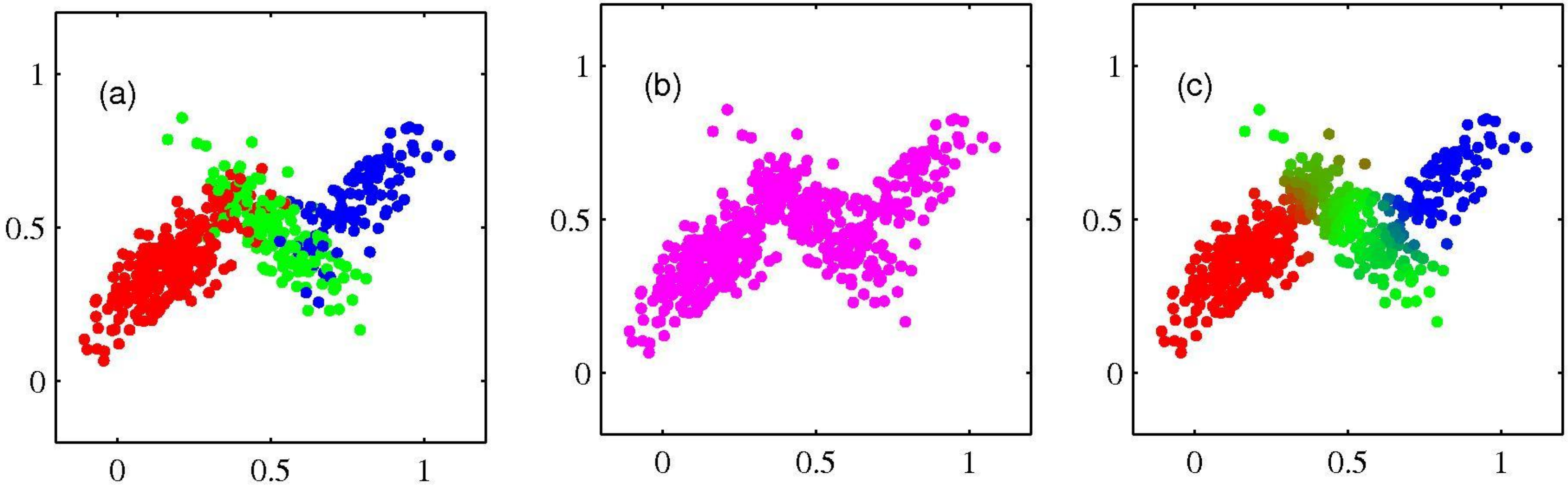
$$\pi_j^{\text{new}} = \frac{\sum_{i=1}^{N} \gamma_j^i}{N}$$

- Repeat E and M steps until convergence

$\boldsymbol{\theta} = [\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}]$

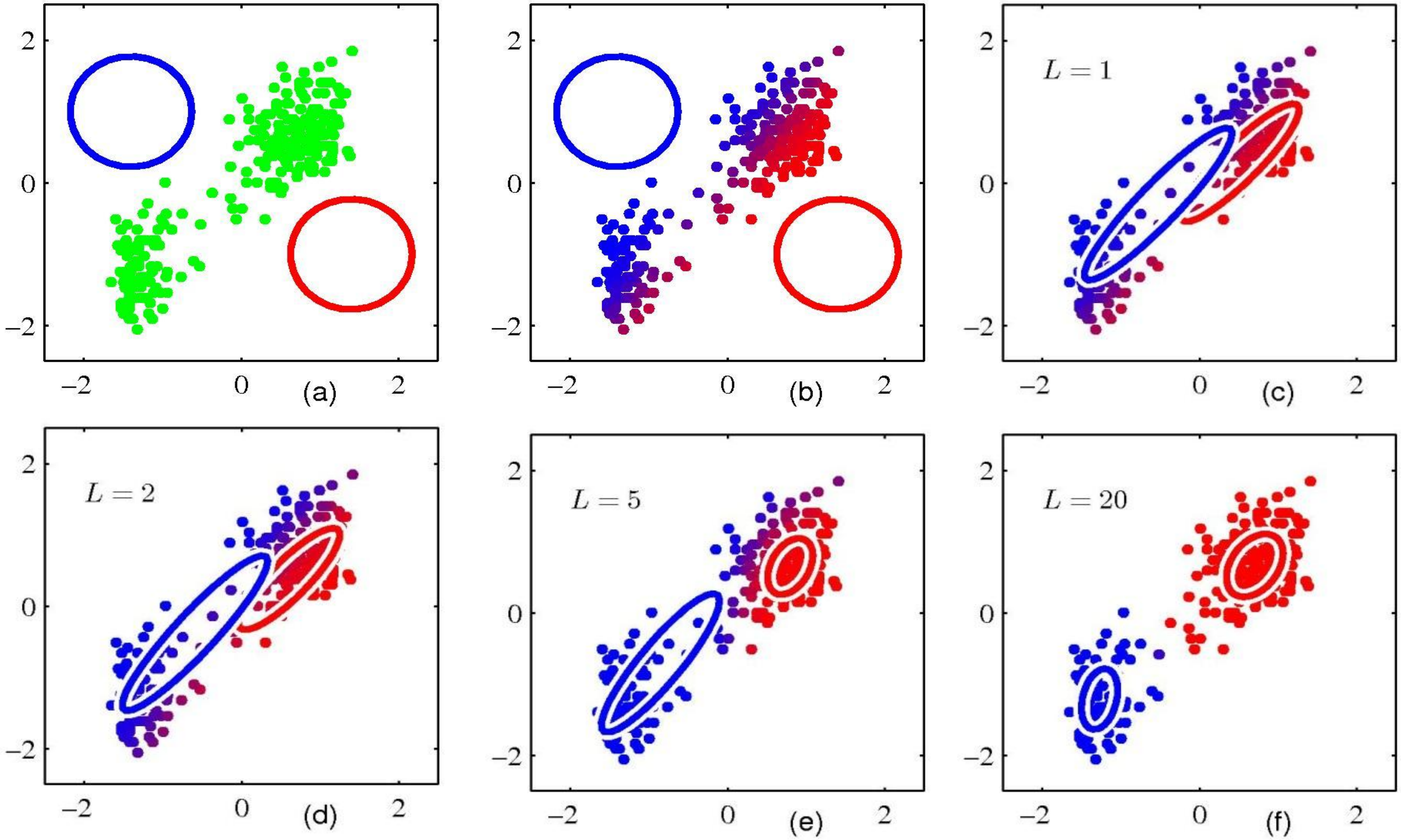
$z^{(i)} \in \{1, 2, \dots, K\}$ shows the mixture from which $x^{(i)}$ is generated

EM & GMM: example



[Bishop]

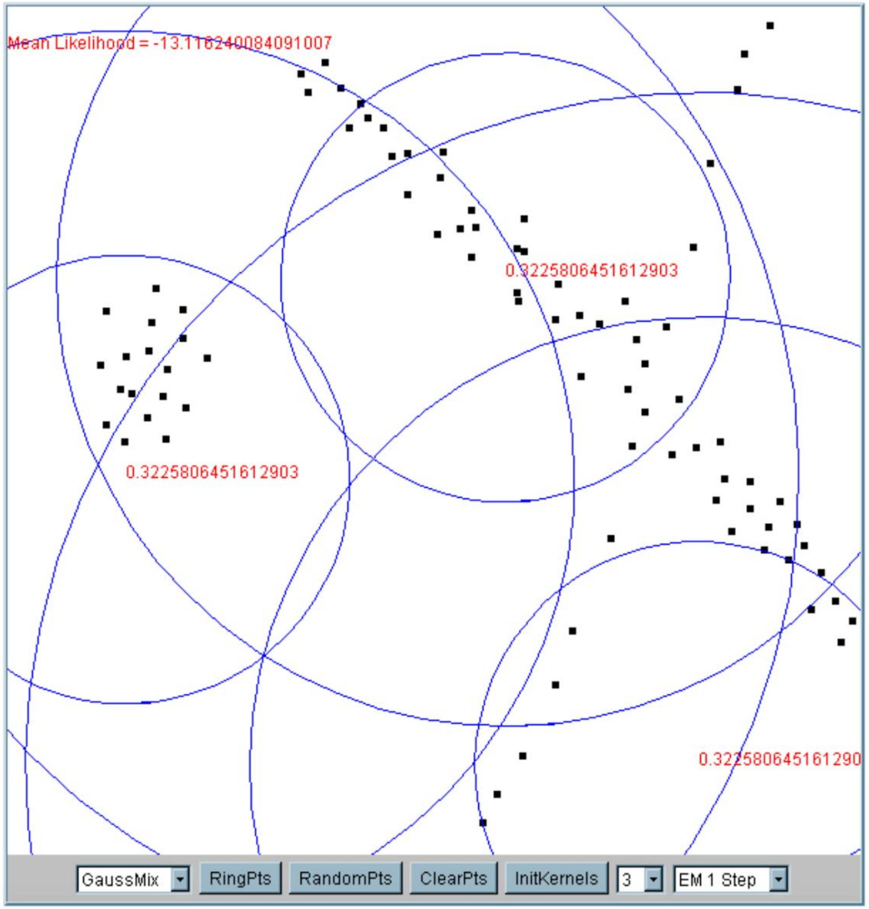
EM & GMM: Example



Example

Iteration 1

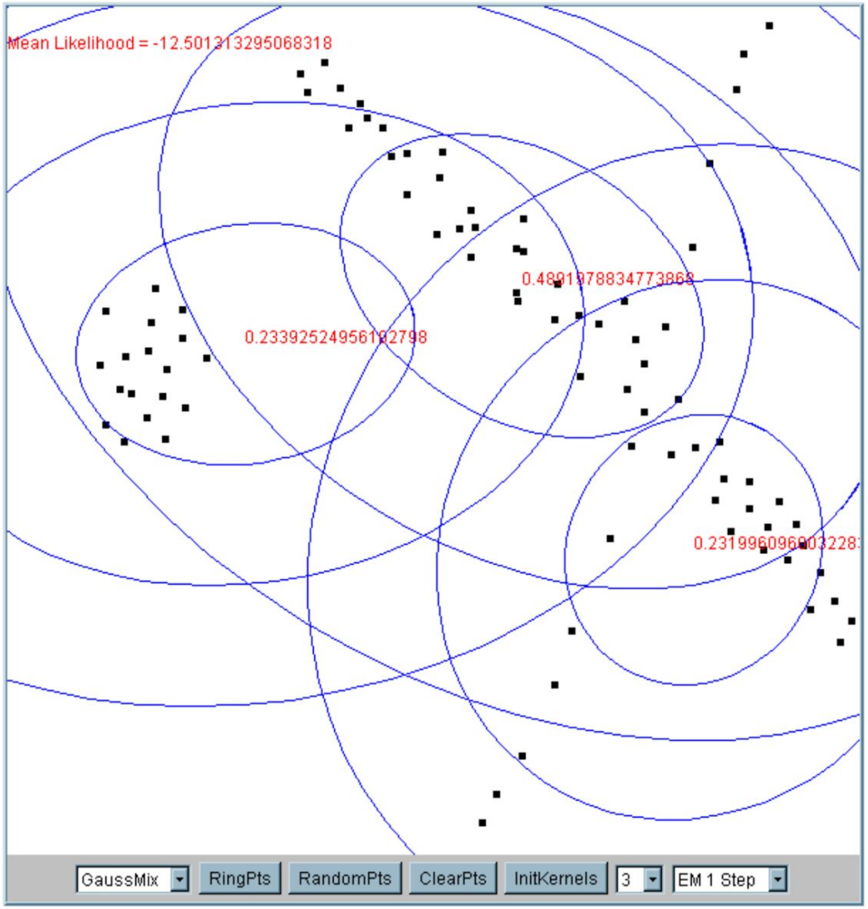
The cluster means are randomly assigned



Mean Likelihood = -13.116240084091007

Example

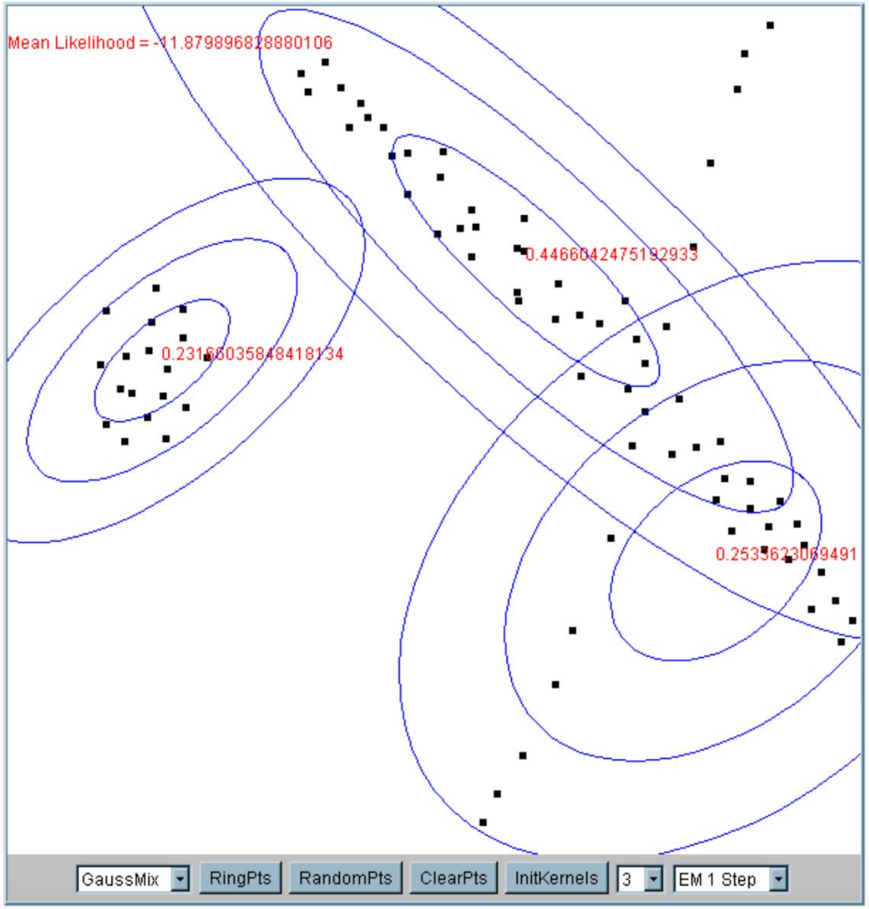
Iteration 2



Mean Likelihood = -12.501313295068318

Example

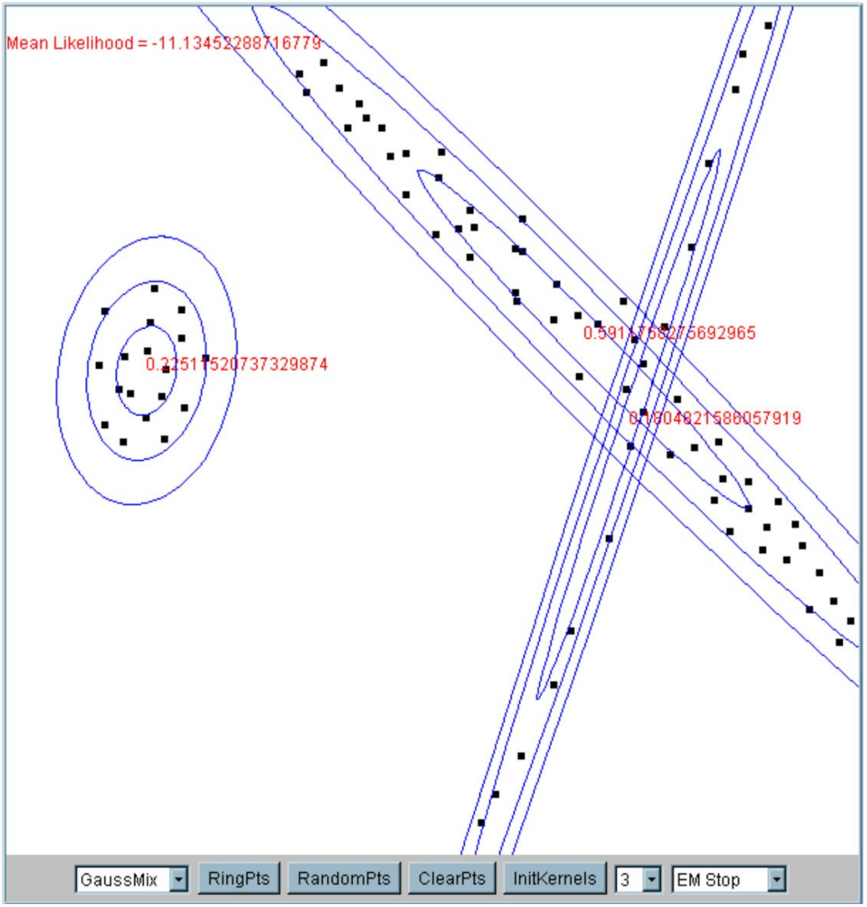
Iteration 5



Mean Likelihood = -11.879896828880106

Example

Iteration 25



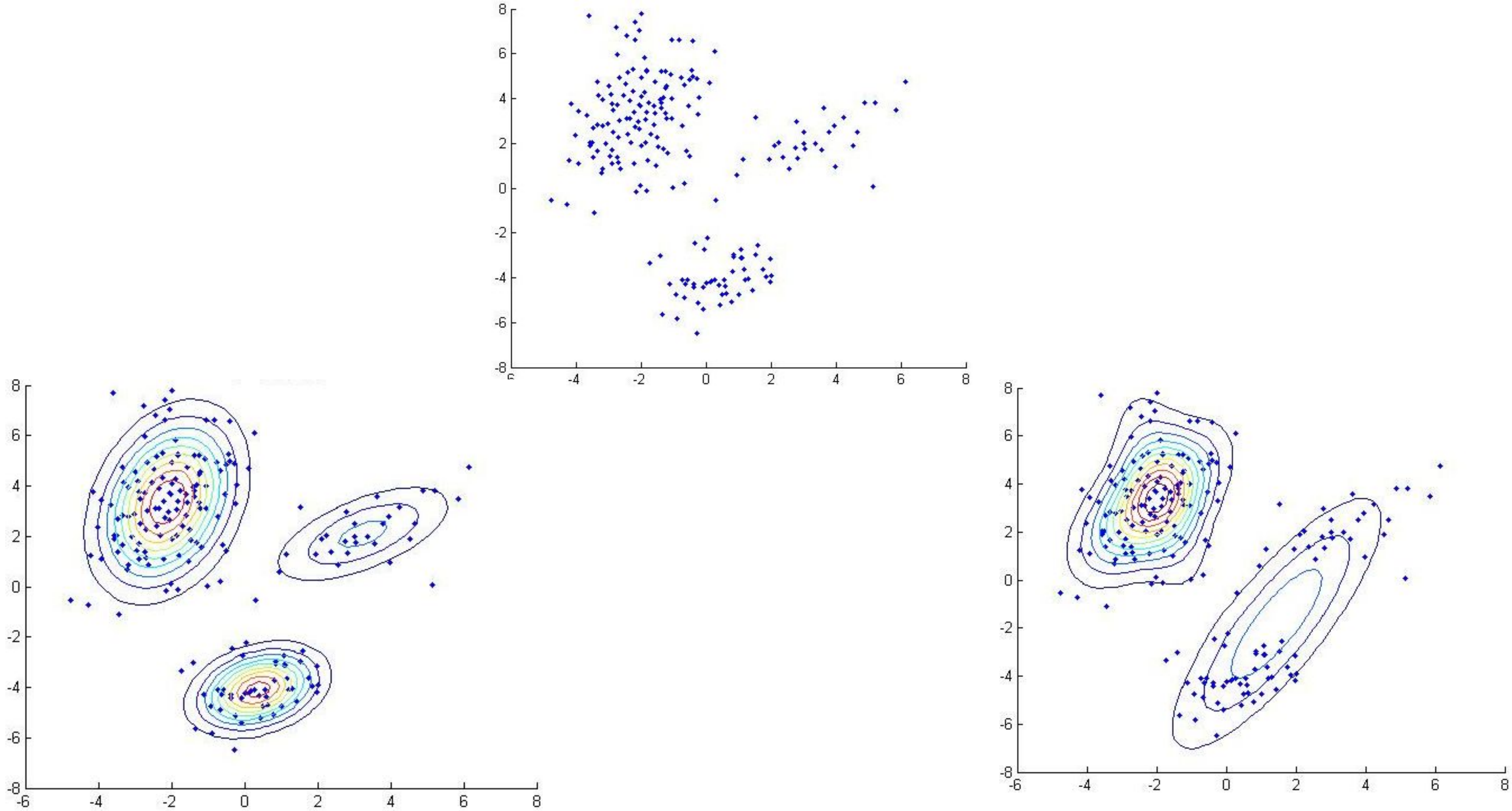
Mean Likelihood = -11.13452288716779

0.22511520737329874

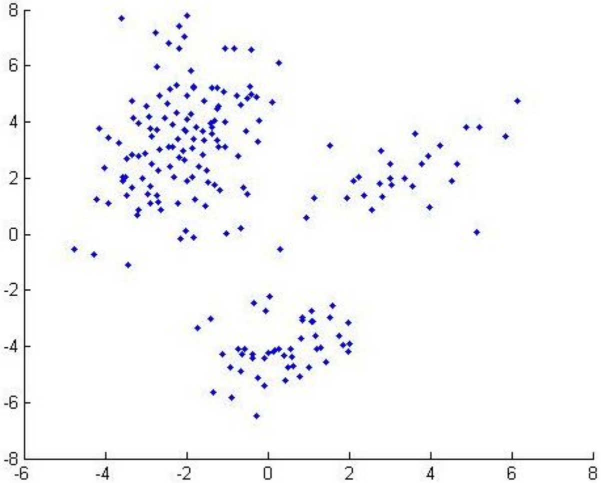
0.5911768275692965

0.1804821586057919

Local Minima



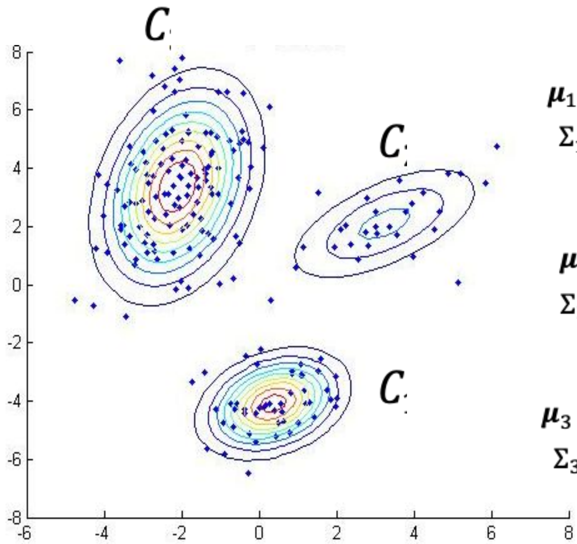
Local Minima



$$\boldsymbol{\mu}_1 = [-2 \quad 3]$$
$$\Sigma_1 = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 4 \end{bmatrix}$$
$$\pi_1 = 0.6$$

$$\boldsymbol{\mu}_2 = [0 \quad -4]$$
$$\Sigma_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$
$$\pi_2 = 0.25$$

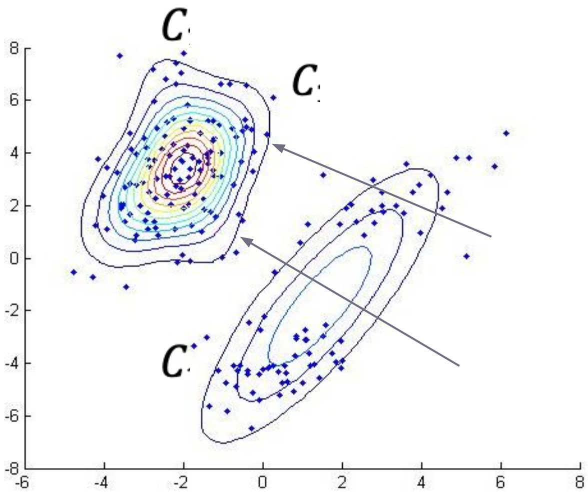
$$\boldsymbol{\mu}_3 = [3 \quad 2]$$
$$\Sigma_3 = \begin{bmatrix} 3 & 1 \\ 1 & 1 \end{bmatrix}$$
$$\pi_3 = 0.15$$



$$\boldsymbol{\mu}_1 = [0.36 \quad -4.09]$$
$$\Sigma_1 = \begin{bmatrix} 0.89 & 0.26 \\ 0.26 & 0.83 \end{bmatrix}$$
$$\pi_1 = 0.249$$

$$\boldsymbol{\mu}_2 = [3.25 \quad 2.09]$$
$$\Sigma_2 = \begin{bmatrix} 2.23 & 1.08 \\ 1.09 & 1.41 \end{bmatrix}$$
$$\pi_2 = 0.146$$

$$\boldsymbol{\mu}_3 = [-2.11 \quad 3.36]$$
$$\Sigma_3 = \begin{bmatrix} 1.12 & 0.61 \\ 0.61 & 3.61 \end{bmatrix}$$
$$\pi_3 = 0.604$$

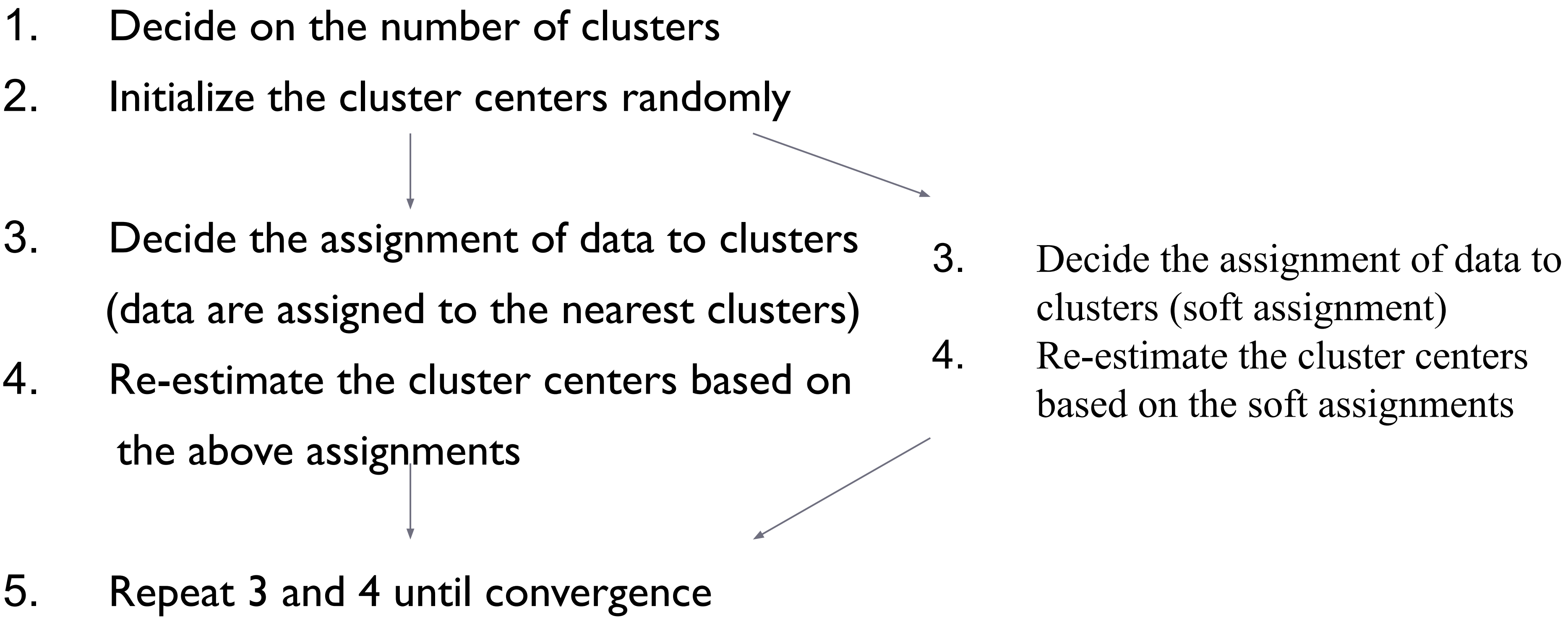


$$\boldsymbol{\mu}_1 = [1.45 \quad -1.81]$$
$$\Sigma_1 = \begin{bmatrix} 3.30 & 4.76 \\ 4.76 & 10.01 \end{bmatrix}$$
$$\pi_1 = 0.392$$

$$\boldsymbol{\mu}_2 = [-2.20 \quad 3.16]$$
$$\Sigma_2 = \begin{bmatrix} 1.30 & 1.10 \\ 1.10 & 2.80 \end{bmatrix}$$
$$\pi_2 = 0.429$$

$$\boldsymbol{\mu}_3 = [-1.88 \quad 3.74]$$
$$\Sigma_3 = \begin{bmatrix} 5.83 & -0.82 \\ -0.82 & 5.83 \end{bmatrix}$$
$$\pi_3 = 0.178$$

K-means and EM on GMM



1. Decide on the number of clusters
2. Initialize the cluster centers randomly
3. Decide the assignment of data to clusters (data are assigned to the nearest clusters)
4. Re-estimate the cluster centers based on the above assignments

3. Decide the assignment of data to clusters (soft assignment)
4. Re-estimate the cluster centers based on the soft assignments

5. Repeat 3 and 4 until convergence

EM+GMM vs. k-means

- k-means:
  - It is not probabilistic
  - Has fewer parameters (and faster)
  - Limited by the underlying assumption of spherical clusters
    - can be extended to use covariance – get "hard EM" (ellipsoidal k-means).

- Both EM and k-means depend on initialization
  - getting stuck in local optima
    - EM+GMM has more local minima
    - Useful trick: first run k-means and then use its result to initialize EM.

How many clusters?

- **Cross validation** to determine the correct number of classes
- Likelihood of the left out data to determine which model (number of clusters) is more accurate

$$\prod_{n \in \text{Validation}} \sum_{k=1}^{K} p\left(x^{(n)} \middle| \theta_k\right) \pi_k$$

Gaussian Mixture Models

- Advantages
  - Optimizes data likelihood
  - Learns a generative model of data
    - can generate new data according to the learned model
  - Relatively efficient: linear in the number of data, number of clusters, number of iterations, and quadratic in the number of dimensions
- Weakness
  - Often terminates at a local optimum.
    - Initialization is important.
  - Not suitable to discover clusters with non-convex shapes

Incomplete log likelihood

- Complete log likelihood
  - Maximizing likelihood (i.e., $\log P(X, Y|\boldsymbol{\theta})$) for labeled data is straightforward

- Incomplete log likelihood
  - With Z unobserved, our objective becomes the log of a marginal probability $\log P(X|\boldsymbol{\theta}) = \log \sum_Z P(X, Z|\boldsymbol{\theta})$
    - This objective will not decouple and we use EM algorithm to solve it
  - $X = \{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(N)}\}$
  - $Z = \{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(N)}\}$

EM Algorithm

•

- Assumptions: X (observed or known variables), Z (unobserved or latent variables), X come from a specific model with unknown parameters $\boldsymbol{\theta}$
  - If Z is relevant to X (in any way), we can hope to extract information about it from X assuming a specific parametric model on the data.

- Steps:
  - Initialization: Initialize the unknown parameters $\boldsymbol{\theta}$
  - Iterate the following steps, until convergence:
    - **Expectation step**: Find the probability of unobserved variables given the current parameters estimates and the observed data.
    - **Maximization step**: from the observed data and the probability of the unobserved data find the most likely parameters (a better estimate for the parameters).

EM algorithm intuition

- When learning with **hidden variables**, we are trying to solve two problems at once:
  - **hypothesizing values for the unobserved variables** in each data sample
  - learning the **parameters**

- Each of these tasks is fairly **easy** when we have the **solution to the other.**
  - Given complete data, we have the statistics, and we can estimate parameters using the MLE formulas.
  - Conversely, computing probability of missing data given the parameters is a probabilistic inference problem

EM algorithm: general

X: observed variables

Z: unobserved variables

$\boldsymbol{\theta}$: parameters

Expectation step (E-step): Given the current parameters, find soft completion of data using probabilistic inference

Maximization step (M-step): Treat the soft completed data as if it were observed and learn a new set of parameters

```
Choose an initial setting $\boldsymbol{\theta}^0, t = 0$
Iterate until convergence:
    E Step: Use $X$ and current $\boldsymbol{\theta}^t$ to calculate $P(Z|X, \boldsymbol{\theta}^t)$
    M Step: $\boldsymbol{\theta}^{t+1} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} E_{Z\sim P(Z|X,\boldsymbol{\theta}^t)}[\log p(X, Z|\boldsymbol{\theta})]$
    $t \leftarrow t + 1$
```

expectation of the log-likelihood evaluated using the current estimate for the parameters $\boldsymbol{\theta}^t$

$$E_{Z\sim P(Z|X,\boldsymbol{\theta}^{\text{old}})}[\log p(X, Z|\boldsymbol{\theta})]$$

$$= \sum_Z P(Z|X, \boldsymbol{\theta}^{\text{old}}) \times \log p(X, Z|\boldsymbol{\theta})$$

EM theoretical foundation: Objective function

$$X = \{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(n)}\}$$
$$Z = \{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(n)}\}$$

$$\ell(\boldsymbol{\theta}; X) = \log P(X|\boldsymbol{\theta}) = \log \sum_{Z} P(X, Z|\boldsymbol{\theta})$$

$$= \log \sum_{Z} Q(Z) \frac{P(X, Z|\boldsymbol{\theta})}{Q(Z)} \overset{\text{Jensen inequality}}{\geq} \underbrace{\sum_{Z} Q(Z) \log \frac{P(X, Z|\boldsymbol{\theta})}{Q(Z)}}_{F[\boldsymbol{\theta}, Q]}$$

$F[\boldsymbol{\theta}, Q]$ is a lower bound on $\ell(\boldsymbol{\theta}; X)$

EM maximizes $F[\boldsymbol{\theta}, Q]$

EM theoretical foundation: Algorithm in general form

- EM is a coordinate ascent algorithm on $F[\boldsymbol{\theta}, Q]$. In the t-th iteration,
  - E-step: maximize $F[\boldsymbol{\theta}, Q]$ w.r.t. Q

$$Q^t = \underset{Q}{\operatorname{argmax}} F[\boldsymbol{\theta}^t, Q]$$

  - M-step:

$$\boldsymbol{\theta}^{t+1} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} F[\boldsymbol{\theta}, Q^t]$$

We will show that each iteration improves the log-likelihood

EM algorithm: general

- EM: general procedure for learning from partly observed data
- Define: $\mathrm{Q}(\boldsymbol{\theta};\boldsymbol{\theta}^{\mathrm{old}}) = E_{Z\sim P(Z|X,\boldsymbol{\theta}^{\mathrm{old}})}[\log p(X,Z|\boldsymbol{\theta})]$

$$= \sum_Z P(Z|X,\boldsymbol{\theta}^{\mathrm{old}}) \times \log p(X,Z|\boldsymbol{\theta})$$

expectation of the log-likelihood evaluated using the current estimate for the parameters $\boldsymbol{\theta}^{\mathrm{old}}$

```
Choose an initial setting $\boldsymbol{\theta}^{\mathrm{old}} = \boldsymbol{\theta}^{0}$
Iterate until convergence:
    E Step: Use $X$ and current $\boldsymbol{\theta}^{\mathrm{old}}$ to calculate $P(Z|X,\boldsymbol{\theta}^{\mathrm{old}})$
    M Step: $\boldsymbol{\theta}^{\mathrm{new}} = \underset{\boldsymbol{\theta}}{\mathrm{argmax}}\, \mathrm{Q}(\boldsymbol{\theta};\boldsymbol{\theta}^{\mathrm{old}})$
    $\boldsymbol{\theta}^{\mathrm{old}} \leftarrow \boldsymbol{\theta}^{\mathrm{new}}$
```

EM theoretical analysis

- What is the underlying theory for the use of the expected complete log likelihood in the M-step?

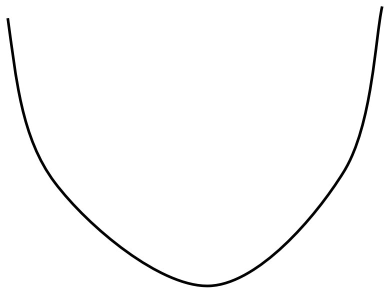
$$E_{P\left(Z\middle|X,\boldsymbol{\theta}^{old}\right)}\left[\log P(X,Z|\boldsymbol{\theta})\right]$$

- Now, we show that maximizing this function also maximizes the likelihood

Jensen's inequality

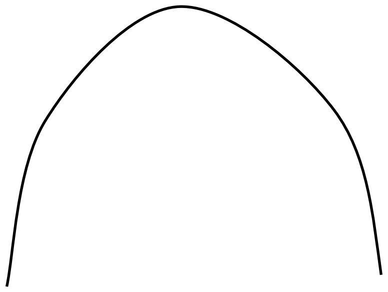
- If f is a convex function

$$E[f(x)] \geq f(E[x])$$



- If f is a concave function

$$E[f(x)] \leq f(E[x])$$



EM theoretical foundation: E-step

- $$Q^t = P(Z|X,\boldsymbol{\theta}^t) \implies Q^t = \underset{Q}{\operatorname{argmax}} F[\boldsymbol{\theta}^t, Q]$$

Proof:

$$F[\boldsymbol{\theta}^t, P(Z|X,\boldsymbol{\theta}^t)] = \sum_{Z} P(Z|X,\boldsymbol{\theta}^t) \log \frac{P(X,Z|\boldsymbol{\theta}^t)}{P(Z|X,\boldsymbol{\theta}^t)}$$

$$= \sum_{Z} P(Z|X,\boldsymbol{\theta}^t) \log P(X|\boldsymbol{\theta}^t) = \log P(X|\boldsymbol{\theta}^t) = \ell(\boldsymbol{\theta}^t; X)$$

EM theoretical foundation: E-step

- $$Q^t = P(Z|X,\boldsymbol{\theta}^t) \implies Q^t = \underset{Q}{\operatorname{argmax}}\, F[\boldsymbol{\theta}^t, Q]$$

Proof:

$$F[\boldsymbol{\theta}^t, P(Z|X,\boldsymbol{\theta}^t)] = \sum_Z P(Z|X,\boldsymbol{\theta}^t) \log \frac{P(X,Z|\boldsymbol{\theta}^t)}{P(Z|X,\boldsymbol{\theta}^t)}$$

$$= \sum_Z P(Z|X,\boldsymbol{\theta}^t) \log P(X|\boldsymbol{\theta}^t) = \log P(X|\boldsymbol{\theta}^t) = \ell(\boldsymbol{\theta}^t; X)$$

- $F[\boldsymbol{\theta}, Q]$ is a lower bound on $\ell(\boldsymbol{\theta}; X)$. Thus, $F[\boldsymbol{\theta}^t, Q]$ has been maximized by setting Q to $P(Z|X,\boldsymbol{\theta}^t)$:

$$F[\boldsymbol{\theta}^t, P(Z|X,\boldsymbol{\theta}^t)] = \ell(\boldsymbol{\theta}^t; X)$$

$$\Rightarrow P(Z|X,\boldsymbol{\theta}^t) = \underset{Q}{\operatorname{argmax}}\, F[\boldsymbol{\theta}^t, Q]$$

EM theoretical foundation: M-step

- M-step can be equivalently viewed as maximizing the expected complete log-likelihood:

$$\boldsymbol{\theta}^{t+1} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, F[\boldsymbol{\theta}, Q^t] = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, E_{Q^t}[\log P(X, Z|\boldsymbol{\theta})]$$

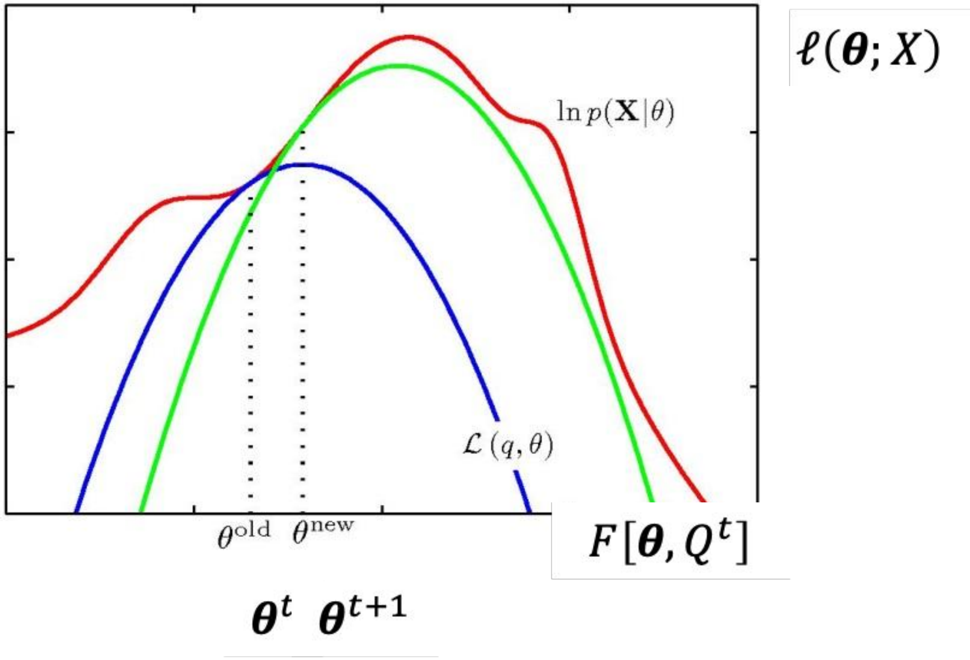
Proof:

$$F[\boldsymbol{\theta}, Q^t] = \sum_Z Q^t(Z) \log \frac{P(X, Z|\boldsymbol{\theta})}{Q^t(Z)}$$

$$= \sum_Z Q^t(Z) \log P(X, Z|\boldsymbol{\theta}) - \sum_Z Q^t(Z) \log Q^t(Z)$$

$$\Rightarrow F[\boldsymbol{\theta}, Q^t] = E_{Q^t}[\log P(X, Z|\boldsymbol{\theta})] + \underbrace{H(Q^t(Z))}_{\text{Independent of } \boldsymbol{\theta}}$$

EM algorithm: illustration



[Bishop]

EM iteration increases $\ell(\boldsymbol{\theta}; X)$

- $$\ell(\boldsymbol{\theta}^t; X) = E_{Q^t}[\log P(X, Z|\boldsymbol{\theta}^t)] + H(Q^t(Z))$$

  $$\ell(\boldsymbol{\theta}^{t+1}; X) \geq E_{Q^t}[\log P(X, Z|\boldsymbol{\theta}^{t+1})] + H(Q^t(Z))$$

  $$\ell(\boldsymbol{\theta}^{t+1}; X) - \ell(\boldsymbol{\theta}^t; X) \geq E_{Q^t}[\log P(X, Z|\boldsymbol{\theta}^{t+1})] - E_{Q^t}[\log P(X, Z|\boldsymbol{\theta}^t)]$$

EM iteration increases $\ell(\boldsymbol{\theta}; X)$

- $$\ell(\boldsymbol{\theta}^t; X) = E_{Q^t}[\log P(X, Z|\boldsymbol{\theta}^t)] + H(Q^t(Z))$$

$$\ell(\boldsymbol{\theta}^{t+1}; X) \geq E_{Q^t}[\log P(X, Z|\boldsymbol{\theta}^{t+1})] + H(Q^t(Z))$$

$$\ell(\boldsymbol{\theta}^{t+1}; X) - \ell(\boldsymbol{\theta}^t; X) \geq E_{Q^t}[\log P(X, Z|\boldsymbol{\theta}^{t+1})] - E_{Q^t}[\log P(X, Z|\boldsymbol{\theta}^t)]$$

Moreover, we have:

$$\boldsymbol{\theta}^{t+1} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} E_{Q^t}[\log P(X, Z|\boldsymbol{\theta})]$$

$$\Rightarrow E_{Q^t}[\log P(X, Z|\boldsymbol{\theta}^{t+1})] \geq E_{Q^t}[\log P(X, Z|\boldsymbol{\theta}^t)]$$

$$\Rightarrow \ell(\boldsymbol{\theta}^{t+1}; X) - \ell(\boldsymbol{\theta}^t; X) \geq 0$$

EM is increasing the log likelihood constantly

EM Convergence

- Objective function is **bounded** (under mild assumptions), so EM is guaranteed to **converge to a stationary point of log likelihood.**

- EM is guaranteed to find a stationary point of the log likelihood
  - A stationary point for the objective of EM is also a stationary point of log-likelihood
  - However, it can be a **local maximum or a saddle point**

EM advantages and disadvantages

- Some good things about EM:
  - **no learning rate** (step-size) parameter
  - automatically enforces parameter constraints
  - very **fast** for low dimensions
  - each iteration is guaranteed to improve likelihood

- Some bad things about EM:
  - can get stuck in **local minima**
  - can be slower than some other iterative gradient-based methods
  - requires expensive inference step (E-step)

EM for GMM : E step details

$$P(\mathbf{z}|\boldsymbol{x},\boldsymbol{\theta}) = \frac{P(\boldsymbol{x},\mathbf{z}|\boldsymbol{\theta})}{\sum_{\mathbf{z}} P(\boldsymbol{x},\mathbf{z}|\boldsymbol{\theta})} = \frac{P(\boldsymbol{x}|\mathbf{z},\boldsymbol{\theta})P(\mathbf{z}|\boldsymbol{\theta})}{\sum_{\mathbf{z}} P(\boldsymbol{x}|\mathbf{z},\boldsymbol{\theta})P(\mathbf{z}|\boldsymbol{\theta})}$$

$$\boldsymbol{\theta} = [\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}]$$

$$Q_j(z_j = 1) = P(z_j = 1|\boldsymbol{x},\boldsymbol{\theta}) = \frac{\pi_j \ \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}{\sum_{k=1}^{K} \pi_k \ \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}$$

EM for GMM
M step: details

$$p(X,Z|\boldsymbol{\theta}) = \prod_{i=1}^{N} p(\boldsymbol{x}^{(i)},\boldsymbol{z}^{(i)}|\boldsymbol{\theta})$$

$$= \prod_{i=1}^{N} p(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)},\boldsymbol{\theta})p(\boldsymbol{z}^{(i)}|\boldsymbol{\pi})$$

$$= \prod_{i=1}^{N}\prod_{j=1}^{K} \mathcal{N}\left(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_j\ ,\boldsymbol{\Sigma}_j\ \right)^{z_j^{(i)}} \pi_j{}^{z_j^{(i)}}$$

$$\boldsymbol{\theta} = [\boldsymbol{\pi},\boldsymbol{\mu},\boldsymbol{\Sigma}]$$

$$\boldsymbol{\theta}^{old} = [\boldsymbol{\pi}^{old},\boldsymbol{\mu}^{old},\boldsymbol{\Sigma}^{\mathbf{old}}]$$

EM for GMM
M step: details

- $p(X,Z|\boldsymbol{\theta}) = \prod_{i=1}^{N} p(\boldsymbol{x}^{(i)},\boldsymbol{z}^{(i)}|\boldsymbol{\theta}) = \prod_{i=1}^{N} p(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)},\boldsymbol{\theta})p(\boldsymbol{z}^{(i)}|\boldsymbol{\pi})$

$$= \prod_{i=1}^{N}\prod_{j=1}^{K} \mathcal{N}\left(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_j\,,\boldsymbol{\Sigma}_j\;\right)^{z_j^{(i)}}\;\pi_j^{\,z_j^{(i)}}$$

$$\boldsymbol{\theta} = [\boldsymbol{\pi},\boldsymbol{\mu},\boldsymbol{\Sigma}]$$

$$\boldsymbol{\theta}^{old} = [\boldsymbol{\pi}^{old},\boldsymbol{\mu}^{old},\boldsymbol{\Sigma}^{\mathbf{old}}]$$

$$\log p(X,Z|\boldsymbol{\theta}) = \sum_{i=1}^{N}\sum_{j=1}^{K} z_j^{(i)}\{\log\mathcal{N}\left(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_j\,,\boldsymbol{\Sigma}_j\;\right)+\log\pi_j\}$$

$$E_{Z\sim P(Z|X,\boldsymbol{\theta}^{\text{old}})}[\log p(X,Z|\boldsymbol{\theta})] =$$

$$= \sum_{i=1}^{N}\sum_{j=1}^{K} \underbrace{E_{P\left(z_j^{(i)}|\boldsymbol{x}^{(i)},\boldsymbol{\theta}^{\text{old}}\right)}\left[z_j^{(i)}\right]}_{\gamma_j^i}\{\log\mathcal{N}\left(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_j\,,\boldsymbol{\Sigma}_j\;\right)+\log\pi_j\}$$

EM for GMM
M step: details

- $$Q(\boldsymbol{\theta};\boldsymbol{\theta}^{\text{old}}) = E_{Z\sim P(Z|X,\boldsymbol{\theta}^{\text{old}})}[\log p(X,Z|\boldsymbol{\theta})]$$
  $$= \sum_{i=1}^{N}\sum_{j=1}^{K}\gamma_j^i\{\log\mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j) + \log\pi_j\}$$

$$\frac{\partial Q(\boldsymbol{\theta};\boldsymbol{\theta}^{old})}{\partial\boldsymbol{\mu}_j} = 0 \Rightarrow \boldsymbol{\mu}_j = \frac{\sum_{i=1}^{N}\gamma_j^i\boldsymbol{x}^{(i)}}{\sum_{i=1}^{N}\gamma_j^i}$$

$$\frac{\partial Q(\boldsymbol{\theta};\boldsymbol{\theta}^{old})}{\partial\boldsymbol{\Sigma}_j} = 0 \Rightarrow \boldsymbol{\Sigma}_j = \frac{1}{\sum_{i=1}^{N}\gamma_j^i}\sum_{i=1}^{N}\gamma_j^i(\boldsymbol{x}^{(i)}-\boldsymbol{\mu}_j)(\boldsymbol{x}^{(i)}-\boldsymbol{\mu}_j)^T$$

$$\frac{\partial\left(Q(\boldsymbol{\theta};\boldsymbol{\theta}^{old}) + \lambda\left(\sum_{l=1}^{k}\pi_l - 1\right)\right)}{\partial\pi_j} = 0 \Rightarrow \pi_j = \frac{\sum_{i=1}^{N}\gamma_j^i}{N}$$

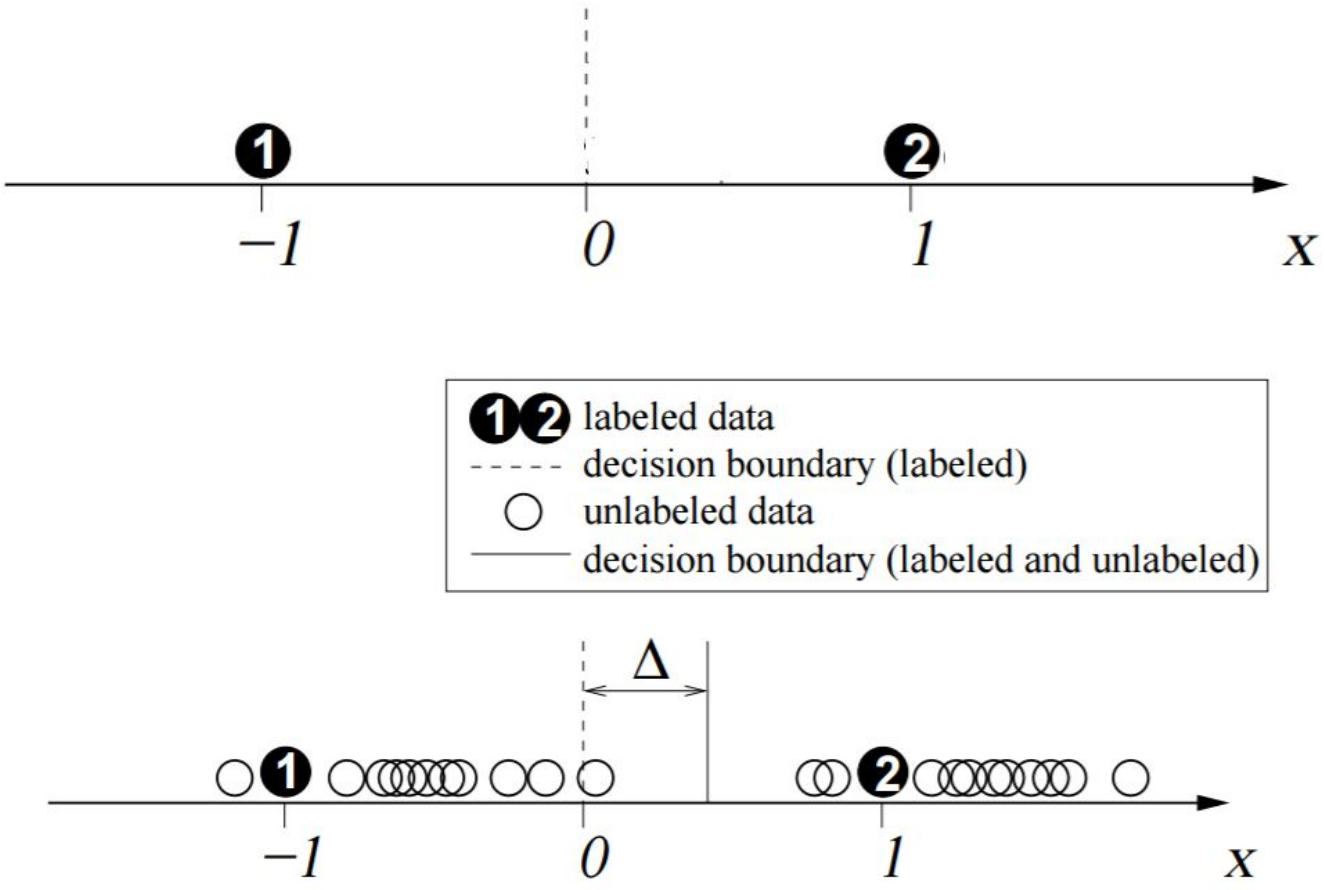
Lagrange multiplier due to the constraint $\sum_{j=1}^{k}\pi_j = 1$

Semi-supervised learning

- Supervised Learning models require labeled data
  - Supervised learning usually requires plenty of labeled data
    - It is usually expensive to have a large set of labeled data
    - Unlabeled data is often abundant with no or low cost

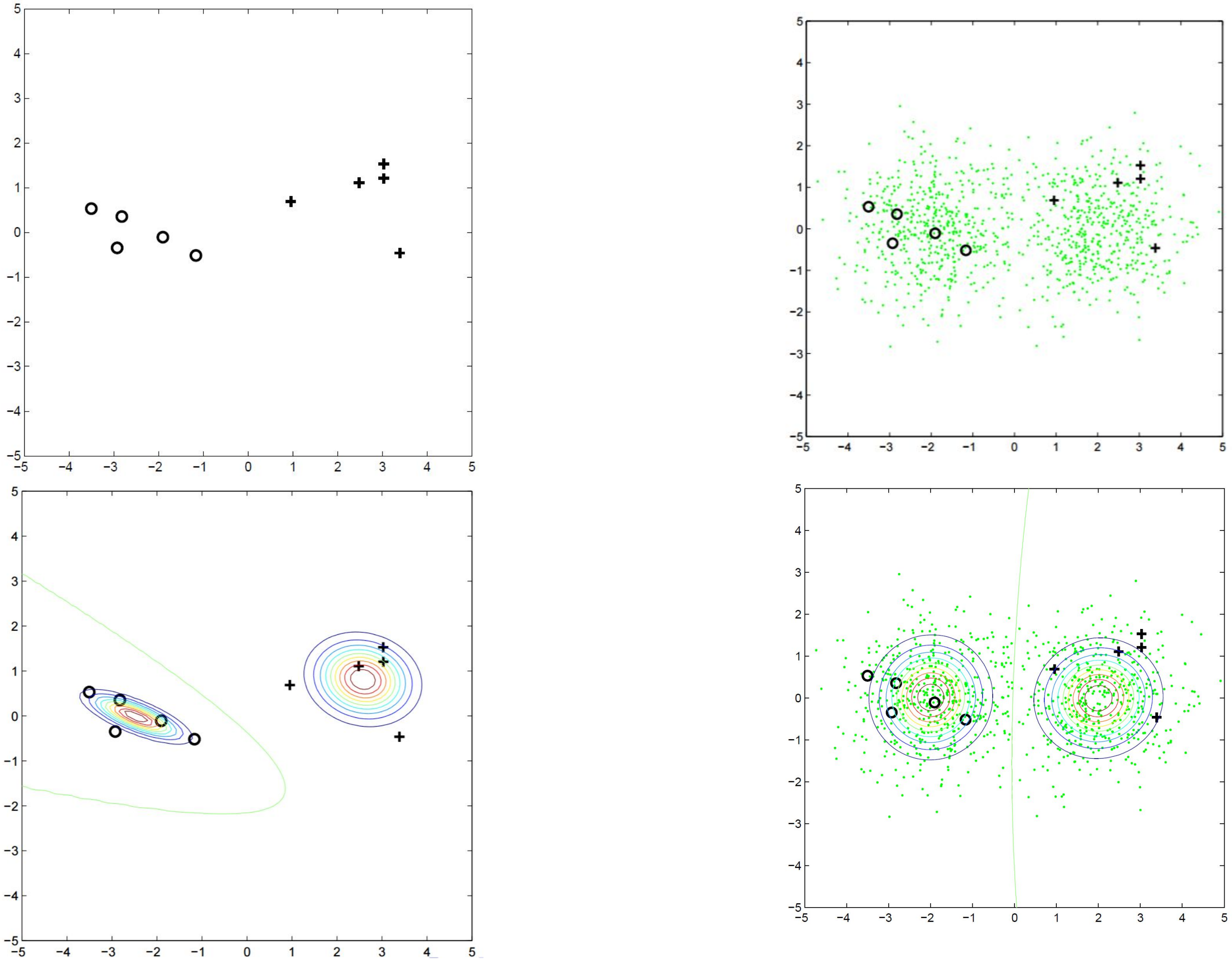
- Learning from both labeled and unlabeled data
  - Labeled training data: $\mathcal{L} = \left\{\left(\boldsymbol{x}^{(n)}, y^{(n)}\right)\right\}_{l=1}^{L}$
  - Unlabeled data available during training: $\mathcal{U} = \left\{\boldsymbol{x}^{(n)}\right\}_{n=L+1}^{L+U}$

Semi-supervised learning: example



1
2
−1
0
1
x
1 2 labeled data
decision boundary (labeled)
unlabeled data
decision boundary (labeled and unlabeled)
Δ
1
2
−1
0
1
x

Zhu, Semi-Supervised Learning Tutorial, ICML, 2007.



Zhu, Semi-Supervised Learning Tutorial, ICML, 2007.

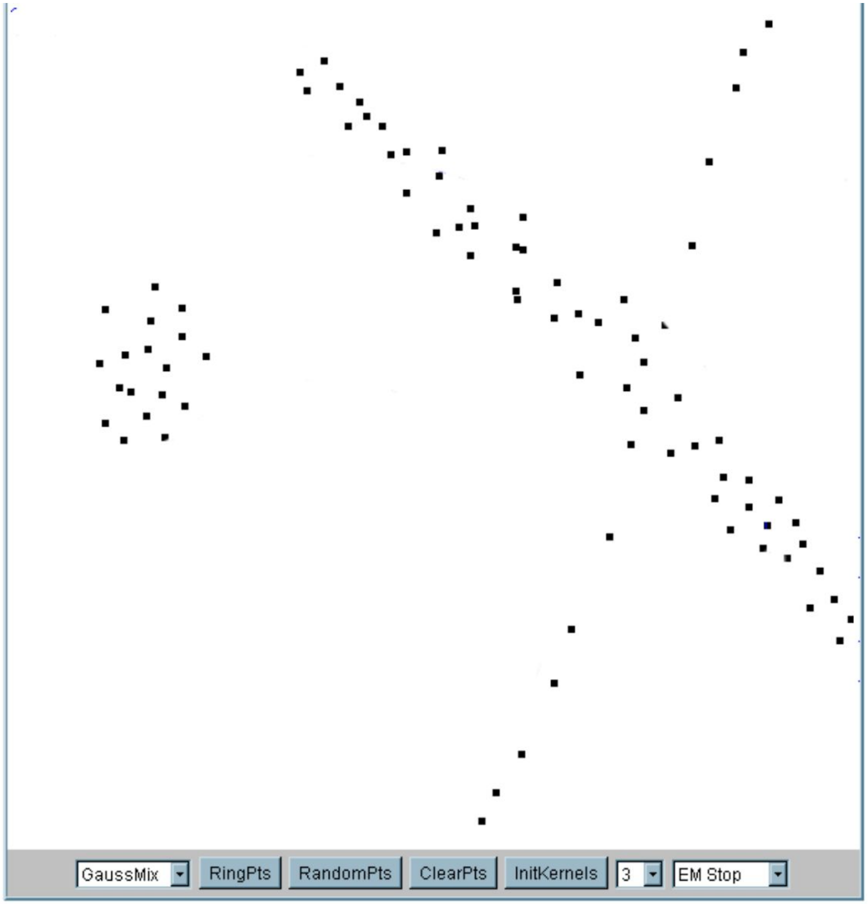
Semi-supervised generative model

- Start from MLE $\boldsymbol{\theta} = [\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}]$ on $\mathcal{L} = \left\{\left(\boldsymbol{x}^{(n)}, y^{(n)}\right)\right\}_{l=1}^{L}$
- Repeat:
  - E-step: compute $p(y^{(n)}|\boldsymbol{x}^{(n)}, \boldsymbol{\theta})$ for $n = L + 1$ to $n = L + U$
  - M-step: compute the parameters $\boldsymbol{\theta} = [\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}]$ considering both labeled data and unlabeled data using the distribution found on their labels in the E-step

Resource

- C. Bishop, “Pattern Recognition and Machine Learning”, Chapter 9.

Example


GaussMix
RingPts
RandomPts
ClearPts
InitKernels
3
EM Stop

EM for GMM
M step: details

$\boldsymbol{\theta} = [\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}]$

$\boldsymbol{\theta}^{old} = [\boldsymbol{\pi}^{old}, \boldsymbol{\mu}^{old}, \boldsymbol{\Sigma}^{\mathbf{old}}]$

- $E_{Z\sim P(Z|X,\boldsymbol{\theta}^{\text{old}})}[\log p(X,Z|\boldsymbol{\theta})]$

$p(X,Z|\boldsymbol{\theta}) = p(X|Z,\boldsymbol{\theta})P(Z|\boldsymbol{\theta})$

$p(X,Z|\boldsymbol{\theta}) = \prod_{i=1}^{N} p(\boldsymbol{x}^{(i)},\boldsymbol{z}^{(i)}|\boldsymbol{\theta})$

$$= \sum_{i=1}^{N} E_{Z\sim P(Z|X,\boldsymbol{\theta}^{\text{old}})}\left[\log\left(\prod_{i=1}^{N} p\left(\boldsymbol{x}^{(i)}\middle|\boldsymbol{z}^{(i)},\boldsymbol{\theta}\right)P\left(\boldsymbol{z}^{(i)}\middle|\boldsymbol{\pi}\right)\right)\right]$$

$$= \sum_{i=1}^{N} E_{\boldsymbol{z}^{(i)}\sim P(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{\theta}^{\text{old}})}\left[\log \prod_{i=1}^{N}\prod_{j=1}^{K} N\left(\boldsymbol{x}^{(i)}\middle|\boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j\right)^{z_j^{(i)}} \pi_j^{z_j^{(i)}}\right]$$

$$= \sum_{i=1}^{N}\sum_{j=1}^{K} \gamma_j^i \log\left(p\left(\boldsymbol{x}^{(i)}\middle|z_j^{(i)} = 1, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right)\pi_j\right)$$

EM for GMM: M step details

$$\boldsymbol{\theta} = [\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}]$$

$$\boldsymbol{\theta}^{old} = [\boldsymbol{\pi}^{old}, \boldsymbol{\mu}^{old}, \boldsymbol{\Sigma}^{\mathbf{old}}]$$

$$Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{\boldsymbol{old}}) = E_{Z\sim P(Z|X,\boldsymbol{\theta}^{\mathrm{old}})}[\log P(X, Z|\boldsymbol{\theta})]$$

EM algorithm: general

•

- Expectation Maximization (EM) seeks to estimate:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, E_{Z\sim P(Z|X,\boldsymbol{\theta})}[\log P(X,Z|\boldsymbol{\theta})]$$

- X: observed variables
- Z: Unobserved variables
- $\boldsymbol{\theta}$: parameters

EM find the maximum likelihood parameters in cases where the models involve latent variables Z in addition to unknown parameters $\boldsymbol{\theta}$ and known data observations X.

EM iteration increases likelihood $p(X|\boldsymbol{\theta})$: general

- $$p(X,Z|\boldsymbol{\theta}) = p(Z|X,\boldsymbol{\theta})p(X|\boldsymbol{\theta})$$
  $$\log p(X,Z|\boldsymbol{\theta}) = \log p(Z|X,\boldsymbol{\theta}) + \log p(X|\boldsymbol{\theta})$$
  $$\log p(X,Z|\boldsymbol{\theta}) - \log p(Z|X,\boldsymbol{\theta}) = \log p(X|\boldsymbol{\theta})$$

$$E_{p(Z|X,\boldsymbol{\theta}^{old})}[\log p(X,Z|\boldsymbol{\theta})] - E_{p(Z|X,\boldsymbol{\theta}^{old})}[\log p(Z|X,\boldsymbol{\theta})] = \log p(X|\boldsymbol{\theta})$$
$$\Rightarrow \log p(X|\boldsymbol{\theta}) = Q(\boldsymbol{\theta};\boldsymbol{\theta}^{old}) - E_{p(Z|X,\boldsymbol{\theta}^{old})}[\log p(Z|X,\boldsymbol{\theta})]$$

$$\log p(X|\boldsymbol{\theta}^{new}) - \log p\left(X|\boldsymbol{\theta}^{old}\right)$$
$$= Q\left(\boldsymbol{\theta}^{new};\boldsymbol{\theta}^{old}\right) - Q\left(\boldsymbol{\theta}^{old};\boldsymbol{\theta}^{old}\right) + E_{p(Z|X,\boldsymbol{\theta}^{old})}\left[\log p\left(Z|X,\boldsymbol{\theta}^{old}\right)\right]$$
$$- E_{p(Z|X,\boldsymbol{\theta}^{old})}[\log p(Z|X,\boldsymbol{\theta}^{new})]$$

- $\boldsymbol{\theta}^{new} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, Q(\boldsymbol{\theta};\boldsymbol{\theta}^{old}) \Rightarrow Q(\boldsymbol{\theta}^{new};\boldsymbol{\theta}^{old}) \geq Q(\boldsymbol{\theta}^{old};\boldsymbol{\theta}^{old})$
- $E_{p(Z|X,\boldsymbol{\theta}^{old})}\left[\log p\left(Z|X,\boldsymbol{\theta}^{old}\right)\right] - E_{p(Z|X,\boldsymbol{\theta}^{old})}[\log p(Z|X,\boldsymbol{\theta}^{new})]$
  $= KL(p\left(Z|X,\boldsymbol{\theta}^{old}\right)||p(Z|X,\boldsymbol{\theta}^{new})) \geq 0$

KL divergence

- Kullback-Leibler divergence between p and q:

$$D_{KL}(p||q) = \int p(x) \log \frac{p(x)}{q(x)} dx$$

- A result from information theory: For any p and q

$$D_{KL}(p||q) \geq 0$$

  - $D_{KL}(p||q) = 0$ if and only if $p \equiv q$
  - D_{KL} is asymmetric

K-means algorithm

- Choose an initial setting for cluster prototypes $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K$
- Iterate until convergence:
  - E step:

$$r_{ik} = \begin{cases} 1 & if\ k = \underset{j}{\operatorname{argmin}} \left\| \boldsymbol{x}^{(i)} - \boldsymbol{\mu}_j \right\|^2 \\ 0 & \text{otherwise} \end{cases}$$

  - M Step:

$$\boldsymbol{\mu}_k = \frac{\sum_{i=1}^{n} r_{ik} \boldsymbol{x}^{(i)}}{\sum_{i=1}^{n} r_{ik}}$$