STA 414/2104: Machine Learning

Mixture models and EM algorithms

Mark Ebden

26 Feb 2018

Based on slides by Russ Salakhutdinov

Mixture Models

• We will look at mixture models, including Gaussian mixture models

• The key idea is to introduce latent variables, which allow complicated distributions to be formed from simpler distributions

• We will see that mixture models can be interpreted in terms of having discrete latent variables (in a directed graphical model)

• Later in class, we will also look at continuous latent variables

Topics

- *K*-means clustering
- Mixture of Gaussians
- An alternative view of EM

K-Means Clustering

- Let us first look at the following problem: Identify clusters, or groups, of data points in a multidimensional space.

- We observe the dataset $\{\mathbf{x_1}, ..., \mathbf{x}_N\}$ consisting of N observations each of D dimensions

- We would like to partition the data into K clusters, where K is given

the center of clusters

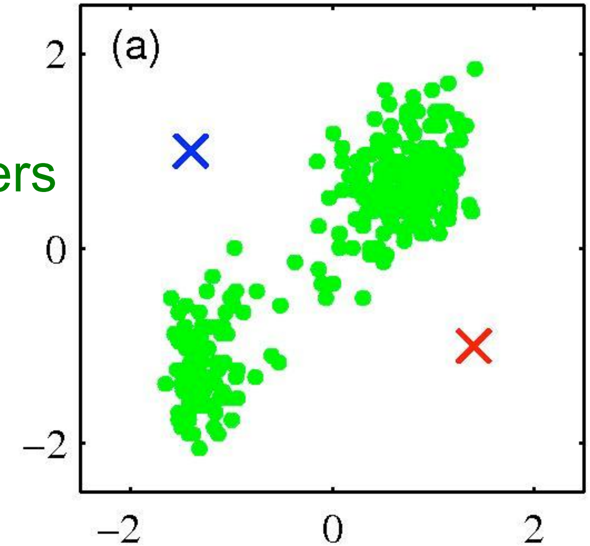
- We next introduce D-dimensional vectors, prototypes $\boldsymbol{\mu}_k, k = 1, ..., K.$

- We can think of $\boldsymbol{\mu}_k$ as representing cluster centres

- Our goal:

  - Find an assignment of data points to clusters
  - Sum of squared distances of each data point to its closest prototype is at the minimum

1. assignment of data points
2. the prototypes, { mu_k }



K-Means Clustering

r_{nk} is binary indicator =1 if we assign x_n to k-th cluster

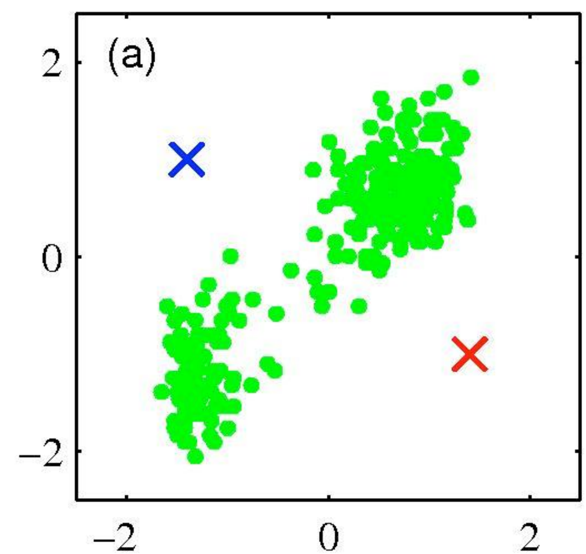
- For each data point $\mathbf{x_n}$ we introduce a binary vector $\mathbf{r_n}$ of length K (1-of-K encoding), which indicates which of the K clusters the data point $\mathbf{x_n}$ is assigned to.
- Define an objective function (distortion measure):

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} ||\mathbf{x}_n - \boldsymbol{\mu}_k||^2.$$

- It represents the sum of squares of the distances of each data point to its assigned prototype $\boldsymbol{\mu}_k$.

the assignments

- Our goal is to find the values of r_{nk} and the cluster centres $\boldsymbol{\mu}_k$ so as to minimize the objective J. the prototypes



x and mu are vectors in input space

Iterative Algorithm

- Define an iterative procedure to minimize:

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} ||\mathbf{x}_n - \boldsymbol{\mu}_k||^2.$$

assignment of n-th point is independent of the rest

- Given $\boldsymbol{\mu}_k$, minimize J with respect to r_{nk} (**E-step**):

Hard assignments of points to clusters.

since J is linear to r, have closed form solution

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

which simply says assign n^{th} data point $\mathbf{x}_n$ to its closest cluster centre

- Given r_{nk}, minimize J with respect to $\boldsymbol{\mu}_k$ (**M-step**):

since J is quadratic to mu_k, compute derivative set = 0 and rearrange.

$$\boldsymbol{\mu}_k = \frac{\sum_n r_{nk}\mathbf{x}_n}{\sum_n r_{nk}}.$$

Number of points assigned to cluster *k*.

Set $\boldsymbol{\mu}_k$ equal to the mean of all the data points assigned to cluster *k*

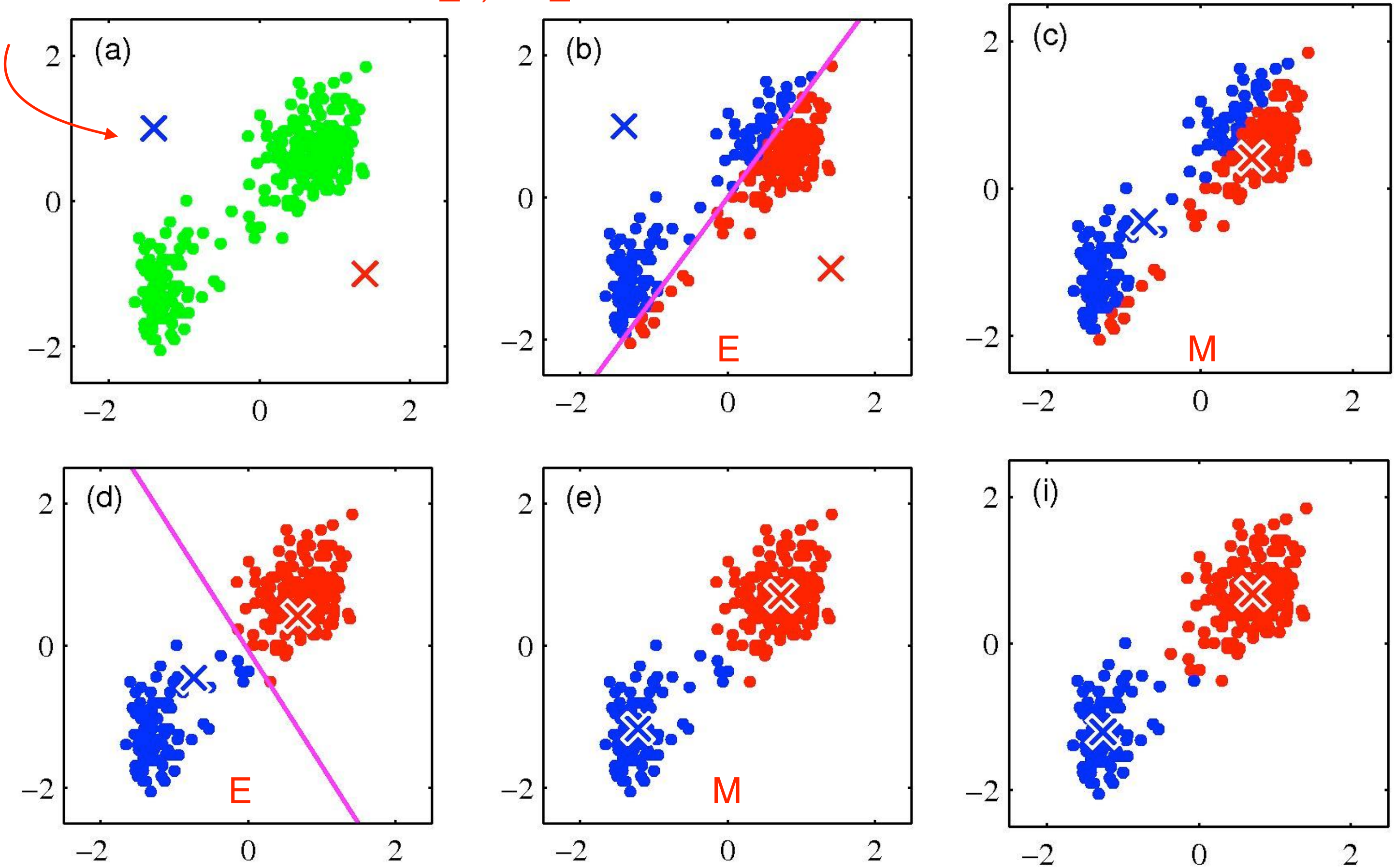
- Guaranteed convergence to a local minimum (not global minimum).

E: re-assigning data points to clusters, given cluster centers
M: re-computing cluster means, given assignments

Example

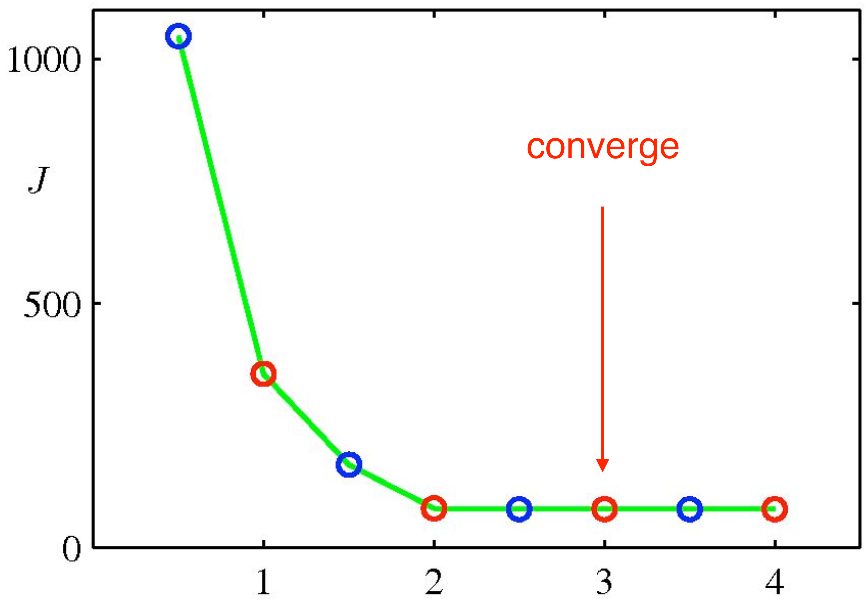
- Example of using *K*-means clustering (*K*=2) on the Old Faithful dataset.

initial value of cluster center mu_1, mu_2



Convergence

• Plot of the cost function after each E-step (blue points) and M-step (red points)



The algorithm has converged after three iterations.

• *K*-means clustering can be generalized by introducing a more general dissimilarity measure:

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} K(\mathbf{x}_n, \boldsymbol{\mu}_k).$$

K defines how different are two points are

Image Segmentation

- Another application of the *K*-means algorithm.
- Partition an image into regions corresponding, for example, to object parts.
- Each pixel in an image is a point in 3-D space, corresponding to R,G,B channels.

- For a given value of *K*, the algorithm represents an image using *K* colours

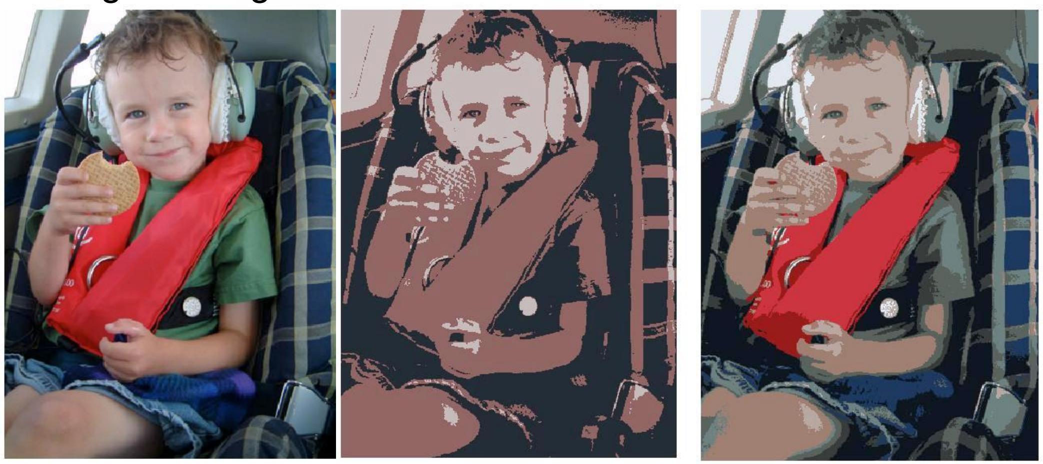
- Another application is image compression.

Image Compression

instead of the entire vector

- For each data point, we store only the identity k of the assigned cluster.
- We also store the values of the cluster centers $\boldsymbol{\mu}_k$.
- Provided $K << N$, we require significantly less data.

Original image K=3 K=10



- The original image has 240 x 180 = 43,200 pixels.
- Each pixel contains {R,G,B} values, each of which requires 8 bits.

24 = 3 channels x 8 bits / channel

- Requires 43,200 x 24 = 1,036,800 bits to transmit directly. set of cluster centers

1000K bits

- With K-means clustering, we need to transmit K code-book vectors $\boldsymbol{\mu}_k$ -- $24K$ bits.

memory required for identity

- For each pixel we need to transmit $\log_2 K$ bits (as there are K vectors).
- Compressed image requires 43,248 (K=2), 86,472 (K=3), and 173,040 (K=10) bits, which amounts to compression ratios of 4.2%, 8.3%, and 16.7%.

total number of bits with compression : 24K + NlogK, without compression: 24N bits

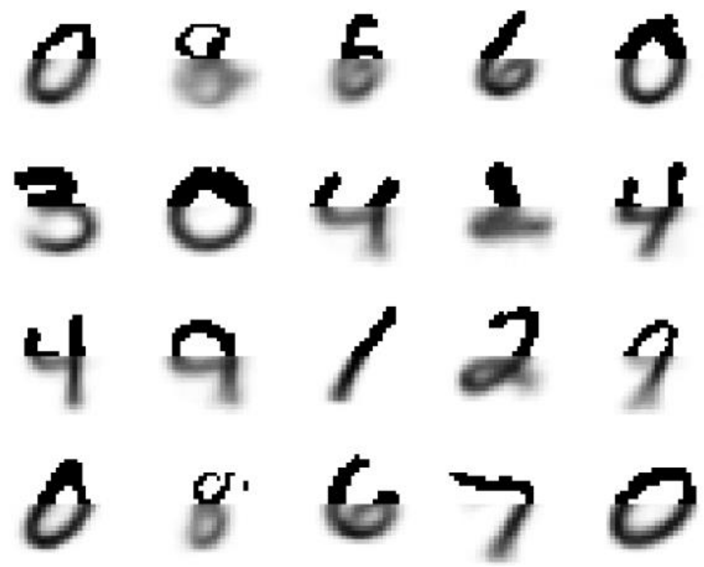
Mixture of *Products of Bernoullis*

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

where $\boldsymbol{\mu} = \{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K\}$, $\boldsymbol{\pi} = \{\pi_1, \ldots, \pi_K\}$, and

$$p(\mathbf{x}|\boldsymbol{\mu}_k) = \prod_{i=1}^{D} \mu_{ki}^{x_i}(1-\mu_{ki})^{(1-x_i)}$$

$p(\mathbf{x}_{i\in\text{bottom}}|\mathbf{x}_{i\in\text{top}}, \boldsymbol{\theta}, \boldsymbol{\pi})$



Topics

- *K*-means clustering
- **Mixture of Gaussians**
- An alternative view of EM

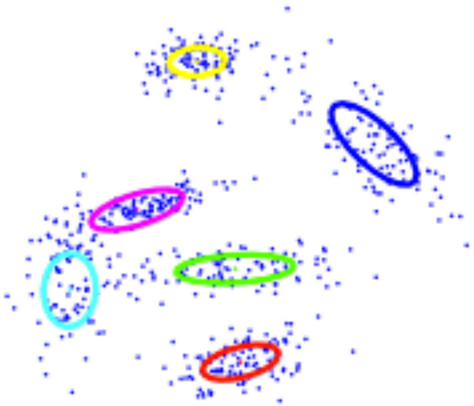
Mixture of Gaussians

- We'll look at a mixture of Gaussians in terms of discrete latent variables

basically convert mixing coefficients to a latent categorical random variable

- The Gaussian mixture can be written as a linear superposition of Gaussians:

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



z is latent, categorical distribution

- Introduce a *K*-dimensional binary random variable **z** having a 1-of-*K* representation:

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

- We will specify the distribution over **z** in terms of mixing coefficients:

$$p(z_k = 1) = \pi_k, \quad 0 \leq \pi_k \leq 1, \quad \sum_k \pi_k = 1.$$

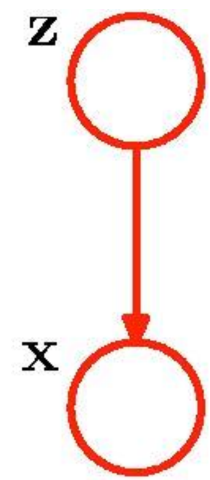
z is a latent variable

Mixture of Gaussians

- Because **z** uses 1-of-*K* encoding, we have:

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

pdf



- We can now specify the conditional distribution:

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

- We have therefore specified the joint distribution:

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

- The marginal distribution over **x** is given by:

over z

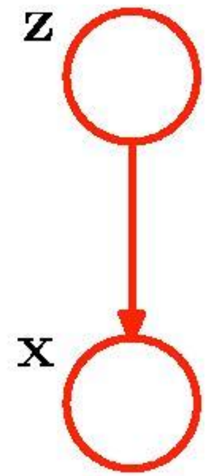
$$p(\mathbf{x}) = \sum_{\mathbf{z}} \underbrace{p(\mathbf{z})p(\mathbf{x}|\mathbf{z})}_{\text{joint distribution } p(x,z)} = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k).$$

- The marginal distribution over **x** is given by a Gaussian mixture.

Mixture of Gaussians

- The marginal distribution is:

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



since p(x) = sum_z p(x,z), each x has a corresponding z

- If we have several observations $\mathbf{x}_1,\ldots,\mathbf{x}_N$, it follows that for every observed data point $\mathbf{x}_n$ there is a corresponding latent variable $\mathbf{z}_n$.

- Let us look at the conditional $p(\mathbf{z}|\mathbf{x})$, responsibilities, which we will need for doing inference:

p(zlx) — p(z) p(xlz)

responsibility that latent variable contribute to explaning x

$$\gamma(z_k) = p(z_k = 1|\mathbf{x}) = \frac{p(z_k = 1)p(\mathbf{x}|z_k = 1)}{\sum_{j=1}^{K} p(z_j = 1)p(\mathbf{x}|z_j = 1)} =$$

conditional probability of z given x

responsibility that component k takes for explaining the data $\mathbf{x}$

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

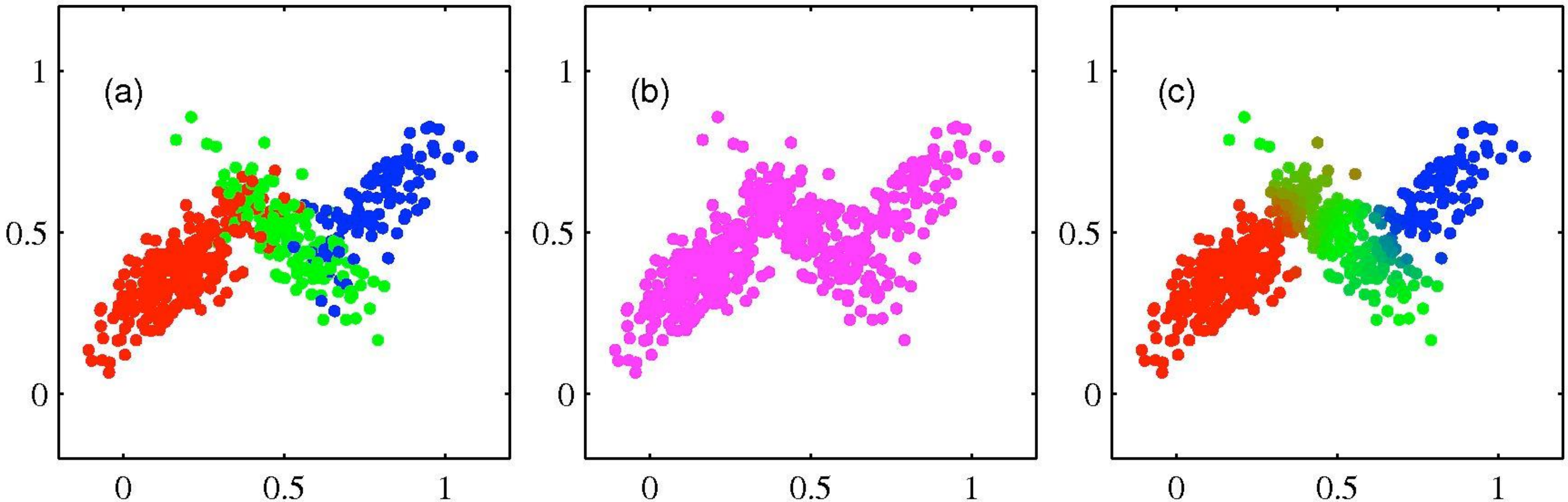
- We will view $\boldsymbol{\mu}_k$ as prior probability that z_k=1, and $\gamma(z_k)$ is the corresponding posterior once we have observed the data.

Example

- 500 points drawn from a mixture of three Gaussians.

3 state of mixture

ignore values of z



Samples from the joint distribution $p(\mathbf{x},\mathbf{z})$.

generated

Samples from the marginal distribution $p(\mathbf{x})$.

real world

Same samples, where colours represent the value of responsibilities.

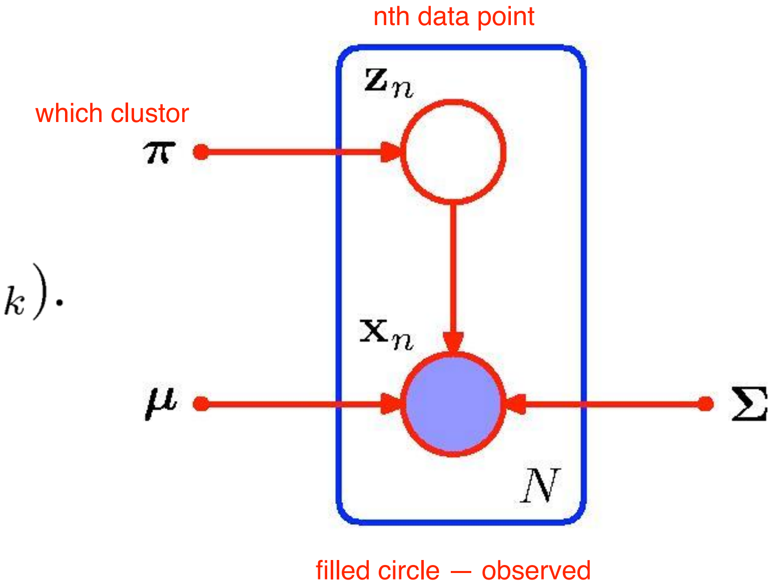
soft partitioning

p(z_k=1 l x) where k = 1,2,3
the responsibilities

Maximum Likelihood

- Suppose we observe a dataset $\{\mathbf{x}_1,\ldots,\mathbf{x}_N\}$, and we model the data using a mixture of Gaussians.
- We represent the dataset as an N x D matrix $\mathbf{X}$.
- The corresponding latent variables will be represented and an N x K matrix $\mathbf{Z}$.
- The log-likelihood takes the form:

$$\ln p(\mathbf{X}|\underbrace{\boldsymbol{\pi},\boldsymbol{\mu},\boldsymbol{\Sigma}}_{\text{Model parameters}}) = \sum_{n=1}^{N} \ln \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k).$$



"Graphical model" for a Gaussian mixture model for a set of i.i.d. data point $\{\mathbf{x}_n\}$, and corresponding latent variables $\{\mathbf{z}_n\}$.

no closed solution, use EM for gaussian mixtures

Maximum Likelihood

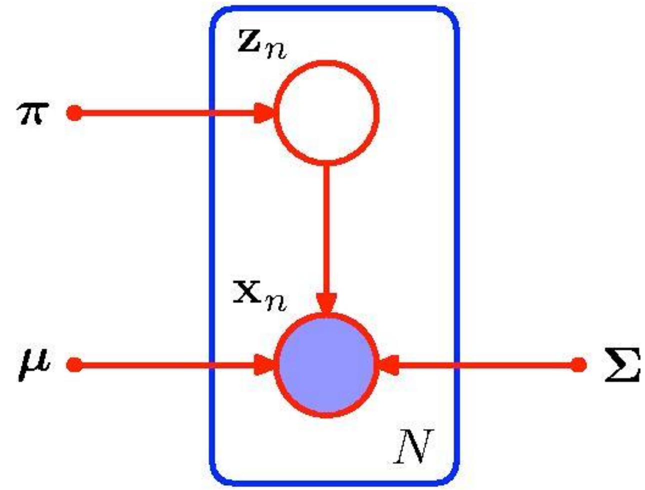
E-step: maximize w.r.t. mu_k

- The log-likelihood:

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

- Differentiating with respect to $\boldsymbol{\mu}_k$ and setting to zero:

$$0 = \sum_{n} \underbrace{\frac{\pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_j \pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j}}_{\gamma(z_{nk})} \boldsymbol{\Sigma}_K^{-1}(\mathbf{x}_n - \boldsymbol{\mu}_k).$$



Soft assignment

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_n \gamma(z_{nk}) \mathbf{x}_n, \quad N_k = \sum_n \gamma(z_{nk}).$$

- We can interpret N_k as the effective number of points assigned to cluster k.
- The mean $\boldsymbol{\mu}_k$ is given by the mean of all the data points weighted by the posterior $\gamma(z_{nk})$ that component k was responsible for generating $\mathbf{x}_n$.

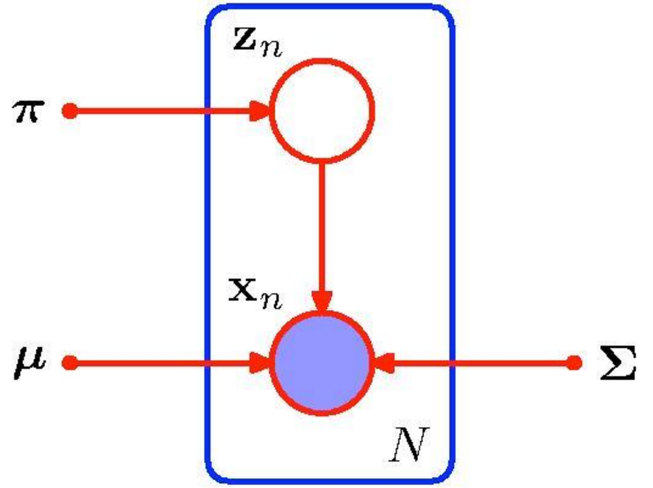
Maximum Likelihood

- The log-likelihood:

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

- Differentiating with respect to Σ_k and setting to zero:

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



- Note that the data points are weighted by the posterior probabilities.

- Maximizing the log-likelihood with respect to the mixing proportions:

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

- The mixing proportion for the k^{th} component is given by the average responsibility which that component takes for explaining the data.

Maximum Likelihood

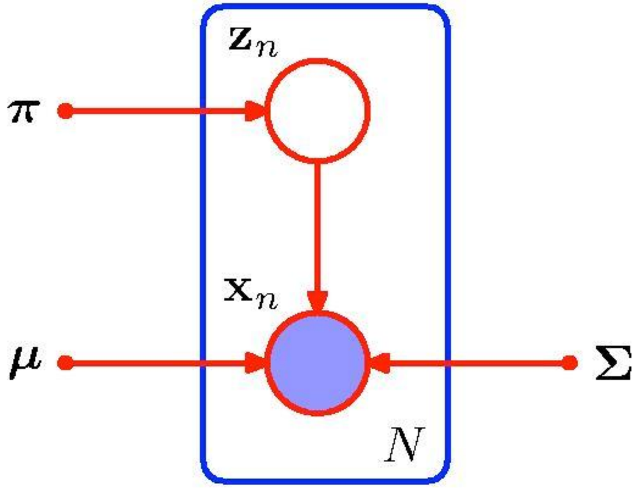
- The log-likelihood:

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

- The maximum likelihood does not have a closed-form solution.

- Parameter updates depend on responsibilities $\gamma(z_{nk})$, which themselves depend on those parameters:

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



- Iterative Solution:

  E-step: Update responsibilities $\gamma(z_{nk})$.
  M-step: Update model parameters $\boldsymbol{\mu}_k$, π_k, $\boldsymbol{\Sigma}_k$, for k=1,…,K.

An EM algorithm

- Initialize the means $\boldsymbol{\mu}_k$, covariances $\boldsymbol{\Sigma}_k$, and mixing proportions π_k.
- E-step: Evaluate responsibilities using current parameter values:

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

the posterior probabilities

- M-step: Re-estimate model parameters using the current responsibilities:

$$\boldsymbol{\mu}_k^{new} = \frac{1}{N_k} \sum_n \gamma(z_{nk}) \mathbf{x}_n, \quad N_k = \sum_n \gamma(z_{nk}),$$

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

$$\pi_k^{new} = \frac{N_k}{N}.$$

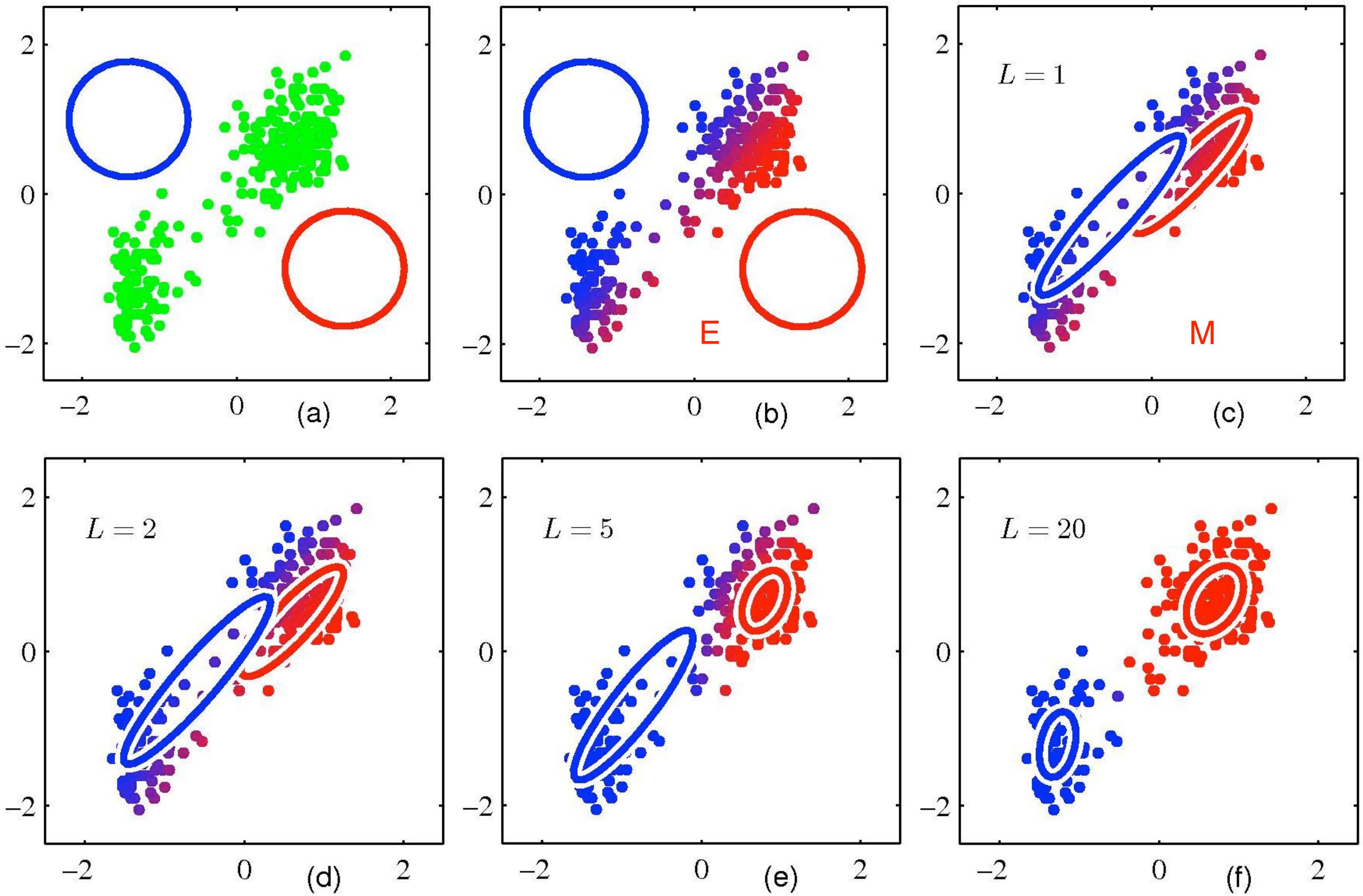
- Evaluate the log-likelihood and check for convergence.

check parameters for convergence

Mixture of Gaussians: Example

• Illustration of an EM algorithm (much slower convergence compared to *K*-means clustering)

result of E



Topics

- *K*-means clustering
- Mixture of Gaussians
- **An alternative view of EM**

An Alternative View of EM

• The goal of EM is to find maximum-likelihood solutions for models with latent variables.

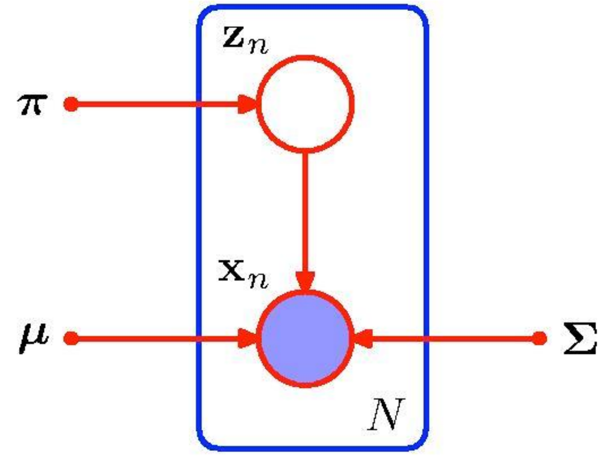
• We represent the observed dataset as an N x D matrix $\mathbf{X}$.

• Latent variables will be represented as an N x K matrix $\mathbf{Z}$.

• The set of all model parameters is denoted here by θ ($\boldsymbol{\theta}$ would be better).

• The log-likelihood takes the form:

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



• Note: even if the joint distribution belongs to the exponential family, the marginal typically does not!

• We will call:

$\{\mathbf{X}, \mathbf{Z}\}$ a complete dataset.

$\{\mathbf{X}\}$ an incomplete dataset.

An Alternative View of EM

- In practice, we are not given a complete dataset {**X**,**Z**}, but only an incomplete dataset {**X**}.
- Our knowledge about the latent variables is given only by the posterior distribution $p(\mathbf{Z}|\mathbf{X},\theta)$.
- Because we cannot use the complete data log-likelihood, we can consider the expected complete-data log-likelihood:

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

- In the E-step, we use the current parameters θ^{old} to compute the posterior over the latent variables $p(\mathbf{Z}|\mathbf{X},\theta^{old})$.
- We use this posterior to compute expected complete log-likelihood.
- In the M-step, we find the revised parameter estimate θ^{new} by maximizing the expected complete log-likelihood:

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

Tractable

The General EM algorithm

- Given a joint distribution $p(\mathbf{Z},\mathbf{X}|\theta)$ over observed and latent variables governed by parameters θ, the goal is to maximize the likelihood function $p(\mathbf{X}|\theta)$ with respect to θ.
- Initialize parameters θ^{old}.
- E-step: Compute posterior over latent variables: $p(\mathbf{Z}|\mathbf{X},\theta^{old})$.
- M-step: Find the new estimate of parameters θ^{new}:

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

where

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

- Check for convergence of either log-likelihood or the parameter values. Otherwise:

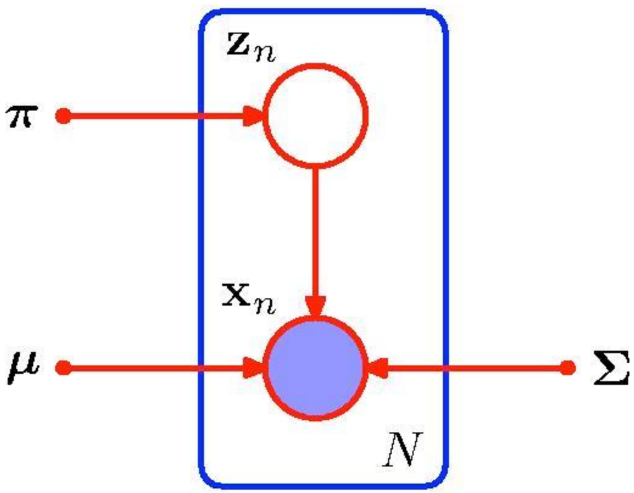
$$\theta^{new} \leftarrow \theta^{old},$$ and iterate.

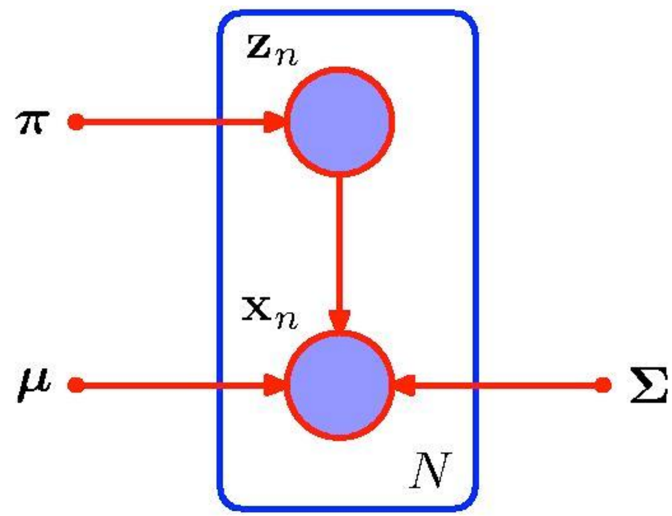
Gaussian Mixtures Revisited

• We now consider the application of the latent variable view of EM to the case of a Gaussian mixture model.

• Recall:

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





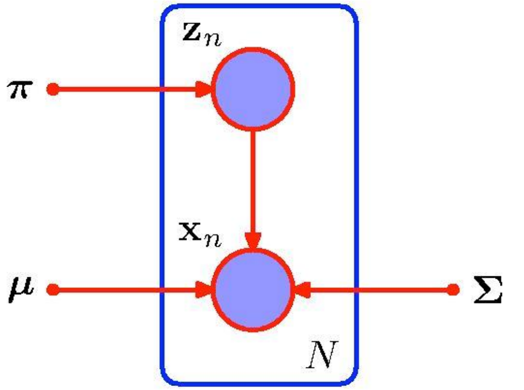
$\{\mathbf{X}\}$ -- incomplete dataset. $\{\mathbf{X}, \mathbf{Z}\}$ -- complete dataset.

Maximizing Complete Data

• Consider the problem of maximizing the likelihood for the complete data:

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

$$\ln p(\mathbf{X}, \mathbf{Z} | \boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \underbrace{\sum_{k=1}^{K}}_{\text{Sum of } K \text{ independent contributions, one for each mixture component.}} \left[\sum_{n=1}^{N} z_{nk} \ln \pi_k + z_{nk} \ln \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right].$$



$\{\mathbf{X}, \mathbf{Z}\}$

-- complete dataset.

• Maximizing with respect to mixing proportions yields:

$$\pi_k = \frac{1}{N} \sum_{n=1}^{N} z_{nk}.$$

• And similarly for the means and covariances.

Posterior Over Latent Variables

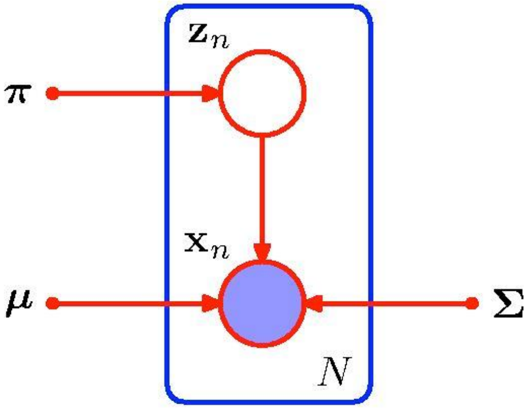
- Remember:

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

- The posterior over latent variables takes the form:

$$p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \propto \prod_{n=1}^{N} \prod_{k=1}^{K} \left[\pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right]^{z_k}.$$

- Note that the posterior factorizes over *n* points, so that under the posterior distribution, $\{\mathbf{z}_n\}$ are independent.



Expected Complete Log-Likelihood

- The expected value of indicator variable z_{nk} under the posterior distribution is:

$$\mathbb{E}[z_{nk}] = \frac{\sum_{\mathbf{z}_n} z_{nk} \prod_j \left[\pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)\right]^{z_{nj}}}{\sum_{\mathbf{z}_n} \prod_j \left[\pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)\right]^{z_{nj}}}$$

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

- This represents the responsibility of component k for data point $\mathbf{x}_n$.
- The complete-data log-likelihood:

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

- The expected complete data log-likelihood is:

$$\mathbb{E}_{\mathbf{Z}} \left[\ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \right] = \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma(z_{nk}) \Big[\ln \pi_k + \ln \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \Big].$$

Expected Complete Log-Likelihood

- The expected complete data log-likelihood is:

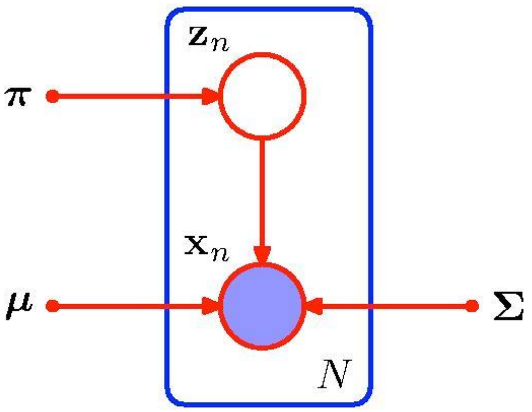
$$\mathbb{E}_{\mathbf{Z}}\left[\ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma})\right] = \sum_{n=1}^{N}\sum_{k=1}^{K}\gamma(z_{nk})\Big[\ln \pi_k + \ln \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\Big].$$

- Maximizing with respect to the model parameters, we obtain:

$$\boldsymbol{\mu}_k^{new} = \frac{1}{N_k}\sum_n \gamma(z_{nk})\mathbf{x}_n, \quad N_k = \sum_n \gamma(z_{nk}),$$

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

$$\pi_k^{new} = \frac{N_k}{N}.$$



Relationship to *K*-Means clustering

• Consider a Gaussian mixture model in which covariances are shared and are given by $\varepsilon\mathbf{I}$.

$$p(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{(2\pi\epsilon)^{D/2}} \exp\left[-\frac{1}{2\epsilon}||\mathbf{x} - \boldsymbol{\mu}_k||^2 \right].$$

• Consider the EM algorithm for a mixture of *K* Gaussians, in which we treat ε as a fixed constant. The posterior responsibilities take the form:

$$\gamma(z_{nk}) = \frac{\pi_k \exp(-||\mathbf{x}_n - \boldsymbol{\mu}_k||^2/2\epsilon)}{\sum_{j=1}^K \pi_j \exp(-||\mathbf{x}_n - \boldsymbol{\mu}_j||^2/2\epsilon)}.$$

• Consider the limit $\varepsilon \rightarrow 0$.

• In the denominator, the term for which $||\mathbf{x}_n - \boldsymbol{\mu}_j||^2$ is smallest will go to zero most slowly. Hence $\gamma(z_{nk}) \rightarrow r_{nk}$, where

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

Relationship to *K*-Means clustering

- In the limit $\varepsilon \rightarrow 0$, the expected complete log-likelihood becomes:

$$\mathbb{E}_{\mathbf{Z}}\left[\ln p(\mathbf{X}, \mathbf{Z} | \boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma})\right] \rightarrow -\frac{1}{2} \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} ||\mathbf{x}_n - \boldsymbol{\mu}_k||^2 + \text{const.}$$

- Hence in the limit, maximizing the expected complete log-likelihood is equivalent to minimizing the distortion measure *J* for the *K*-means algorithm.