# Advanced Machine Learning
# Lecture 9

## Mixture Models

### 14.11.2012

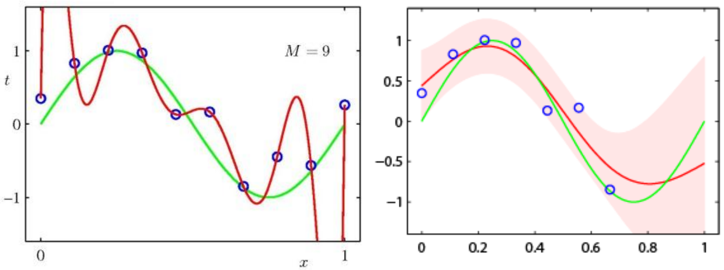
**Bastian Leibe**
**RWTH Aachen**
http://www.vision.rwth-aachen.de/

**leibe@vision.rwth-aachen.de**

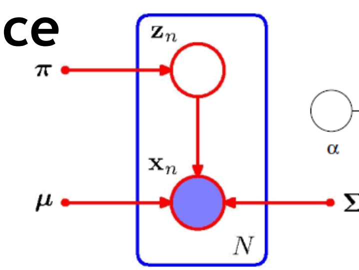
# This Lecture: *Advanced Machine Learning*

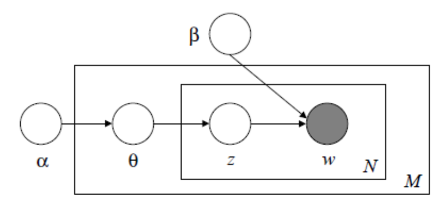
- Regression Approaches
  - Linear Regression
  - Regularization (Ridge, Lasso)
  - Kernels (Kernel Ridge Regression)
  - Gaussian Processes

$$f : \mathcal{X} \rightarrow \mathbb{R}$$




- Bayesian Estimation & Bayesian Non-Parametrics
  - Prob. Distributions, Approx. Inference
  - Mixture Models & EM
  - Dirichlet Processes
  - Latent Factor Models
  - Beta Processes




- SVMs and Structured Output Learning
  - SV Regression, SVDD
  - Large-margin Learning

$$f : \mathcal{X} \rightarrow \mathcal{Y}$$

# Recap: Importance Sampling

- Approach
  - Approximate expectations directly (but does not enable to draw samples from $p(\mathbf{z})$ directly).
  - Goal: $$\mathbb{E}[f] = \int f(\mathbf{z})p(\mathbf{z})d\mathbf{z}$$
- Idea
  - Use a proposal distribution $q(\mathbf{z})$ from which it is easy to sample.
  - Express expectations in the form of a finite sum over samples $\{\mathbf{z}^{(l)}\}$ drawn from $q(\mathbf{z})$.

$$\mathbb{E}[f] = \int f(\mathbf{z})p(\mathbf{z})d\mathbf{z} = \int f(\mathbf{z})\frac{p(\mathbf{z})}{q(\mathbf{z})}q(\mathbf{z})d\mathbf{z}$$

$$\simeq \frac{1}{L}\sum_{l=1}^{L}\underbrace{\frac{p(\mathbf{z}^{(l)})}{q(\mathbf{z}^{(l)})}}_{\text{Importance weights}}f(\mathbf{z}^{(l)})$$

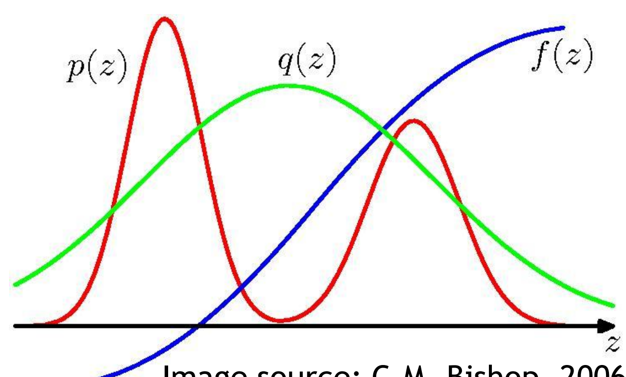



Image source: C.M. Bishop, 2006

# Recap: MCMC – Markov Chain Monte Carlo

- Overview
  - Allows to sample from a large class of distributions.
  - Scales well with the dimensionality of the sample space.
- Idea
  - We maintain a record of the current state $\mathbf{z}^{(\tau)}$
  - The proposal distribution depends on the current state: $q(\mathbf{z}|\mathbf{z}^{(\tau)})$
  - The sequence of samples forms a Markov chain $\mathbf{z}^{(1)}, \mathbf{z}^{(2)},\ldots$
- Approach
  - At each time step, we generate a candidate sample from the proposal distribution and accept the sample according to a criterion.
  - Different variants of MCMC for different criteria.

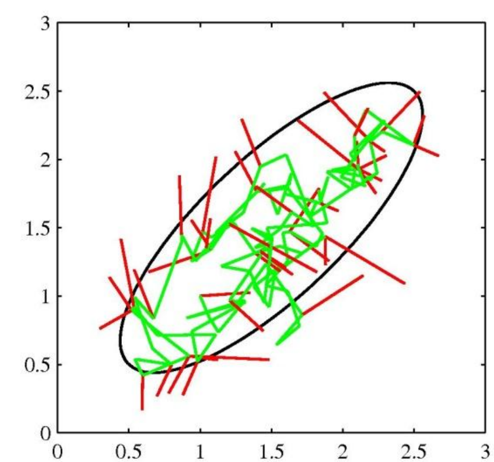



Image source: C.M. Bishop, 2006

# Recap: Markov Chains – Properties

- Invariant distribution
  - A distribution is said to be invariant (or stationary) w.r.t. a Markov chain if each step in the chain leaves that distribution invariant.
  - Transition probabilities:
$$T\left(\mathbf{z}^{(m)}, \mathbf{z}^{(m+1)}\right) = p\left(\mathbf{z}^{(m+1)} | \mathbf{z}^{(m)}\right)$$
  - For homogeneous Markov chain, distribution $p^*(\mathbf{z})$ is invariant if:
$$p^\star(\mathbf{z}) = \sum_{\mathbf{z}'} T(\mathbf{z}', \mathbf{z})\, p^\star(\mathbf{z}')$$
- Detailed balance
  - Sufficient (but not necessary) condition to ensure that a distribution is invariant:
$$p^\star(\mathbf{z}) T(\mathbf{z}, \mathbf{z}') = p^\star(\mathbf{z}') T(\mathbf{z}', \mathbf{z})$$
  - A Markov chain which respects *detailed balance* is reversible.

# Detailed Balance

- **Detailed balance** means
  - If we pick a state from the target distribution $p(\mathbf{z})$ and make a transition under $T$ to another state, it is just as likely that we will pick $\mathbf{z}_A$ and go from $\mathbf{z}_A$ to $\mathbf{z}_B$ than that we will pick $\mathbf{z}_B$ and go from $\mathbf{z}_B$ to $\mathbf{z}_A$.
  - It can easily be seen that a transition probability that satisfies detailed balance w.r.t. a particular distribution will leave that distribution invariant, because

$$\sum_{\mathbf{z}'} p^{\star}(\mathbf{z}')T(\mathbf{z}', \mathbf{z}) = \sum_{\mathbf{z}'} p^{\star}(\mathbf{z})T(\mathbf{z}, \mathbf{z}')$$

$$= p^{\star}(\mathbf{z}) \sum_{\mathbf{z}'} p(\mathbf{z}'|\mathbf{z}) = p^{\star}(\mathbf{z})$$

# Recap: MCMC – Metropolis Algorithm

- **Metropolis algorithm** [Metropolis et al., 1953]
  - Proposal distribution is symmetric: $q(\mathbf{z}_A|\mathbf{z}_B) = q(\mathbf{z}_B|\mathbf{z}_A)$
  - The new candidate sample $\mathbf{z}^*$ is accepted with probability

$$A(\mathbf{z}^\star, \mathbf{z}^{(\tau)}) = \min\left(1, \frac{\tilde{p}(\mathbf{z}^\star)}{\tilde{p}(\mathbf{z}^{(\tau)})}\right)$$

  $\Rightarrow$ New candidate samples always accepted if $\tilde{p}(\mathbf{z}^\star) \geq \tilde{p}(\mathbf{z}^{(\tau)})$.
  - The algorithm sometimes accepts a state with lower probability.

- **Metropolis-Hastings algorithm**
  - Generalization: Proposal distribution not necessarily symmetric.
  - The new candidate sample $\mathbf{z}^*$ is accepted with probability

$$A(\mathbf{z}^\star, \mathbf{z}^{(\tau)}) = \min\left(1, \frac{\tilde{p}(\mathbf{z}^\star)q_k(\mathbf{z}^{(\tau)}|\mathbf{z}^\star)}{\tilde{p}(\mathbf{z}^{(\tau)})q_k(\mathbf{z}^\star|\mathbf{z}^{(\tau)})}\right)$$

  - where $k$ labels the members of the set of considered transitions.

# Recap: MCMC - Metropolis-Hastings Algorithm

- **Properties**
  - We can show that $p(\mathbf{z})$ is an invariant distribution of the Markov chain defined by the Metropolis-Hastings algorithm.
  - We show detailed balance:

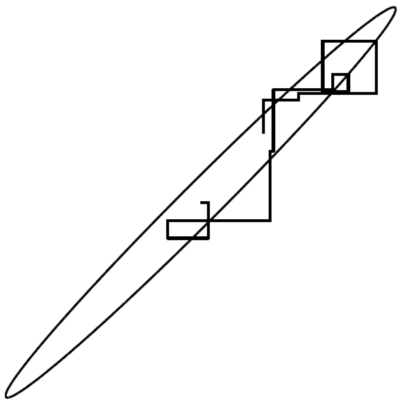
$$A(\mathbf{z}', \mathbf{z}) = \min\left\{1, \frac{\tilde{p}(\mathbf{z}')q_k(\mathbf{z}|\mathbf{z}')}{\tilde{p}(\mathbf{z})q_k(\mathbf{z}'|\mathbf{z})}\right\}$$

$$\begin{aligned}
\tilde{p}(\mathbf{z})q_k(\mathbf{z}'|\mathbf{z})A_k(\mathbf{z}', \mathbf{z}) &= \min\left\{\tilde{p}(\mathbf{z})q_k(\mathbf{z}'|\mathbf{z}), \tilde{p}(\mathbf{z}')q_k(\mathbf{z}|\mathbf{z}')\right\} \\
&= \min\left\{\tilde{p}(\mathbf{z}')q_k(\mathbf{z}|\mathbf{z}'), \tilde{p}(\mathbf{z})q_k(\mathbf{z}'|\mathbf{z})\right\} \\
\tilde{p}(\mathbf{z})q_k(\mathbf{z}'|\mathbf{z})A_k(\mathbf{z}', \mathbf{z}) &= \tilde{p}(\mathbf{z}')q_k(\mathbf{z}|\mathbf{z}')A_k(\mathbf{z}, \mathbf{z}') \\
\tilde{p}(\mathbf{z})T(\mathbf{z}, \mathbf{z}') &= \tilde{p}(\mathbf{z}')T(\mathbf{z}', \mathbf{z})
\end{aligned}$$

Update: This was wrong on the first version of the slides (also wrong in the Bishop book)!

# Recap: Gibbs Sampling

- Approach
  - MCMC-algorithm that is simple and widely applicable.
  - May be seen as a special case of Metropolis-Hastings.
- Idea
  - Sample variable-wise: replace $\mathbf{z}_i$ by a value drawn from the distribution $p(z_i | \mathbf{z}_{\setminus i})$.
    - This means we update one coordinate at a time.
  - Repeat procedure either by cycling through all variables or by choosing the next variable.
- Properties
  - The algorithm always accepts!
  - Completely parameter free.
  - Can also be applied to subsets of variables.

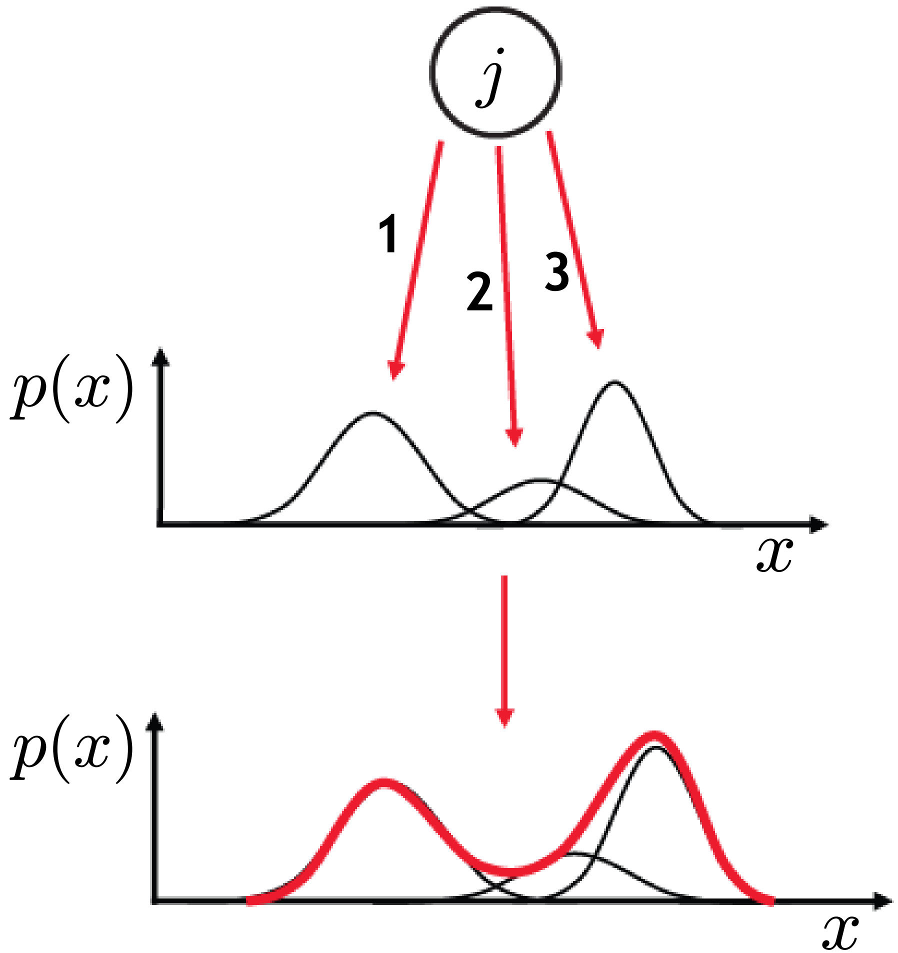


# Topics of This Lecture

- Recap: Mixtures of Gaussians
  - Mixtures of Gaussians
  - ML estimation
  - EM algorithm for MoGs
- An alternative view of EM
  - Latent variables
  - General EM
  - Mixtures of Gaussians revisited
  - Mixtures of Bernoulli distributions
- The EM algorithm in general
  - Generalized EM
  - Monte Carlo EM

# Recap: Mixture of Gaussians (MoG)

- "Generative model"




$p(j) = \pi_j$ "Weight" of mixture component

$p(x|\theta_j)$ Mixture component

Mixture density

$$p(x|\theta) = \sum_{j=1}^{M} p(x|\theta_j)p(j)$$

# Recap: Mixture of Multivariate Gaussians

- **Multivariate Gaussians**

$$p(\mathbf{x}|\theta) = \sum_{j=1}^{M} p(\mathbf{x}|\theta_j)p(j)$$

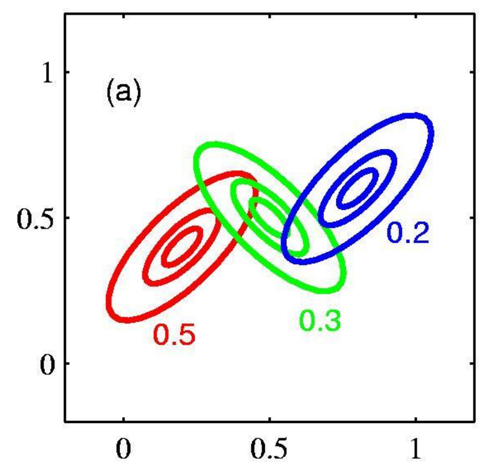
$$p(\mathbf{x}|\theta_j) = \frac{1}{(2\pi)^{D/2}|\mathbf{\Sigma}_j|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)^{\mathrm{T}}\mathbf{\Sigma}_j^{-1}(\mathbf{x}-\boldsymbol{\mu}_j)\right\}$$

  - **Mixture weights / mixture coefficients:**

$p(j) = \pi_j$ **with** $0 \cdot \ \pi_j \cdot \ 1$ **and** $\sum_{j=1}^{M} \pi_j = 1$

  - **Parameters:**

$$\theta = (\pi_1, \boldsymbol{\mu}_1, \mathbf{\Sigma}_1, \ldots, \pi_M, \boldsymbol{\mu}_M, \mathbf{\Sigma}_M)$$



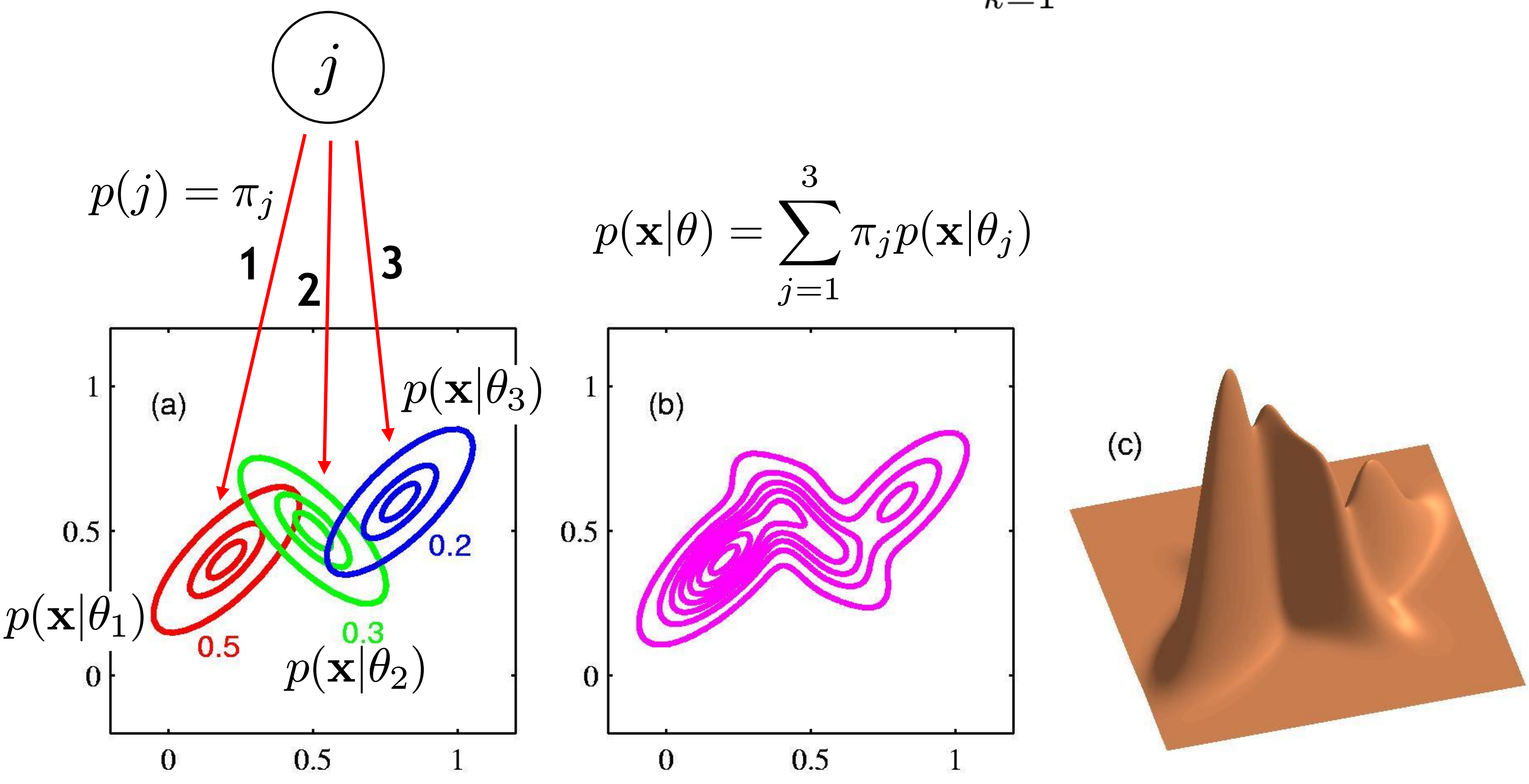
# Recap: Mixture of Multivariate Gaussians

- **"Generative model"**

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

$$p(j) = \pi_j$$

$$p(\mathbf{x}|\theta) = \sum_{j=1}^{3} \pi_j p(\mathbf{x}|\theta_j)$$



# Recap: ML for Mixtures of Gaussians

- **Maximum Likelihood**
  - **Minimize** $E = -\ln L(\theta) = -\sum_{n=1}^{N} \ln p(\mathbf{x}_n|\theta)$
  - **We can already see that this will be difficult, since**

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

This will cause problems!

# Recap: ML for Mixture of Gaussians

$$\boxed{\frac{\partial}{\partial \boldsymbol{\mu}_j}\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k) = \boldsymbol{\Sigma}^{-1}(\mathbf{x}_n-\boldsymbol{\mu}_j)\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)}$$

- **Minimization:**

$$\frac{\partial E}{\partial \boldsymbol{\mu}_j} = -\sum_{n=1}^{N} \frac{\frac{\partial}{\partial \boldsymbol{\mu}_j} p(\mathbf{x}_n|\theta_j)}{\sum_{k=1}^{K} p(\mathbf{x}_n|\theta_k)}$$

$$= -\sum_{n=1}^{N} \left( \boldsymbol{\Sigma}^{-1}(\mathbf{x}_n-\boldsymbol{\mu}_j) \frac{p(\mathbf{x}_n|\theta_j)}{\sum_{k=1}^{K} p(\mathbf{x}_n|\theta_k)} \right)$$

$$= -\cancel{\boldsymbol{\Sigma}}^{-1} \sum_{n=1}^{N} (\mathbf{x}_n-\boldsymbol{\mu}_j) \underbrace{\frac{\pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j)}{\sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)}}_{=\gamma_j(\mathbf{x}_n)} \stackrel{!}{=} 0$$

"responsibility" of component $j$ for $\mathbf{x}_n$

- **We thus obtain**

$$\Rightarrow \boldsymbol{\mu}_j = \frac{\sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)\mathbf{x}_n}{\sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)}$$

RWTH AACHEN UNIVERSITY

# Recap: ML for Mixtures of Gaussians

- But...

$$\boldsymbol{\mu}_j = \frac{\sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)\mathbf{x}_n}{\sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)} \qquad \gamma_j(\mathbf{x}_n) = \frac{\pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}{\sum_{k=1}^{N} \pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}$$

- I.e. there is no direct analytical solution!

$$\frac{\partial E}{\partial \boldsymbol{\mu}_j} = f\left(\pi_1, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1, \ldots, \pi_M, \boldsymbol{\mu}_M, \boldsymbol{\Sigma}_M\right)$$

  - Complex gradient function (non-linear mutual dependencies)
  - Optimization of one Gaussian depends on all other Gaussians!
  - It is possible to apply iterative numerical optimization here, but the EM algorithm provides a simpler alternative.

# Recap: EM Algorithm

- **Expectation-Maximization (EM) Algorithm**
  - **E-Step**: softly assign samples to mixture components

$$\gamma_j(\mathbf{x}_n) \leftarrow \frac{\pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}{\sum_{k=1}^{N} \pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)} \qquad \forall j = 1, \ldots, K, \quad n = 1, \ldots, N$$

  - **M-Step**: re-estimate the parameters (separately for each mixture component) based on the soft assignments

$$\hat{N}_j \leftarrow \sum_{n=1}^{N} \gamma_j(\mathbf{x}_n) \text{ = soft number of samples labeled } j$$

$$\hat{\pi}_j^{\text{new}} \leftarrow \frac{\hat{N}_j}{N}$$

$$\hat{\boldsymbol{\mu}}_j^{\text{new}} \leftarrow \frac{1}{\hat{N}_j} \sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)\mathbf{x}_n$$

$$\hat{\boldsymbol{\Sigma}}_j^{\text{new}} \leftarrow \frac{1}{\hat{N}_j} \sum_{n=1}^{N} \gamma_j(\mathbf{x}_n)(\mathbf{x}_n - \hat{\boldsymbol{\mu}}_j^{\text{new}})(\mathbf{x}_n - \hat{\boldsymbol{\mu}}_j^{\text{new}})^{\mathrm{T}}$$

# Recap: EM Algorithm – An Example

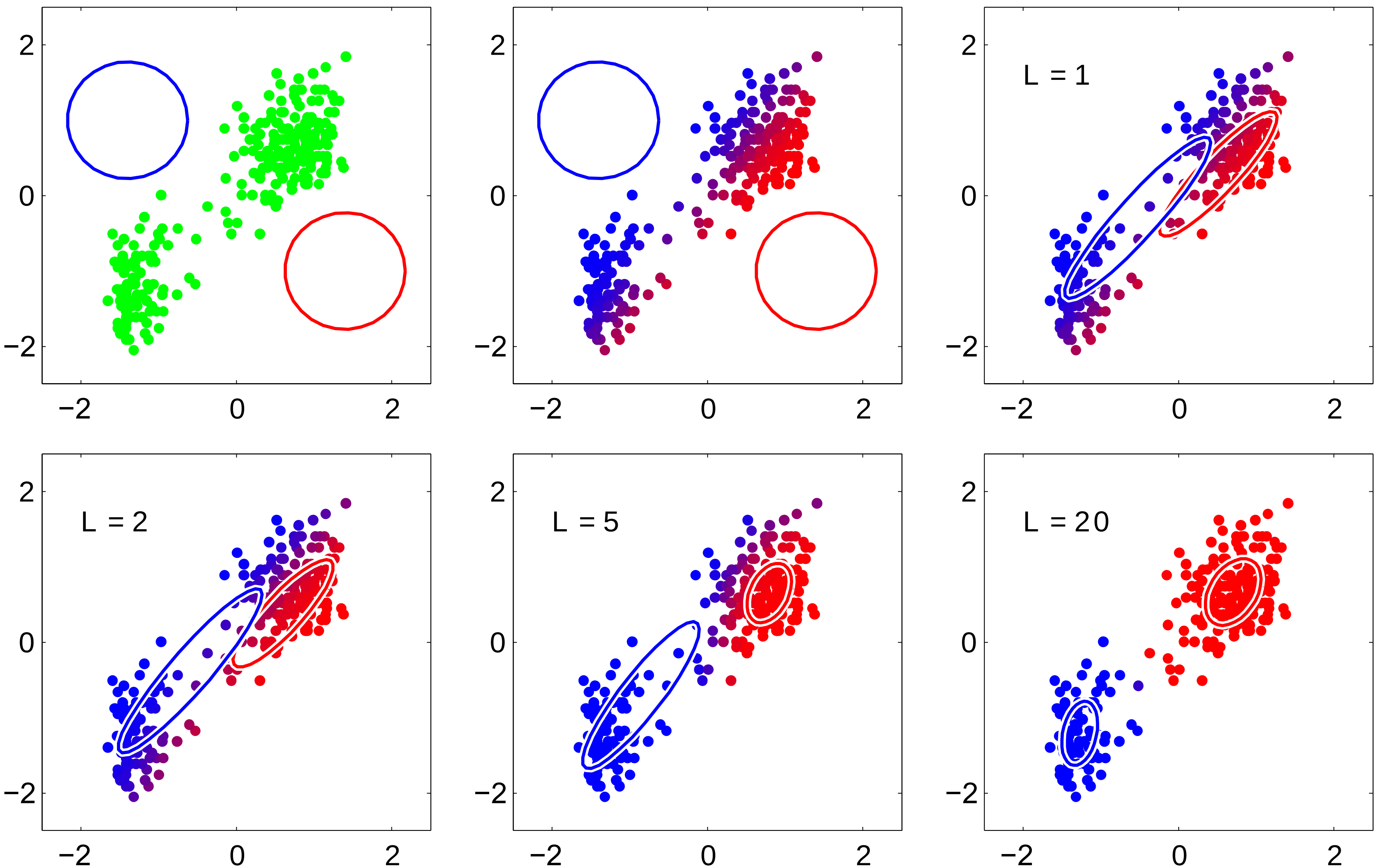


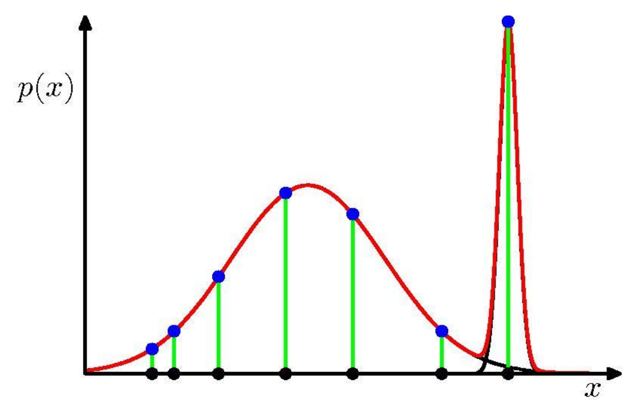



Image source: C.M. Bishop, 2006

# Recap: EM – Caveats

- **When implementing EM, we need to take care to avoid singularities in the estimation!**
  - Mixture components may collapse on single data points.
  - E.g. consider the case $\Sigma_k = \sigma_k^2 \mathbf{I}$ (this also holds in general)
  - Assume component $j$ is exactly centered on data point $\mathbf{x}_n$. This data point will then contribute a term in the likelihood function

$$\mathcal{N}(\mathbf{x}_n|\mathbf{x}_n, \sigma_j^2 \mathbf{I}) = \frac{1}{\sqrt{2\pi}\sigma_j}$$




  - For $\sigma_j \to 0$, this term goes to infinity!

$\Rightarrow$ **Need to introduce regularization**
  - Enforce minimum width for the Gaussians


Image source: C.M. Bishop, 2006

# Application: Image Segmentation

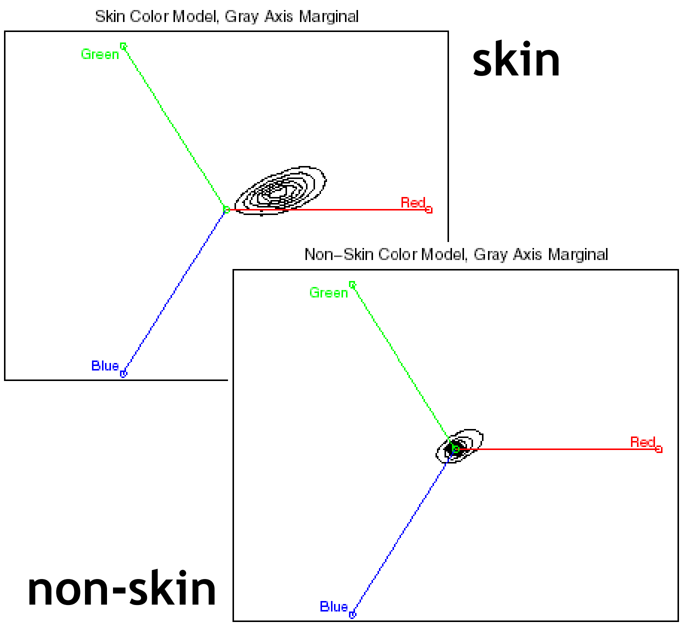
(a) input image (b) user input (c) inferred segmentation

- **User assisted image segmentation**
  - User marks two regions for foreground and background.
  - Learn a MoG model for the color values in each region.
  - Use those models to classify all other pixels.

  $\Rightarrow$ Simple segmentation procedure
  (building block for more complex applications)

# Application: Color-Based Skin Detection

- Collect training samples for skin/non-skin pixels.
- Estimate MoG to represent the skin/ non-skin densities




Classify skin color pixels in novel images

M. Jones and J. Rehg, [Statistical Color Models with Application to Skin Detection](), IJCV 2002.

# Outlook for Today

- Criticism
  - This is all very nice, but in the ML lecture, the EM algorithm miraculously fell out of the air.
  - Why do we actually solve it this way?
- This lecture
  - We will take a more general view on EM
    - Different interpretation in terms of latent variables
    - Detailed derivation
  - This will allow us to derive EM algorithms also for other cases.
  - In particular, we will use it for estimating mixtures of Bernoulli distributions in the next lecture.

# Topics of This Lecture

- Recap: Mixtures of Gaussians
  - Mixtures of Gaussians
  - ML estimation
  - EM algorithm for MoGs
- An alternative view of EM
  - Latent variables
  - General EM
  - Mixtures of Gaussians revisited
  - Mixtures of Bernoulli distributions
- The EM algorithm in general
  - Generalized EM
  - Monte Carlo EM

# Gaussian Mixtures as Latent Variable Model

- Mixture of Gaussians
  - Can be written as linear superposition of Gaussians in the form

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

- Let's write this in a different form…
  - Introduce a $K$-dimensional binary random variable $\mathbf{z}$ with a 1-of-$K$ coding, i.e., $z_k = 1$ and all other elements are zero.
  - Define the joint distribution over $\mathbf{x}$ and $\mathbf{z}$ as

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

  - This corresponds to the following graphical model:

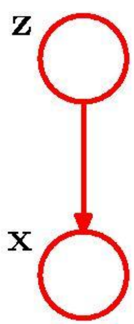




Image source: C.M. Bishop, 2006

# Gaussian Mixtures as Latent Variable Models

- **Marginal distribution over $\mathbf{z}$**
  - Specified in terms of the mixing coefficients $\pi_k$, such that

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

where $0 \cdot \pi_j \cdot 1$ and $\sum_{j=1}^{K} \pi_j = 1.$

  - Since $\mathbf{z}$ uses a 1-of-$K$ representation, we can also write this as

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

  - Similarly, we can write for the conditional distribution

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

# Gaussian Mixtures as Latent Variable Models

- Marginal distribution of $\mathbf{x}$
  - Summing the joint distribution over all possible states of $\mathbf{z}$

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

- What have we gained by this?
  - The resulting formula looks still the same after all...

  $\Rightarrow$ We have represented the marginal distribution in terms of latent variables $\mathbf{z}$.

  - Since $p(\mathbf{x}) = \sum_{\mathbf{z}} p(\mathbf{x}, \mathbf{z})$, there is a corresponding latent variable $\mathbf{z}_n$ for each data point $\mathbf{x}_n$.
  - We are now able to work with the joint distribution $p(\mathbf{x}, \mathbf{z})$ instead of the marginal distribution $p(\mathbf{x})$.

  $\Rightarrow$ This will lead to significant simplifications...

# Gaussian Mixtures as Latent Variable Models

- **Conditional probability of $\mathbf{z}$ given $\mathbf{x}$:**
  - **Use again the "responsibility" notation $\gamma_k(z_k)$**

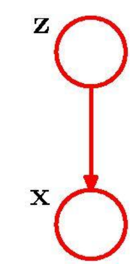
$$\gamma(z_k) \equiv p(z_k = 1|\mathbf{x}) = \frac{p(z_k = 1)p(\mathbf{x}|z_k = 1)}{\sum_{j=1}^{K} p(z_j = 1)p(\mathbf{x}|z_j = 1)}$$

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

  - **We can view $\pi_k$ as the prior probability of $z_k = 1$ and $\gamma(z_k)$ as the corresponding posterior once we have observed $\mathbf{x}$.**

# Sidenote: Sampling from a Gaussian Mixture

- MoG Sampling
  - We can use ancestral sampling to generate random samples from a Gaussian mixture model.
    1. Generate a value $\hat{\mathbf{z}}$ from the marginal distribution $p(\mathbf{z})$.
    2. Generate a value $\hat{\mathbf{x}}$ from the conditional distribution $p(\mathbf{x}|\hat{\mathbf{z}})$.




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

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

Evaluating the responsibilities $\gamma(z_{nk})$

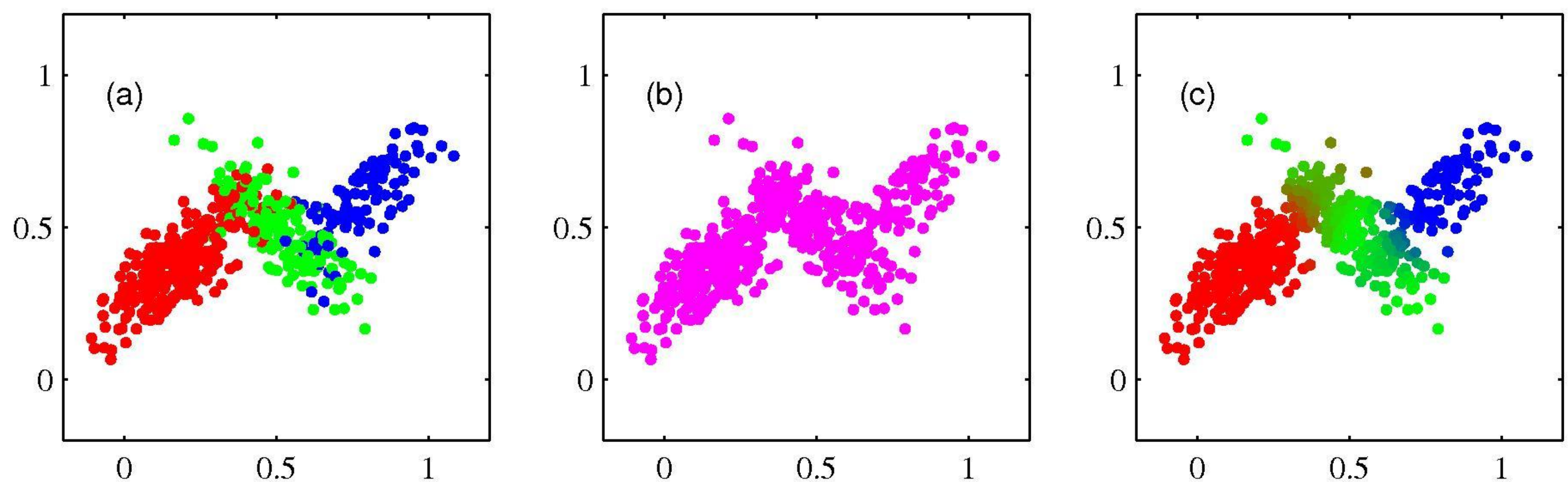




Image source: C.M. Bishop, 2006

# Alternative View of EM

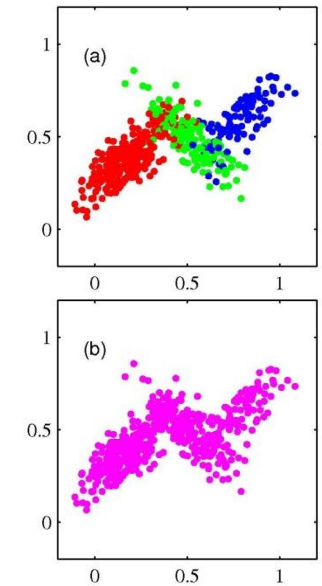
- **Complementary view of the EM algorithm**
  - The goal of EM is to find ML solutions for models having latent variables.
  - Notation
    - Set of all data $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^T$
    - Set of all latent variables $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_N]^T$
    - Set of all model parameters $\theta$
  - Log-likelihood function

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

  - Key observation: summation inside logarithm $\Rightarrow$ difficult.

# Alternative View of EM

- Now, suppose we were told for each observation in $\mathbf{X}$ the corresponding value of the latent variable $\mathbf{Z}$…
  - Call $\{\mathbf{X},\mathbf{Z}\}$ the complete data set and refer to the actual observed data $\mathbf{X}$ as incomplete.




  - The likelihood for the complete data set now takes the form

$$\log p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$$

$\Rightarrow$ Straightforward to marginalize…


Image source: C.M. Bishop, 2006

# Alternative View of EM

- In practice, however,...
  - We are not given the complete data set $\{\mathbf{X},\mathbf{Z}\}$, but only the incomplete data $\mathbf{X}$.
  - Our knowledge of the latent variable values in $\mathbf{Z}$ is given only by the posterior distribution $p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta})$.
  - Since we cannot use the complete-data log-likelihood, we consider instead its expected value under the posterior distribution of the latent variable:

$$\mathcal{Q}(\boldsymbol{\theta},\boldsymbol{\theta}^{\text{old}}) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}}) \log p(\mathbf{X},\mathbf{Z}|\boldsymbol{\theta})$$

  - This corresponds to the E-step of the EM algorithm.
  - In the subsequent M-step, we then maximize the expectation to obtain the revised parameter set $\boldsymbol{\theta}^{\text{new}}$.

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

# General EM Algorithm

- Algorithm
  1. Choose an initial setting for the parameters $\boldsymbol{\theta}^{\text{old}}$
  2. E-step: Evaluate $p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{\text{old}})$
  3. M-step: Evaluate $\boldsymbol{\theta}^{\text{new}}$ given by

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

where

$$\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{\text{old}}) \log p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$$

  4. While not converged, let $\boldsymbol{\theta}^{\text{old}} \leftarrow \boldsymbol{\theta}^{\text{new}}$ and return to step 2.

# Remark: MAP-EM

- **Modification for MAP**
  - The EM algorithm can be adapted to find MAP solutions for models for which a prior $p(\boldsymbol{\theta})$ is defined over the parameters.
  - Only changes needed:

2. E-step: Evaluate $p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{\text{old}})$

3. M-step: Evaluate $\boldsymbol{\theta}^{\text{new}}$ given by

$$\boldsymbol{\theta}^{\text{new}} = \arg\max_{\boldsymbol{\theta}} \; \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}) + \log p(\boldsymbol{\theta})$$

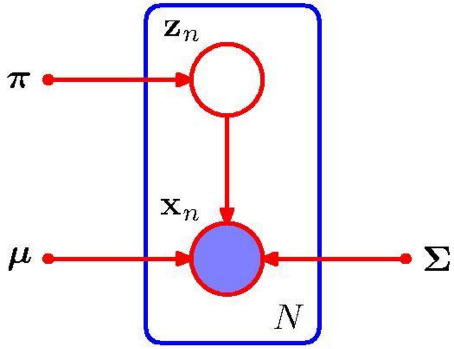
$\Rightarrow$ Suitable choices for the prior will remove the ML singularities!

# Gaussian Mixtures Revisited

- Applying the latent variable view of EM
  - Goal is to maximize the log-likelihood using the observed data $\mathbf{X}$

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

  - Corresponding graphical model:




  - Suppose we are additionally given the values of the latent variables $\mathbf{Z}$.
  - The corresponding graphical model for the complete data now looks like this:

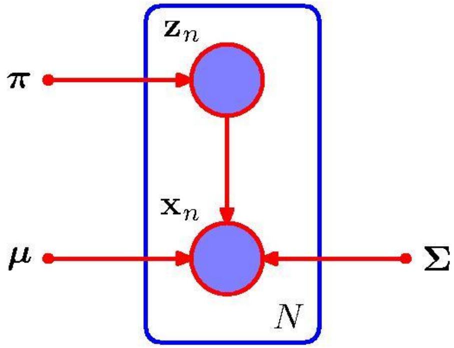




Image source: C.M. Bishop, 2006

# Gaussian Mixtures Revisited

- **Maximize the likelihood**
  - **For the complete-data set $\{\mathbf{X},\mathbf{Z}\}$, the likelihood has the form**

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

  - **Taking the logarithm, we obtain**

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

  - **Compared to the incomplete-data case, the order of the sum and logarithm has been interchanged.**

$\Rightarrow$ **Much simpler solution to the ML problem.**

  - **Maximization w.r.t. a mean or covariance is exactly as for a single Gaussian, except that it involves only the subset of data points that are “assigned” to that component.**

# Gaussian Mixtures Revisited

- **Maximization w.r.t. mixing coefficients**
  - More complex, since the $\pi_k$ are coupled by the summation constraint

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

  - Solve with a Lagrange multiplier

$$\log p(\mathbf{X}, \mathbf{Z} | \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi}) + \lambda \left( \sum_{k=1}^{K} \pi_k - 1 \right)$$

  - Solution (after a longer derivation):

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

$\Rightarrow$ The complete-data log-likelihood can be maximized trivially in closed form.

# Gaussian Mixtures Revisited

- **In practice, we don't have values for the latent variables**
  - Consider the expectation w.r.t. the posterior distribution of the latent variables instead.
  - The posterior distribution takes the form

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

  and factorizes over $n$, so that the $\{\mathbf{z}_n\}$ are independent under the posterior.

  Expected value of indicator variable $z_{nk}$ under the posterior.

$$\begin{aligned}
\mathbb{E}[z_{nk}] &= \frac{\sum_{z_{nk}} z_{nk} \left[\pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right]^{z_{nk}}}{\sum_{z_{nj}} \left[\pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)\right]^{z_{nj}}} \\
&= \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} = \gamma(z_{nk})
\end{aligned}$$

# Gaussian Mixtures Revisited

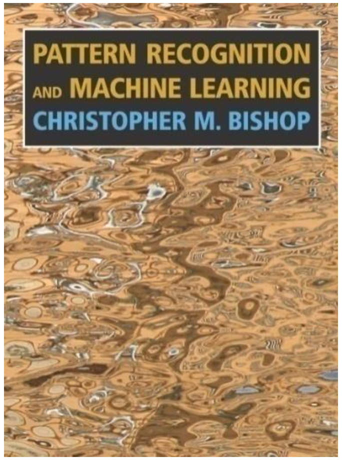
- **Continuing the estimation**
  - **The complete-data log-likelihood is therefore**

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

$\Rightarrow$ **This is precisely the EM algorithm for Gaussian mixtures as derived before.**

# References and Further Reading

- More information about EM and MoG estimation is available in Chapter 9 of Bishop's book (recommendable to read).




Christopher M. Bishop
Pattern Recognition and Machine Learning
Springer, 2006

- Additional information
  - Original EM paper:
    - A.P. Dempster, N.M. Laird, D.B. Rubin, „Maximum-Likelihood from incomplete data via EM algorithm", In Journal Royal Statistical Society, Series B. Vol 39, 1977
  - EM tutorial:
    - J.A. Bilmes, "A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models", TR-97-021, ICSI, U.C. Berkeley, CA,USA