These are the slides of the lecture

Pattern Recognition
Winter term 2020/21
Friedrich-Alexander University of Erlangen-Nuremberg.

Erlangen, January 8, 2021
Prof. Dr.-Ing. Andreas Maier

Pattern Recognition (PR)

Prof. Dr.-Ing. Andreas Maier
Pattern Recognition Lab (CS 5), Friedrich-Alexander-Universität Erlangen-Nürnberg
Winter Term 2020/21

The Expectation Maximization Algorithm

Parameter Estimation Methods

Goal: Derivation of a parameter estimation technique that can deal with

- high dimensional parameter spaces and
- latent, hidden, incomplete data.

Parameter Estimation Methods

Goal: Derivation of a parameter estimation technique that can deal with

- high dimensional parameter spaces and
- latent, hidden, incomplete data.

Parameter estimation techniques known from statistics:

1. Maximum likelihood estimation (ML estimation)
   - All observations are assumed to be mutually statistically independent.
   - The observations are kept fixed.
   - The (log-)likelihood function is optimized regarding the parameters.

Parameter Estimation Methods

Goal: Derivation of a parameter estimation technique that can deal with

- high dimensional parameter spaces and
- latent, hidden, incomplete data.

Parameter estimation techniques known from statistics:

1. Maximum likelihood estimation (ML estimation)
   - All observations are assumed to be mutually statistically independent.
   - The observations are kept fixed.
   - The (log-)likelihood function is optimized regarding the parameters.
2. Maximum a-posteriori estimation (MAP estimation)
   - The probability density function of the parameters $p(\boldsymbol{\theta})$ to be estimated is known.

Parameter Estimation

Let X be the observed random variable and $\boldsymbol{\theta}$ the parameter set.

The estimates of $\boldsymbol{\theta}$ are denoted by $\hat{\boldsymbol{\theta}}$.

Let x be an event assigned to the random variable X.

Parameter Estimation

Let X be the observed random variable and $\boldsymbol{\theta}$ the parameter set.
The estimates of $\boldsymbol{\theta}$ are denoted by $\hat{\boldsymbol{\theta}}$.
Let x be an event assigned to the random variable X.

- ML estimation: $\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\; p(x; \boldsymbol{\theta}) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\; \log p(x; \boldsymbol{\theta})$

Parameter Estimation

Let X be the observed random variable and $\boldsymbol{\theta}$ the parameter set.
The estimates of $\boldsymbol{\theta}$ are denoted by $\hat{\theta}$.
Let x be an event assigned to the random variable X.

- ML estimation: $\hat{\theta} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\ p(x;\boldsymbol{\theta}) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\ \log p(x;\boldsymbol{\theta})$
- MAP estimation:

$$
\begin{aligned}
\hat{\theta} &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\ p(\boldsymbol{\theta}|x) \\
&= \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\ \frac{p(\boldsymbol{\theta})\,p(x|\boldsymbol{\theta})}{\sum_{\boldsymbol{\theta}} p(\boldsymbol{\theta})\,p(x|\boldsymbol{\theta})} \\
&= \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\ \log p(\boldsymbol{\theta}) + \log p(x|\boldsymbol{\theta})
\end{aligned}
$$

Here $\boldsymbol{\theta}$ is considered as a random variable and its probability density function $p(\boldsymbol{\theta})$ is known.

ML Estimation: Example

Example

Let us assume a Gaussian distributed random vector:

$$p(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{\det(2\pi\boldsymbol{\Sigma})}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}$$

ML Estimation: Example

Example

Let us assume a Gaussian distributed random vector:

$$p(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{\sqrt{\det(2\pi\boldsymbol{\Sigma})}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}$$

- We observe the random vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_m$ (training data).

ML Estimation: Example

Example

Let us assume a Gaussian distributed random vector:

$$p(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{\det(2\pi\boldsymbol{\Sigma})}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}-\boldsymbol{\mu})}$$

- We observe the random vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_m$ (training data).
- Based on these training data, we have to estimate the mean vector $\boldsymbol{\mu}$ and the covariance matrix $\boldsymbol{\Sigma}$.

ML Estimation: Example (cont.)

Example (cont.)

The ML estimator assumes mutually independent observations and optimizes the pdf for the given set of training data:

$$\{\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}\} = \underset{\boldsymbol{\mu}, \boldsymbol{\Sigma}}{\operatorname{argmax}} \prod_{i=1}^{m} p(\boldsymbol{x}_i; \boldsymbol{\mu}, \boldsymbol{\Sigma})$$

ML Estimation: Example (cont.)

Example (cont.)

The ML estimator assumes mutually independent observations and optimizes the pdf for the given set of training data:

$$
\begin{aligned}
\{\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}\} &= \operatorname*{argmax}_{\boldsymbol{\mu},\boldsymbol{\Sigma}} \prod_{i=1}^{m} p(\boldsymbol{x}_i; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \\
&= \operatorname*{argmax}_{\boldsymbol{\mu},\boldsymbol{\Sigma}} \sum_{i=1}^{m} \log p(\boldsymbol{x}_i; \boldsymbol{\mu}, \boldsymbol{\Sigma})
\end{aligned}
$$

ML Estimation: Example (cont.)

Example (cont.)

The ML estimator assumes mutually independent observations and optimizes the pdf for the given set of training data:

$$
\begin{aligned}
\{\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}\} &= \underset{\boldsymbol{\mu},\boldsymbol{\Sigma}}{\operatorname{argmax}} \prod_{i=1}^{m} p(\boldsymbol{x}_i; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \\
&= \underset{\boldsymbol{\mu},\boldsymbol{\Sigma}}{\operatorname{argmax}} \sum_{i=1}^{m} \log p(\boldsymbol{x}_i; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \\
&= \underset{\boldsymbol{\mu},\boldsymbol{\Sigma}}{\operatorname{argmax}} L(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m; \boldsymbol{\mu}, \boldsymbol{\Sigma})
\end{aligned}
$$

ML Estimation: Example (cont.)

Example (cont.)

The ML estimator assumes mutually independent observations and optimizes the pdf for the given set of training data:

$$
\begin{aligned}
\{\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}\} &= \underset{\boldsymbol{\mu},\boldsymbol{\Sigma}}{\operatorname{argmax}} \prod_{i=1}^{m} p(\boldsymbol{x}_i; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \\
&= \underset{\boldsymbol{\mu},\boldsymbol{\Sigma}}{\operatorname{argmax}} \sum_{i=1}^{m} \log p(\boldsymbol{x}_i; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \\
&= \underset{\boldsymbol{\mu},\boldsymbol{\Sigma}}{\operatorname{argmax}} L(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m; \boldsymbol{\mu}, \boldsymbol{\Sigma})
\end{aligned}
$$

where the log-likelihood function is defined by

$$
L := L(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{i=1}^{m} \log p(\boldsymbol{x}_i; \boldsymbol{\mu}, \boldsymbol{\Sigma})
$$

ML Estimation: Example (cont.)

Example (cont.)

Necessary conditions for the estimation of the parameters are:

$$\frac{\partial L}{\partial \boldsymbol{\mu}} \overset{!}{=} 0 \qquad \text{and} \qquad \frac{\partial L}{\partial \boldsymbol{\Sigma}} \overset{!}{=} 0$$

ML Estimation: Example (cont.)

Example (cont.)

Necessary conditions for the estimation of the parameters are:

$$\frac{\partial L}{\partial \boldsymbol{\mu}} \stackrel{!}{=} 0 \quad \text{and} \quad \frac{\partial L}{\partial \boldsymbol{\Sigma}} \stackrel{!}{=} 0$$

Now we get for the mean vector:

$$\frac{\partial L}{\partial \boldsymbol{\mu}} = \sum_{i=1}^{m} \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}) \stackrel{!}{=} 0$$

ML Estimation: Example (cont.)

Example (cont.)

Necessary conditions for the estimation of the parameters are:

$$\frac{\partial L}{\partial \boldsymbol{\mu}} \stackrel{!}{=} 0 \quad \text{and} \quad \frac{\partial L}{\partial \boldsymbol{\Sigma}} \stackrel{!}{=} 0$$

Now we get for the mean vector:

$$\frac{\partial L}{\partial \boldsymbol{\mu}} = \sum_{i=1}^{m} \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}) \stackrel{!}{=} 0$$

and thus the ML estimate for the mean vector meets our expectation:

$$\hat{\mu} = \frac{1}{m} \sum_{i=1}^{m} \boldsymbol{x}_i$$

ML Estimation: Example (cont.)

Example (cont.)

Along the same lines, we get the estimator of the covariance matrix
by computation of the zero crossings of the partial derivatives w. r. t. the components of the covariance matrix:

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{m} \sum_{i=1}^{m} (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}})(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}})^T$$

Gaussian Mixture Models

So far, we have considered parameter estimation for statistical models with:

Gaussian Mixture Models

So far, we have considered parameter estimation for statistical models with:

- one class-dependent distribution component

Gaussian Mixture Models

So far, we have considered parameter estimation for statistical models with:

- one class-dependent distribution component
- uni- or multivariate feature vectors

Gaussian Mixture Models

So far, we have considered parameter estimation for statistical models with:

- one class-dependent distribution component
- uni- or multivariate feature vectors
- the type was mostly Gaussian (normally distributed features)

Gaussian Mixture Models

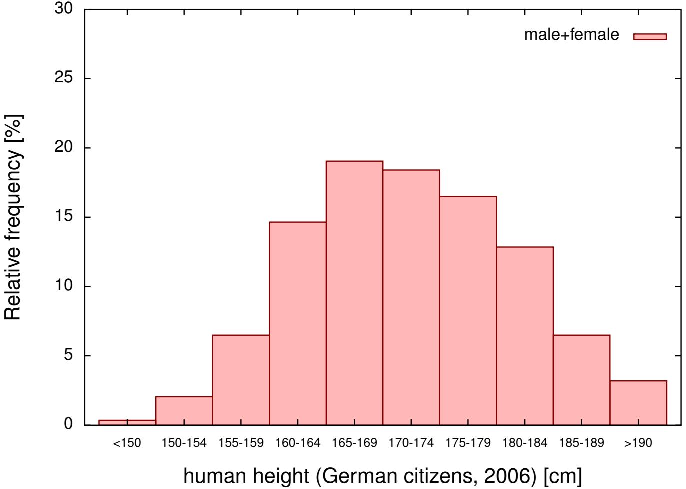
So far, we have considered parameter estimation for statistical models with:

- one class-dependent distribution component
- uni- or multivariate feature vectors
- the type was mostly Gaussian (normally distributed features)

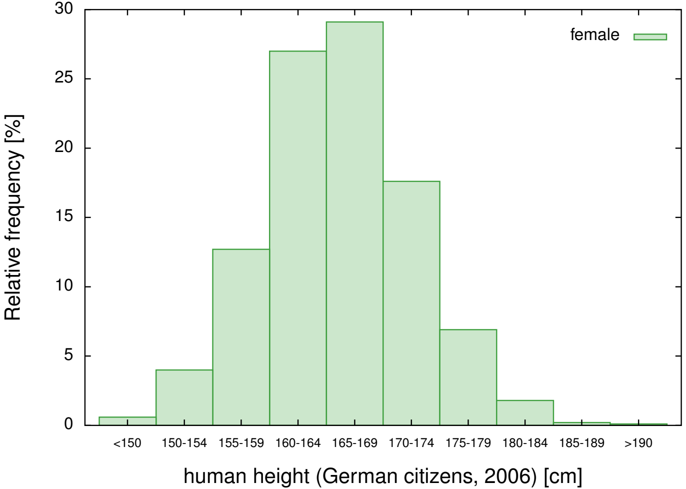
Now we extend this model by representing the observations with a set of K multivariate Gaussian distributions:

Gaussian Mixture Model (GMM)

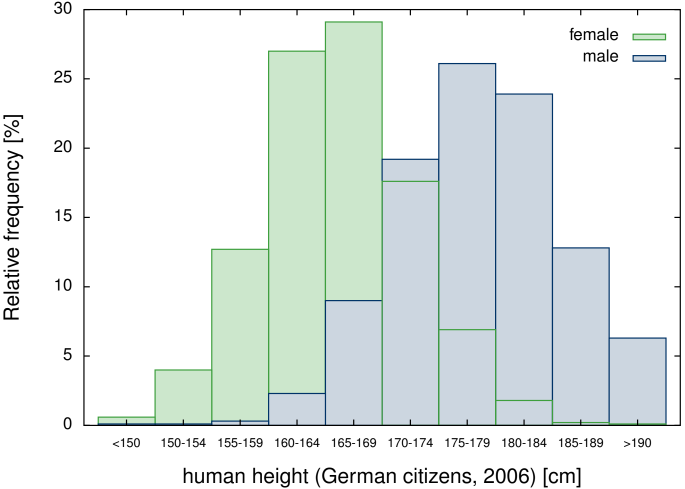
Gaussian Mixture Models (cont.)



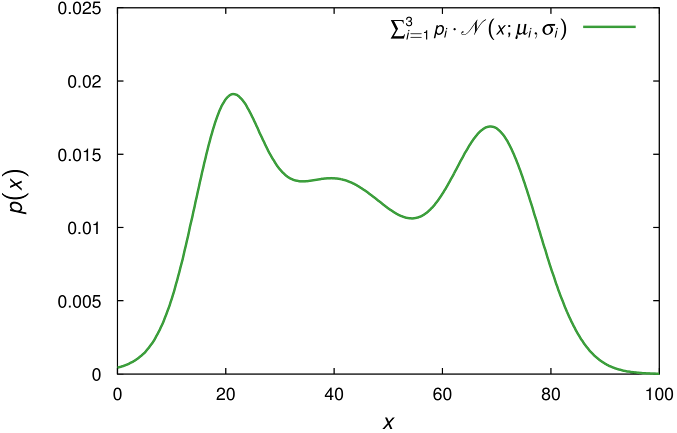
Gaussian Mixture Models (cont.)



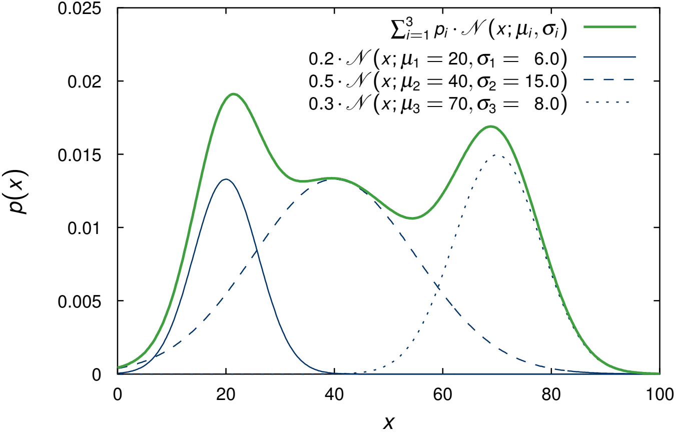
Gaussian Mixture Models (cont.)



Gaussian Mixture Models (cont.)



Gaussian Mixture Models (cont.)



Gaussian Mixture Models (cont.)

Problem description:

Given m feature vectors in an d dimensional space, find a set of K multivariate Gaussian distributions that best represent the observations.

Gaussian Mixture Models (cont.)

Problem description:

Given m feature vectors in an d dimensional space, find a set of K multivariate Gaussian distributions that best represent the observations.

GMMs are an example of classification by *unsupervised learning*:

Gaussian Mixture Models (cont.)

Problem description:

Given m feature vectors in an d dimensional space, find a set of K multivariate Gaussian distributions that best represent the observations.

GMMs are an example of classification by *unsupervised learning*:

- It is not known which feature vectors are generated by which of the K Gaussians

Gaussian Mixture Models (cont.)

Problem description:

Given m feature vectors in an d dimensional space, find a set of K multivariate Gaussian distributions that best represent the observations.

GMMs are an example of classification by *unsupervised learning*:

- It is not known which feature vectors are generated by which of the K Gaussians
- The desired output is, for each feature vector, an estimate of the probability that it is generated by distribution k

Gaussian Mixture Models (cont.)

GMM parameter estimation:

$\boldsymbol{\mu}_k$	the K means
$\boldsymbol{\Sigma}_k$	the K covariance matrices of size $d \times d$
p_k	fraction of all features in component k
$p(k \mid i) \equiv p_{ik}$	the K probabilities for each of the m feature vectors $\boldsymbol{x}_i$

Gaussian Mixture Models (cont.)

GMM parameter estimation:

$\boldsymbol{\mu}_k$	the K means
$\boldsymbol{\Sigma}_k$	the K covariance matrices of size $d \times d$
p_k	fraction of all features in component k
$p(k \mid i) \equiv p_{ik}$	the K probabilities for each of the m feature vectors $\boldsymbol{x}_i$

Additional estimates:

$p(\boldsymbol{x})$	probability distribution of observing a feature vector $\boldsymbol{x}$
L	overall log-likelihood function of the estimated parameter set

GMM – Expectation

The key to the estimation problem is the overall log-likelihood objective function L:

$$L = \sum_{i=1}^{m} \log p(\boldsymbol{x}_i)$$

GMM – Expectation

The key to the estimation problem is the overall log-likelihood objective function L:

$$L = \sum_{i=1}^{m} \log p(\boldsymbol{x}_i)$$

Split $p(\boldsymbol{x}_i)$ into its contributions from the K Gaussians:

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

GMM – Expectation

The key to the estimation problem is the overall log-likelihood objective function L:

$$L = \sum_{i=1}^{m} \log p(\boldsymbol{x}_i)$$

Split $p(\boldsymbol{x}_i)$ into its contributions from the K Gaussians:

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

Individual probabilities for the K contributions:

$$p_{ik} \equiv p(k|i) = \frac{p_k \mathcal{N}(\boldsymbol{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{p(\boldsymbol{x}_i)}$$

GMM – Maximization

Problem: How do we get $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$ and p_k?

GMM – Maximization

Problem: How do we get $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$ and p_k?

- Similar to the ML estimate for the Gaussian, we maximize the log-likelihood by deriving w. r. t. the unknowns.

GMM – Maximization

Problem: How do we get $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$ and p_k?

- Similar to the ML estimate for the Gaussian, we maximize the log-likelihood by deriving w. r. t. the unknowns.
- The ML estimates are:

$$\hat{\boldsymbol{\mu}}_k = \frac{\sum_i p_{ik}\boldsymbol{x}_i}{\sum_i p_{ik}}$$

$$\hat{\boldsymbol{\Sigma}}_k = \frac{\sum_i p_{ik}(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_k)(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_k)^T}{\sum_i p_{ik}}$$

$$\hat{p}_k = \frac{1}{m}\sum_{i=1}^{m} p_{ik}$$

GMM Parameter Estimation

Observations:

- If we know the values for the parameters $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, p_k)$, we can compute the expectations (E-step).

GMM Parameter Estimation

Observations:

- If we know the values for the parameters $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, p_k)$, we can compute the expectations (E-step).
- Once we have the expectations we can compute improved values for the parameters (M-step).

GMM Parameter Estimation

Observations:

- If we know the values for the parameters $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, p_k)$, we can compute the expectations (E-step).
- Once we have the expectations we can compute improved values for the parameters (M-step).

We have found an iterative solution scheme for the nonlinear GMM parameter estimation problem:

- *Right at* the ML solution both E- and M-step relations hold.

GMM Parameter Estimation

Observations:

- If we know the values for the parameters $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, p_k)$, we can compute the expectations (E-step).
- Once we have the expectations we can compute improved values for the parameters (M-step).

We have found an iterative solution scheme for the nonlinear GMM parameter estimation problem:

- *Right at* the ML solution both E- and M-step relations hold.
- The ML parameters are a stationary point for the E- and M-step.

GMM Parameter Estimation

Observations:

- If we know the values for the parameters $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, p_k)$, we can compute the expectations (E-step).
- Once we have the expectations we can compute improved values for the parameters (M-step).

We have found an iterative solution scheme for the nonlinear GMM parameter estimation problem:

- *Right at* the ML solution both E- and M-step relations hold.
- The ML parameters are a stationary point for the E- and M-step.
- Starting from any parameter values, an iteration of the E-step combined with an M-step will increase L

GMM Parameter Estimation (cont.)

EM algorithm for GMM parameter estimation:

Initialization: $\boldsymbol{\mu}_k^{(0)}, \boldsymbol{\Sigma}_k^{(0)}, p_k^{(0)}$
$j \leftarrow 0$
Expectation step: compute new values for p_{ik}, L
Maximization step: update values for $\boldsymbol{\mu}_k^{(j)}, \boldsymbol{\Sigma}_k^{(j)}, p_k^{(j)}$
$j \leftarrow j+1$
L is no longer changing
Output: estimates $\hat{\boldsymbol{\mu}}_k, \hat{\boldsymbol{\Sigma}}_k, \hat{p}_k$

Pattern Recognition Lab

FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG

TECHNISCHE FAKULTÄT

Next Time in

Pattern Recognition

Missing Information Principle

A colloquial formulation of the missing information principle (MIP) is as simple as:

$$\text{observable information} = \text{complete information} - \text{hidden information}$$

Missing Information Principle (cont.)

Mathematical formalization of the MIP:

- observable random variable: X
- hidden random variable: Y
- parameter set: $\boldsymbol{\theta}$

Missing Information Principle (cont.)

Mathematical formalization of the MIP:

- observable random variable: X
- hidden random variable: Y
- parameter set: $\boldsymbol{\theta}$

The joint probability density of the events x (observation) and y (hidden) is:

$$p(x, y; \boldsymbol{\theta}) = p(x; \boldsymbol{\theta})\, p(y | x; \boldsymbol{\theta})$$

Missing Information Principle (cont.)

Mathematical formalization of the MIP:

- observable random variable: X
- hidden random variable: Y
- parameter set: $\boldsymbol{\theta}$

The joint probability density of the events x (observation) and y (hidden) is:

$$p(x, y; \boldsymbol{\theta}) = p(x; \boldsymbol{\theta})\, p(y|x; \boldsymbol{\theta})$$

and thus:

$$p(x; \boldsymbol{\theta}) = \frac{p(x, y; \boldsymbol{\theta})}{p(y|x; \boldsymbol{\theta})}$$

Missing Information Principle (cont.)

Mathematical formalization of the MIP:

- observable random variable: X
- hidden random variable: Y
- parameter set: $\boldsymbol{\theta}$

The joint probability density of the events x (observation) and y (hidden) is:

$$p(x, y; \boldsymbol{\theta}) = p(x; \boldsymbol{\theta})\, p(y|x; \boldsymbol{\theta})$$

and thus:

$$p(x; \boldsymbol{\theta}) = \frac{p(x, y; \boldsymbol{\theta})}{p(y|x; \boldsymbol{\theta})}$$

The mathematical formulation of the MIP is:

$$-\log p(x; \boldsymbol{\theta}) = -\log p(x, y; \boldsymbol{\theta}) - (-\log p(y|x; \boldsymbol{\theta}))$$

Key Equation

We now consider the mathematical formulation of the key equation and derive an iterative parameter estimation scheme:

Key Equation

We now consider the mathematical formulation of the key equation and derive an iterative parameter estimation scheme:

- Let i denote the iteration parameter.

Key Equation

We now consider the mathematical formulation of the key equation and derive an iterative parameter estimation scheme:

- Let i denote the iteration parameter.
- Consider the key equation $(i+1)$-st iteration

$$\log p\left(x; \hat{\theta}^{(i+1)}\right) = \log p\left(x, y; \hat{\theta}^{(i+1)}\right) - \log p\left(y|x; \hat{\theta}^{(i+1)}\right) \ ,$$

where $\hat{\theta}^{(i+1)}$ denotes the estimation in iteration step $(i+1)$.

Key Equation

We now consider the mathematical formulation of the key equation and derive an iterative parameter estimation scheme:

- Let i denote the iteration parameter.
- Consider the key equation $(i+1)$-st iteration

$$\log p\left(x;\hat{\theta}^{(i+1)}\right) = \log p\left(x,y;\hat{\theta}^{(i+1)}\right) - \log p\left(y|x;\hat{\theta}^{(i+1)}\right) ,$$

  where $\hat{\theta}^{(i+1)}$ denotes the estimation in iteration step $(i+1)$.

- Now we multiply both sides with $p\left(y|x;\hat{\theta}^{(i)}\right)$ and integrate over the hidden event y:

$$\int p\left(y|x;\hat{\theta}^{(i)}\right) \log p\left(x;\hat{\theta}^{(i+1)}\right) \mathrm{d}y$$

Key Equation

We now consider the mathematical formulation of the key equation and derive an iterative parameter estimation scheme:

- Let i denote the iteration parameter.
- Consider the key equation $(i+1)$-st iteration

$$\log p\left(x; \hat{\theta}^{(i+1)}\right) = \log p\left(x, y; \hat{\theta}^{(i+1)}\right) - \log p\left(y|x; \hat{\theta}^{(i+1)}\right) \quad ,$$

  where $\hat{\theta}^{(i+1)}$ denotes the estimation in iteration step $(i+1)$.

- Now we multiply both sides with $p\left(y|x; \hat{\theta}^{(i)}\right)$ and integrate over the hidden event y:

$$\begin{aligned}\int p\left(y|x; \hat{\theta}^{(i)}\right) \log p\left(x; \hat{\theta}^{(i+1)}\right) \mathrm{d}y \quad = \quad & \int p\left(y|x; \hat{\theta}^{(i)}\right) \log p\left(x, y; \hat{\theta}^{(i+1)}\right) \mathrm{d}y - \\ & \int p\left(y|x; \hat{\theta}^{(i)}\right) \log p\left(y|x; \hat{\theta}^{(i+1)}\right) \mathrm{d}y\end{aligned}$$

Key Equation (cont.)

Now consider the left hand side of this equation:

$$\int p\left(y|x;\hat{\theta}^{(i)}\right)\log p\left(x;\hat{\theta}^{(i+1)}\right)\,\mathrm{d}y =$$

Key Equation (cont.)

Now consider the left hand side of this equation:

$$
\begin{aligned}
& \int p\left(y|x;\hat{\theta}^{(i)}\right) \log p\left(x;\hat{\theta}^{(i+1)}\right) \, \mathrm{d}y = \\
= \quad & \log p\left(x;\hat{\theta}^{(i+1)}\right) \int p\left(y|x;\hat{\theta}^{(i)}\right) \, \mathrm{d}y =
\end{aligned}
$$

Key Equation (cont.)

Now consider the left hand side of this equation:

$$\begin{aligned}
& \int p\left(y|x;\hat{\theta}^{(i)}\right) \log p\left(x;\hat{\theta}^{(i+1)}\right) \,\mathrm{d}y = \\
= \quad & \log p\left(x;\hat{\theta}^{(i+1)}\right) \int p\left(y|x;\hat{\theta}^{(i)}\right) \,\mathrm{d}y = \\
= \quad & \log p\left(x;\hat{\theta}^{(i+1)}\right)
\end{aligned}$$

Key Equation (cont.)

Now consider the left hand side of this equation:

$$
\begin{aligned}
& \int p\left(y|x;\hat{\theta}^{(i)}\right) \log p\left(x;\hat{\theta}^{(i+1)}\right) \, \mathrm{d}y = \\
= \quad & \log p\left(x;\hat{\theta}^{(i+1)}\right) \int p\left(y|x;\hat{\theta}^{(i)}\right) \, \mathrm{d}y = \\
= \quad & \log p\left(x;\hat{\theta}^{(i+1)}\right)
\end{aligned}
$$

- Observation: The left side of the key equation is the log likelihood function of observations.

Key Equation (cont.)

Now consider the left hand side of this equation:

$$
\begin{aligned}
& \int p\left(y|x;\hat{\theta}^{(i)}\right) \log p\left(x;\hat{\theta}^{(i+1)}\right) \, \mathrm{d}y = \\
= \quad & \log p\left(x;\hat{\theta}^{(i+1)}\right) \int p\left(y|x;\hat{\theta}^{(i)}\right) \, \mathrm{d}y = \\
= \quad & \log p\left(x;\hat{\theta}^{(i+1)}\right)
\end{aligned}
$$

- Observation: The left side of the key equation is the log likelihood function of observations.
- Conclusion: The maximization of the right hand side of the above key equation corresponds to a ML estimation

Kullback-Leibler Statistics and Entropy

For the terms on the right hand side we introduce the following notation (formally this is incorrect due to the differences in the iteration index):

- Kullback-Leibler Statistics

$$Q(\hat{\theta}^{(i)};\hat{\theta}^{(i+1)}) = \int p(y|x;\hat{\theta}^{(i)}) \log p(x,y;\hat{\theta}^{(i+1)})\,\mathrm{d}y$$

Kullback-Leibler Statistics and Entropy

For the terms on the right hand side we introduce the following notation (formally this is incorrect due to the differences in the iteration index):

- Kullback-Leibler Statistics

$$Q(\hat{\theta}^{(i)}; \hat{\theta}^{(i+1)}) = \int p(y|x; \hat{\theta}^{(i)}) \log p(x, y; \hat{\theta}^{(i+1)}) \, \mathrm{d}y$$

- Entropy:

$$H(\hat{\theta}^{(i)}; \hat{\theta}^{(i+1)}) = -\int p(y|x; \hat{\theta}^{(i)}) \log p(y|x; \hat{\theta}^{(i+1)}) \, \mathrm{d}y$$

Kullback-Leibler Statistics

Let us first take a closer look at the Kullback-Leibler statistics:

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}') = \int p(y|x; \boldsymbol{\theta}) \log p(x, y; \boldsymbol{\theta}') \, \mathrm{d}y$$

The Kullback-Leibler statistics (also called Q-function) w. r. t. $\boldsymbol{\theta}'$ given $\boldsymbol{\theta}$ is the conditional expectation:

$$E[\log p(x, y; \boldsymbol{\theta}') \mid x, \boldsymbol{\theta}] = \int p(y|x; \boldsymbol{\theta}) \log p(x, y; \boldsymbol{\theta}') \, \mathrm{d}y$$

Key Equation

The key equation of the Expectation Maximization algorithm (EM algorithm) can be rewritten:

$$\log p\left(x; \hat{\theta}^{(i+1)}\right) = Q\left(\hat{\theta}^{(i)}; \hat{\theta}^{(i+1)}\right) + H\left(\hat{\theta}^{(i)}; \hat{\theta}^{(i+1)}\right)$$

- Below we will motivate that the maximization of the Kullback-Leibler statistics can replace the optimization of the log-likelihood function.
- A complete proof can be found in the literature (see Further Readings).

Entropy Changes with Iterations

For the entropy we get the inequality:

$$H(\boldsymbol{\theta};\boldsymbol{\theta}') \geq H(\boldsymbol{\theta};\boldsymbol{\theta})$$

Entropy Changes with Iterations

For the entropy we get the inequality:

$$H(\boldsymbol{\theta};\boldsymbol{\theta}') \geq H(\boldsymbol{\theta};\boldsymbol{\theta})$$

This is shown rather straightforward:

$$H(\boldsymbol{\theta};\boldsymbol{\theta}') - H(\boldsymbol{\theta};\boldsymbol{\theta})$$

Entropy Changes with Iterations

For the entropy we get the inequality:

$$H(\boldsymbol{\theta};\boldsymbol{\theta}') \geq H(\boldsymbol{\theta};\boldsymbol{\theta})$$

This is shown rather straightforward:

$$\begin{aligned} & H(\boldsymbol{\theta};\boldsymbol{\theta}') - H(\boldsymbol{\theta};\boldsymbol{\theta}) \\ & \quad = \quad -\int p(y|x;\boldsymbol{\theta}) \log p(y|x;\boldsymbol{\theta}') \,\mathrm{d}y + \int p(y|x;\boldsymbol{\theta}) \log p(y|x;\boldsymbol{\theta}) \,\mathrm{d}y \end{aligned}$$

Entropy Changes with Iterations

For the entropy we get the inequality:

$$H(\boldsymbol{\theta};\boldsymbol{\theta}') \geq H(\boldsymbol{\theta};\boldsymbol{\theta})$$

This is shown rather straightforward:

$$\begin{aligned}
&H(\boldsymbol{\theta};\boldsymbol{\theta}') - H(\boldsymbol{\theta};\boldsymbol{\theta}) \\
&\quad = \quad -\int p(y|x;\boldsymbol{\theta}) \log p(y|x;\boldsymbol{\theta}')\,\mathrm{d}y + \int p(y|x;\boldsymbol{\theta}) \log p(y|x;\boldsymbol{\theta})\,\mathrm{d}y \\
&\quad = \quad -\int p(y|x;\boldsymbol{\theta}) \log \frac{p(y|x;\boldsymbol{\theta}')}{p(y|x;\boldsymbol{\theta})}\,\mathrm{d}y
\end{aligned}$$

Entropy Changes with Iterations

For the entropy we get the inequality:

$$H(\boldsymbol{\theta};\boldsymbol{\theta}') \geq H(\boldsymbol{\theta};\boldsymbol{\theta})$$

This is shown rather straightforward:

$$\begin{aligned}
&H(\boldsymbol{\theta};\boldsymbol{\theta}') - H(\boldsymbol{\theta};\boldsymbol{\theta}) \\
&\quad = \quad -\int p(y|x;\boldsymbol{\theta}) \log p(y|x;\boldsymbol{\theta}')\,\mathrm{d}y + \int p(y|x;\boldsymbol{\theta}) \log p(y|x;\boldsymbol{\theta})\,\mathrm{d}y \\
&\quad = \quad -\int p(y|x;\boldsymbol{\theta}) \log \frac{p(y|x;\boldsymbol{\theta}')}{p(y|x;\boldsymbol{\theta})}\,\mathrm{d}y \\
&\quad = \quad \int p(y|x;\boldsymbol{\theta}) \log \frac{p(y|x;\boldsymbol{\theta})}{p(y|x;\boldsymbol{\theta}')}\,\mathrm{d}y
\end{aligned}$$

Entropy Changes with Iterations (cont.)

The difference of the considered entropies

$$
\begin{aligned}
& H(\boldsymbol{\theta};\boldsymbol{\theta}') - H(\boldsymbol{\theta};\boldsymbol{\theta}) = \\
& = \int p(y|x;\boldsymbol{\theta}) \log \frac{p(y|x;\boldsymbol{\theta})}{p(y|x;\boldsymbol{\theta}')} \, \mathrm{d}y \geq 0
\end{aligned}
$$

is thus the Kullback-Leibler divergence of the pdf's $p(y|x;\boldsymbol{\theta})$ and $p(y|x;\boldsymbol{\theta}')$, and the Kullback-Leibler divergence is known to be non-negative.

Entropy Changes with Iterations (cont.)

The best to see this is to make use of the inequality

$$\log(x) \leq x - 1$$

and conclude:

$$\int p(x) \log \frac{p(x)}{q(x)} \, \mathrm{d}x \quad = \quad -\int p(x) \log \frac{q(x)}{p(x)} \, \mathrm{d}x$$

Entropy Changes with Iterations (cont.)

The best to see this is to make use of the inequality

$$\log(x) \leq x - 1$$

and conclude:

$$\begin{aligned}
\int p(x) \log \frac{p(x)}{q(x)} \, \mathrm{d}x &= -\int p(x) \log \frac{q(x)}{p(x)} \, \mathrm{d}x \\
&\geq \int p(x) \left(1 - \frac{q(x)}{p(x)} \right) \mathrm{d}x
\end{aligned}$$

Entropy Changes with Iterations (cont.)

The best to see this is to make use of the inequality

$$\log(x) \leq x - 1$$

and conclude:

$$\begin{aligned} \int p(x) \log \frac{p(x)}{q(x)} \, \mathrm{d}x &= -\int p(x) \log \frac{q(x)}{p(x)} \, \mathrm{d}x \\ &\geq \int p(x) \left(1 - \frac{q(x)}{p(x)}\right) \mathrm{d}x \\ &= 1 - 1 = 0 \end{aligned}$$

Expectation Maximization Algorithm

The basic idea of the EM algorithm:

Instead of maximizing the log-likelihood function on the left hand side of the key-equation, we maximize the Kullback-Leibler statistics iteratively while ignoring the entropy term.

Expectation Maximization Algorithm (cont.)

Initialization: $\hat{\theta}^{(0)}$

$i \leftarrow -1$

$i \leftarrow i+1$

Expectation step:

$$Q\left(\hat{\theta}^{(i)}; \boldsymbol{\theta}\right) := \int p\left(y|x; \hat{\theta}^{(i)}\right) \log p(x, y; \boldsymbol{\theta})\, dy$$

Maximization step:

$$\hat{\theta}^{(i+1)} \leftarrow \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, Q\left(\hat{\theta}^{(i)}; \boldsymbol{\theta}\right)$$

$\hat{\theta}^{(i+1)} = \hat{\theta}^{(i)}$

Output: estimate $\hat{\theta} \leftarrow \hat{\theta}^{(i)}$

Advantages of the EM Algorithm

A few practical positive aspects regarding the EM algorithm:

- The maximum of the KL statistics is usually computed using zero crossings of the gradient.

Advantages of the EM Algorithm

A few practical positive aspects regarding the EM algorithm:

- The maximum of the KL statistics is usually computed using zero crossings of the gradient.
- Mostly we find closed form iteration schemes.

Advantages of the EM Algorithm

A few practical positive aspects regarding the EM algorithm:

- The maximum of the KL statistics is usually computed using zero crossings of the gradient.
- Mostly we find closed form iteration schemes.
- Easy to implement closed form iteration formulas (if these exist).

Advantages of the EM Algorithm

A few practical positive aspects regarding the EM algorithm:

- The maximum of the KL statistics is usually computed using zero crossings of the gradient.
- Mostly we find closed form iteration schemes.
- Easy to implement closed form iteration formulas (if these exist).
- Iteration scheme is numerically robust.

Advantages of the EM Algorithm

A few practical positive aspects regarding the EM algorithm:

- The maximum of the KL statistics is usually computed using zero crossings of the gradient.
- Mostly we find closed form iteration schemes.
- Easy to implement closed form iteration formulas (if these exist).
- Iteration scheme is numerically robust.
- Closed form iterations have constant memory requirements.

Advantages of the EM Algorithm

A few practical positive aspects regarding the EM algorithm:

- The maximum of the KL statistics is usually computed using zero crossings of the gradient.
- Mostly we find closed form iteration schemes.
- Easy to implement closed form iteration formulas (if these exist).
- Iteration scheme is numerically robust.
- Closed form iterations have constant memory requirements.
- If the argument in the logarithm can be factorized properly, we observe a decomposition of the parameter space (independent lower dimensional sub-spaces)

Drawbacks of EM

The EM algorithm has a few major drawbacks:

Drawbacks of EM

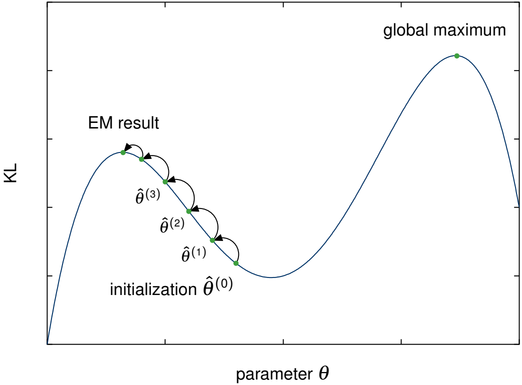
The EM algorithm has a few major drawbacks:

- slow, slow, slow convergence
  (should not be used in run time critical applications)

Drawbacks of EM

The EM algorithm has a few major drawbacks:

- slow, slow, slow convergence
  (should not be used in run time critical applications)
- local optimization method, i. e. the initialization $\hat{\theta}^{(0)}$ has to lie in the area of attraction of the global maximum.



Constrained Optimization

Many optimization problems in the context of the EM algorithm are of the following form:

Example

Optimize the multivariate function

$$f_0(p_1, p_2, \ldots, p_K) = \sum_{k=1}^{K} a_k \log p_k$$

subject to

$$\begin{aligned} \sum_{k=1}^{K} p_k &= 1 \\ p_k &\geq 0 \end{aligned}$$

Constrained Optimization (cont.)

Example

Application of the Lagrange multiplier method:

$$L(p_1, p_2, \dots, p_K) = \sum_{k=1}^{K} a_k \log p_k + \nu \left(\sum_{k=1}^{K} p_k - 1 \right)$$

Constrained Optimization (cont.)

Example

Application of the Lagrange multiplier method:

$$L(p_1, p_2, \ldots, p_K) = \sum_{k=1}^{K} a_k \log p_k + \nu \left(\sum_{k=1}^{K} p_k - 1 \right)$$

The optimization can be done using the partial derivative:

$$\frac{\partial L(p_1, p_2, \ldots, p_K)}{\partial p_k} = \frac{a_k}{p_k} + \nu \stackrel{!}{=} 0 \quad .$$

Constrained Optimization (cont.)

Example (cont.)

The Lagrange multiplier is:

$$a_k = -\nu p_k \ .$$

Constrained Optimization (cont.)

Example (cont.)

The Lagrange multiplier is:

$$a_k = -\nu p_k \ .$$

Due to the fact that the p_k's are unknown, we have to apply a trick to get ν.
We just sum both sides of the above equation over all k and get:

$$\nu = -\sum_{k=1}^{K} a_k \ .$$

Constrained Optimization (cont.)

Example (cont.)

The Lagrange multiplier is:

$$a_k = -\nu p_k \ .$$

Due to the fact that the p_k's are unknown, we have to apply a trick to get ν.
We just sum both sides of the above equation over all k and get:

$$\nu = -\sum_{k=1}^{K} a_k \ .$$

The estimator for p_k now is:

$$\hat{p}_k = \frac{a_k}{\sum_{l=1}^{K} a_l}$$

EM Algorithm: Example

Example

Estimate the priors p_k of classes $k = 1, 2, \dots, K$ from the observation x
where the probability density function of observations is given by the marginal over all classes:

$$p(x; \boldsymbol{\beta}) = \sum_{k=1}^{K} p_k \, p(x|k; \boldsymbol{\beta})$$

EM Algorithm: Example

Example

Estimate the priors p_k of classes $k = 1, 2, \ldots, K$ from the observation x
where the probability density function of observations is given by the marginal over all classes:

$$p(x; \boldsymbol{\beta}) = \sum_{k=1}^{K} p_k \, p(x|k; \boldsymbol{\beta})$$

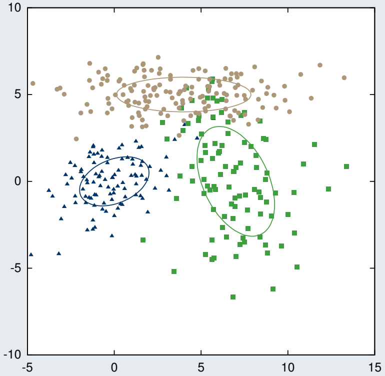
Application of the EM scheme:

- observable random measurement: x
- hidden random measurement: k
- parameter set: $\boldsymbol{\theta} = \{p_k; k = 1, \ldots, K\}$

EM Algorithm: Example (cont.)

Example

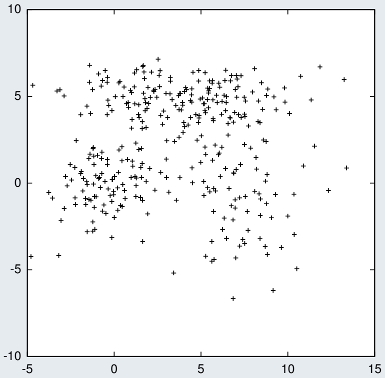
For illustration purposes let us consider three classes. If events, in this case 2-D points, are labeled by colors representing different classes, the priors are easily estimated by relative frequencies.



EM Algorithm: Example (cont.)

Example (cont.)

The problem appears quite difficult, if the class (color) labels are missing.



EM Algorithm: Example (cont.)

Example

The Kullback-Leibler statistics results in:

$$Q\left(\hat{\theta}^{(i)};\hat{\theta}^{(i+1)}\right) = \sum_{k=1}^{K} a_k \log\left(\hat{p}_k^{(i+1)} p(x|k;\boldsymbol{\beta})\right)$$

EM Algorithm: Example (cont.)

Example

The Kullback-Leibler statistics results in:

$$
\begin{aligned}
Q\left(\hat{\theta}^{(i)};\hat{\theta}^{(i+1)}\right) &= \sum_{k=1}^{K} a_k \log\left(\hat{p}_k^{(i+1)}\, p(x|k;\boldsymbol{\beta})\right) \\
&= \sum_{k=1}^{K} a_k \left(\log \hat{p}_k^{(i+1)} + \log p(x|k;\boldsymbol{\beta})\right)
\end{aligned}
$$

EM Algorithm: Example (cont.)

Example

The Kullback-Leibler statistics results in:

$$\begin{aligned}
Q\left(\hat{\theta}^{(i)};\hat{\theta}^{(i+1)}\right) &= \sum_{k=1}^{K} a_k \log\left(\hat{p}_k^{(i+1)} p(x|k;\boldsymbol{\beta})\right) \\
&= \sum_{k=1}^{K} a_k \left(\log \hat{p}_k^{(i+1)} + \log p(x|k;\boldsymbol{\beta})\right) \\
&= \sum_{k=1}^{K} a_k \log \hat{p}_k^{(i+1)} + \sum_{k=1}^{K} a_k \log p(x|k;\boldsymbol{\beta})
\end{aligned}$$

where

$$a_k = \frac{\hat{p}_k^{(i)} p(x|k;\boldsymbol{\beta})}{\sum_j \hat{p}_j^{(i)} p(x|j;\boldsymbol{\beta})}$$

EM Algorithm: Example (cont.)

Example (cont.)

Now we compute the gradient with respect to $\hat{p}_k^{(i+1)}$ and its zero crossing.
The final estimator for priors now is a closed form iteration scheme:

$$\hat{p}_k^{(i+1)} = \frac{\frac{\hat{p}_k^{(i)}\, p(x|k;\boldsymbol{\beta})}{\sum_j \hat{p}_j^{(i)}\, p(x|j,\boldsymbol{\beta})}}{\sum_l \frac{\hat{p}_l^{(i)}\, p(x|l;\boldsymbol{\beta})}{\sum_j \hat{p}_j^{(i)}\, p(x|j;\boldsymbol{\beta})}}$$

EM Algorithm: Example (cont.)

Example (cont.)

Now we compute the gradient with respect to $\hat{p}_k^{(i+1)}$ and its zero crossing.
The final estimator for priors now is a closed form iteration scheme:

$$\hat{p}_k^{(i+1)} = \frac{\frac{\hat{p}_k^{(i)} p(x|k;\boldsymbol{\beta})}{\sum_j \hat{p}_j^{(i)} p(x|j,\boldsymbol{\beta})}}{\sum_l \frac{\hat{p}_l^{(i)} p(x|l;\boldsymbol{\beta})}{\sum_j \hat{p}_j^{(i)} p(x|j;\boldsymbol{\beta})}} = \frac{\hat{p}_k^{(i)} p(x|k;\boldsymbol{\beta})}{\sum_j \hat{p}_j^{(i)} p(x|j;\boldsymbol{\beta})}$$

Initialization of Priors:

- Use prior medical knowledge about the frequency of tissue classes
- If no prior information is available, assume uniform distribution

Lessons Learned

- Standard parameter estimation method: ML estimation
- If the prior pdf of the parameters is known: MAP estimation
- In the presence of latent random variables: EM algorithm
- EM advantages: decomposition of search space, closed form iteration schemes
- EM disadvantage: slow convergence, local method

FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG

TECHNISCHE FAKULTÄT

Pattern Recognition Lab

Next Time in

Pattern Recognition

Further Readings

- Easy to understand tutorial on ML estimation:

  In Jae Myung:
  ☞ Tutorial on maximum likelihood estimation,
  Journal of Mathematical Psychology, 47(1):90-100, 2003

- The classics for an introduction to the EM algorithm is:

  A. P. Dempster, N. M. Laird, D. B. Rubin:
  ☞ Maximum Likelihood Estimation from Incomplete Data via the EM Algorithm,
  Journal of the Royal Statistical Society, Series B, 39(1):1-38.

- W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery:
  ☞ Numerical Recipes,
  3rd Edition, Cambridge University Press, 2007.

Comprehensive Questions

- What is a Gaussian Mixture Model?
- What is the missing information principle?
- Write down the key equation for the EM algorithm:
- Is the EM algorithm a local or a global parameter estimation method?