Parametric Models
Part II: Expectation-Maximization and Mixture Density Estimation

Selim Aksoy

Department of Computer Engineering
Bilkent University
saksoy@cs.bilkent.edu.tr

CS 551, Spring 2012

Missing Features

- Suppose that we have a Bayesian classifier that uses the feature vector $\mathbf{x}$ but a subset $\mathbf{x}_{\boldsymbol{g}}$ of $\mathbf{x}$ are observed and the values for the remaining features $\mathbf{x}_{\boldsymbol{b}}$ are missing.
- How can we make a decision?
  - Throw away the observations with missing values.
  - Or, substitute $\mathbf{x}_{\boldsymbol{b}}$ by their average $\bar{\mathbf{x}}_{\boldsymbol{b}}$ in the training data, and use $\mathbf{x} = (\mathbf{x}_{\boldsymbol{g}}, \bar{\mathbf{x}}_{\boldsymbol{b}})$.
  - Or, marginalize the posterior over the missing features, and use the resulting posterior

$$P(w_i|\mathbf{x}_{\boldsymbol{g}}) = \frac{\int P(w_i|\mathbf{x}_{\boldsymbol{g}}, \mathbf{x}_{\boldsymbol{b}})\, p(\mathbf{x}_{\boldsymbol{g}}, \mathbf{x}_{\boldsymbol{b}})\, d\mathbf{x}_{\boldsymbol{b}}}{\int p(\mathbf{x}_{\boldsymbol{g}}, \mathbf{x}_{\boldsymbol{b}})\, d\mathbf{x}_{\boldsymbol{b}}}.$$

©2012, Selim Aksoy (Bilkent University)

Expectation-Maximization

- We can also extend maximum likelihood techniques to allow learning of parameters when some training patterns have missing features.
- The *Expectation-Maximization (EM)* algorithm is a general iterative method of finding the maximum likelihood estimates of the parameters of a distribution from training data.

Expectation-Maximization

- There are two main applications of the EM algorithm:
  - Learning when the data is incomplete or has missing values.
  - Optimizing a likelihood function that is analytically intractable but can be simplified by assuming the existence of and values for additional but missing (or hidden) parameters.
- The second problem is more common in pattern recognition applications.

Expectation-Maximization

- Assume that the observed data $\mathcal{X}$ is generated by some distribution.
- Assume that a complete dataset $\mathcal{Z} = (\mathcal{X}, \mathcal{Y})$ exists as a combination of the observed but incomplete data $\mathcal{X}$ and the missing data $\mathcal{Y}$.
- The observations in $\mathcal{Z}$ are assumed to be i.i.d. from the joint density

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

Expectation-Maximization

- We can define a new likelihood function

  $$L(\boldsymbol{\Theta}|\mathcal{Z}) = L(\boldsymbol{\Theta}|\mathcal{X},\mathcal{Y}) = p(\mathcal{X},\mathcal{Y}|\boldsymbol{\Theta})$$

  called the complete-data likelihood where $L(\boldsymbol{\Theta}|\mathcal{X})$ is referred to as the incomplete-data likelihood.
- The EM algorithm:
  - First, finds the expected value of the complete-data log-likelihood using the current parameter estimates (expectation step).
  - Then, maximizes this expectation (maximization step).

Expectation-Maximization

- Define

$$Q(\mathbf{\Theta}, \mathbf{\Theta}^{(i-1)}) = E\big[\log p(\mathcal{X}, \mathcal{Y}|\mathbf{\Theta}) \,|\, \mathcal{X}, \mathbf{\Theta}^{(i-1)}\big]$$

  as the expected value of the complete-data log-likelihood w.r.t. the unknown data $\mathcal{Y}$ given the observed data $\mathcal{X}$ and the current parameter estimates $\mathbf{\Theta}^{(i-1)}$.

- The expected value can be computed as

$$E\big[\log p(\mathcal{X}, \mathcal{Y}|\mathbf{\Theta})|\mathcal{X}, \mathbf{\Theta}^{(i-1)}\big] = \int \log p(\mathcal{X}, \mathbf{y}|\mathbf{\Theta})\, p(\mathbf{y}|\mathcal{X}, \mathbf{\Theta}^{(i-1)})\, d\mathbf{y}.$$

- This is called the *E-step*.

Expectation-Maximization

- Then, the expectation can be maximized by finding optimum values for the new parameters $\boldsymbol{\Theta}$ as

$$\boldsymbol{\Theta}^{(i)} = \arg\max_{\boldsymbol{\Theta}} Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(i-1)}).$$

- This is called the *M-step*.
- These two steps are repeated iteratively where each iteration is guaranteed to increase the log-likelihood.
- The EM algorithm is also guaranteed to converge to a local maximum of the likelihood function.

Mixture Densities

- A mixture model is a linear combination of m densities

$$p(\mathbf{x}|\mathbf{\Theta}) = \sum_{j=1}^{m} \alpha_j p_j(\mathbf{x}|\boldsymbol{\theta_j})$$

  where $\mathbf{\Theta} = (\alpha_1, \ldots, \alpha_m, \boldsymbol{\theta_1}, \ldots, \boldsymbol{\theta_m})$ such that $\alpha_j \geq 0$ and $\sum_{j=1}^{m} \alpha_j = 1$.

- $\alpha_1, \ldots, \alpha_m$ are called the mixing parameters.
- $p_j(\mathbf{x}|\boldsymbol{\theta_j})$, $j = 1, \ldots, m$ are called the component densities.

Mixture Densities

- Suppose that $\mathcal{X} = \{\mathbf{x_1}, \ldots, \mathbf{x_n}\}$ is a set of observations i.i.d. with distribution $p(\mathbf{x}|\mathbf{\Theta})$.
- The log-likelihood function of $\mathbf{\Theta}$ becomes

$$\log L(\mathbf{\Theta}|\mathcal{X}) = \log \prod_{i=1}^{n} p(\mathbf{x_i}|\mathbf{\Theta}) = \sum_{i=1}^{n} \log \left(\sum_{j=1}^{m} \alpha_j p_j(\mathbf{x_i}|\boldsymbol{\theta_j}) \right).$$

- We cannot obtain an analytical solution for $\mathbf{\Theta}$ by simply setting the derivatives of $\log L(\mathbf{\Theta}|\mathcal{X})$ to zero because of the logarithm of the sum.

Mixture Density Estimation via EM

- Consider $\mathcal{X}$ as incomplete and define hidden variables $\mathcal{Y} = \{y_i\}_{i=1}^n$ where y_i corresponds to which mixture component generated the data vector $\mathbf{x}_{\boldsymbol{i}}$.
- In other words, $y_i = j$ if the i'th data vector was generated by the j'th mixture component.
- Then, the log-likelihood becomes

$$\begin{aligned}\log L(\boldsymbol{\Theta}|\mathcal{X},\mathcal{Y}) &= \log p(\mathcal{X},\mathcal{Y}|\boldsymbol{\Theta}) \\ &= \sum_{i=1}^{n} \log(p(\mathbf{x}_{\boldsymbol{i}}|y_i,\boldsymbol{\theta}_{\boldsymbol{i}})p(y_i|\boldsymbol{\theta}_{\boldsymbol{i}})) \\ &= \sum_{i=1}^{n} \log(\alpha_{y_i} p_{y_i}(\mathbf{x}_{\boldsymbol{i}}|\boldsymbol{\theta}_{\boldsymbol{y_i}})).\end{aligned}$$

Mixture Density Estimation via EM

- Assume we have the initial parameter estimates $\boldsymbol{\Theta}^{(g)} = (\alpha_1^{(g)}, \ldots, \alpha_m^{(g)}, \boldsymbol{\theta_1}^{(g)}, \ldots, \boldsymbol{\theta_m}^{(g)})$.
- Compute

$$p(y_i|\mathbf{x_i}, \boldsymbol{\Theta}^{(g)}) = \frac{\alpha_{y_i}^{(g)} p_{y_i}(\mathbf{x_i}|\boldsymbol{\theta_{y_i}}^{(g)})}{p(\mathbf{x_i}|\boldsymbol{\Theta}^{(g)})} = \frac{\alpha_{y_i}^{(g)} p_{y_i}(\mathbf{x_i}|\boldsymbol{\theta_{y_i}}^{(g)})}{\sum_{j=1}^{m} \alpha_j^{(g)} p_j(\mathbf{x_i}|\boldsymbol{\theta_j}^{(g)})}$$

and

$$p(\mathcal{Y}|\mathcal{X}, \boldsymbol{\Theta}^{(g)}) = \prod_{i=1}^{n} p(y_i|\mathbf{x_i}, \boldsymbol{\Theta}^{(g)}).$$

Mixture Density Estimation via EM

- Then, $Q(\mathbf{\Theta}, \mathbf{\Theta}^{(g)})$ takes the form

$$
\begin{aligned}
Q(\mathbf{\Theta}, \mathbf{\Theta}^{(g)}) &= \sum_{\mathbf{y}} \log p(\mathcal{X}, \mathbf{y} | \mathbf{\Theta}) p(\mathbf{y} | \mathcal{X}, \mathbf{\Theta}^{(g)}) \\
&= \sum_{j=1}^{m} \sum_{i=1}^{n} \log(\alpha_j p_j(\mathbf{x_i} | \boldsymbol{\theta_j})) p(j | \mathbf{x_i}, \mathbf{\Theta}^{(g)}) \\
&= \sum_{j=1}^{m} \sum_{i=1}^{n} \log(\alpha_j) p(j | \mathbf{x_i}, \mathbf{\Theta}^{(g)}) \\
&\qquad + \sum_{j=1}^{m} \sum_{i=1}^{n} \log(p_j(\mathbf{x_i} | \boldsymbol{\theta_j})) p(j | \mathbf{x_i}, \mathbf{\Theta}^{(g)}).
\end{aligned}
$$

©2012, Selim Aksoy (Bilkent University)

Mixture Density Estimation via EM

- We can maximize the two sets of summations for α_j and $\boldsymbol{\theta_j}$ independently because they are not related.
- The estimate for α_j can be computed as

$$\hat{\alpha}_j = \frac{1}{n}\sum_{i=1}^{n} p(j|\mathbf{x}_{\boldsymbol{i}}, \boldsymbol{\Theta}^{(g)})$$

where

$$p(j|\mathbf{x}_{\boldsymbol{i}}, \boldsymbol{\Theta}^{(g)}) = \frac{\alpha_j^{(g)} p_j(\mathbf{x}_{\boldsymbol{i}}|\boldsymbol{\theta}_{\boldsymbol{j}}^{(g)})}{\sum_{t=1}^{m} \alpha_t^{(g)} p_t(\mathbf{x}_{\boldsymbol{i}}|\boldsymbol{\theta}_{\boldsymbol{t}}^{(g)})}.$$

Mixture of Gaussians

- We can obtain analytical expressions for $\boldsymbol{\theta_j}$ for the special case of a Gaussian mixture where $\boldsymbol{\theta_j} = (\boldsymbol{\mu_j}, \boldsymbol{\Sigma_j})$ and

$$\begin{aligned} p_j(\mathbf{x}|\boldsymbol{\theta_j}) &= p_j(\mathbf{x}|\boldsymbol{\mu_j}, \boldsymbol{\Sigma_j}) \\ &= \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma_j}|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu_j})^T \boldsymbol{\Sigma_j}^{-1}(\mathbf{x}-\boldsymbol{\mu_j})\right]. \end{aligned}$$

- Equating the partial derivative of $Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(g)})$ with respect to $\boldsymbol{\mu_j}$ to zero gives

$$\hat{\boldsymbol{\mu}}_{\boldsymbol{j}} = \frac{\sum_{i=1}^{n} p(j|\mathbf{x_i}, \boldsymbol{\Theta}^{(g)})\mathbf{x_i}}{\sum_{i=1}^{n} p(j|\mathbf{x_i}, \boldsymbol{\Theta}^{(g)})}.$$

Mixture of Gaussians

- We consider five models for the covariance matrix $\boldsymbol{\Sigma_j}$:
  - $\boldsymbol{\Sigma_j} = \sigma^2 \mathbf{I}$

$$\hat{\sigma}^2 = \frac{1}{nd} \sum_{j=1}^{m} \sum_{i=1}^{n} p(j|\mathbf{x_i}, \boldsymbol{\Theta}^{(g)}) \|\mathbf{x_i} - \boldsymbol{\hat{\mu}_j}\|^2$$

  - $\boldsymbol{\Sigma_j} = \sigma_j^2 \mathbf{I}$

$$\hat{\sigma}_j^2 = \frac{\sum_{i=1}^{n} p(j|\mathbf{x_i}, \boldsymbol{\Theta}^{(g)}) \|\mathbf{x_i} - \boldsymbol{\hat{\mu}_j}\|^2}{d \sum_{i=1}^{n} p(j|\mathbf{x_i}, \boldsymbol{\Theta}^{(g)})}$$

©2012, Selim Aksoy (Bilkent University)

Mixture of Gaussians

- Covariance models continued:
  - $\boldsymbol{\Sigma_j} = \text{diag}(\{\sigma_{jk}^2\}_{k=1}^d)$

$$\hat{\sigma}_{jk}^2 = \frac{\sum_{i=1}^n p(j|\mathbf{x_i}, \boldsymbol{\Theta}^{(g)})(\mathbf{x_{\mathit{i}k}} - \boldsymbol{\hat{\mu}_{j}}_k)^2}{\sum_{i=1}^n p(j|\mathbf{x_i}, \boldsymbol{\Theta}^{(g)})}$$

  - $\boldsymbol{\Sigma_j} = \boldsymbol{\Sigma}$

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{n} \sum_{j=1}^m \sum_{i=1}^n p(j|\mathbf{x_i}, \boldsymbol{\Theta}^{(g)})(\mathbf{x_i} - \boldsymbol{\hat{\mu}_j})(\mathbf{x_i} - \boldsymbol{\hat{\mu}_j})^T$$

  - $\boldsymbol{\Sigma_j} =$ arbitrary

$$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{j}} = \frac{\sum_{i=1}^n p(j|\mathbf{x_i}, \boldsymbol{\Theta}^{(g)})(\mathbf{x_i} - \boldsymbol{\hat{\mu}_j})(\mathbf{x_i} - \boldsymbol{\hat{\mu}_j})^T}{\sum_{i=1}^n p(j|\mathbf{x_i}, \boldsymbol{\Theta}^{(g)})}$$

Mixture of Gaussians

- Summary:
  - Estimates for α_j, $\boldsymbol{\mu_j}$ and $\boldsymbol{\Sigma_j}$ perform both expectation and maximization steps simultaneously.
  - EM iterations proceed by using the current estimates as the initial estimates for the next iteration.
  - The priors are computed from the proportion of examples belonging to each mixture component.
  - The means are the component centroids.
  - The covariance matrices are calculated as the sample covariance of the points associated with each component.

Examples

- Mixture of Gaussians examples
- 1-D Bayesian classification examples
- 2-D Bayesian classification examples

©2012, Selim Aksoy (Bilkent University)

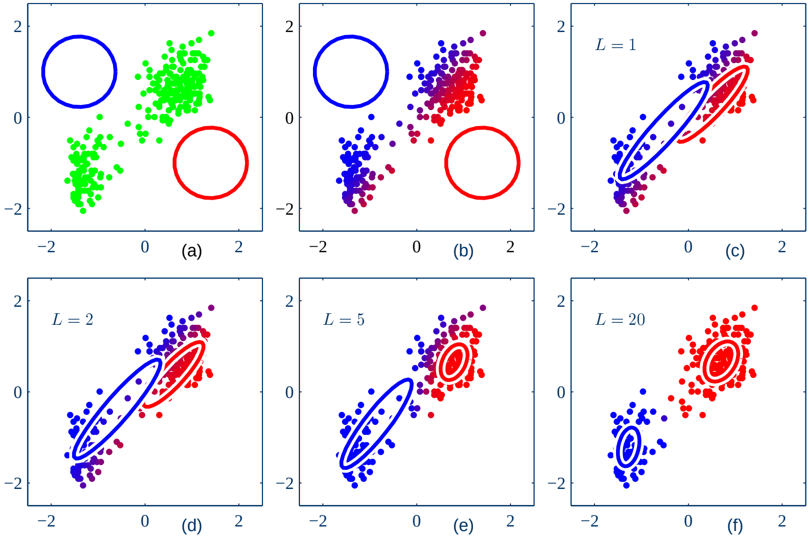
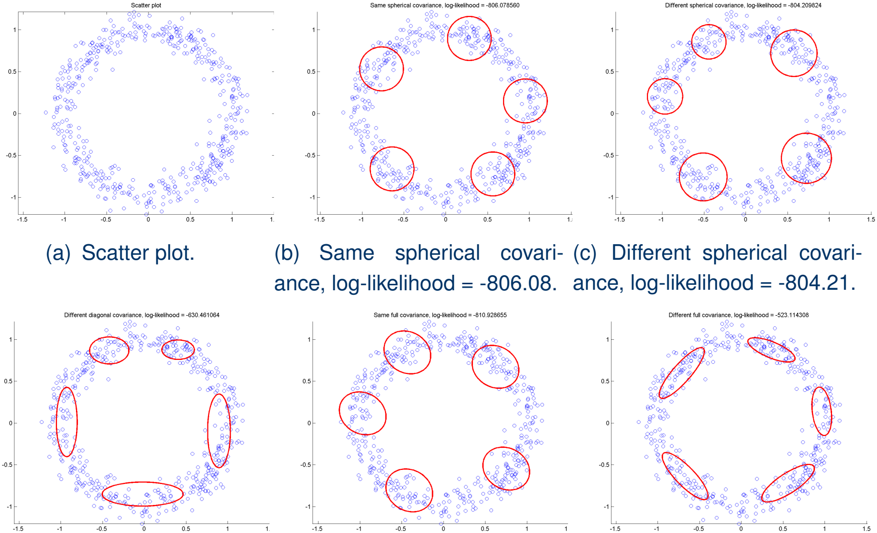


Figure 1: Illustration of the EM algorithm iterations for a mixture of two Gaussians.



(a) Scatter plot.

(b) Same spherical covariance, log-likelihood = -806.08.

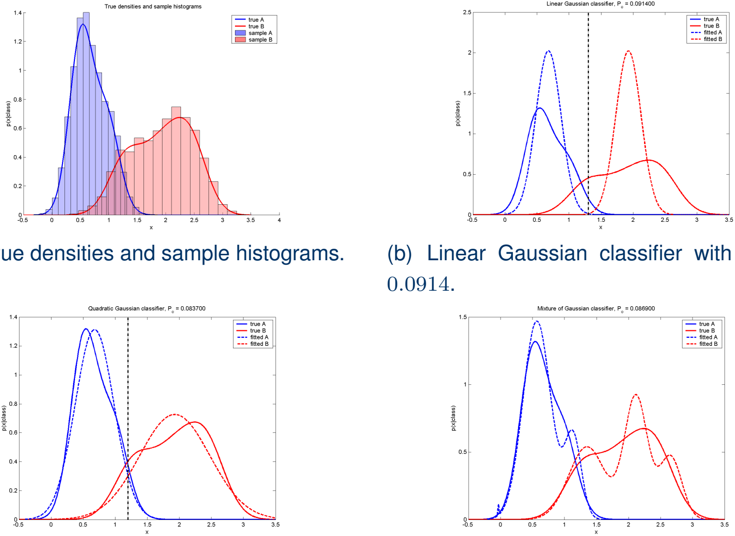
(c) Different spherical covariance, log-likelihood = -804.21.

(d) Different diagonal covariance, log-likelihood = -630.46.

(e) Same arbitrary covariance, log-likelihood = -810.93.

(f) Different arbitrary covariance, log-likelihood = -523.11.

Figure 2: Fitting mixtures of 5 Gaussians to data from a circular distribution



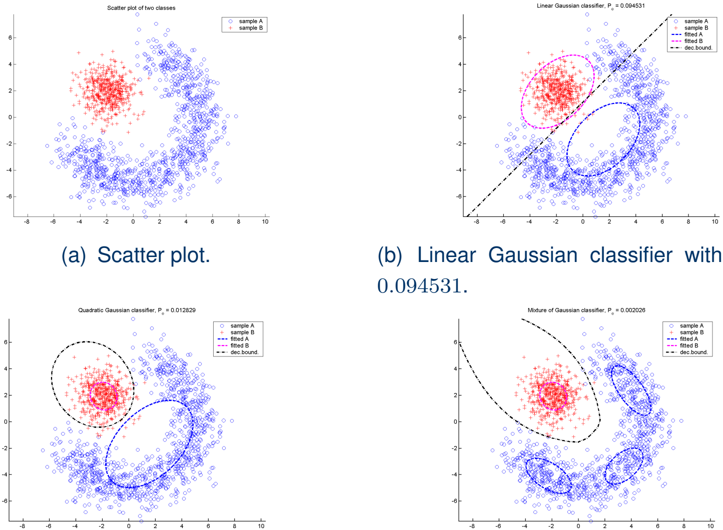
(a) True densities and sample histograms.

(b) Linear Gaussian classifier with $P_e = 0.0914$.

(c) Quadratic Gaussian classifier with $P_e = 0.0837$.

(d) Mixture of Gaussian classifier with $P_e = 0.0869$.

Figure 3: 1-D Bayesian classification examples where the data for each class come from a mixture of three Gaussians. Bayes error is $P_e = 0.0828$.



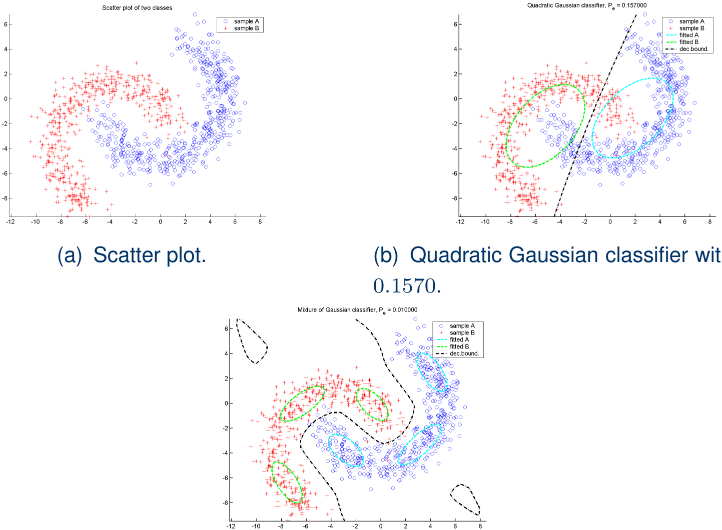
(a) Scatter plot.

(b) Linear Gaussian classifier with $P_e = 0.094531$.

(c) Quadratic Gaussian classifier with $P_e = 0.012829$.

(d) Mixture of Gaussian classifier with $P_e = 0.002026$.

Figure 4: 2-D Bayesian classification examples where the data for the classes come from a banana shaped distribution and a bivariate Gaussian



(a) Scatter plot.

(b) Quadratic Gaussian classifier with $P_e = 0.1570$.

(c) Mixture of Gaussian classifier with $P_e = 0.0100$.

Figure 5: 2-D Bayesian classification examples where the data for each class come from a banana shaped distribution.

Mixture of Gaussians

- Questions:
  - How can we find the initial estimates for $\boldsymbol{\Theta}$?
    - Choose random data points, make them the initial means, assign all points to these means, and compute the priors and covariance matrices.
    - Or, run a clustering algorithm for an initial grouping of all points, and compute the initial estimates from these groups.
  - How do we know when to stop the iterations?
    - Stop if the change in log-likelihood between two iterations is less than a threshold.
    - Or, use a threshold for the number of iterations.
  - How can we find the number of components in the mixture?

Minimum Description Length Principle

- The *Minimum Description Length (MDL)* principle tries to find a compromise between the model complexity (still having a good data approximation) and the complexity of the data approximation (while using a simple model).
- Under the MDL principle, the best model is the one that minimizes the sum of the model's complexity $\mathcal{L}(\mathcal{M})$ and the efficiency of the description of the training data with respect to that model $\mathcal{L}(\mathcal{D}|\mathcal{M})$, i.e.,

$$\mathcal{L}(\mathcal{D}, \mathcal{M}) = \mathcal{L}(\mathcal{M}) + \mathcal{L}(\mathcal{D}|\mathcal{M}).$$

Minimum Description Length Principle

- According to Shannon, the shortest code-length to encode data $\mathcal{D}$ with a distribution $p(\mathcal{D}|\mathcal{M})$ under model $\mathcal{M}$ is given by

$$\mathcal{L}(\mathcal{D}|\mathcal{M}) = -\log L(\mathcal{M}|\mathcal{D}) = -\log p(\mathcal{D}|\mathcal{M})$$

  where $L(\mathcal{M}|\mathcal{D})$ is the likelihood function for model $\mathcal{M}$ given the sample $\mathcal{D}$.

Minimum Description Length Principle

- The model complexity is measured as the number of bits required to describe the model parameters.
- According to Rissanen, the code-length to encode $\kappa_{\mathcal{M}}$ real-valued parameters characterizing n data points is

$$\mathcal{L}(\mathcal{M}) = \frac{\kappa_{\mathcal{M}}}{2} \log n$$

where $\kappa_{\mathcal{M}}$ is the number of free parameters in model $\mathcal{M}$ and n is the size of the sample used to estimate those parameters.

Minimum Description Length Principle

- Once the description lengths for different models have been calculated, we select the one having the smallest such length.
- It can be shown theoretically that classifiers designed with a minimum description length principle are guaranteed to converge to the ideal or true model in the limit of more and more data.

Minimum Description Length Principle

- As an example, let's derive the description lengths for Gaussian mixture models with m components.
- The total number of free parameters for different covariance matrix models are:

$$
\begin{aligned}
\boldsymbol{\Sigma_j} &= \sigma^2 \mathbf{I} & \kappa_{\mathcal{M}} &= (m-1) + md + 1 \\
\boldsymbol{\Sigma_j} &= \sigma_j^2 \mathbf{I} & \kappa_{\mathcal{M}} &= (m-1) + md + m \\
\boldsymbol{\Sigma_j} &= \text{diag}(\{\sigma_{jk}^2\}_{k=1}^{d}) & \kappa_{\mathcal{M}} &= (m-1) + md + md \\
\boldsymbol{\Sigma_j} &= \boldsymbol{\Sigma} & \kappa_{\mathcal{M}} &= (m-1) + md + \frac{d(d+1)}{2} \\
\boldsymbol{\Sigma_j} &= \text{arbitrary} & \kappa_{\mathcal{M}} &= (m-1) + md + m\frac{d(d+1)}{2}
\end{aligned}
$$

where d is the dimension of the feature vectors.

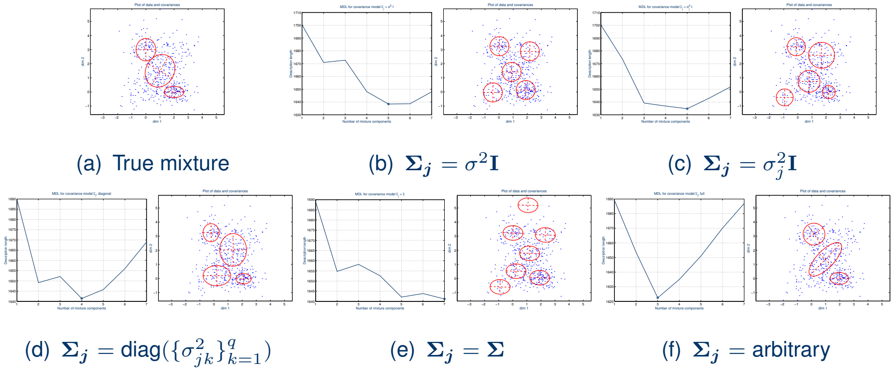
Minimum Description Length Principle

- The first term describes the mixture weights $\{\alpha_j\}_{j=1}^m$, the second term describes the means $\{\boldsymbol{\mu_j}\}_{j=1}^m$ and the third term describes the covariance matrices $\{\boldsymbol{\Sigma_j}\}_{j=1}^m$.
- Hence, the best m can be found as

$$m^* = \arg\min_m \left[\frac{\kappa_{\mathcal{M}}}{2} \log n - \sum_{i=1}^n \log \left(\sum_{j=1}^m \alpha_j p_j(\mathbf{x}_{\boldsymbol{i}} | \boldsymbol{\mu_j}, \boldsymbol{\Sigma_j}) \right) \right].$$

©2012, Selim Aksoy (Bilkent University)

Minimum Description Length Principle



(a) True mixture

(b) $\boldsymbol{\Sigma_j} = \sigma^2\mathbf{I}$

(c) $\boldsymbol{\Sigma_j} = \sigma_j^2\mathbf{I}$

(d) $\boldsymbol{\Sigma_j} = \text{diag}(\{\sigma_{jk}^2\}_{k=1}^q)$

(e) $\boldsymbol{\Sigma_j} = \boldsymbol{\Sigma}$

(f) $\boldsymbol{\Sigma_j} = \text{arbitrary}$

Figure 6: Example fits for a sample from a mixture of three bivariate Gaussians. For each covariance model, description length vs. the number of components (left) and fitted Gaussians as ellipses at one standard deviations (right) are shown. Using MDL with the arbitrary covariance matrix gave the smallest description length and also could capture the true number of components.