Gaussian Mixture Models (GMM) and ML Estimation Examples

Mean and Variance of Gaussian

- Consider the Gaussian PDF:

$$f(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\},$$

Given the observations (sample) $X_1, \cdots, X_n.$

Form the log-likelihood function

$$l(\mu,\sigma) = \sum_{i=1}^{n}\left[-\log\sigma - \frac{1}{2}\log 2\pi - \frac{1}{2\sigma^2}(X_i-\mu)^2\right] = -n\log\sigma - \frac{n}{2}\log 2\pi - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(X_i-\mu)^2$$

Take the derivatives wrt $\mu\ and\ \sigma$ and set it to zero

Solution

- Sample mean and variance:

$$\frac{\partial l(\mu,\sigma)}{\partial \mu} = \frac{1}{\sigma^2}\sum_{i=1}^{n}(X_i-\mu)=0$$

$$\frac{\partial l(\mu,\sigma)}{\partial \sigma} = -\frac{n}{\sigma}+\sigma^{-3}\sum_{i=1}^{n}(X_i-\mu)^2=0$$

Solving these equations will give us the MLE for μ and σ:

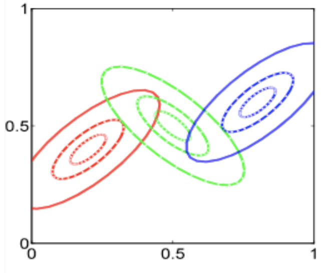
$$\hat{\mu}=\overline{X} \quad \text{and} \quad \hat{\sigma}=\sqrt{\frac{1}{n}\sum_{i=1}^{n}(X_i-\overline{X})^2}$$

Gaussian Mixture Model

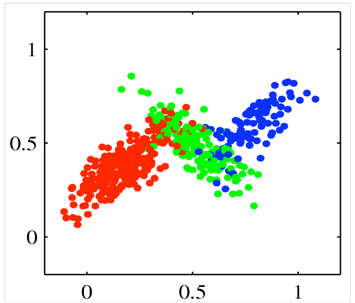
- GMM

$$\Pr(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \text{ where}$$

$$\sum_{k=1}^{K} \pi_k = 1, 0 \leq \pi_k \leq 1.$$



➔



X is multidimensional.
Ref: https://people.eecs.berkeley.edu/~jordan/courses/294-fall09/lectures/clustering/slides.pdf

Can we use the ML estimation method to estimate the unknown parameters, $\mu_k, \sigma_k, \pi_\mathrm{k}$?

- It is not easy:
  - Loss function is the negative log likelihood

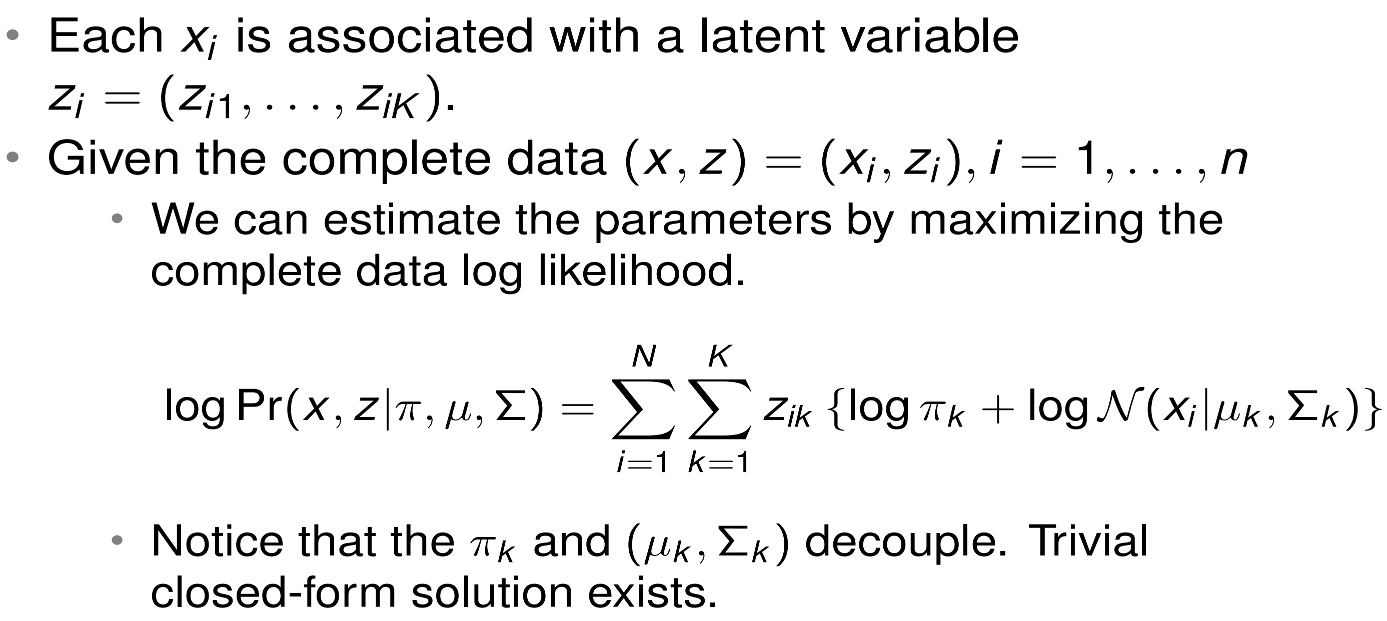
$$-\log \Pr(x|\pi, \mu, \Sigma) = -\sum_{i=1}^{n} \log \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \right\}$$

  - Why is this function difficult to optimize?
    - Notice that the sum over the components appears inside the log, thus coupling all the parameters.

However, it is possible to obtain an iterative solution!

We can estimate the parameters using iterative Expectation-Maximization (EM) algorithm

- Given the observations *x_i, i=1,2,…,n*



- Each x_i is associated with a latent variable $z_i = (z_{i1}, \ldots, z_{iK})$.
- Given the complete data $(x, z) = (x_i, z_i), i = 1, \ldots, n$
  - We can estimate the parameters by maximizing the complete data log likelihood.

$$\log \Pr(x, z|\pi, \mu, \Sigma) = \sum_{i=1}^{N} \sum_{k=1}^{K} z_{ik} \{\log \pi_k + \log \mathcal{N}(x_i|\mu_k, \Sigma_k)\}$$

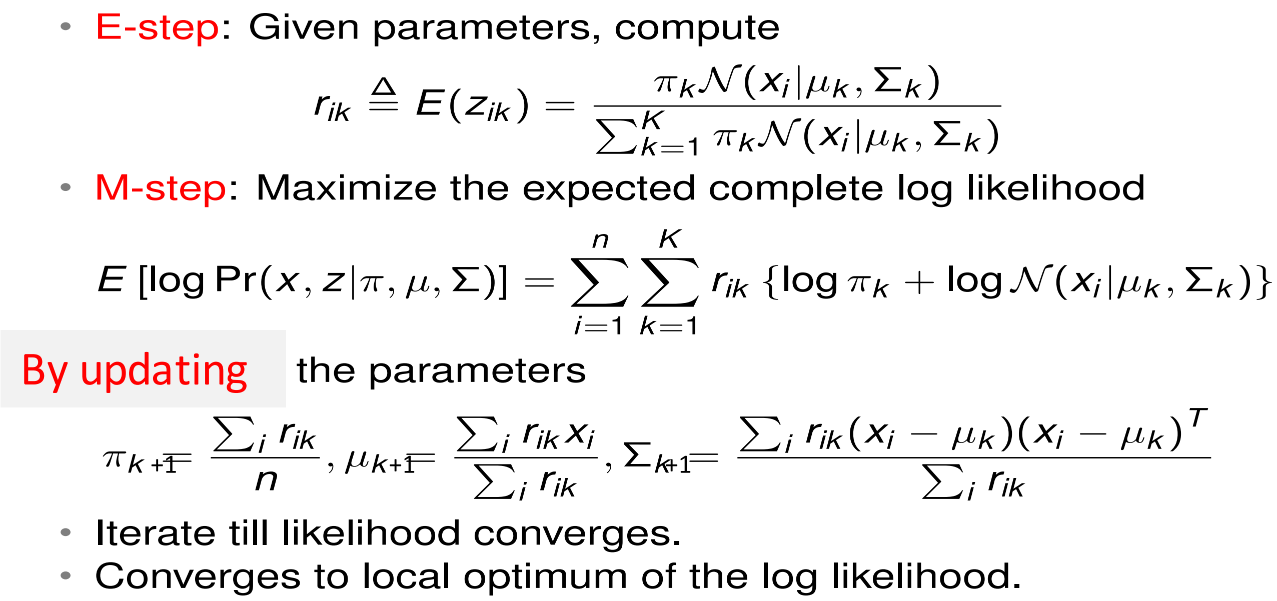
  - Notice that the π_k and (μ_k, Σ_k) decouple. Trivial closed-form solution exists.

The latent variable parameter z_{ik} represents the contribution of k-th Gaussian to x_i

Take the derivative of the log-likelihood wrt μ_k, σ_k, π_k and set it to zero to get equations to be used in EM algorithm

Iterate for *k=1,2,…*

- Initialize with $\mu_o, \sigma_o I, \pi_o$
- Update equations at the k-th iteration:



- E-step: Given parameters, compute

$$r_{ik} \triangleq E(z_{ik}) = \frac{\pi_k \mathcal{N}(x_i|\mu_k, \Sigma_k)}{\sum_{k=1}^{K} \pi_k \mathcal{N}(x_i|\mu_k, \Sigma_k)}$$

- M-step: Maximize the expected complete log likelihood

$$E\left[\log \Pr(x, z|\pi, \mu, \Sigma)\right] = \sum_{i=1}^{n} \sum_{k=1}^{K} r_{ik} \left\{\log \pi_k + \log \mathcal{N}(x_i|\mu_k, \Sigma_k)\right\}$$

By updating the parameters

$$\pi_{k+1} = \frac{\sum_i r_{ik}}{n}, \mu_{k+1} = \frac{\sum_i r_{ik} x_i}{\sum_i r_{ik}}, \Sigma_{k+1} = \frac{\sum_i r_{ik}(x_i - \mu_k)(x_i - \mu_k)^T}{\sum_i r_{ik}}$$

- Iterate till likelihood converges.
- Converges to local optimum of the log likelihood.

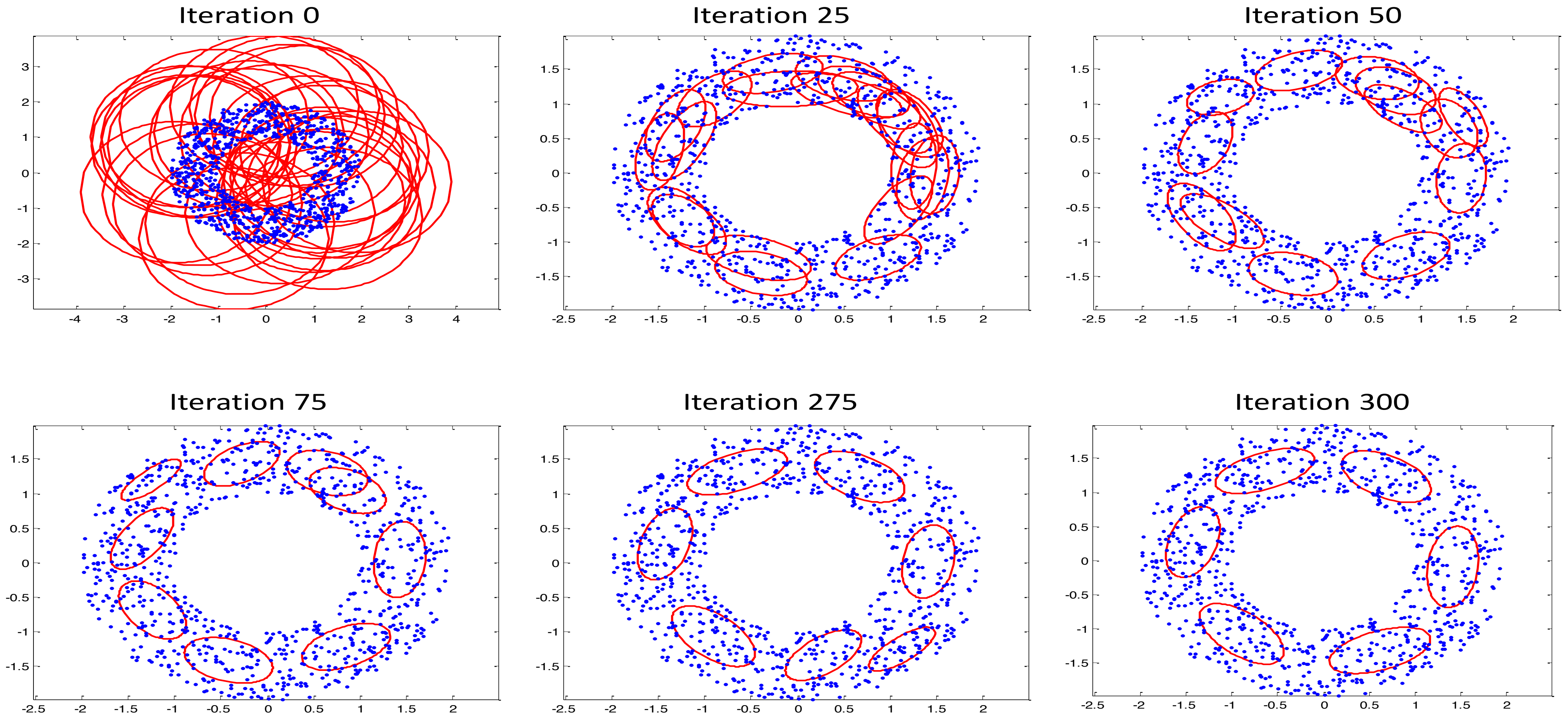
It may not converge to the global optimum!

Example

GMM example

- Training set: $n = 900$ examples from a uniform pdf inside an annulus
- Model: GMM with $C = 30$ Gaussian components
- Training procedure
  - Gaussians centers initialized by choosing 30 arbitrary training examples
  - Covariance matrices initialized to be diagonal, with large variance compared to that of the training data
  - To avoid singularities, at every iteration the covariance matrices computed with EM were regularized with a small multiple of the identity matrix
  - Components whose mixing coefficients fell below a threshold are removed

Observations: blue dots



Ellipses represent 2-D Gaussians

Vector quantization = K-means clustering

k-means clustering

- The k-means algorithm is a simple procedure that attempts to group a collection of unlabeled examples $X = \{x_1 \dots x_n\}$ into one of C clusters
  - k-means seeks to find compact clusters, measured as

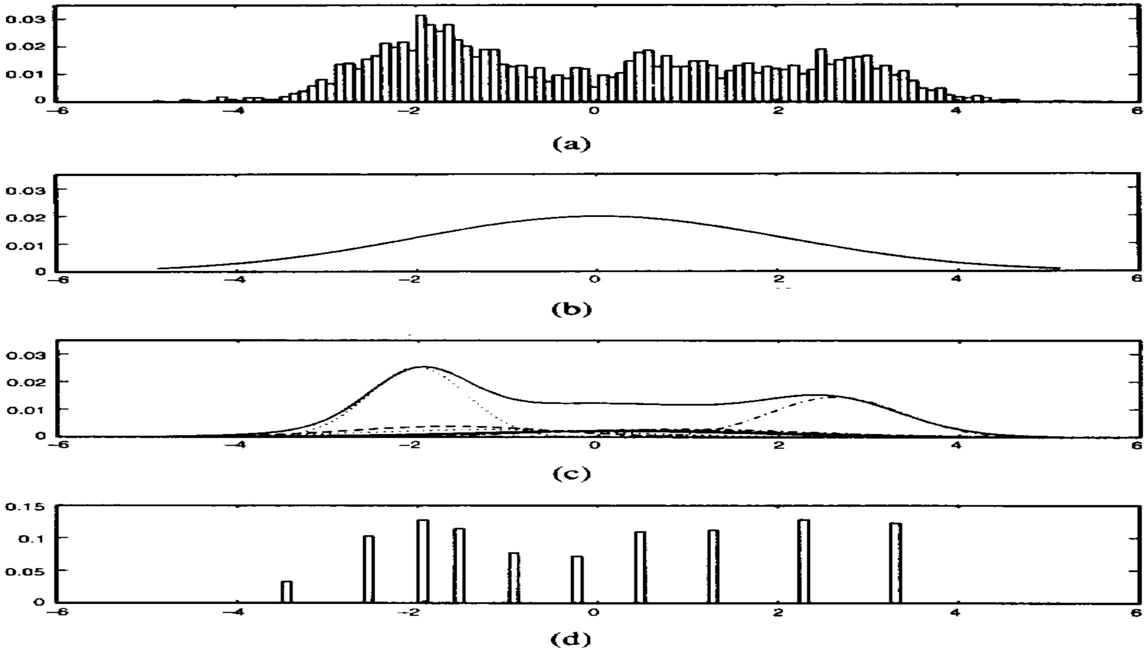
$$J_{MSE} = \sum_{c=1}^{C} \sum_{x \in \omega_c} \|x - \mu_c\|^2 \,;\; \mu_c = \frac{1}{n_c} \sum_{x \in \omega_c} x$$

  - It can be shown that k-means is a special case of the GMM-EM algorithm
- Procedure

1. Define the number of clusters
2. Initialize clusters by
   a) an arbitrary assignment of examples to clusters or
   b) an arbitrary set of cluster centers (i.e., use some examples as centers)
3. Compute the sample mean of each cluster
4. Reassign each example to the cluster with the nearest mean
5. If the classification of all samples has not changed, stop, else go to step 3

You may initialize the GMM algorithm from K-means cluster centers !

Example



(a)

(b)

(c)

(d)

Fig. 3. Comparison of distribution modeling: (a) Histogram of a single cepstral coefficient from a 25 second utterance by a male speaker; (b) maximum likelihood unimodal Gaussian model; (c) GMM and its 10 underlying component densities; (d) histogram of the data assigned to the VQ centroid locations of a 10-element codebook.

Speaker Identification

- Feature extractions from data using mel-cepstrum

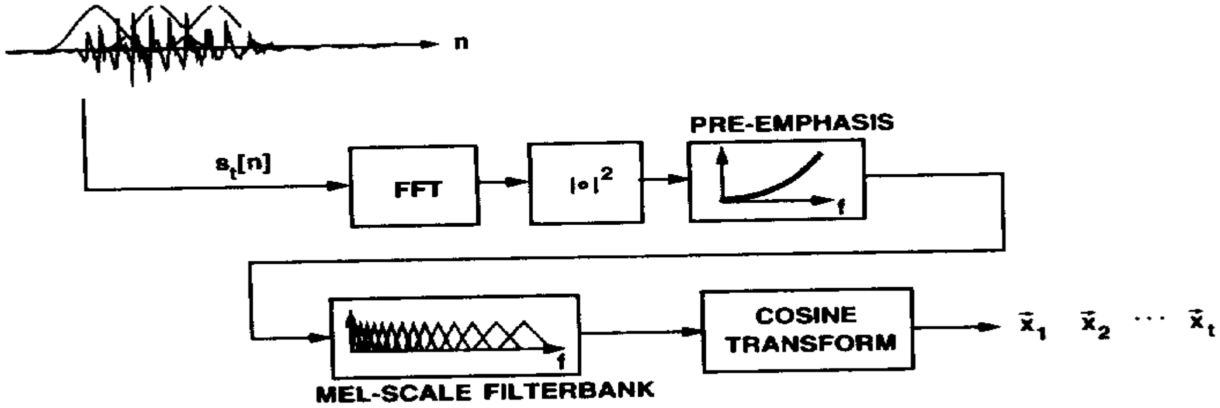
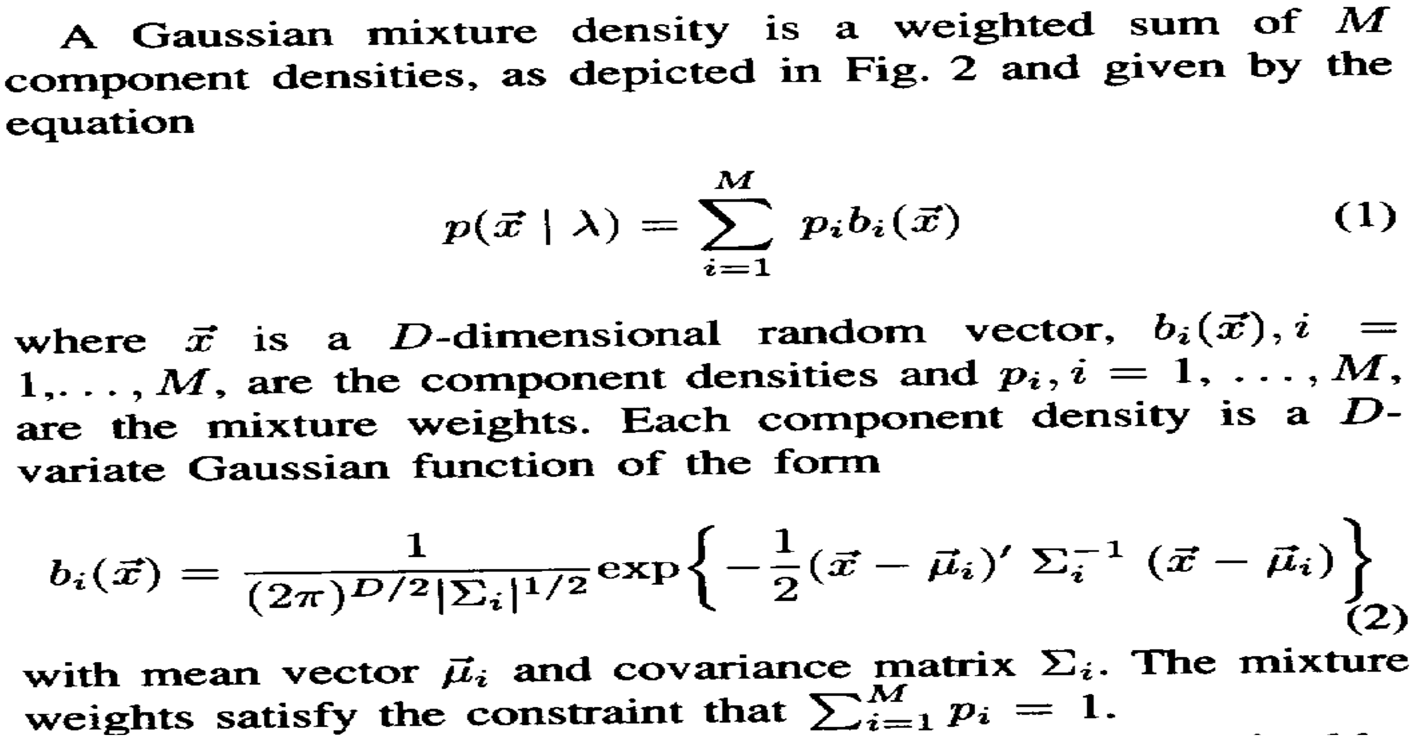


Fig. 1. Mel-scale cepstral feature analysis.

- Extract feature vectors for each speaker (e.g., 90 sec long data)
- Frame length 10ms

Speaker model GMM



A Gaussian mixture density is a weighted sum of M component densities, as depicted in Fig. 2 and given by the equation

$$p(\vec{x} \mid \lambda) = \sum_{i=1}^{M} p_i b_i(\vec{x}) \tag{1}$$

where $\vec{x}$ is a D-dimensional random vector, $b_i(\vec{x}), i = 1,. . . , M$, are the component densities and $p_i, i = 1, . . . , M$, are the mixture weights. Each component density is a D-variate Gaussian function of the form

$$b_i(\vec{x}) = \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} \exp\left\{-\frac{1}{2}(\vec{x} - \vec{\mu}_i)' \, \Sigma_i^{-1} \, (\vec{x} - \vec{\mu}_i)\right\} \tag{2}$$

with mean vector $\vec{\mu}_i$ and covariance matrix Σ_i. The mixture weights satisfy the constraint that $\sum_{i=1}^{M} p_i = 1$.

\Lambda represents a person (speaker)

GMM Model

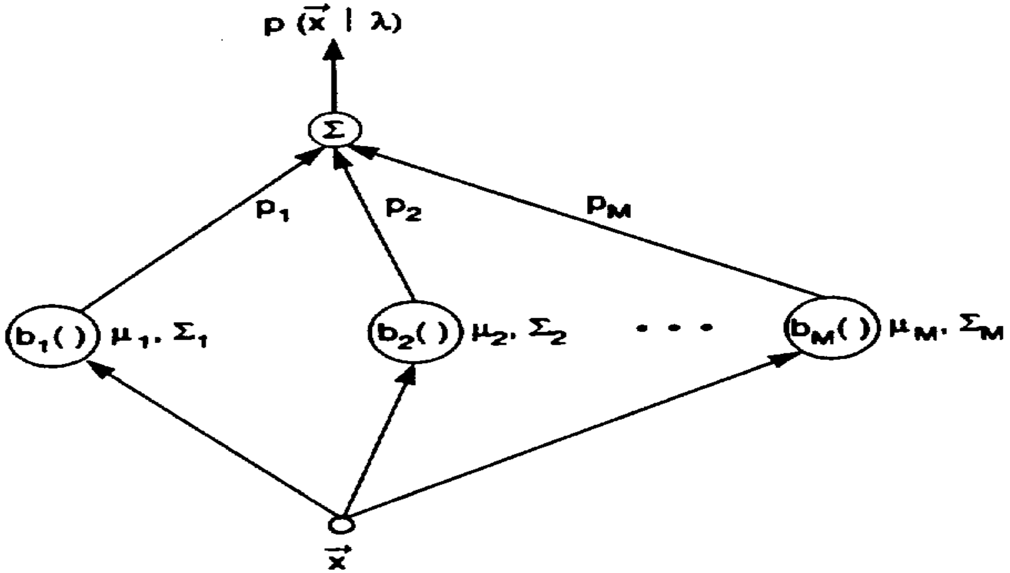
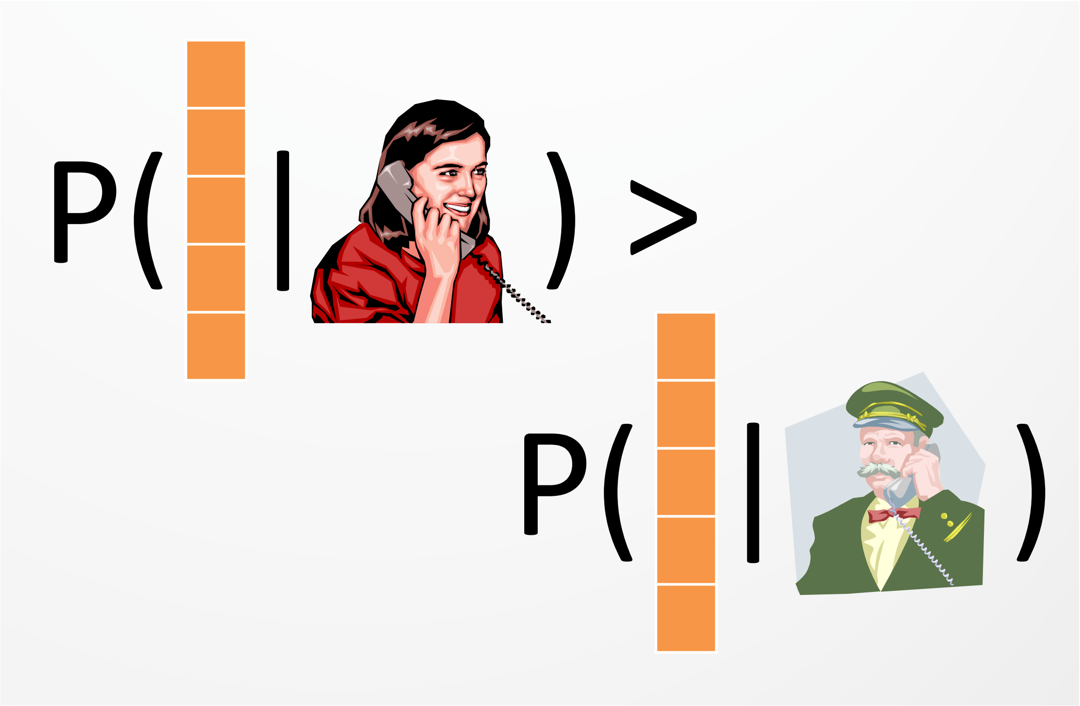

$p(\vec{x} \mid \lambda)$
Σ
p_1
p_2
p_M
$b_1()$ μ_1, Σ_1
$b_2()$ μ_2, Σ_2
• • •
$b_M()$ μ_M, Σ_M
$\vec{x}$

Fig. 2. Depiction of an M component Gaussian mixture density. A Gaussian mixture density is a weighted sum of Gaussian densities, where p_i, $i = 1, \ldots, M$, are the mixture weights and $b_i()$, $i = 1, \ldots, M$, are the component Gaussians.

Mixture weights \sum_i p_i =1

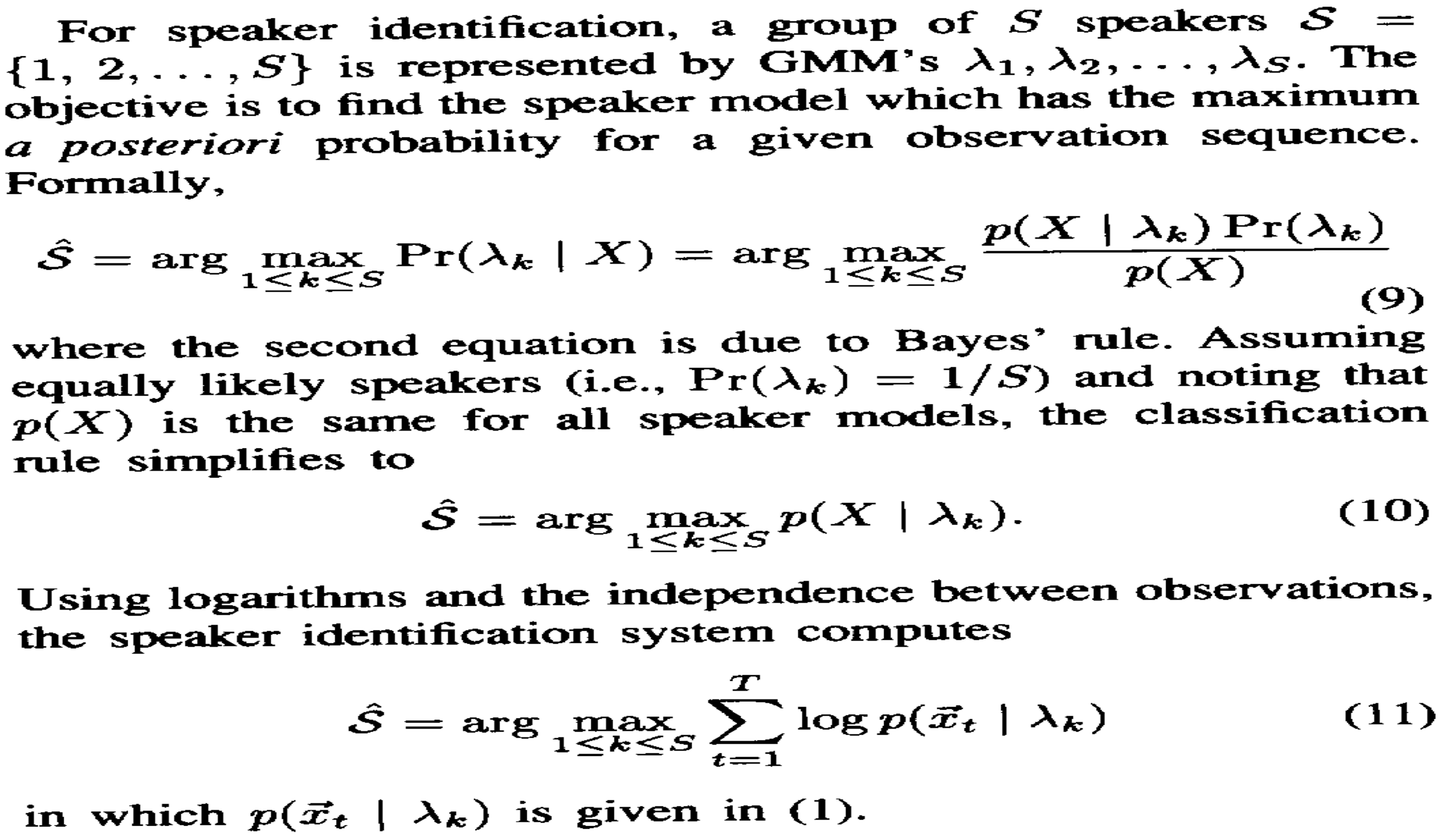
Train the model from observations using the iterative algorithm for each speaker

How do we decide?



She is the person!

Speaker identification problem solution



For speaker identification, a group of S speakers $\mathcal{S} = \{1, 2, \ldots, S\}$ is represented by GMM's $\lambda_1, \lambda_2, \ldots, \lambda_S$. The objective is to find the speaker model which has the maximum *a posteriori* probability for a given observation sequence. Formally,

$$\hat{S} = \arg \max_{1 \leq k \leq S} \Pr(\lambda_k \mid X) = \arg \max_{1 \leq k \leq S} \frac{p(X \mid \lambda_k) \Pr(\lambda_k)}{p(X)} \tag{9}$$

where the second equation is due to Bayes' rule. Assuming equally likely speakers (i.e., $\Pr(\lambda_k) = 1/S$) and noting that $p(X)$ is the same for all speaker models, the classification rule simplifies to

$$\hat{S} = \arg \max_{1 \leq k \leq S} p(X \mid \lambda_k). \tag{10}$$

Using logarithms and the independence between observations, the speaker identification system computes

$$\hat{S} = \arg \max_{1 \leq k \leq S} \sum_{t=1}^{T} \log p(\vec{x}_t \mid \lambda_k) \tag{11}$$

in which $p(\vec{x}_t \mid \lambda_k)$ is given in (1).

Speaker Identification result

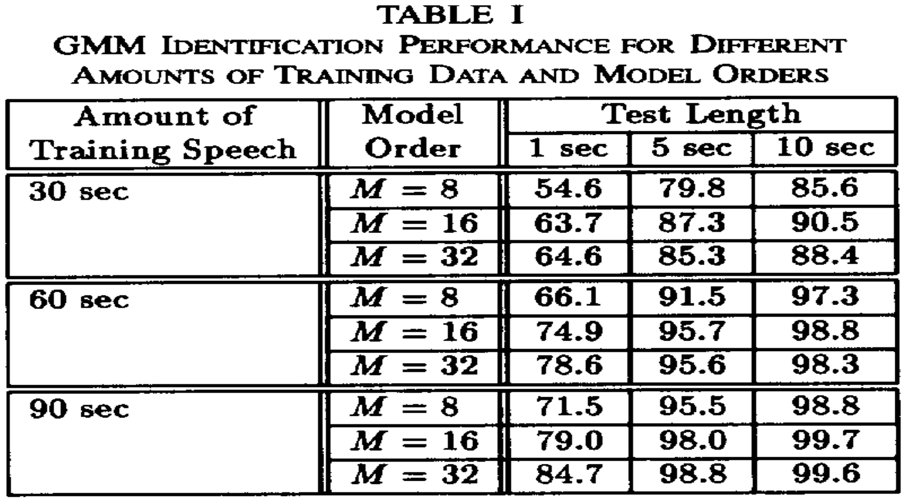
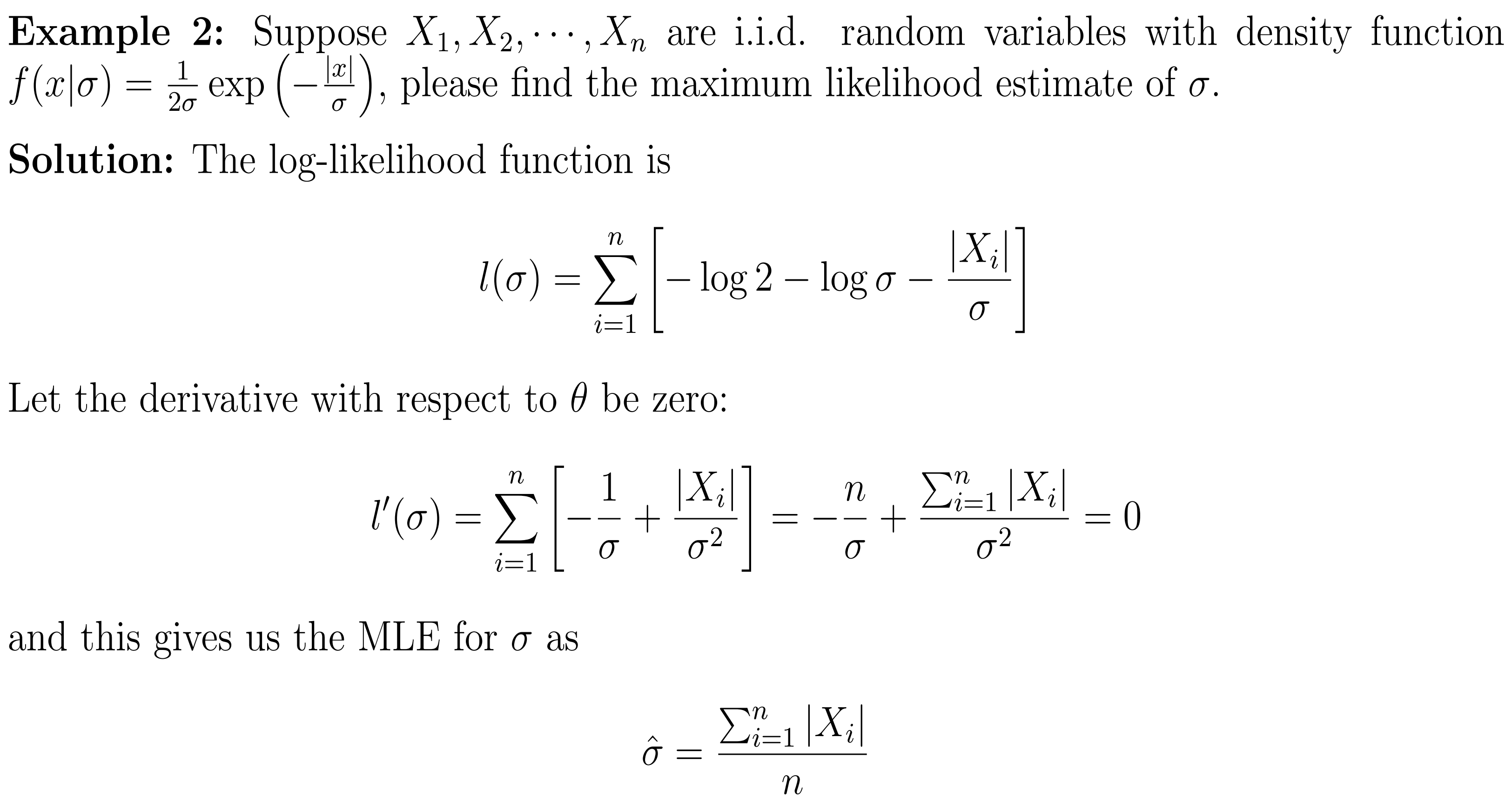


TABLE I
GMM Identification Performance for Different Amounts of Training Data and Model Orders

Amount of Training Speech	Model Order	Test Length		
		1 sec	5 sec	10 sec
30 sec	$M = 8$	54.6	79.8	85.6
	$M = 16$	63.7	87.3	90.5
	$M = 32$	64.6	85.3	88.4
60 sec	$M = 8$	66.1	91.5	97.3
	$M = 16$	74.9	95.7	98.8
	$M = 32$	78.6	95.6	98.3
90 sec	$M = 8$	71.5	95.5	98.8
	$M = 16$	79.0	98.0	99.7
	$M = 32$	84.7	98.8	99.6

16 speakers

Exponential Distribution Example



Example 2: Suppose $X_1, X_2, \cdots, X_n$ are i.i.d. random variables with density function $f(x|\sigma) = \frac{1}{2\sigma}\exp\left(-\frac{|x|}{\sigma}\right)$, please find the maximum likelihood estimate of σ.

Solution: The log-likelihood function is

$$l(\sigma) = \sum_{i=1}^{n}\left[-\log 2 - \log\sigma - \frac{|X_i|}{\sigma}\right]$$

Let the derivative with respect to θ be zero:

$$l'(\sigma) = \sum_{i=1}^{n}\left[-\frac{1}{\sigma} + \frac{|X_i|}{\sigma^2}\right] = -\frac{n}{\sigma} + \frac{\sum_{i=1}^{n}|X_i|}{\sigma^2} = 0$$

and this gives us the MLE for σ as

$$\hat{\sigma} = \frac{\sum_{i=1}^{n}|X_i|}{n}$$

You get a different estimate for the standard deviation

References

- Robust text-independent **speaker identification** using Gaussian mixture **speaker** models, DA Reynolds, RC **Rose** - Speech and Audio Processing, IEEE ..., 1995 - ieeexplore.ieee.org Gaussian mixture models (GMM) for robust text-independent speaker identification. The individual Gaussian components of a GMM are shown to represent some general speaker-dependent spectral shapes that are effective for modeling speaker identity. The focus of ...Cited by 2855 Related articles All 11 versions Web of Science: 976 Cite Save

- **Speaker** verification using adapted Gaussian mixture models, DA Reynolds, TF Quatieri, RB Dunn - Digital signal processing, 2000 – Elsevier DA Reynolds

  PDF] Pattern Recognition

- CM **Bishop** - Machine Learning, 2006 - academia.edu, PDF is available!
- https://people.eecs.berkeley.edu/~jordan/courses/294-fall09/lectures/clustering/slides.pdf