

# *E9 205 Machine Learning for Signal Processing*

---

**MLE for Gaussian Model**

06-09-2017

---

Instructor - Sriram Ganapathy ([sriram@ee.iisc.ernet.in](mailto:sriram@ee.iisc.ernet.in))

Teaching Assistant - Aravind Illa ([aravindece77@gmail.com](mailto:aravindece77@gmail.com)).



# Finding the parameters of the Model

- The Gaussian model has the following parameters

$$\theta = (\mu, \Sigma)$$

- **Total number of parameters** to be learned for D dimensional data is  $D^2 + D$
- Given N data points  $\{\mathbf{x}_i\}_{i=1}^N$  how do we estimate the **parameters of model**.
  - Several criteria can be used
  - The most popular method is the **maximum likelihood estimation (MLE)**.

# MLE

Define the likelihood function as  $L(\boldsymbol{\theta}) = \prod_{i=1}^N p(\mathbf{x}_i | \boldsymbol{\theta})$

The **maximum likelihood estimator (MLE)** is

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{arg\,max}} L(\boldsymbol{\theta})$$

The MLE satisfies **nice properties** like

- Consistency (convergence to true value)
- Efficiency (has the least Mean squared error).



# MLE

For the Gaussian distribution

$$p(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^D |\boldsymbol{\Sigma}|}} \exp\left\{-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^* \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right\}$$

$$L(\boldsymbol{\theta}) = \prod_{i=1}^N p(\mathbf{x}_i|\boldsymbol{\theta})$$

$$\log L(\boldsymbol{\theta}) = -\frac{ND}{2} - \frac{N}{2} \log |\boldsymbol{\Sigma}| - \frac{1}{2} \sum_{i=1}^N \left( (\mathbf{x}_i - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}) \right)$$

To estimate the parameters  $\frac{\partial \log L}{\partial \boldsymbol{\mu}} = 0$

# MLE

Using matrix differentiation rules, for a symmetric matrix  $\mathbf{A}$

$$\frac{\partial \mathbf{x}^T \mathbf{A} \mathbf{x}}{\partial \mathbf{x}} = 2\mathbf{A} \mathbf{x} \quad \mu^* = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i$$

Using matrix differentiation rules for log determinant and trace

$$\frac{\partial \log(|\mathbf{A}|)}{\partial \mathbf{A}} = 2\mathbf{A}^{-1} - \text{diag}(\mathbf{A}^{-1})$$

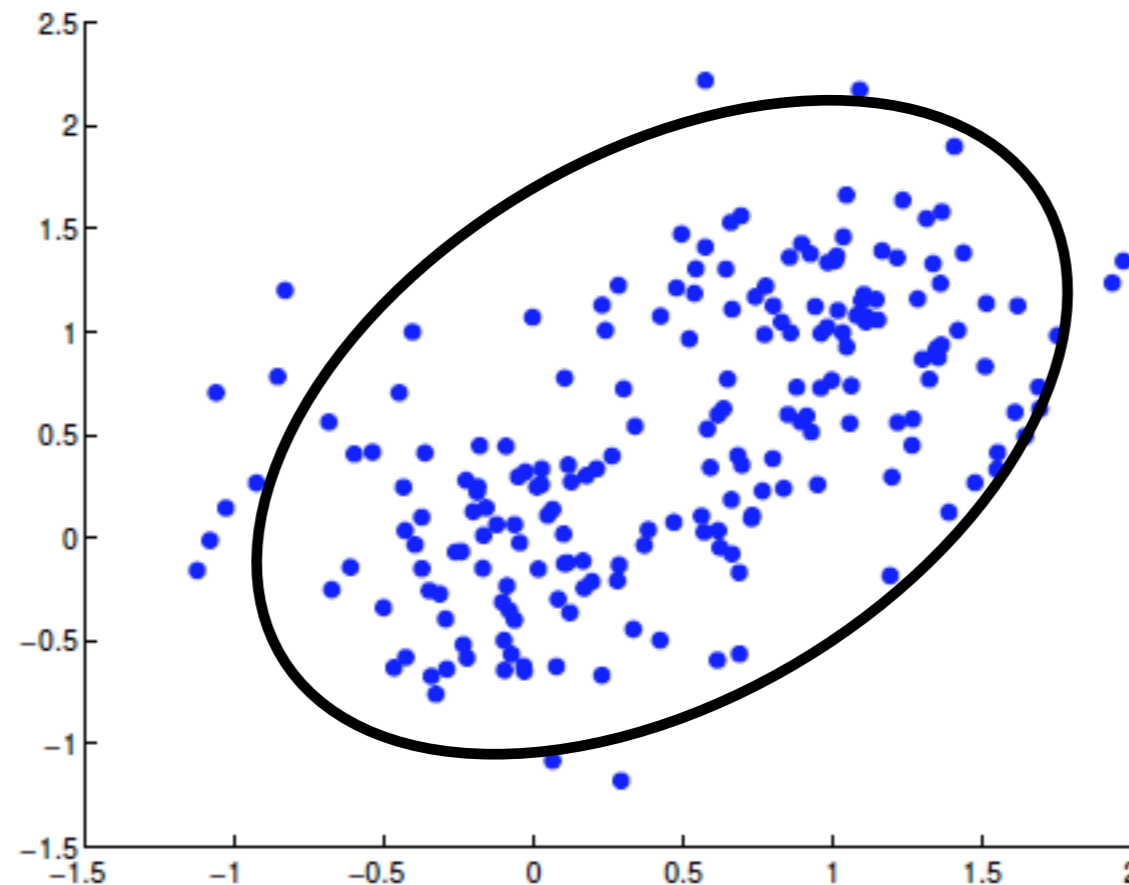
$$\frac{\partial \text{tr}(\mathbf{A} \mathbf{B})}{\partial \mathbf{A}} = \mathbf{B} + \mathbf{B}^T - \text{diag}(\mathbf{B})$$

$$\Sigma^* = \frac{1}{N} \sum_{i=1}^N (\mathbf{x}_i - \mu^*) (\mathbf{x}_i - \mu^*)^T$$

Sample mean and Sample Covariance

# Gaussian Distribution

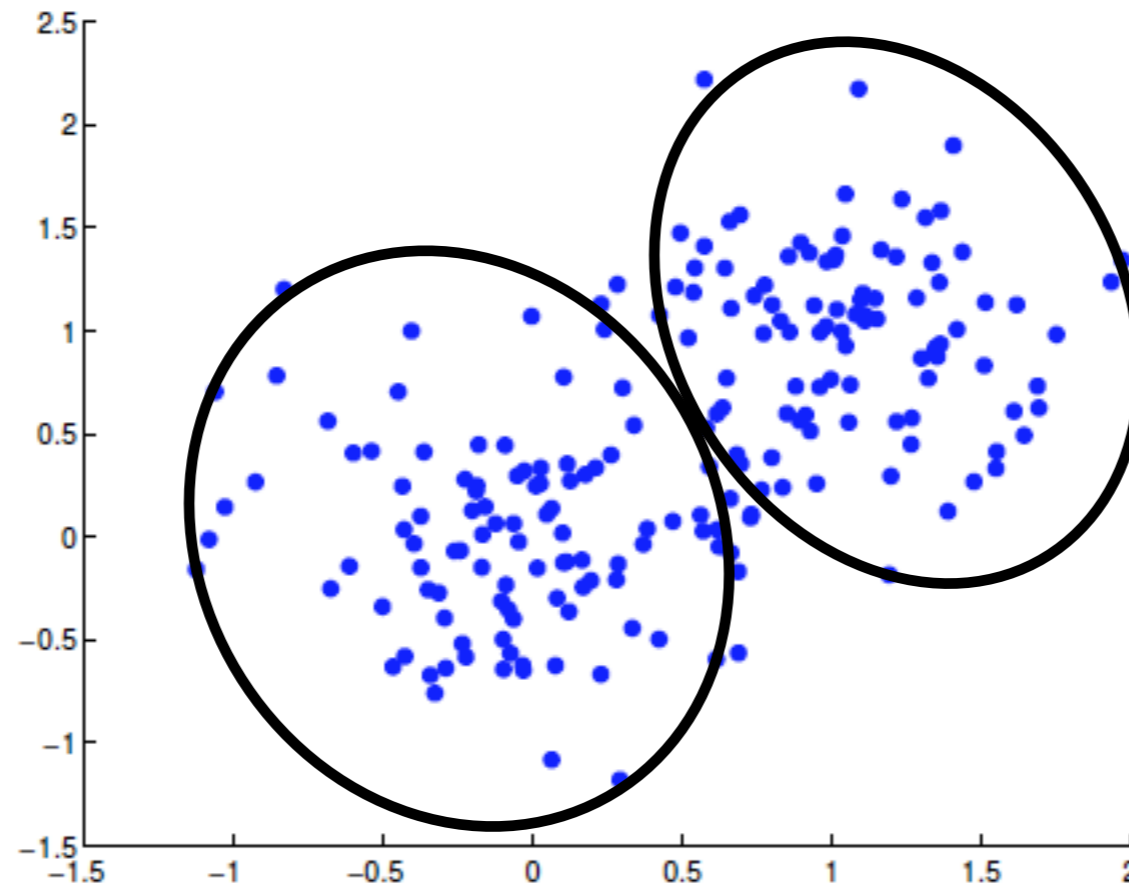
Often the data lies in clusters (2-D example)



Fitting a single Gaussian model may be **too broad**.

# Gaussian Distribution

Need mixture models

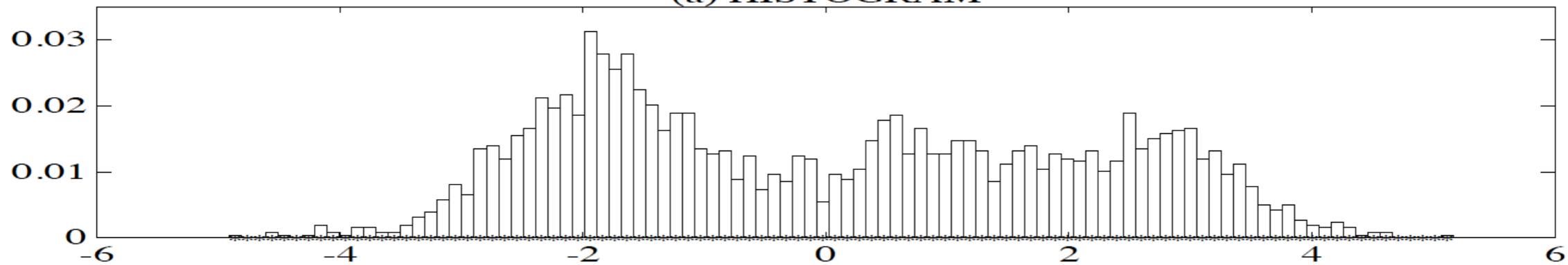


Can fit any arbitrary distribution.

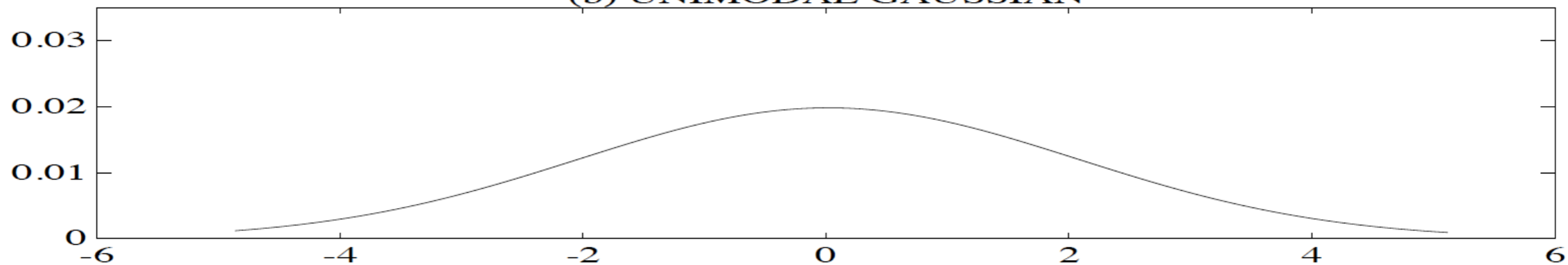
# Gaussian Distribution

## 1-D example

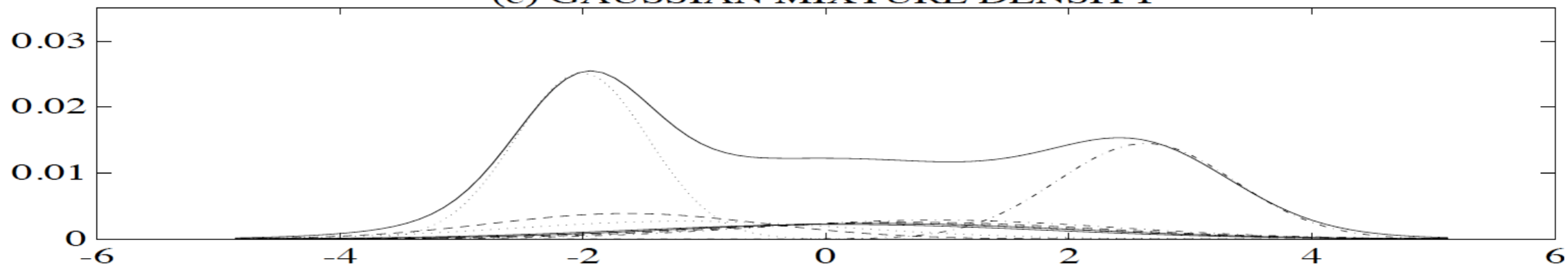
(a) HISTOGRAM



(b) UNIMODAL GAUSSIAN



(c) GAUSSIAN MIXTURE DENSITY





# Gaussian Distribution Summary

- The Gaussian model - parametric distributions
- **Simple and useful** properties.
- Can model unimodal (single peak distributions)
- **MLE** gives intuitive results
- Issues with Gaussian model
  - Multi-modal data
  - Not useful for complex data distributions
- Need for **mixture models**

# Gaussian Mixture Models

A Gaussian Mixture Model (GMM) is defined as

$$p(\mathbf{x}|\Theta) = \sum_{k=1}^K \alpha_k p(\mathbf{x}|\theta_k)$$

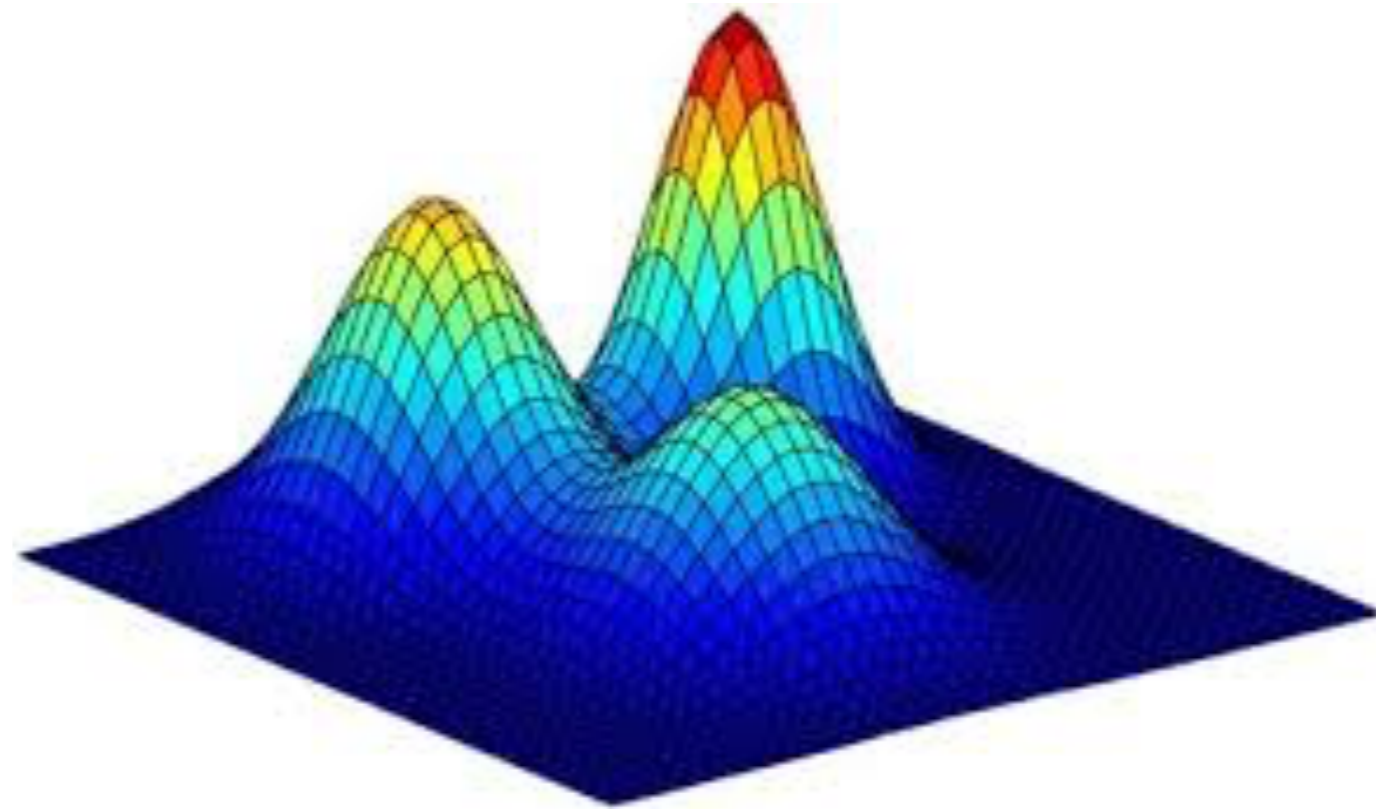
$$p(\mathbf{x}|\theta_k) = \frac{1}{\sqrt{(2\pi)^D |\Sigma_k|}} \exp\left\{ -\frac{1}{2}(\mathbf{x} - \mu_k)^* \Sigma_k^{-1} (\mathbf{x} - \mu_k) \right\}$$

The weighting coefficients have the property

$$\sum_{k=1}^K \alpha_k = 1$$

# Gaussian Mixture Models

- Properties of GMM
  - Can model multi-modal data.
  - Identify data clusters.
  - Can model arbitrarily complex data distributions



The set of parameters for the model are

$$\Theta_k = \{\alpha_k, \theta_k\}_{k=1}^K \quad \theta_k = \{\mu_k, \Sigma_k\}$$

The number of parameters is  $KD^2 + KD + K$

# MLE for GMM

- The log-likelihood function over the entire data in this case will have a **logarithm of a summation**

$$\log L(\Theta) = \sum_{i=1}^N \log \left( \sum_{k=1}^K \alpha_k p(\mathbf{x}_i | \theta_k) \right)$$

- Solving for the optimal parameters using MLE for GMM is not straight forward.
- Resort to the **Expectation Maximization (EM)** algorithm