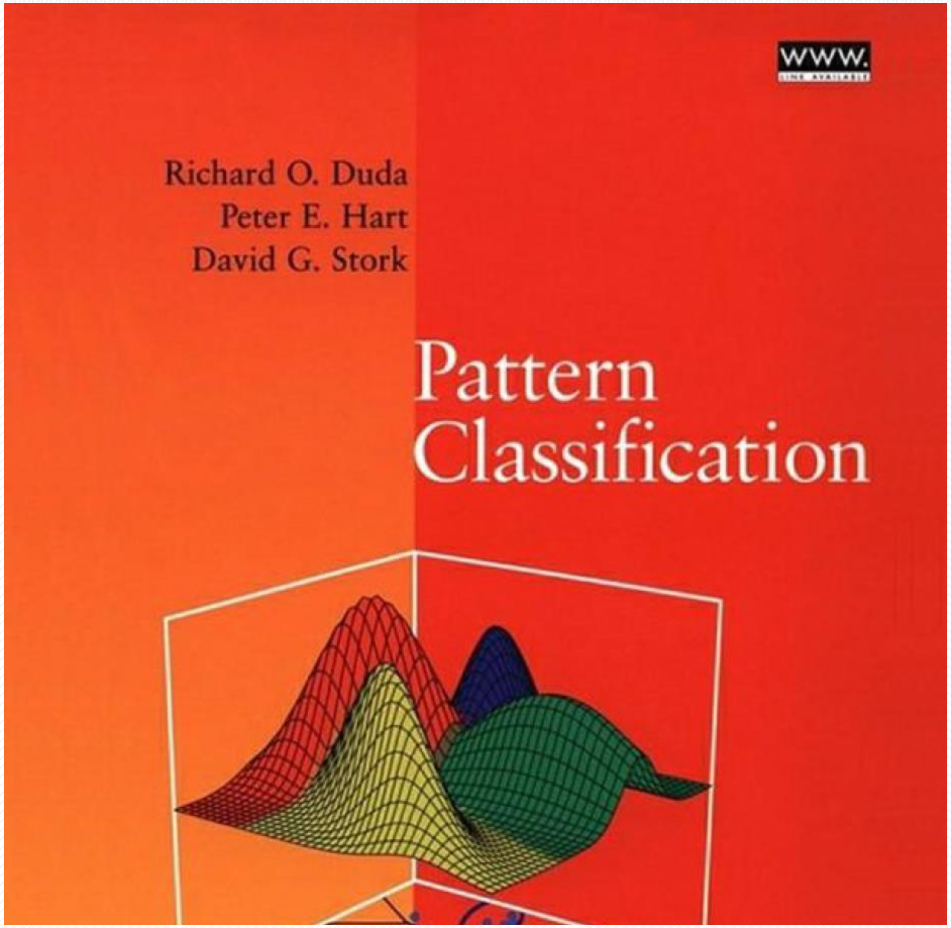
Classificação de Padrões - Pattern Classification

Apresentação com base no Material do MIT e no livro Pattern Classification, R. Duda, P. Hart, D. Stork

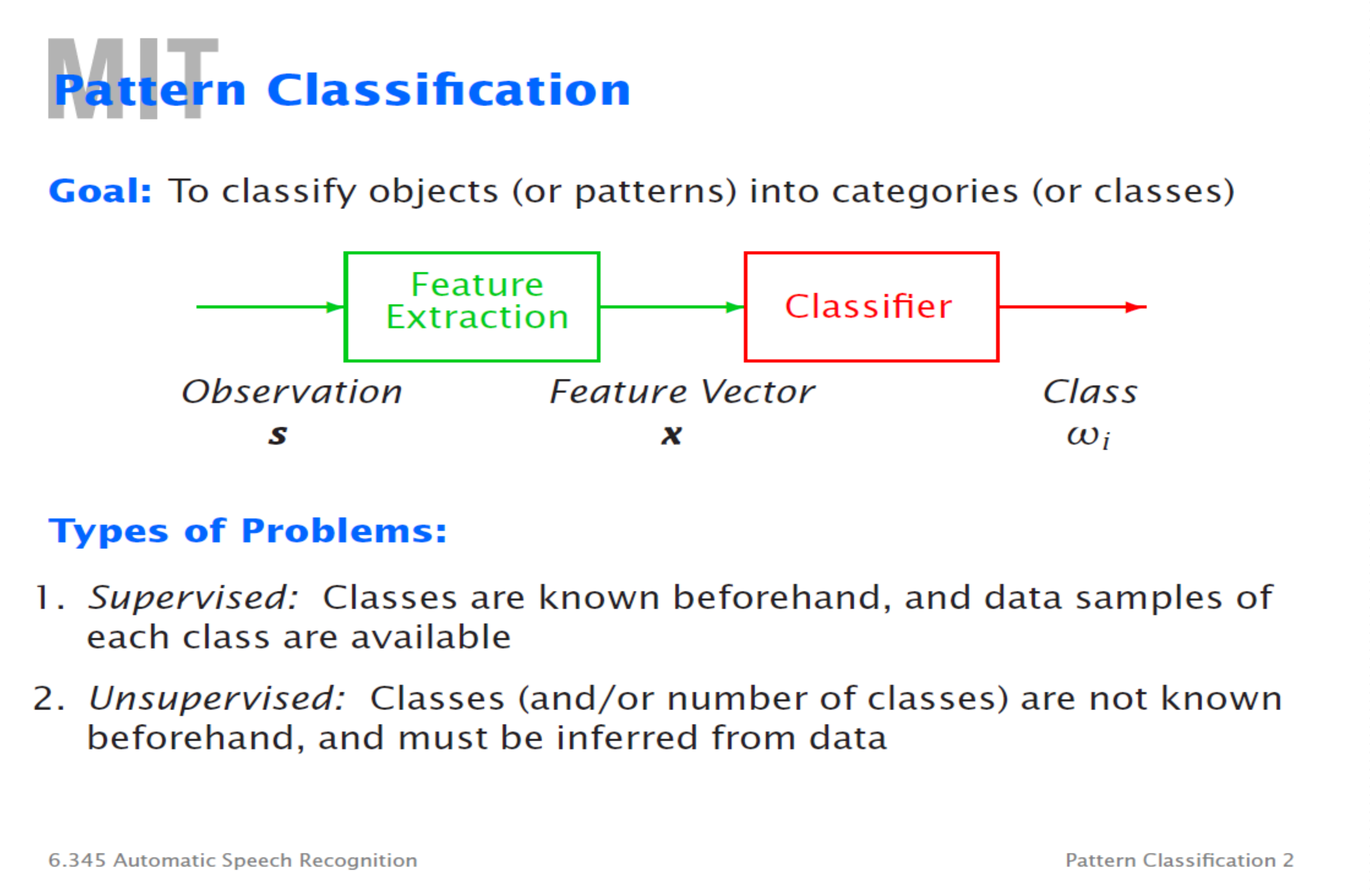

WWW.
Richard O. Duda
Peter E. Hart
David G. Stork
Pattern Classification

Pattern Classification

- **Pattern Classification**

- Introduction
- Parametric classifiers
- Semi-parametric classifiers

Pattern Classification



Pattern Classification

Goal: To classify objects (or patterns) into categories (or classes)

Feature Extraction

Classifier

Observation $\boldsymbol{s}$

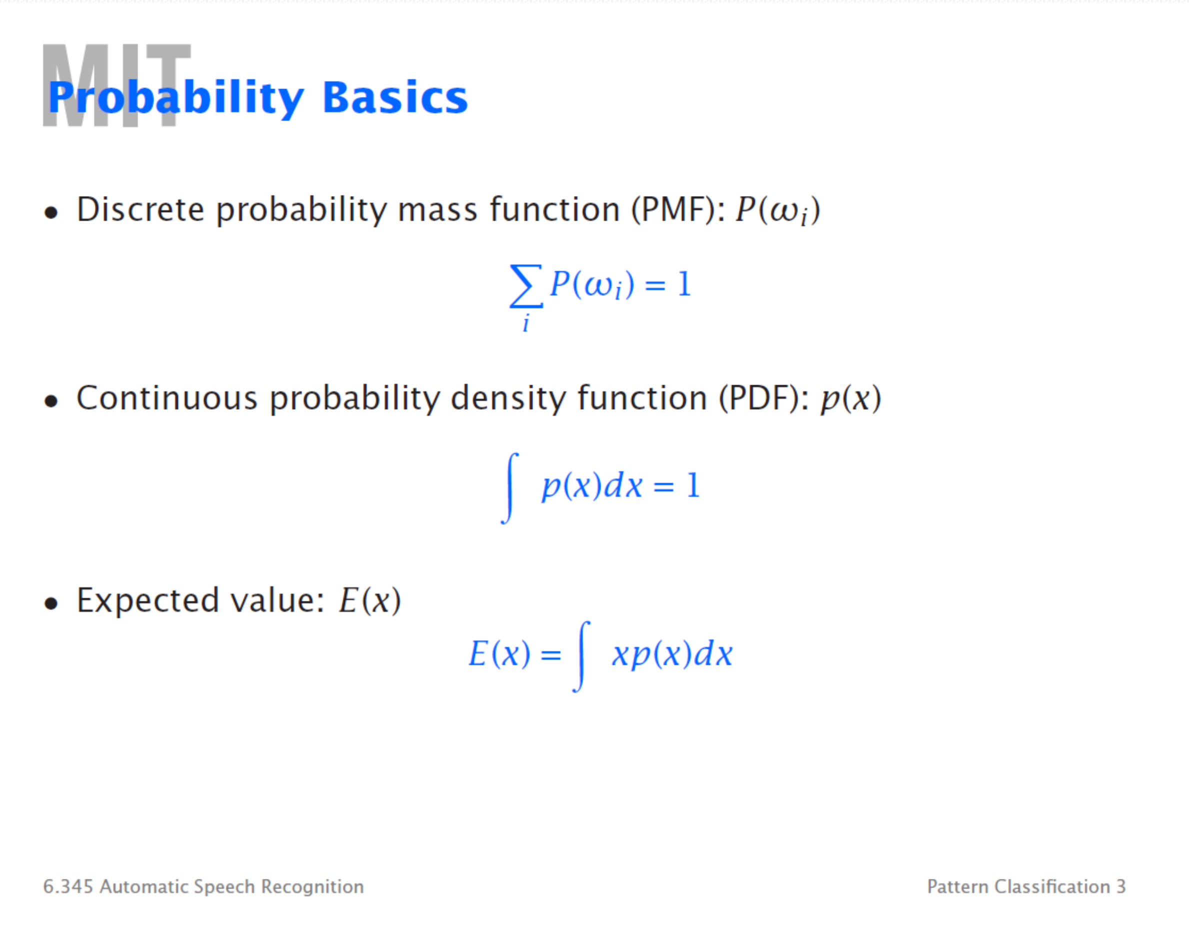
Feature Vector $\boldsymbol{x}$

Class ω_i

Types of Problems:

1. *Supervised:* Classes are known beforehand, and data samples of each class are available
2. *Unsupervised:* Classes (and/or number of classes) are not known beforehand, and must be inferred from data

Pattern Classification



Probability Basics

- Discrete probability mass function (PMF): $P(\omega_i)$

$$\sum_i P(\omega_i) = 1$$

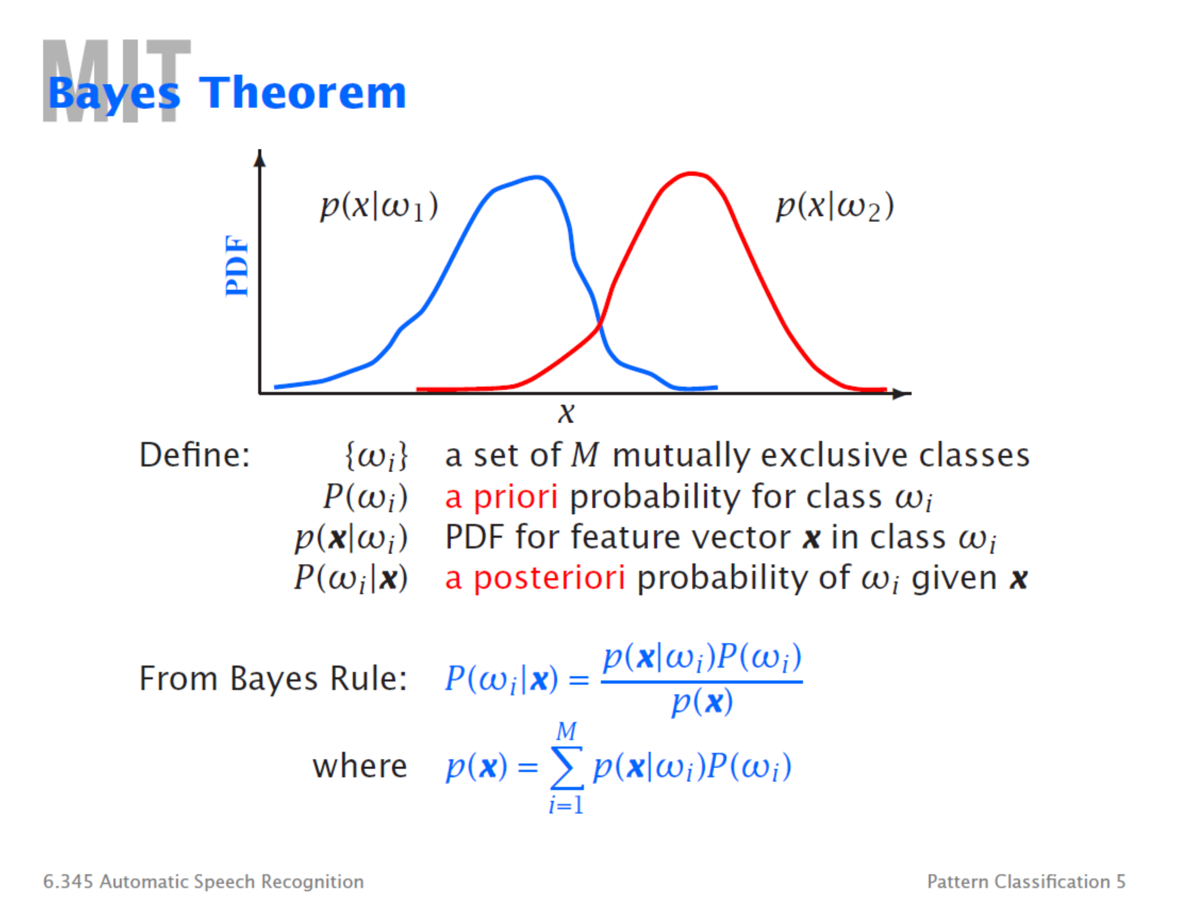
- Continuous probability density function (PDF): $p(x)$

$$\int p(x)dx = 1$$

- Expected value: $E(x)$

$$E(x) = \int xp(x)dx$$

Pattern Classification



Bayes Theorem

PDF
$p(x|\omega_1)$
$p(x|\omega_2)$
x

Define: $\{\omega_i\}$ a set of M mutually exclusive classes

$P(\omega_i)$ a priori probability for class ω_i

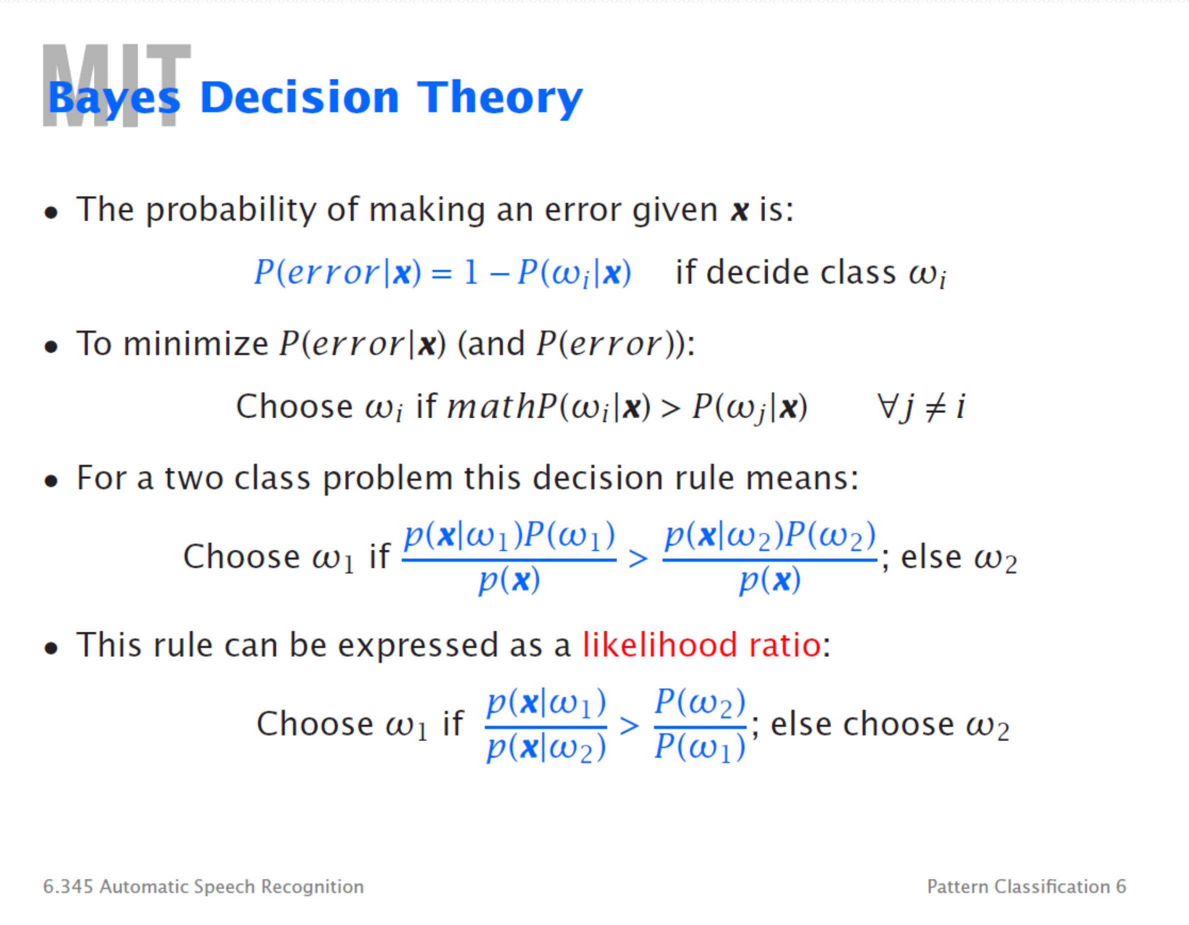
$p(\mathbf{x}|\omega_i)$ PDF for feature vector $\mathbf{x}$ in class ω_i

$P(\omega_i|\mathbf{x})$ a posteriori probability of ω_i given $\mathbf{x}$

From Bayes Rule: $$P(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)P(\omega_i)}{p(\mathbf{x})}$$

where $$p(\mathbf{x}) = \sum_{i=1}^{M} p(\mathbf{x}|\omega_i)P(\omega_i)$$

Pattern Classification



Bayes Decision Theory

- The probability of making an error given $\boldsymbol{x}$ is:

$$P(error|\mathbf{x}) = 1 - P(\omega_i|\mathbf{x}) \qquad \text{if decide class } \omega_i$$

- To minimize $P(error|\mathbf{x})$ (and $P(error)$):

$$\text{Choose } \omega_i \text{ if } mathP(\omega_i|\mathbf{x}) > P(\omega_j|\mathbf{x}) \qquad \forall j \neq i$$

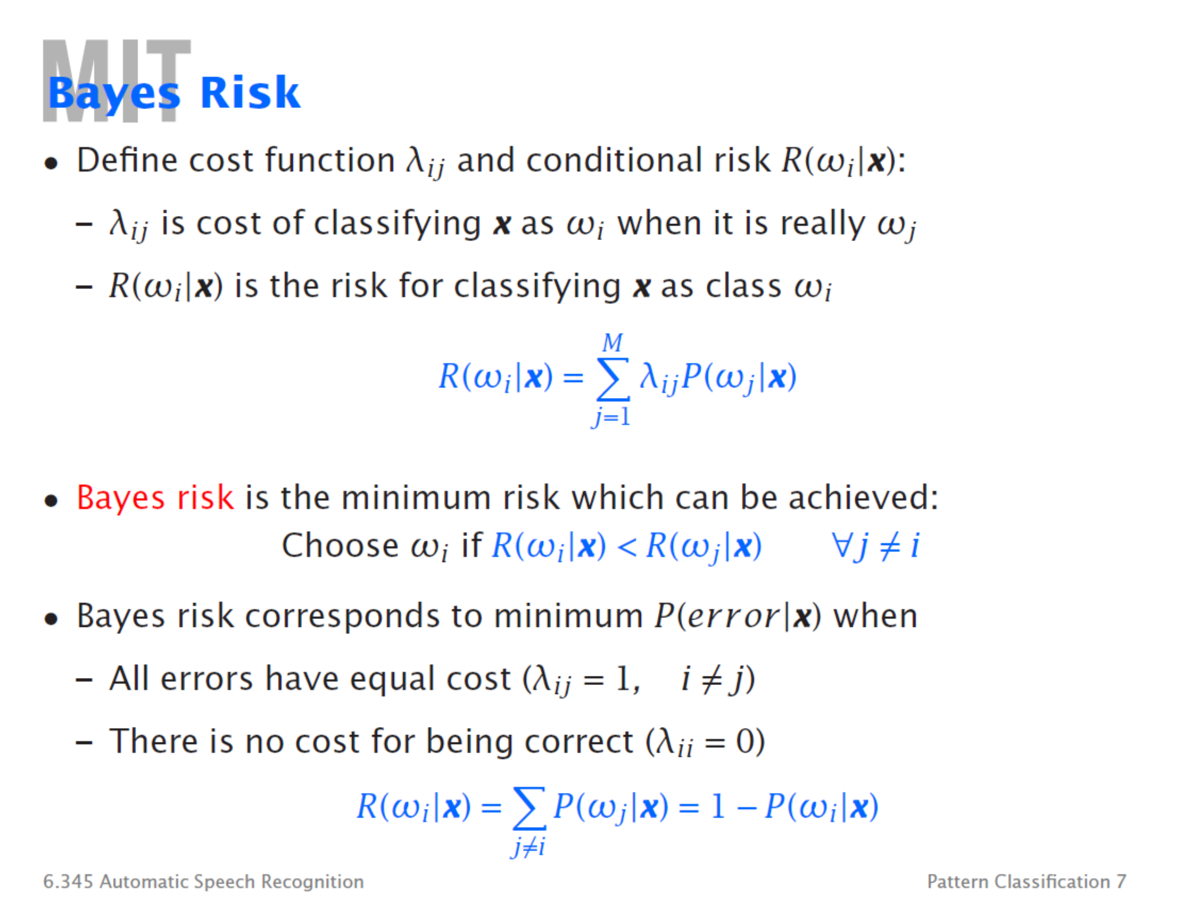
- For a two class problem this decision rule means:

$$\text{Choose } \omega_1 \text{ if } \frac{p(\mathbf{x}|\omega_1)P(\omega_1)}{p(\mathbf{x})} > \frac{p(\mathbf{x}|\omega_2)P(\omega_2)}{p(\mathbf{x})}; \text{ else } \omega_2$$

- This rule can be expressed as a likelihood ratio:

$$\text{Choose } \omega_1 \text{ if } \frac{p(\mathbf{x}|\omega_1)}{p(\mathbf{x}|\omega_2)} > \frac{P(\omega_2)}{P(\omega_1)}; \text{ else choose } \omega_2$$

Pattern Classification



Bayes Risk

- Define cost function λ_{ij} and conditional risk $R(\omega_i|\boldsymbol{x})$:
  - λ_{ij} is cost of classifying $\boldsymbol{x}$ as ω_i when it is really ω_j
  - $R(\omega_i|\boldsymbol{x})$ is the risk for classifying $\boldsymbol{x}$ as class ω_i

$$R(\omega_i|\boldsymbol{x}) = \sum_{j=1}^{M} \lambda_{ij} P(\omega_j|\boldsymbol{x})$$

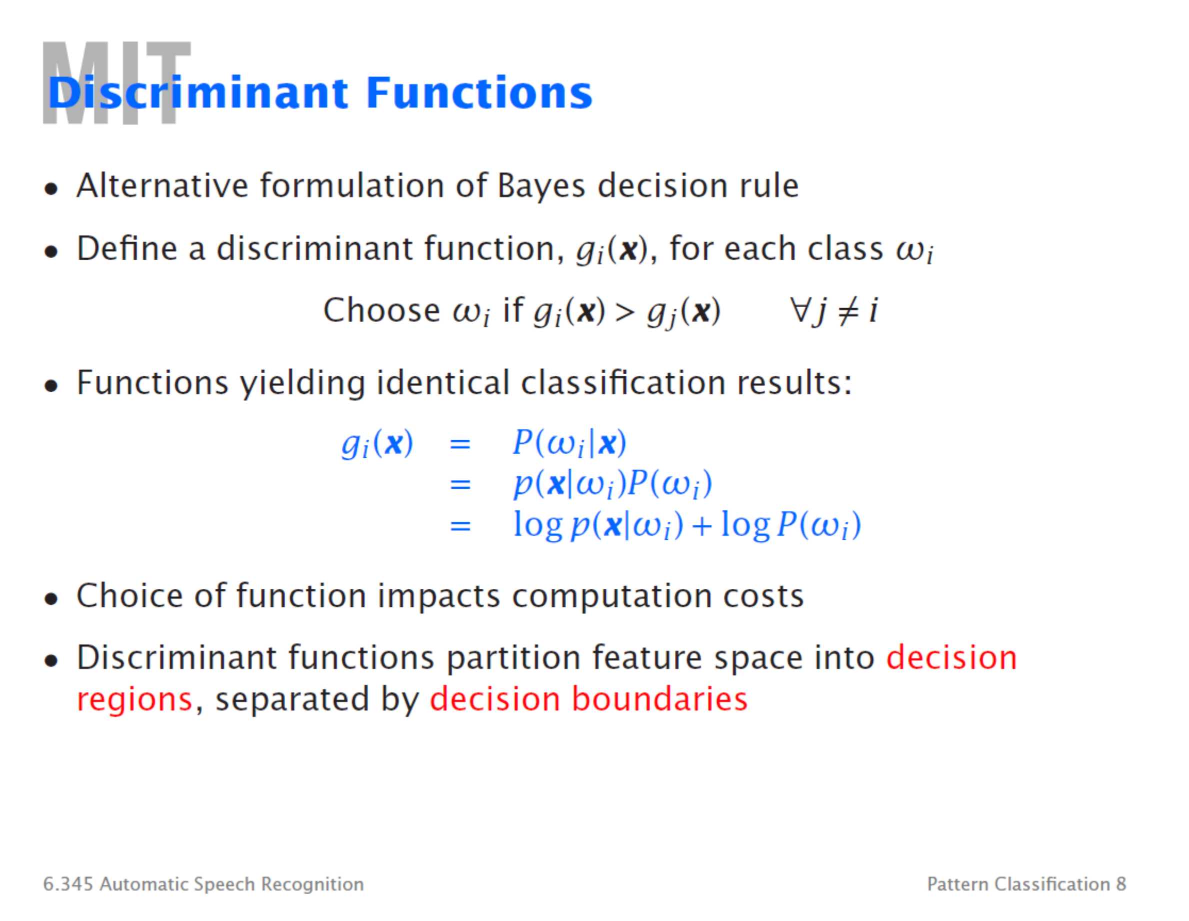
- Bayes risk is the minimum risk which can be achieved:

  Choose ω_i if $R(\omega_i|\boldsymbol{x}) < R(\omega_j|\boldsymbol{x}) \qquad \forall j \neq i$

- Bayes risk corresponds to minimum $P(error|\boldsymbol{x})$ when
  - All errors have equal cost ($\lambda_{ij} = 1, \quad i \neq j$)
  - There is no cost for being correct ($\lambda_{ii} = 0$)

$$R(\omega_i|\boldsymbol{x}) = \sum_{j \neq i} P(\omega_j|\boldsymbol{x}) = 1 - P(\omega_i|\boldsymbol{x})$$

Pattern Classification



Discriminant Functions

- Alternative formulation of Bayes decision rule
- Define a discriminant function, $g_i(\boldsymbol{x})$, for each class ω_i

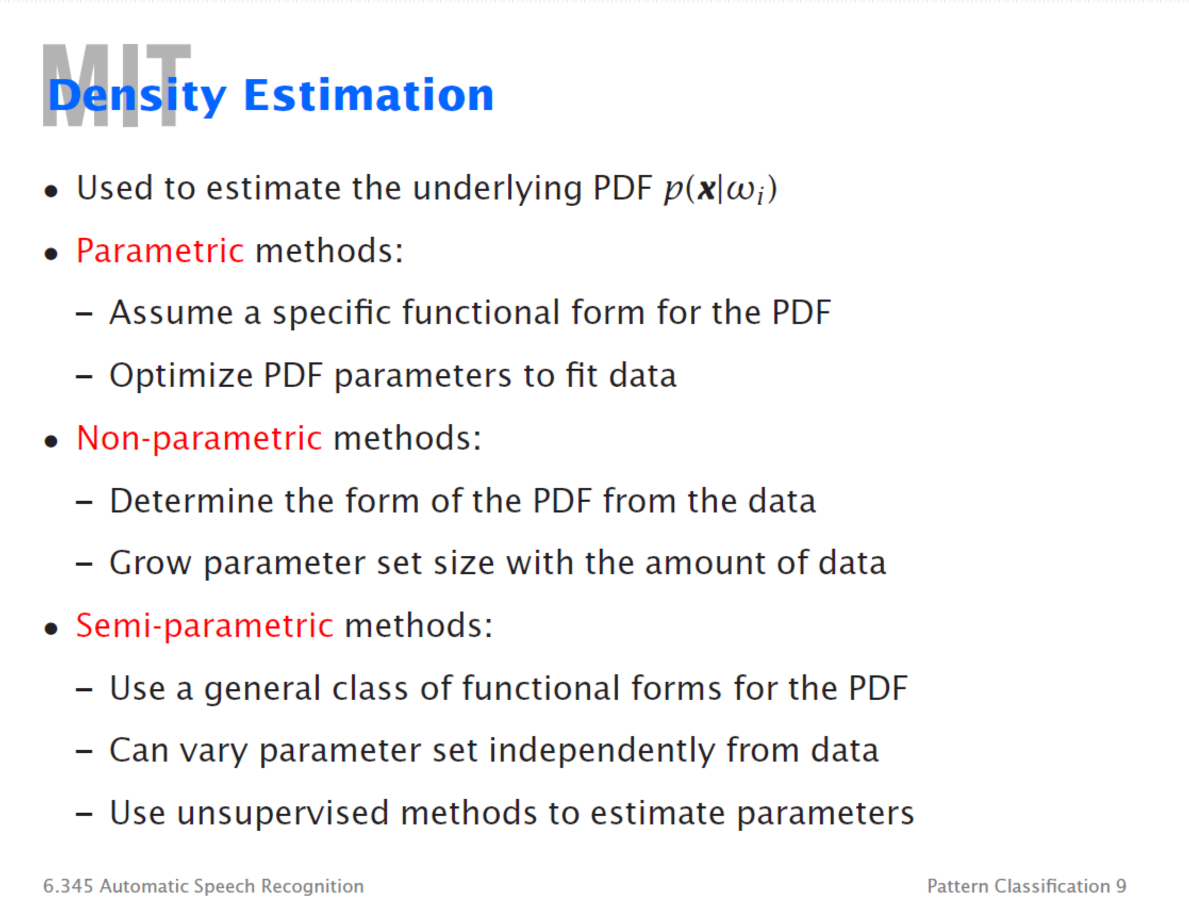
$$\text{Choose } \omega_i \text{ if } g_i(\boldsymbol{x}) > g_j(\boldsymbol{x}) \qquad \forall j \neq i$$

- Functions yielding identical classification results:

$$\begin{aligned} g_i(\boldsymbol{x}) &= P(\omega_i|\boldsymbol{x}) \\ &= p(\boldsymbol{x}|\omega_i)P(\omega_i) \\ &= \log p(\boldsymbol{x}|\omega_i) + \log P(\omega_i) \end{aligned}$$

- Choice of function impacts computation costs
- Discriminant functions partition feature space into decision regions, separated by decision boundaries

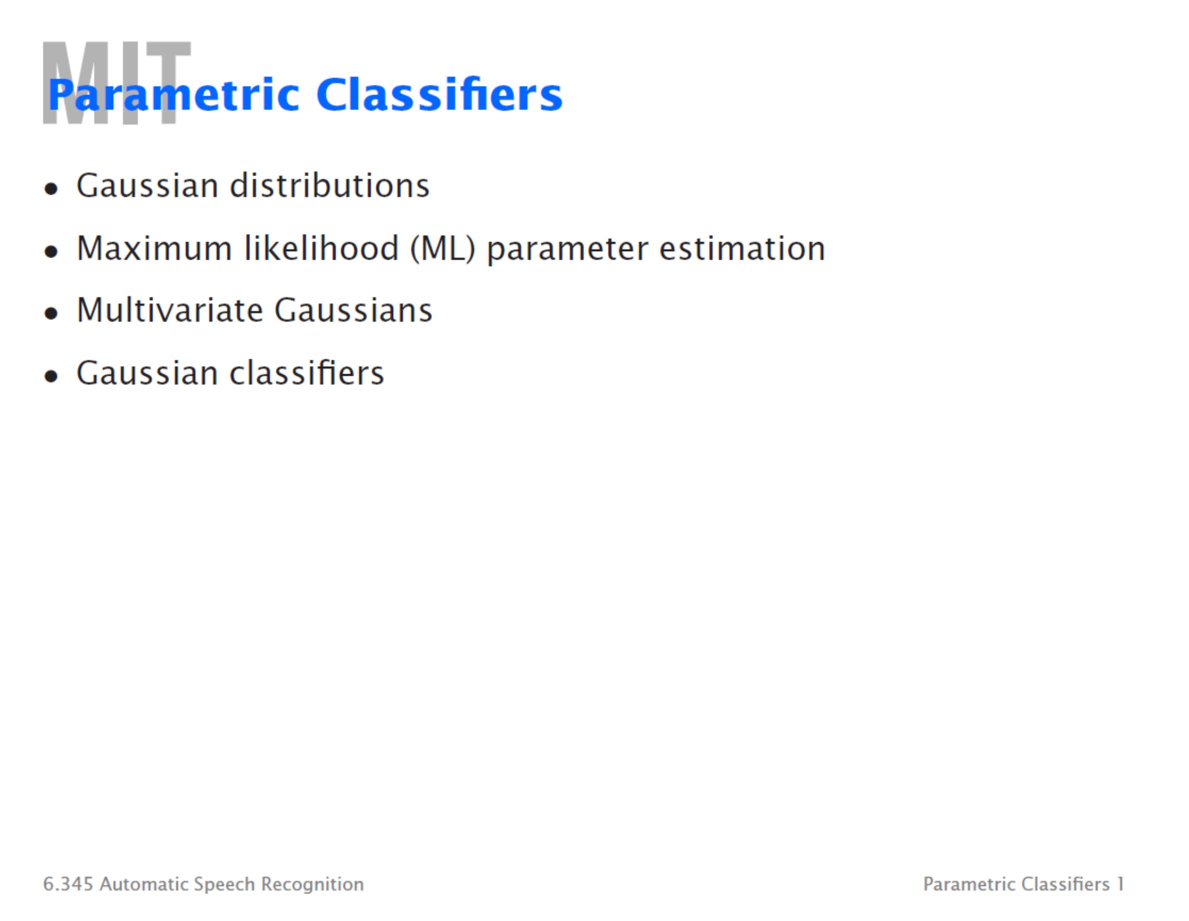
Pattern Classification



Density Estimation

- Used to estimate the underlying PDF $p(\boldsymbol{x}|\omega_i)$
- Parametric methods:
  - Assume a specific functional form for the PDF
  - Optimize PDF parameters to fit data
- Non-parametric methods:
  - Determine the form of the PDF from the data
  - Grow parameter set size with the amount of data
- Semi-parametric methods:
  - Use a general class of functional forms for the PDF
  - Can vary parameter set independently from data
  - Use unsupervised methods to estimate parameters

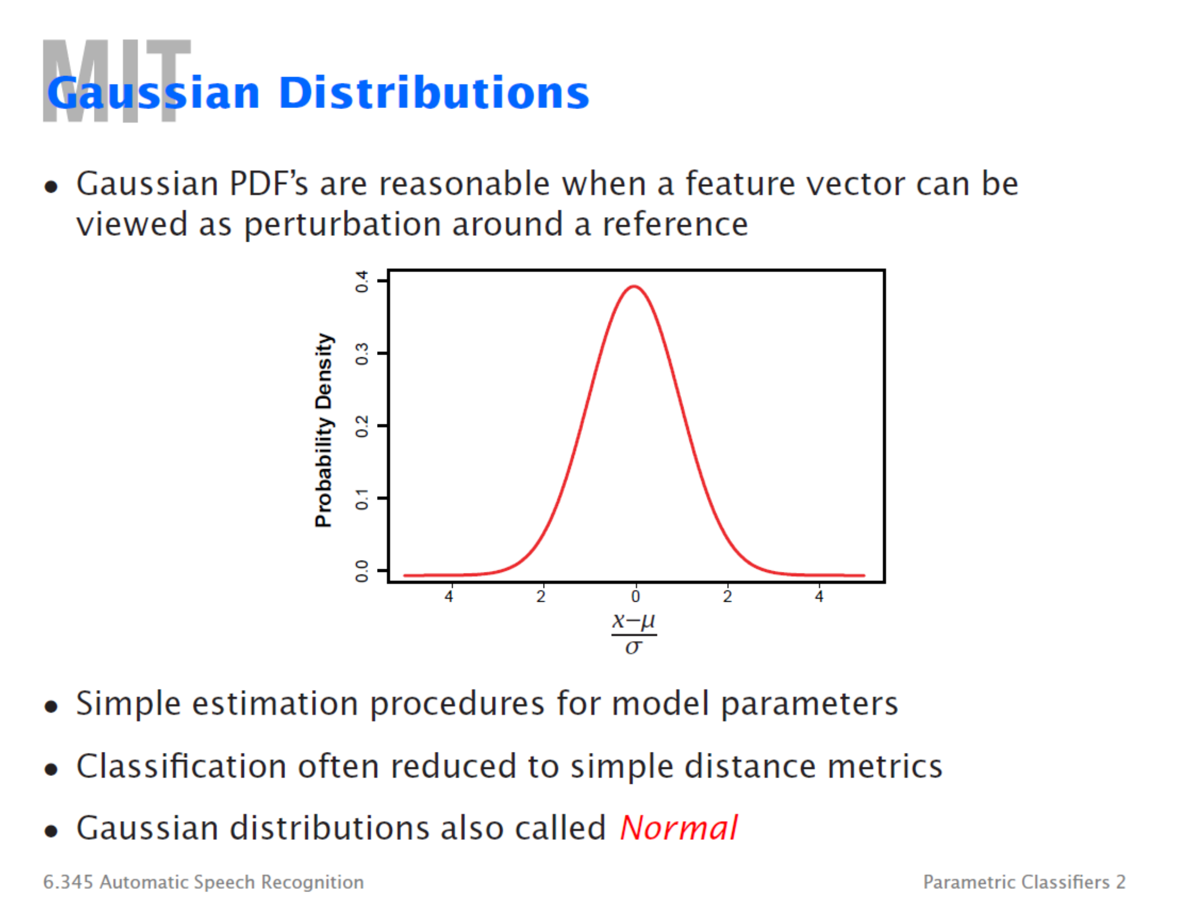
Pattern Classification



Parametric Classifiers

- Gaussian distributions
- Maximum likelihood (ML) parameter estimation
- Multivariate Gaussians
- Gaussian classifiers

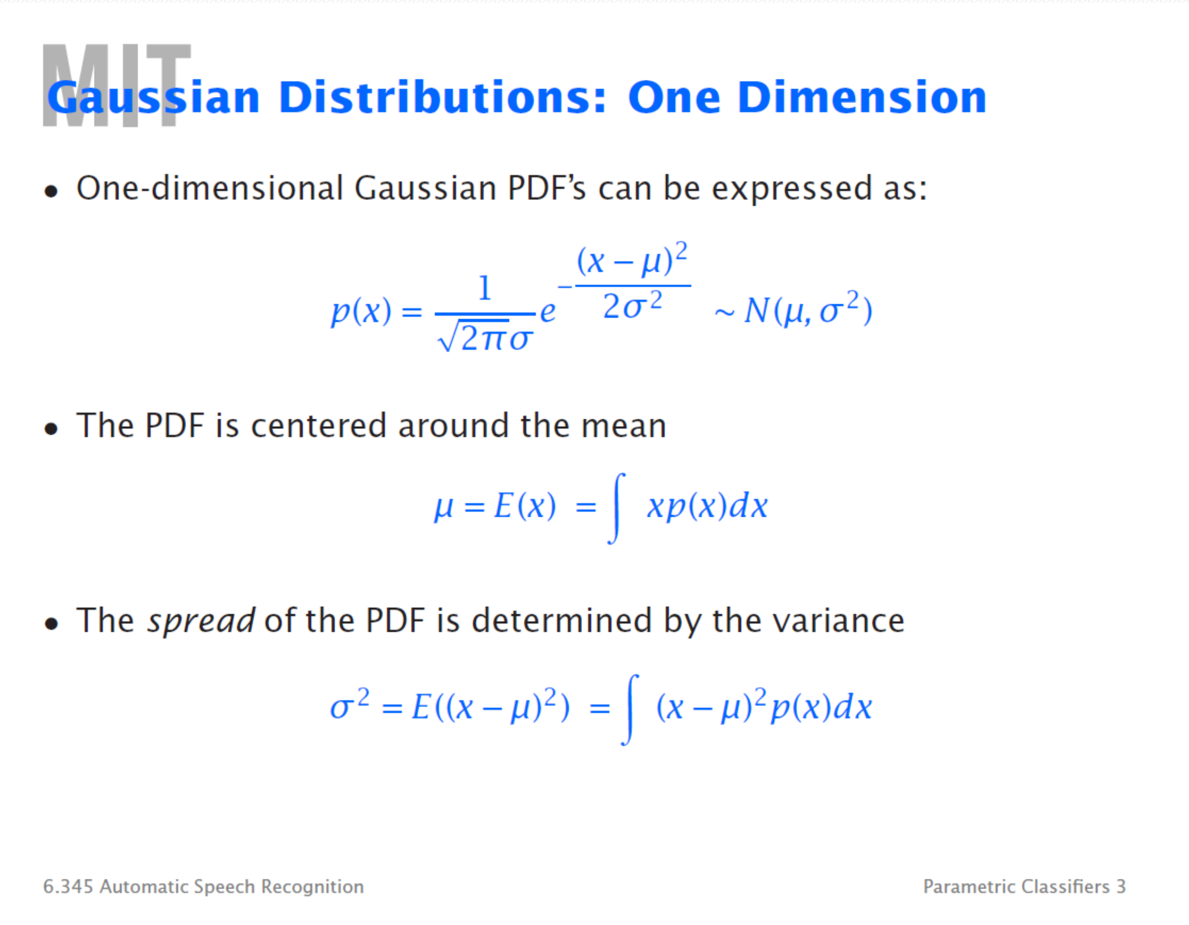
Pattern Classification


MIT
0.0
0.1
0.2
0.3
0.4
Probability Density
4
2
0
2
4
$\frac{x-\mu}{\sigma}$

Gaussian Distributions

- Gaussian PDF’s are reasonable when a feature vector can be viewed as perturbation around a reference
- Simple estimation procedures for model parameters
- Classification often reduced to simple distance metrics
- Gaussian distributions also called *Normal*

Pattern Classification



Gaussian Distributions: One Dimension

- One-dimensional Gaussian PDF's can be expressed as:

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \sim N(\mu, \sigma^2)$$

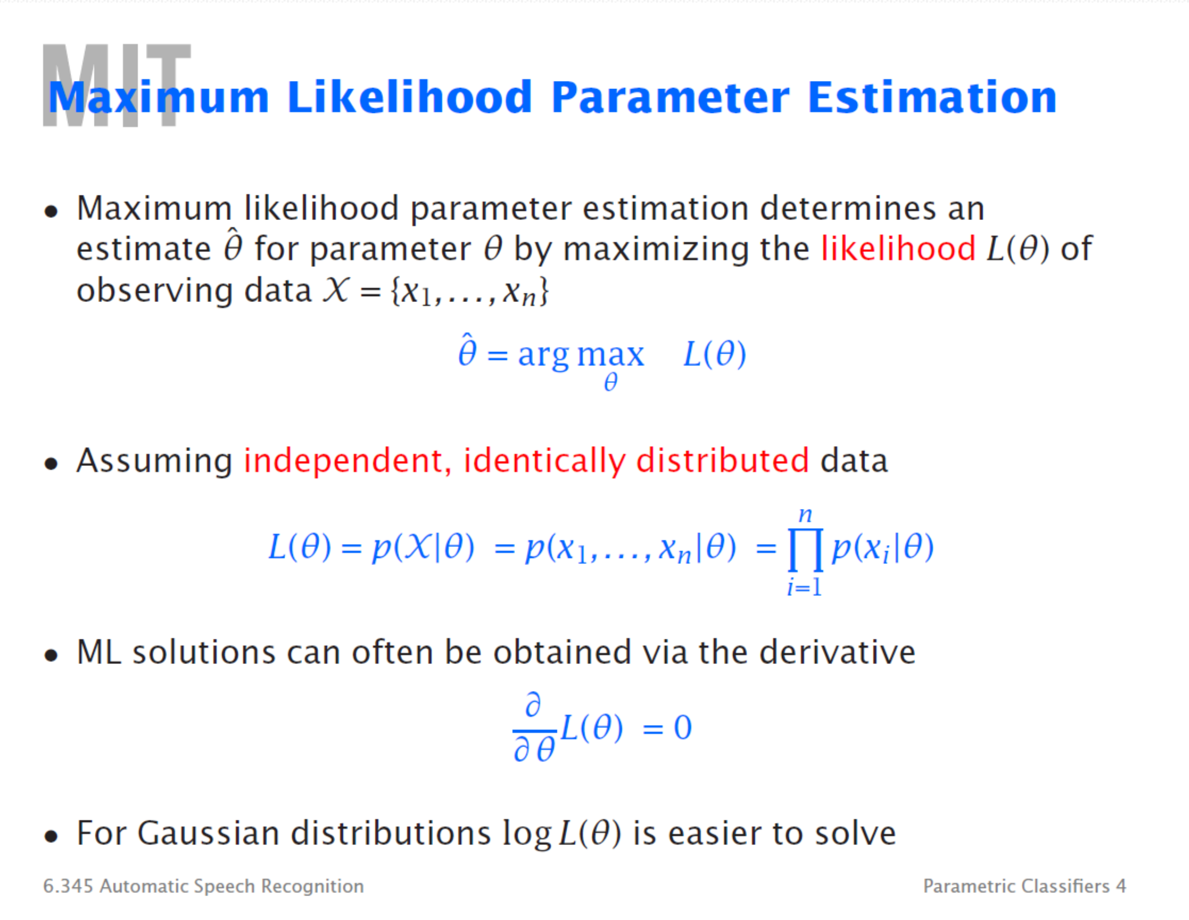
- The PDF is centered around the mean

$$\mu = E(x) = \int xp(x)dx$$

- The *spread* of the PDF is determined by the variance

$$\sigma^2 = E((x-\mu)^2) = \int (x-\mu)^2 p(x)dx$$

Pattern Classification



Maximum Likelihood Parameter Estimation

- Maximum likelihood parameter estimation determines an estimate $\hat{\theta}$ for parameter θ by maximizing the likelihood $L(\theta)$ of observing data $\mathcal{X} = \{x_1, \ldots, x_n\}$

$$\hat{\theta} = \arg\max_{\theta} \quad L(\theta)$$

- Assuming independent, identically distributed data

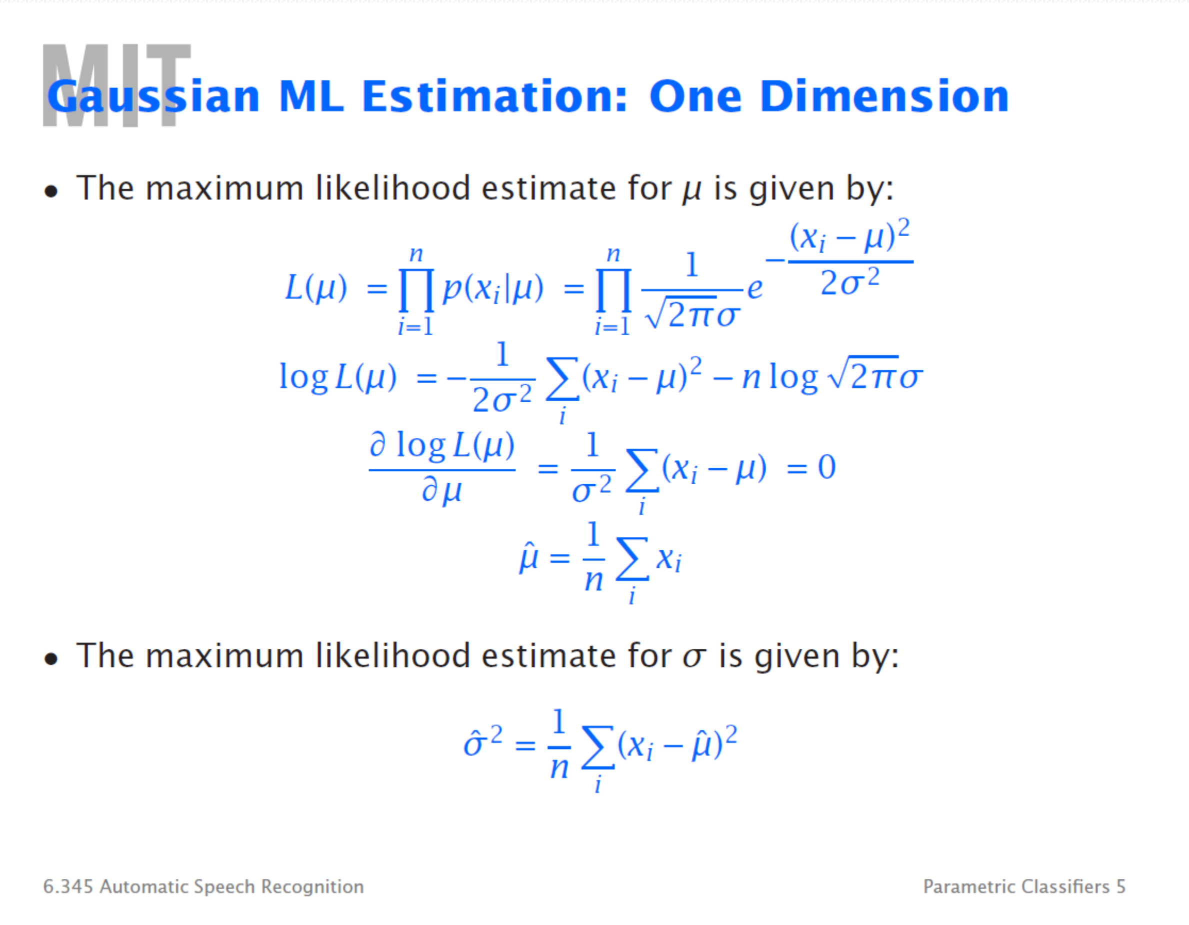
$$L(\theta) = p(\mathcal{X}|\theta) \ = p(x_1, \ldots, x_n|\theta) \ = \prod_{i=1}^{n} p(x_i|\theta)$$

- ML solutions can often be obtained via the derivative

$$\frac{\partial}{\partial \theta} L(\theta) \ = 0$$

- For Gaussian distributions $\log L(\theta)$ is easier to solve

Pattern Classification



Gaussian ML Estimation: One Dimension

- The maximum likelihood estimate for μ is given by:

$$\begin{aligned} L(\mu) &= \prod_{i=1}^{n} p(x_i|\mu) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}} \\ \log L(\mu) &= -\frac{1}{2\sigma^2}\sum_i (x_i-\mu)^2 - n\log\sqrt{2\pi}\sigma \\ \frac{\partial \log L(\mu)}{\partial \mu} &= \frac{1}{\sigma^2}\sum_i (x_i-\mu) = 0 \\ \hat{\mu} &= \frac{1}{n}\sum_i x_i \end{aligned}$$

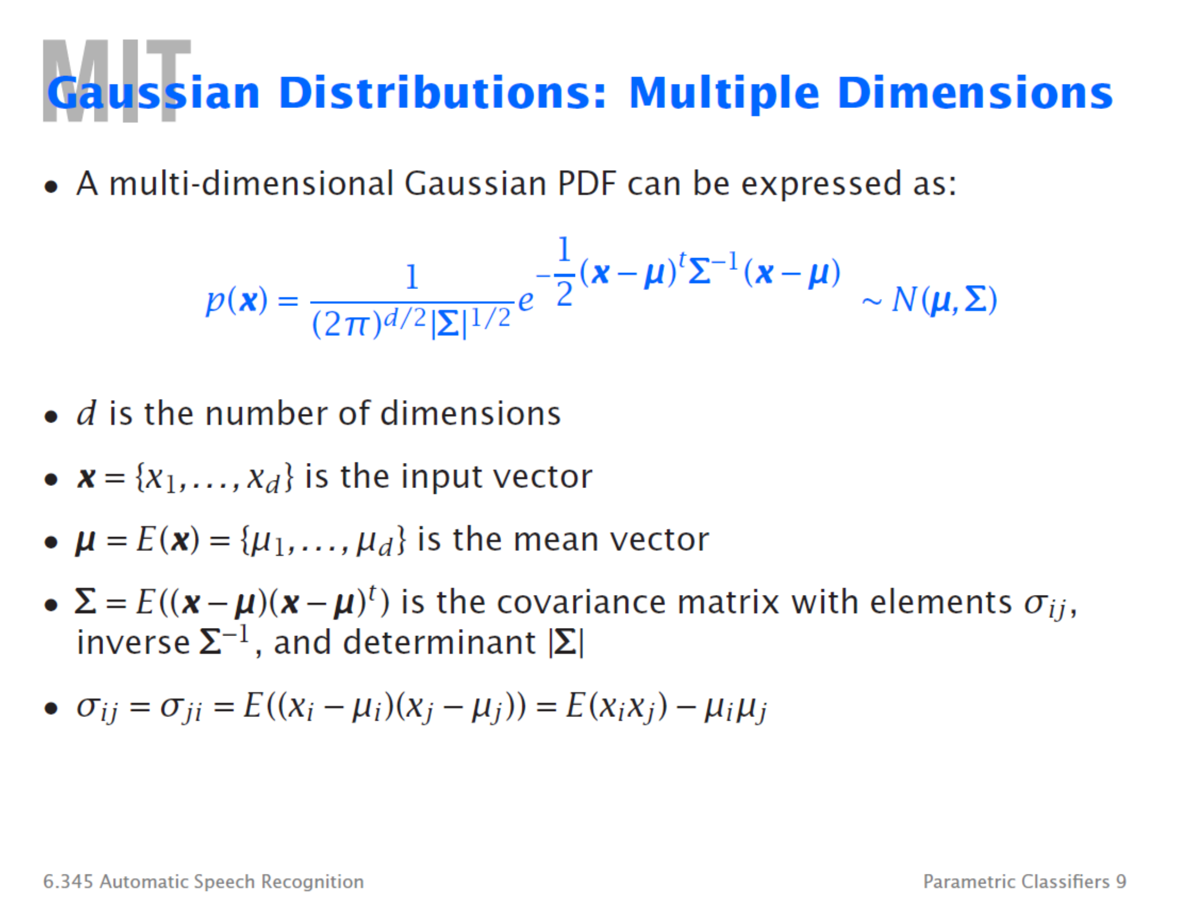
- The maximum likelihood estimate for σ is given by:

$$\hat{\sigma}^2 = \frac{1}{n}\sum_i (x_i-\hat{\mu})^2$$

6.345 Automatic Speech Recognition

Parametric Classifiers 5

Pattern Classification



MIT

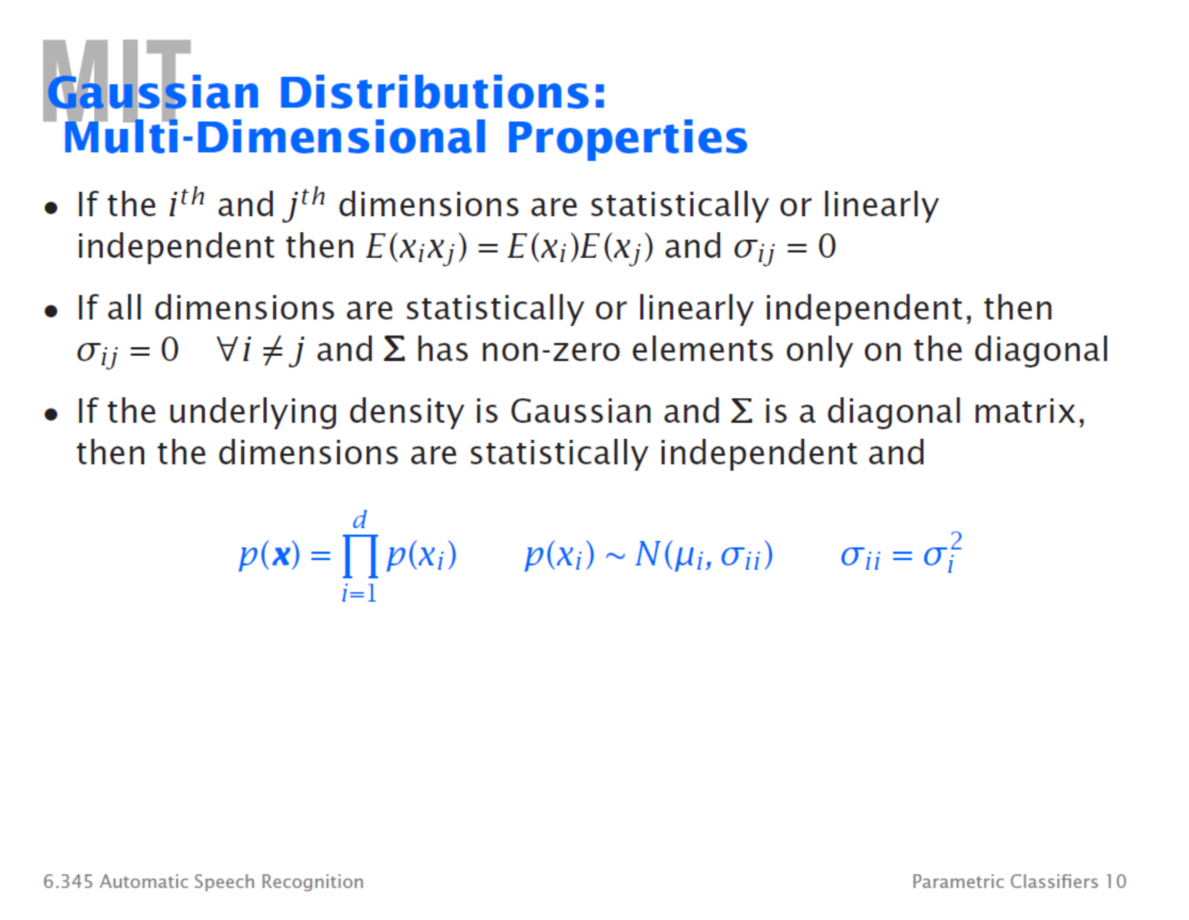
Gaussian Distributions: Multiple Dimensions

- A multi-dimensional Gaussian PDF can be expressed as:

$$p(\boldsymbol{x}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^t\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})} \sim N(\boldsymbol{\mu},\boldsymbol{\Sigma})$$

- d is the number of dimensions
- $\boldsymbol{x} = \{x_1, \ldots, x_d\}$ is the input vector
- $\boldsymbol{\mu} = E(\boldsymbol{x}) = \{\mu_1, \ldots, \mu_d\}$ is the mean vector
- $\boldsymbol{\Sigma} = E((\boldsymbol{x}-\boldsymbol{\mu})(\boldsymbol{x}-\boldsymbol{\mu})^t)$ is the covariance matrix with elements σ_{ij}, inverse $\boldsymbol{\Sigma}^{-1}$, and determinant $|\boldsymbol{\Sigma}|$
- $\sigma_{ij} = \sigma_{ji} = E((x_i-\mu_i)(x_j-\mu_j)) = E(x_i x_j) - \mu_i\mu_j$

Pattern Classification



Gaussian Distributions: Multi-Dimensional Properties

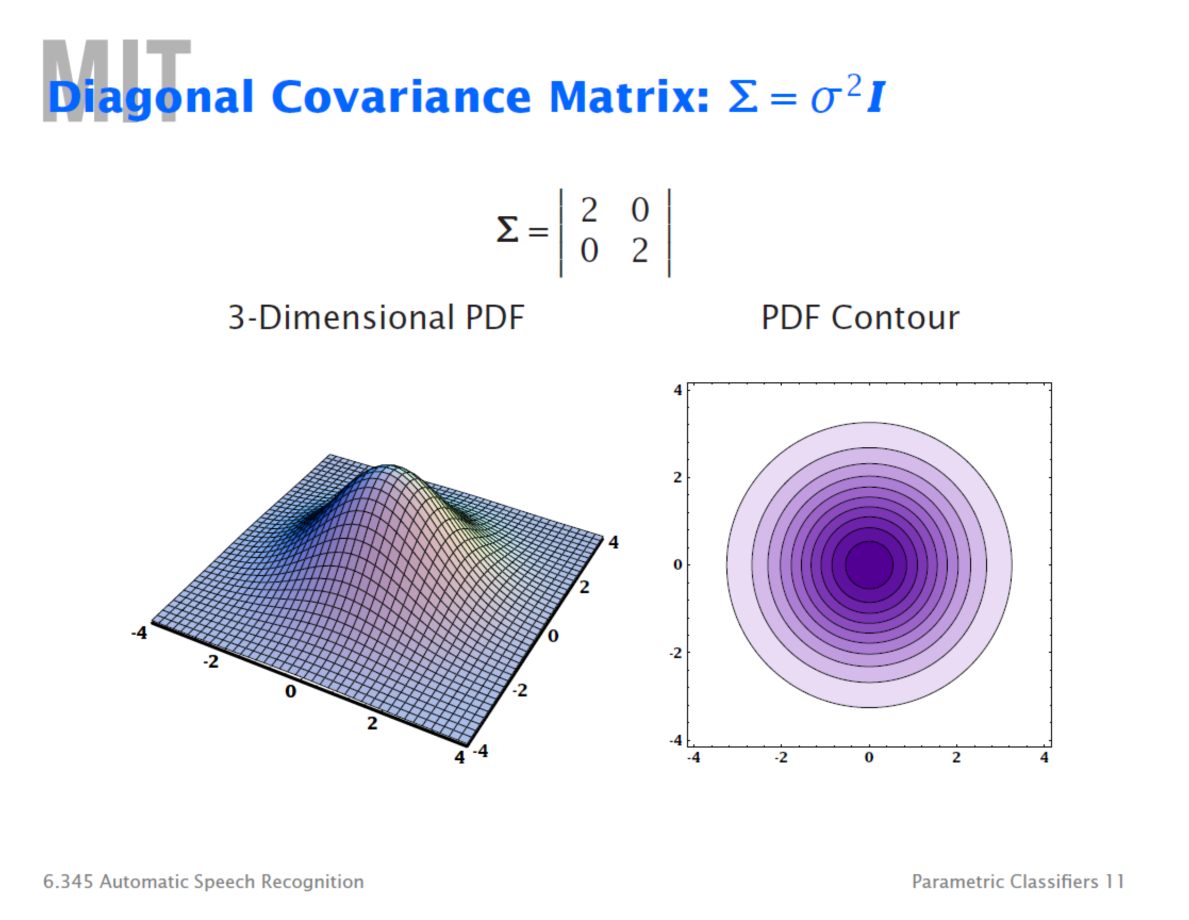
- If the i^{th} and j^{th} dimensions are statistically or linearly independent then $E(x_i x_j) = E(x_i)E(x_j)$ and $\sigma_{ij} = 0$
- If all dimensions are statistically or linearly independent, then $\sigma_{ij} = 0 \quad \forall i \neq j$ and $\mathbf{\Sigma}$ has non-zero elements only on the diagonal
- If the underlying density is Gaussian and $\mathbf{\Sigma}$ is a diagonal matrix, then the dimensions are statistically independent and

$$p(\mathbf{x}) = \prod_{i=1}^{d} p(x_i) \qquad p(x_i) \sim N(\mu_i, \sigma_{ii}) \qquad \sigma_{ii} = \sigma_i^2$$

6.345 Automatic Speech Recognition

Parametric Classifiers 10

Pattern Classification



Diagonal Covariance Matrix: $\Sigma = \sigma^2 \boldsymbol{I}$

$$\Sigma = \begin{vmatrix} 2 & 0 \\ 0 & 2 \end{vmatrix}$$

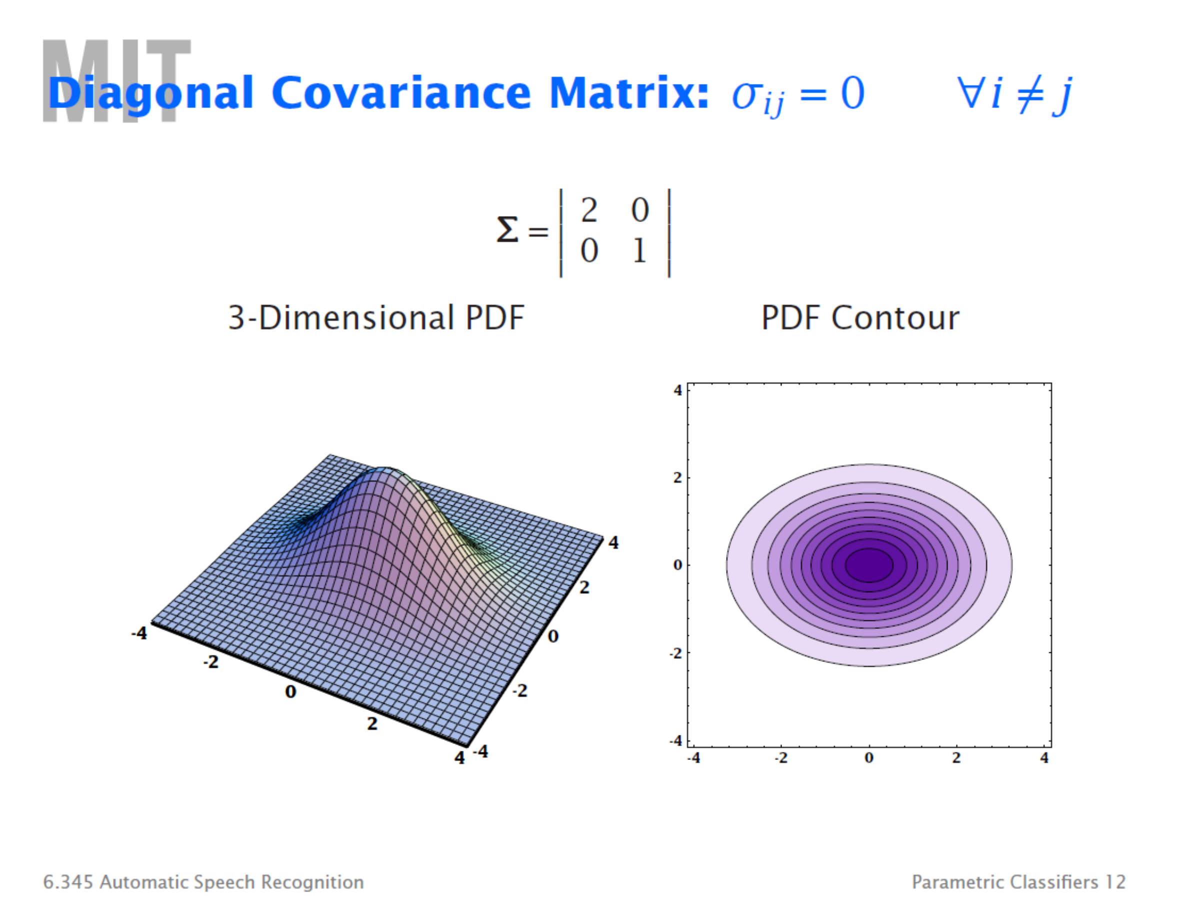
3-Dimensional PDF

PDF Contour

6.345 Automatic Speech Recognition

Parametric Classifiers 11

Pattern Classification



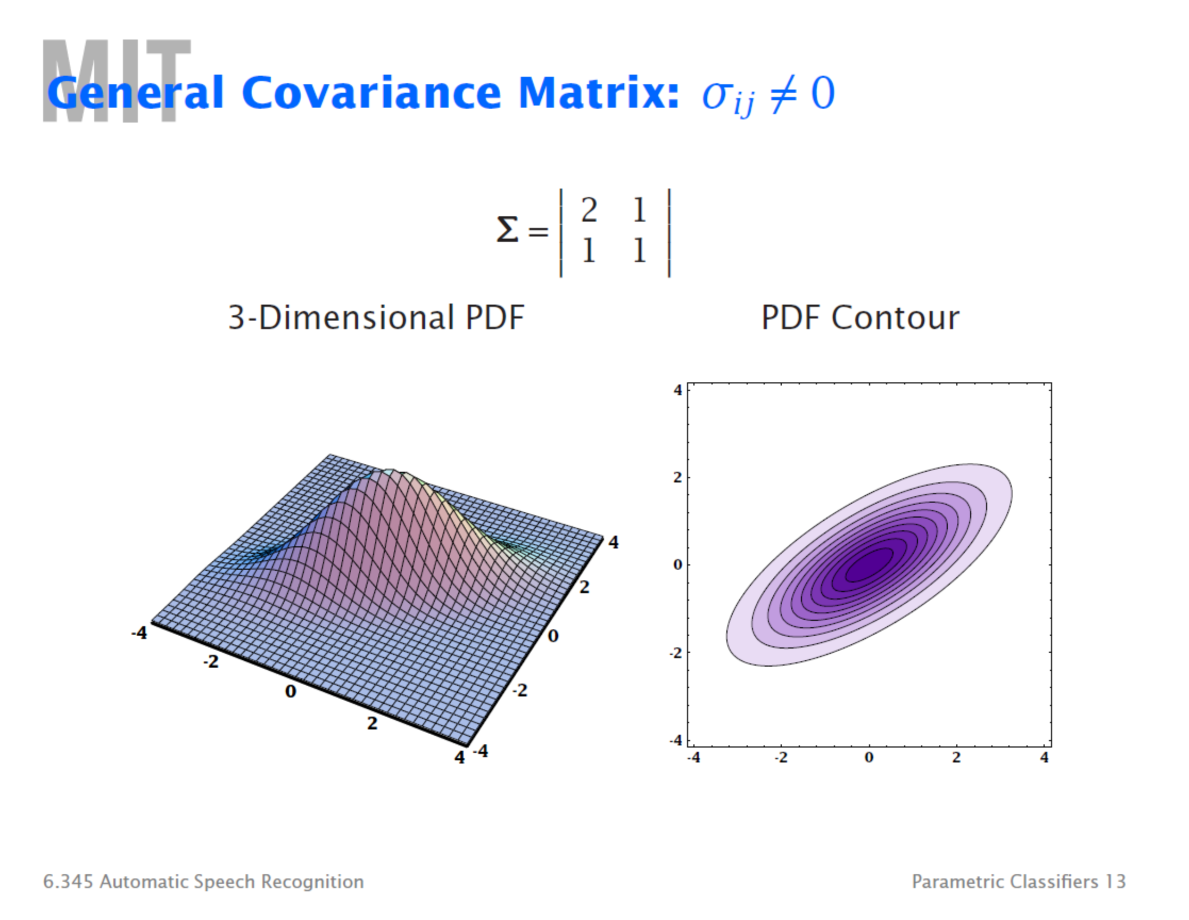
Diagonal Covariance Matrix: $\sigma_{ij} = 0 \qquad \forall i \neq j$

$$\Sigma = \begin{vmatrix} 2 & 0 \\ 0 & 1 \end{vmatrix}$$

3-Dimensional PDF

PDF Contour

Pattern Classification



General Covariance Matrix: $\sigma_{ij} \neq 0$

$$\Sigma = \begin{vmatrix} 2 & 1 \\ 1 & 1 \end{vmatrix}$$

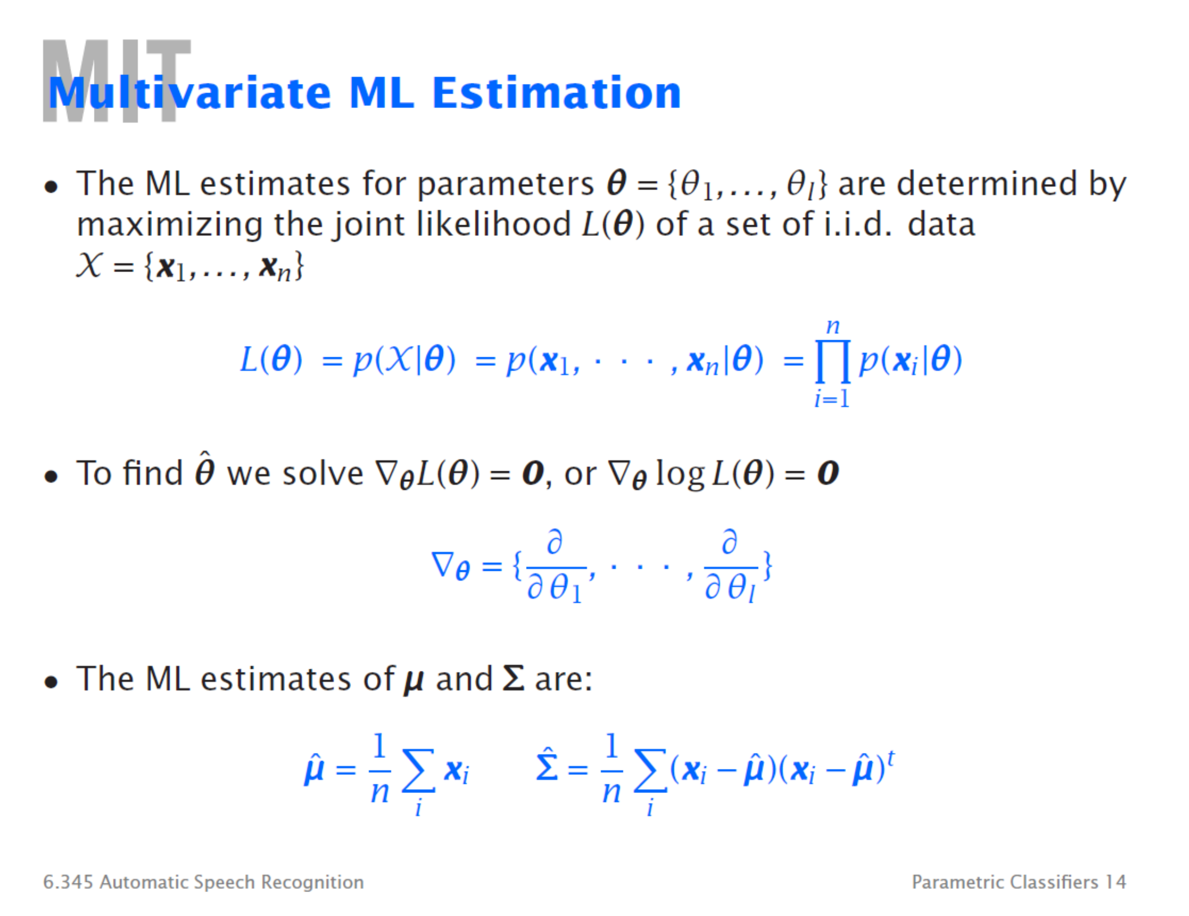
3-Dimensional PDF

PDF Contour

6.345 Automatic Speech Recognition

Parametric Classifiers 13

Pattern Classification



Multivariate ML Estimation

- The ML estimates for parameters $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_l\}$ are determined by maximizing the joint likelihood $L(\boldsymbol{\theta})$ of a set of i.i.d. data $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$

$$L(\boldsymbol{\theta}) = p(\mathcal{X}|\boldsymbol{\theta}) = p(\mathbf{x}_1, \cdots, \mathbf{x}_n|\boldsymbol{\theta}) = \prod_{i=1}^{n} p(\mathbf{x}_i|\boldsymbol{\theta})$$

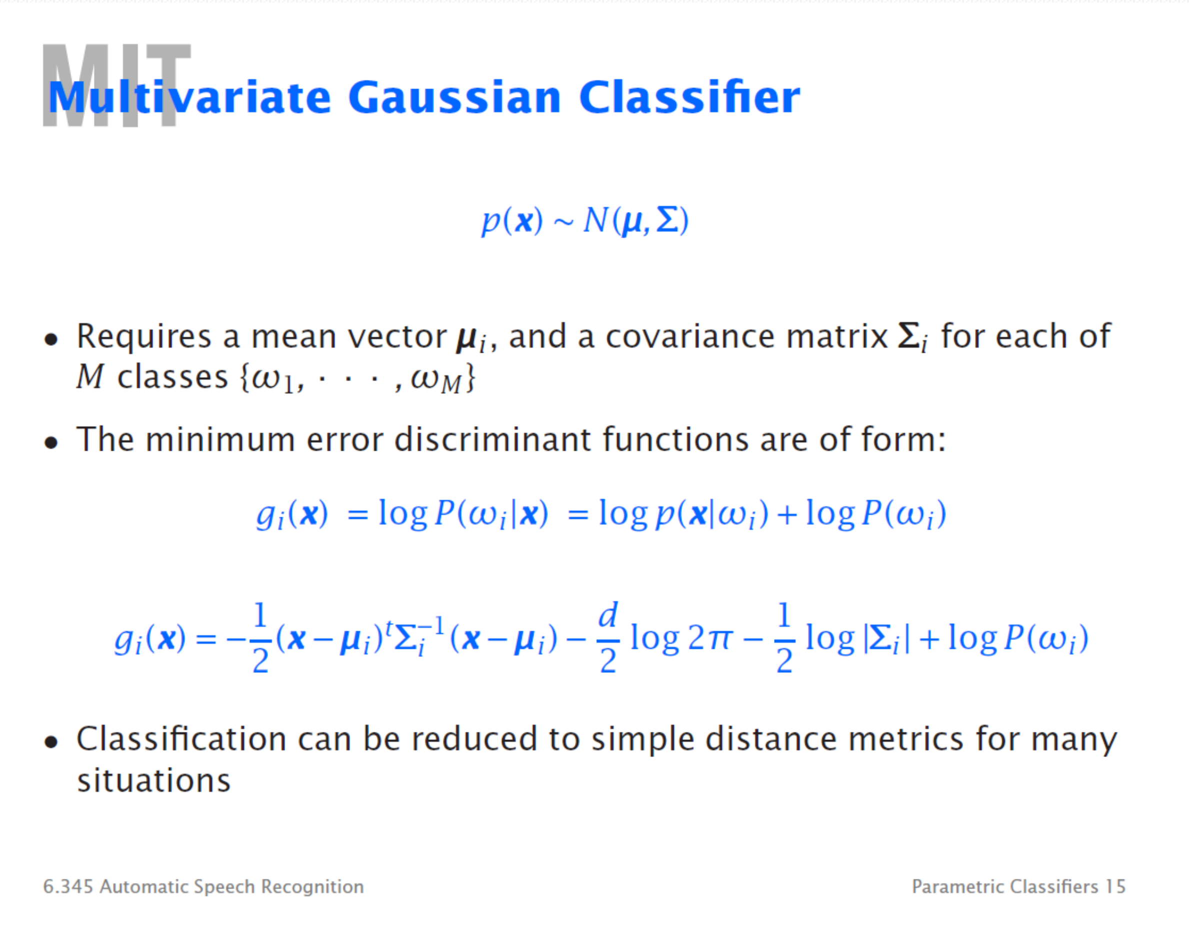
- To find $\hat{\theta}$ we solve $\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) = \mathbf{0}$, or $\nabla_{\boldsymbol{\theta}} \log L(\boldsymbol{\theta}) = \mathbf{0}$

$$\nabla_{\boldsymbol{\theta}} = \{\frac{\partial}{\partial \theta_1}, \cdots, \frac{\partial}{\partial \theta_l}\}$$

- The ML estimates of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are:

$$\hat{\boldsymbol{\mu}} = \frac{1}{n} \sum_i \mathbf{x}_i \qquad \hat{\boldsymbol{\Sigma}} = \frac{1}{n} \sum_i (\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})^t$$

Pattern Classification



MIT

Multivariate Gaussian Classifier

$$p(\boldsymbol{x}) \sim N(\boldsymbol{\mu}, \Sigma)$$

- Requires a mean vector $\boldsymbol{\mu}_i$, and a covariance matrix $\boldsymbol{\Sigma}_i$ for each of M classes $\{\omega_1, \cdots, \omega_M\}$
- The minimum error discriminant functions are of form:

$$g_i(\boldsymbol{x}) \; = \log P(\omega_i|\boldsymbol{x}) \; = \log p(\boldsymbol{x}|\omega_i) + \log P(\omega_i)$$

$$g_i(\boldsymbol{x}) = -\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^t\Sigma_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i) - \frac{d}{2}\log 2\pi - \frac{1}{2}\log|\Sigma_i| + \log P(\omega_i)$$

- Classification can be reduced to simple distance metrics for many situations

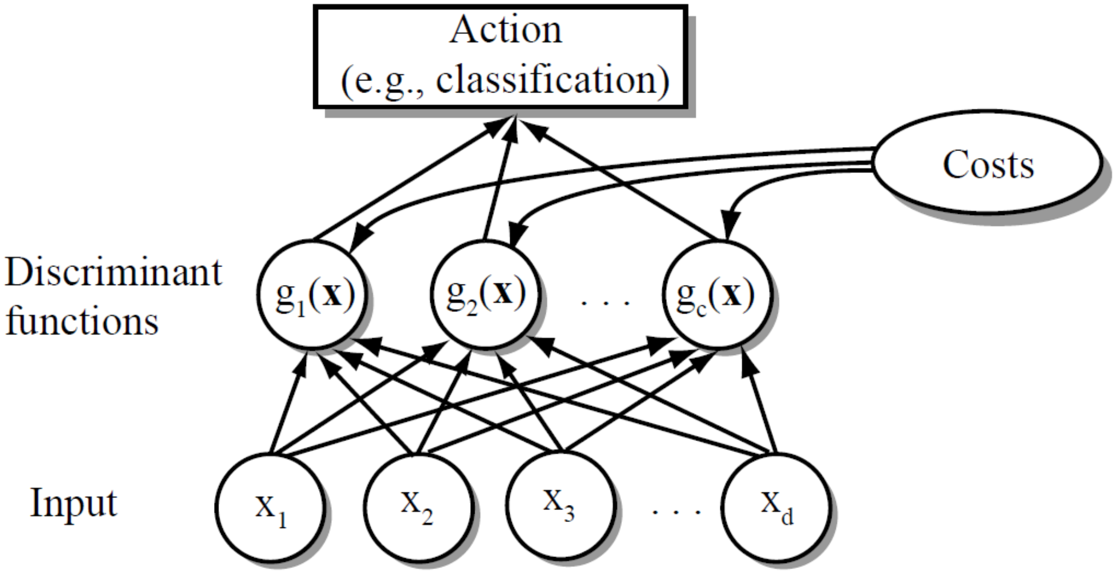


Figure 2.5: The functional structure of a general statistical pattern classifier which includes d inputs and c discriminant functions $g_i(\mathbf{x})$. A subsequent step determines which of the discriminant values is the maximum, and categorizes the input pattern accordingly. The arrows show the direction of the flow of information, though frequently the arrows are omitted when the direction of flow is self-evident.

$$g_i(\mathbf{x}) = P(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)P(\omega_i)}{\sum_{j=1}^{c} p(\mathbf{x}|\omega_j)P(\omega_j)} \tag{25}$$

$$g_i(\mathbf{x}) = p(\mathbf{x}|\omega_i)P(\omega_i) \tag{26}$$

$$g_i(\mathbf{x}) = \ln p(\mathbf{x}|\omega_i) + \ln P(\omega_i), \tag{27}$$

where ln denotes natural logarithm.

DECISION REGION

Even though the discriminant functions can be written in a variety of forms, the decision rules are equivalent. The effect of any decision rule is to divide the feature space into c *decision regions*, $\mathcal{R}_1,...,\mathcal{R}_c$. If $g_i(\mathbf{x}) > g_j(\mathbf{x})$ for all $j \neq i$, then $\mathbf{x}$ is in $\mathcal{R}_i$, and the decision rule calls for us to assign $\mathbf{x}$ to ω_i. The regions are separated by *decision boundaries*, surfaces in feature space where ties occur among the largest discriminant functions (Fig. 2.6).

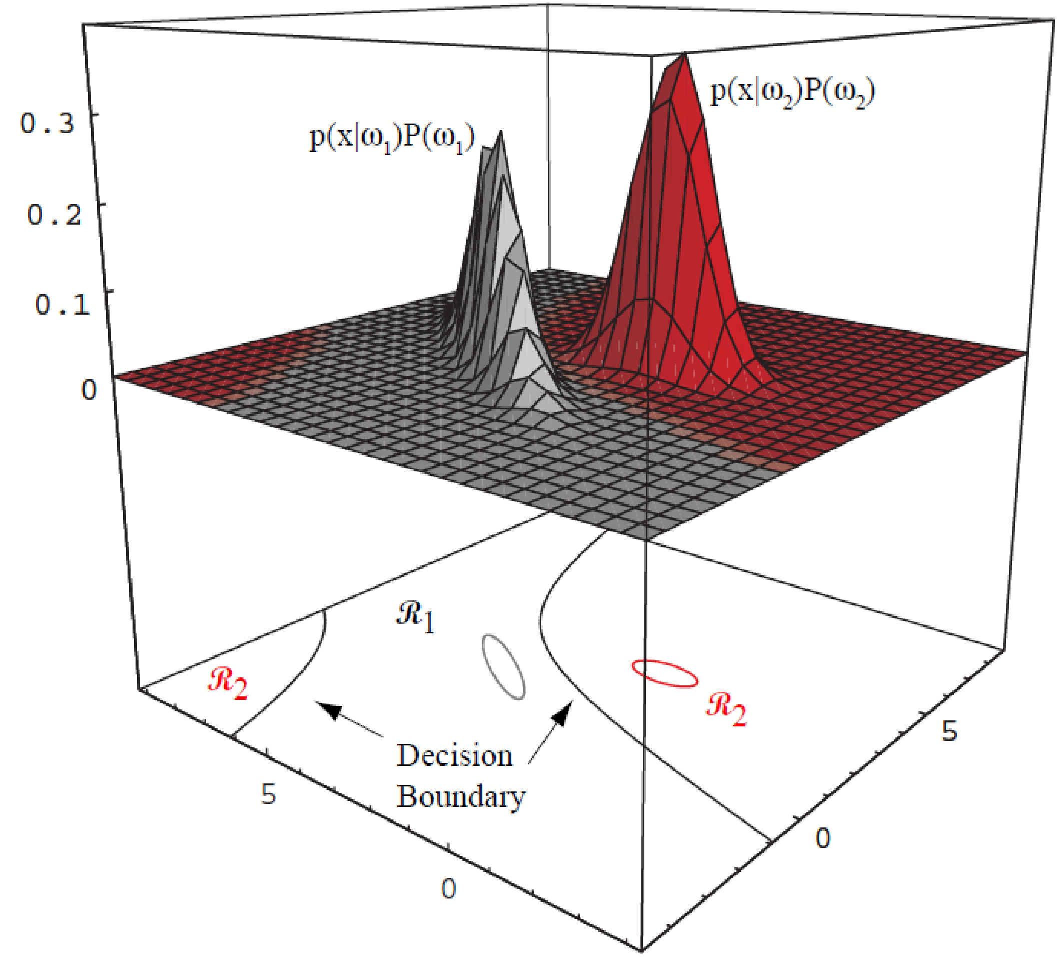
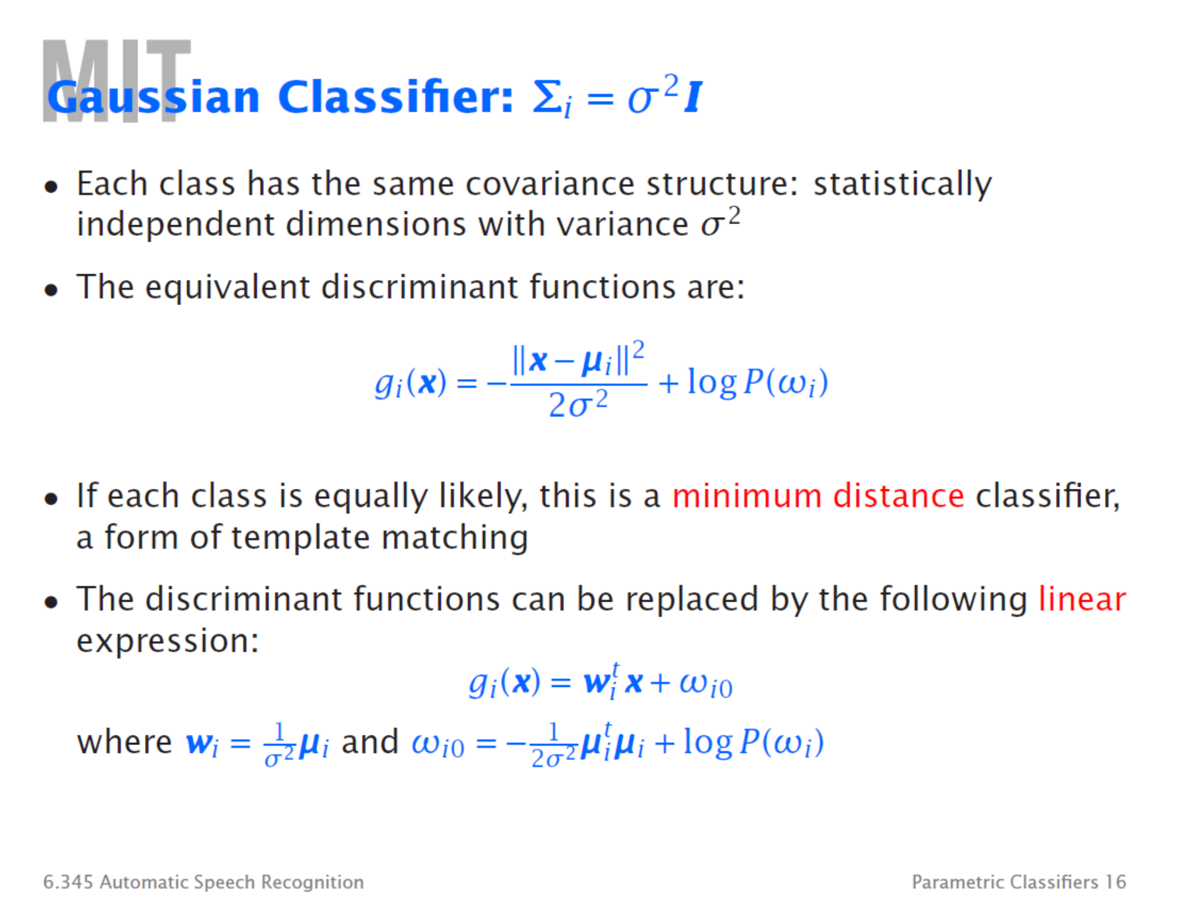


Figure 2.6: In this two-dimensional two-category classifier, the probability densities are Gaussian (with $1/e$ ellipses shown), the decision boundary consists of two hyperbolas, and thus the decision region $\mathcal{R}_2$ is not simply connected.

Pattern Classification



Gaussian Classifier: $\Sigma_i = \sigma^2 \boldsymbol{I}$

- Each class has the same covariance structure: statistically independent dimensions with variance σ^2
- The equivalent discriminant functions are:

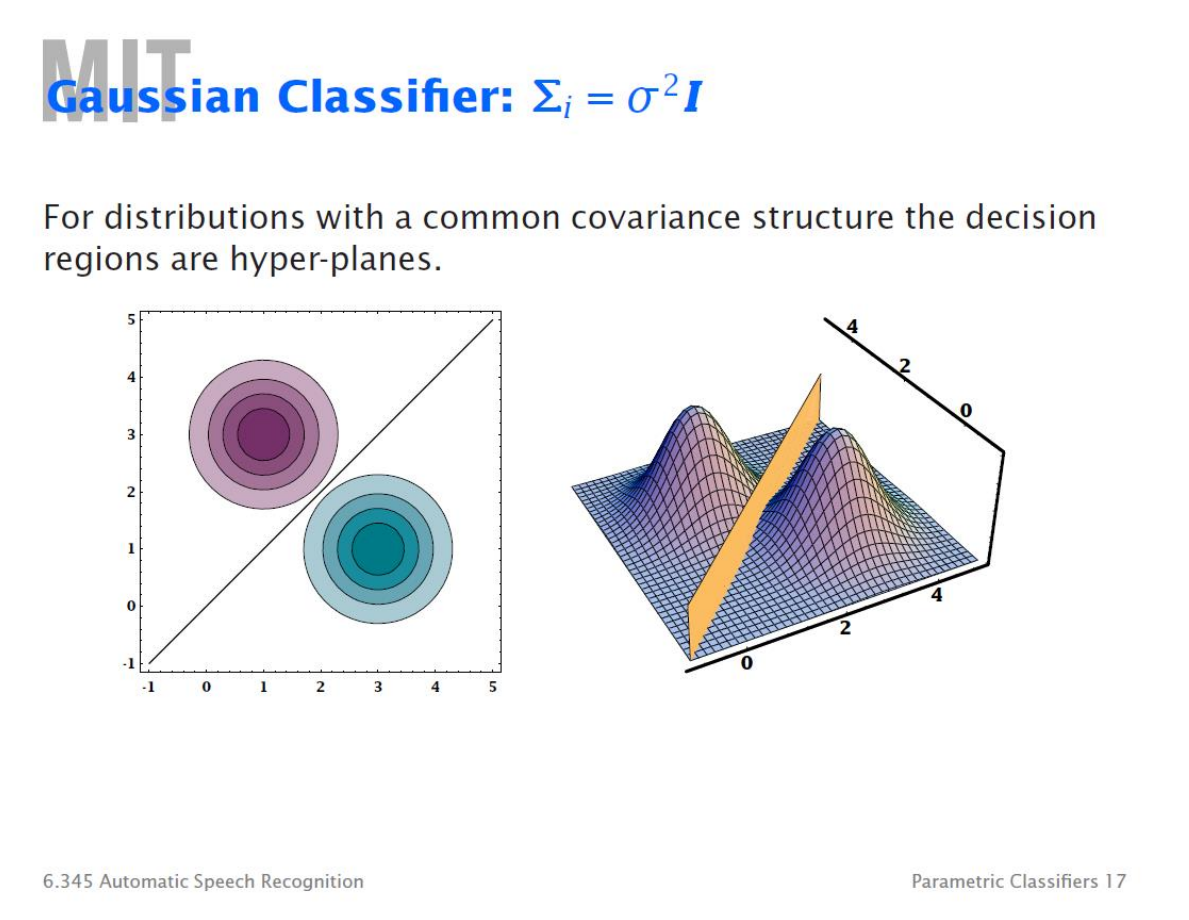
$$g_i(\boldsymbol{x}) = -\frac{\|\boldsymbol{x} - \boldsymbol{\mu}_i\|^2}{2\sigma^2} + \log P(\omega_i)$$

- If each class is equally likely, this is a minimum distance classifier, a form of template matching
- The discriminant functions can be replaced by the following linear expression:

$$g_i(\boldsymbol{x}) = \mathbf{w}_i^t \boldsymbol{x} + \omega_{i0}$$

where $\mathbf{w}_i = \frac{1}{\sigma^2}\boldsymbol{\mu}_i$ and $\omega_{i0} = -\frac{1}{2\sigma^2}\boldsymbol{\mu}_i^t\boldsymbol{\mu}_i + \log P(\omega_i)$

Pattern Classification



Gaussian Classifier: $\Sigma_i = \sigma^2 \boldsymbol{I}$

For distributions with a common covariance structure the decision regions are hyper-planes.

6.345 Automatic Speech Recognition

Parametric Classifiers 17

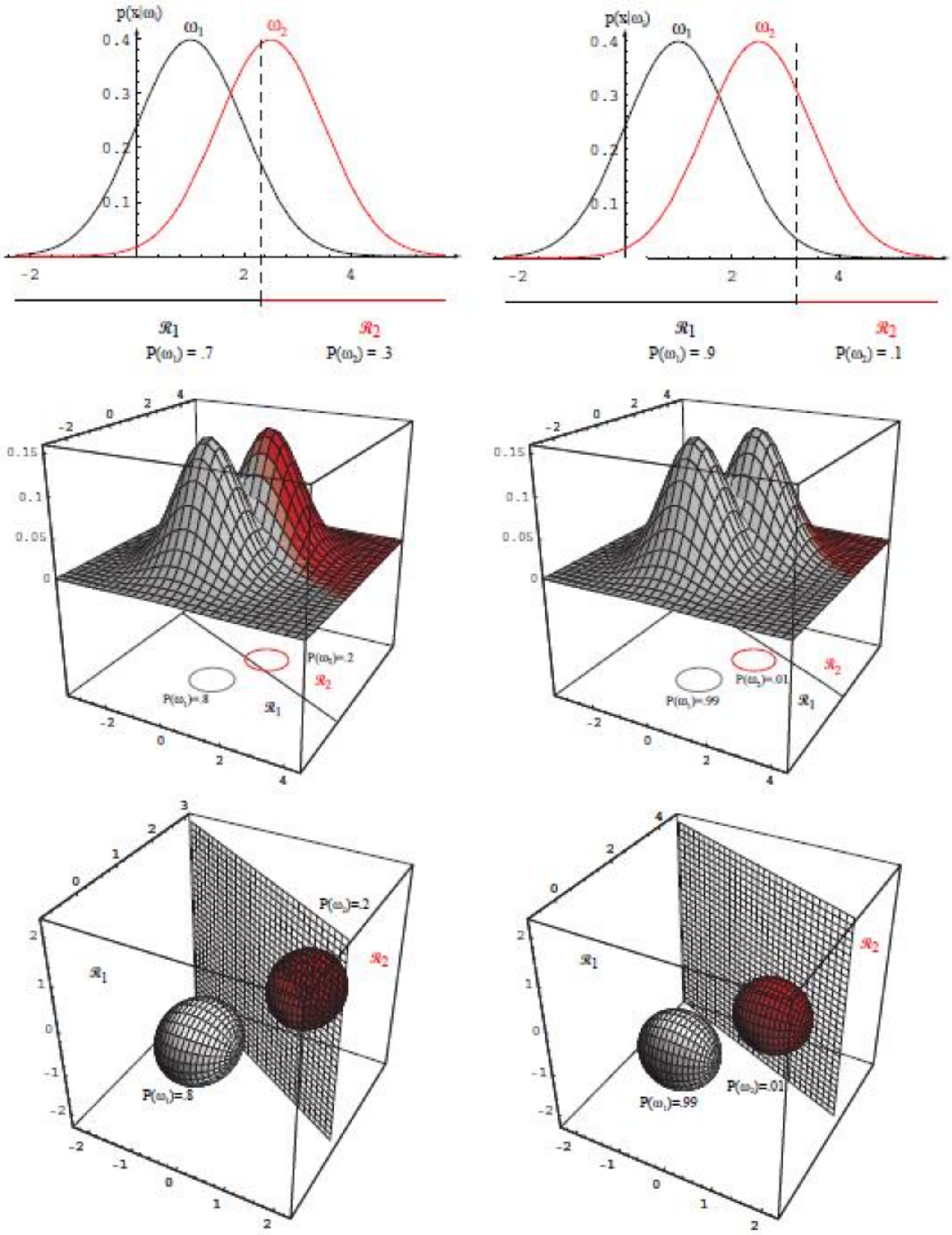


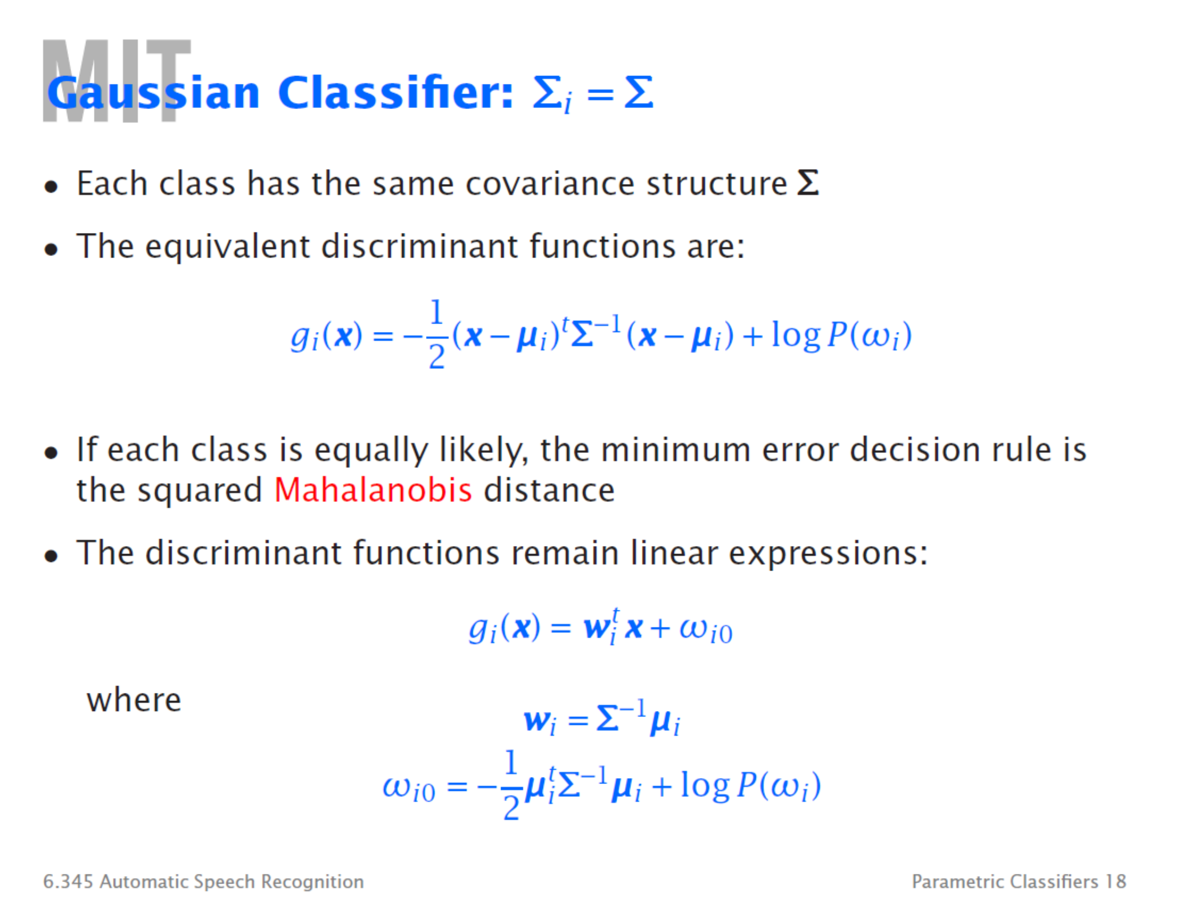
Figure 2.11: As the priors are changed, the decision boundary shifts: for sufficiently

If the prior probabilities $P(\omega_i)$ are the same for all c classes, then the ln $P(\omega_i)$ term becomes another unimportant additive constant that can be ignored. When this happens, the optimum decision rule can be stated very simply: to classify a feature vector $\mathbf{x}$, measure the Euclidean distance $\|\mathbf{x} - \boldsymbol{\mu}_i\|$ from each $\mathbf{x}$ to each of the c mean vectors, and assign $\mathbf{x}$ to the category of the nearest mean. Such a classifier is called a *minimum distance classifier*. If each mean vector is thought of as being an ideal prototype or template for patterns in its class, then this is essentially a *template-matching* procedure (Fig. 2.10), a technique we will consider again in Chap. ?? Sect. ?? on the nearest-neighbor algorithm.

MINIMUM DISTANCE CLASSIFIER

TEMPLATE-MATCHING

Pattern Classification



Gaussian Classifier: $\Sigma_i = \Sigma$

- Each class has the same covariance structure $\boldsymbol{\Sigma}$
- The equivalent discriminant functions are:

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^t\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_i) + \log P(\omega_i)$$

- If each class is equally likely, the minimum error decision rule is the squared Mahalanobis distance
- The discriminant functions remain linear expressions:

$$g_i(\mathbf{x}) = \mathbf{w}_i^t\mathbf{x} + \omega_{i0}$$

where

$$\mathbf{w}_i = \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_i$$

$$\omega_{i0} = -\frac{1}{2}\boldsymbol{\mu}_i^t\Sigma^{-1}\boldsymbol{\mu}_i + \log P(\omega_i)$$

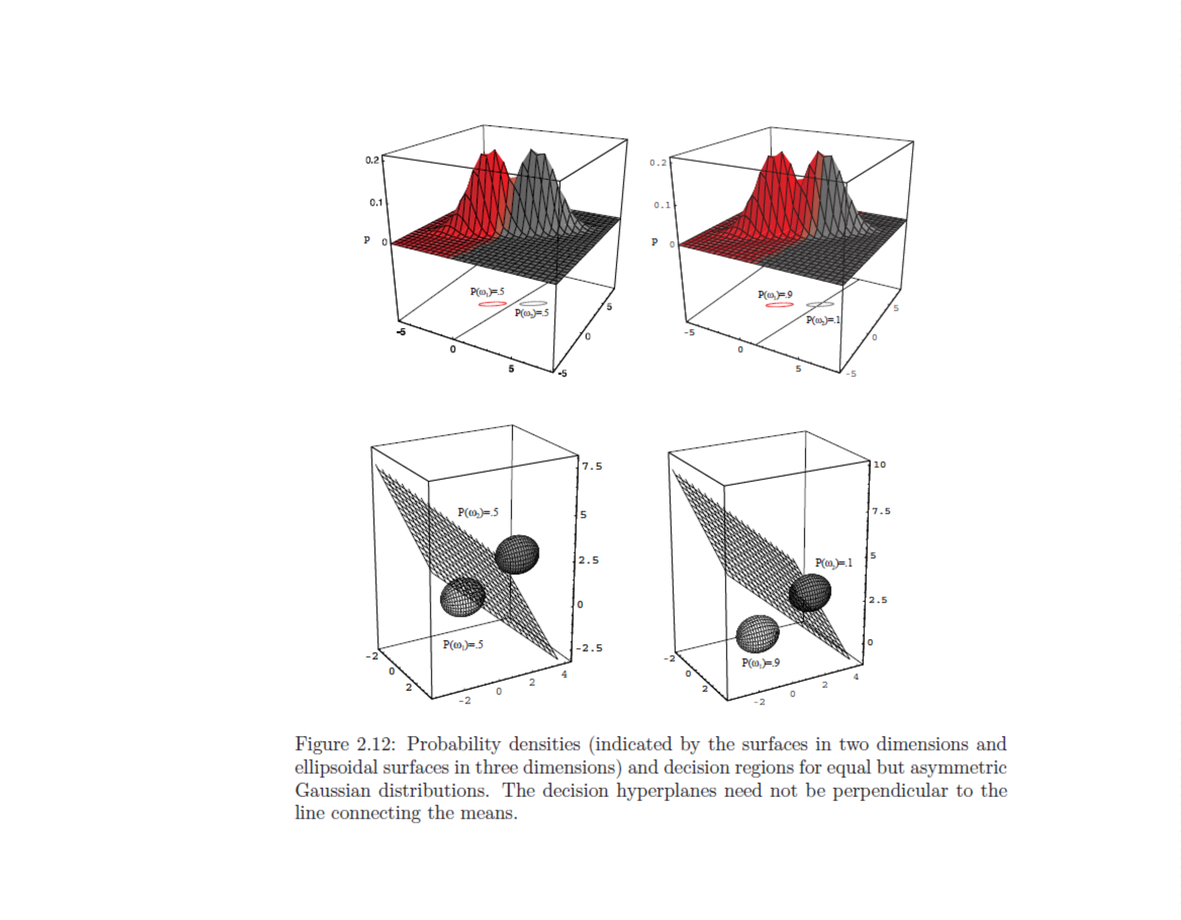
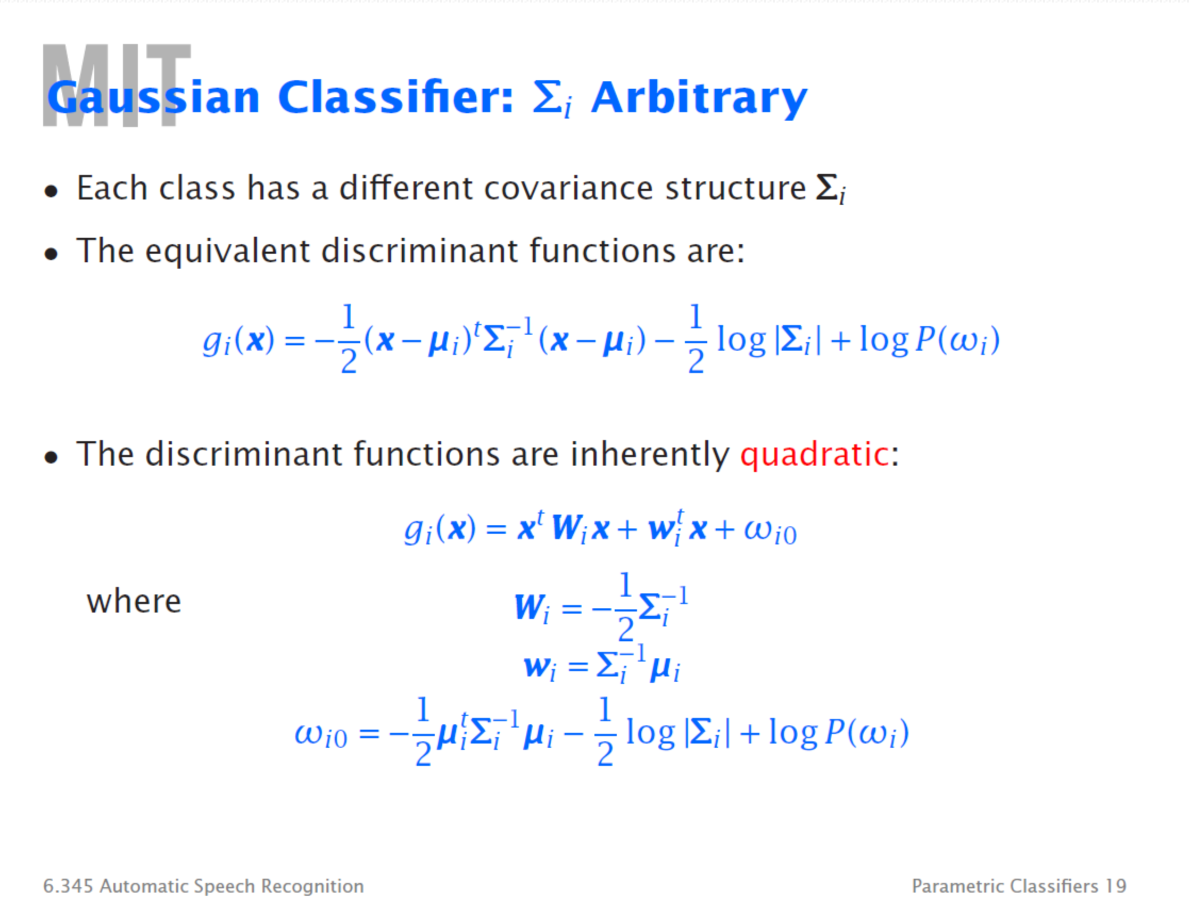


Figure 2.12: Probability densities (indicated by the surfaces in two dimensions and ellipsoidal surfaces in three dimensions) and decision regions for equal but asymmetric Gaussian distributions. The decision hyperplanes need not be perpendicular to the line connecting the means.

Pattern Classification



Gaussian Classifier: Σ_i Arbitrary

- Each class has a different covariance structure $\boldsymbol{\Sigma}_i$
- The equivalent discriminant functions are:

$$g_i(\boldsymbol{x}) = -\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^t\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i) - \frac{1}{2}\log|\boldsymbol{\Sigma}_i| + \log P(\omega_i)$$

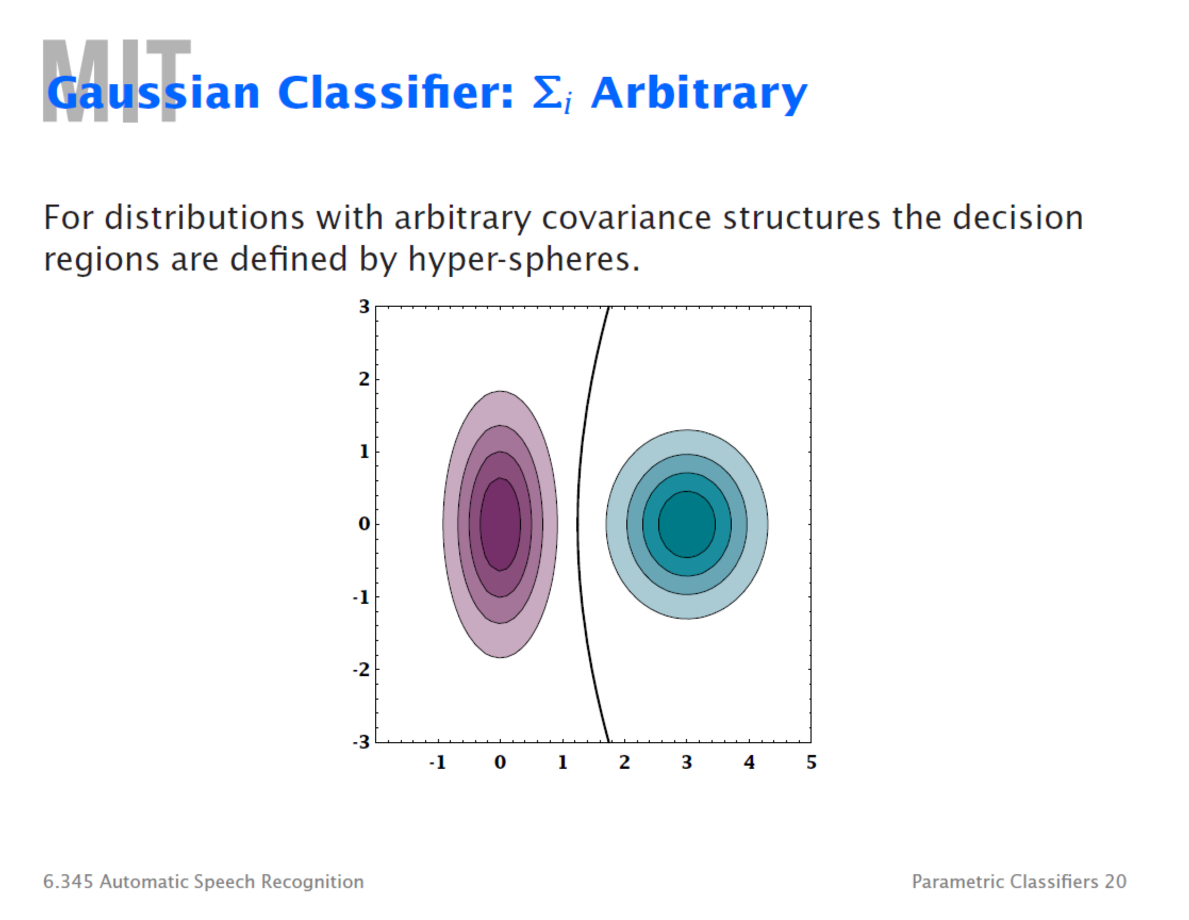
- The discriminant functions are inherently quadratic:

$$g_i(\boldsymbol{x}) = \boldsymbol{x}^t\,\boldsymbol{W}_i\boldsymbol{x} + \boldsymbol{w}_i^t\boldsymbol{x} + \omega_{i0}$$

where

$$\boldsymbol{W}_i = -\frac{1}{2}\boldsymbol{\Sigma}_i^{-1}$$
$$\boldsymbol{w}_i = \boldsymbol{\Sigma}_i^{-1}\boldsymbol{\mu}_i$$
$$\omega_{i0} = -\frac{1}{2}\boldsymbol{\mu}_i^t\boldsymbol{\Sigma}_i^{-1}\boldsymbol{\mu}_i - \frac{1}{2}\log|\boldsymbol{\Sigma}_i| + \log P(\omega_i)$$

Pattern Classification



Gaussian Classifier: Σ_i Arbitrary

For distributions with arbitrary covariance structures the decision regions are defined by hyper-spheres.

3
2
1
0
-1
-2
-3
-1
0
1
2
3
4
5

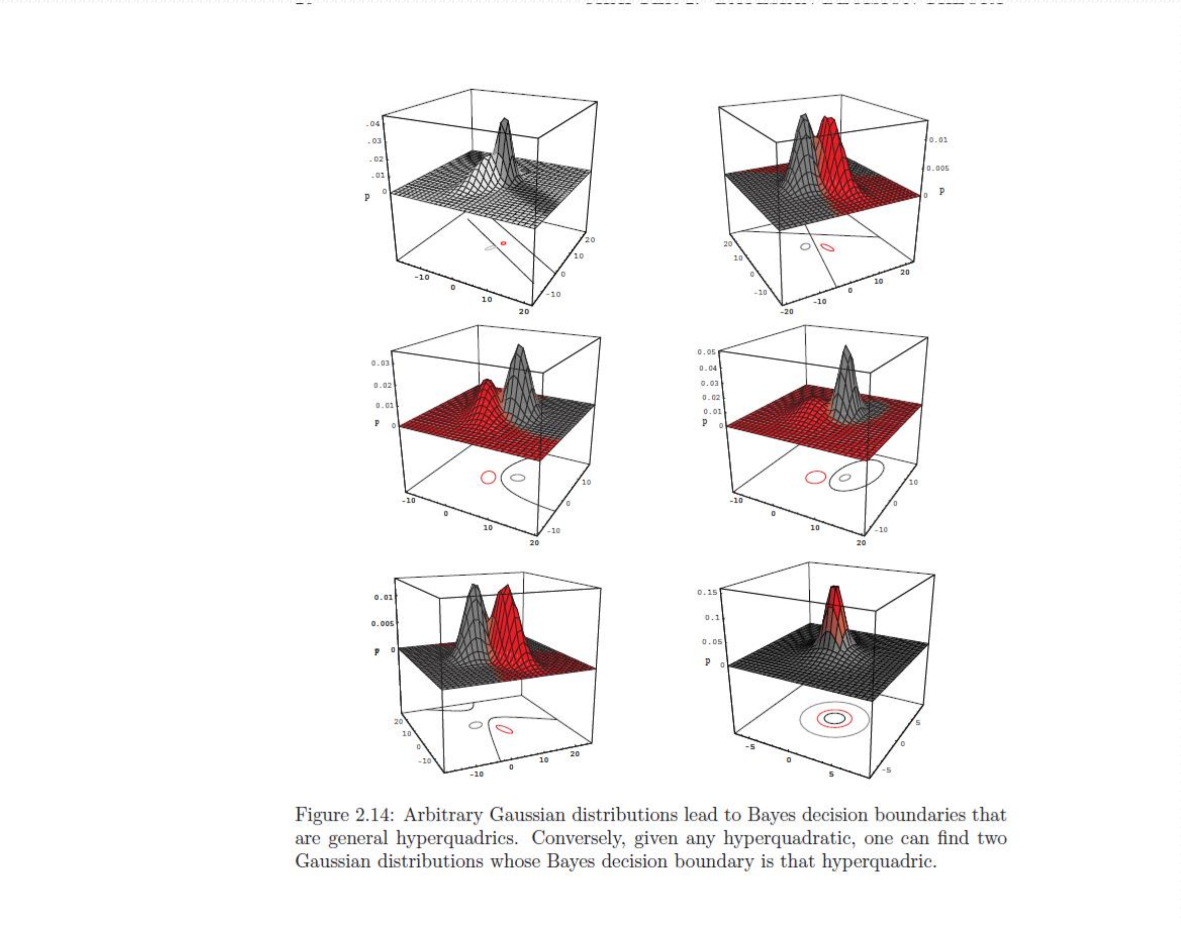
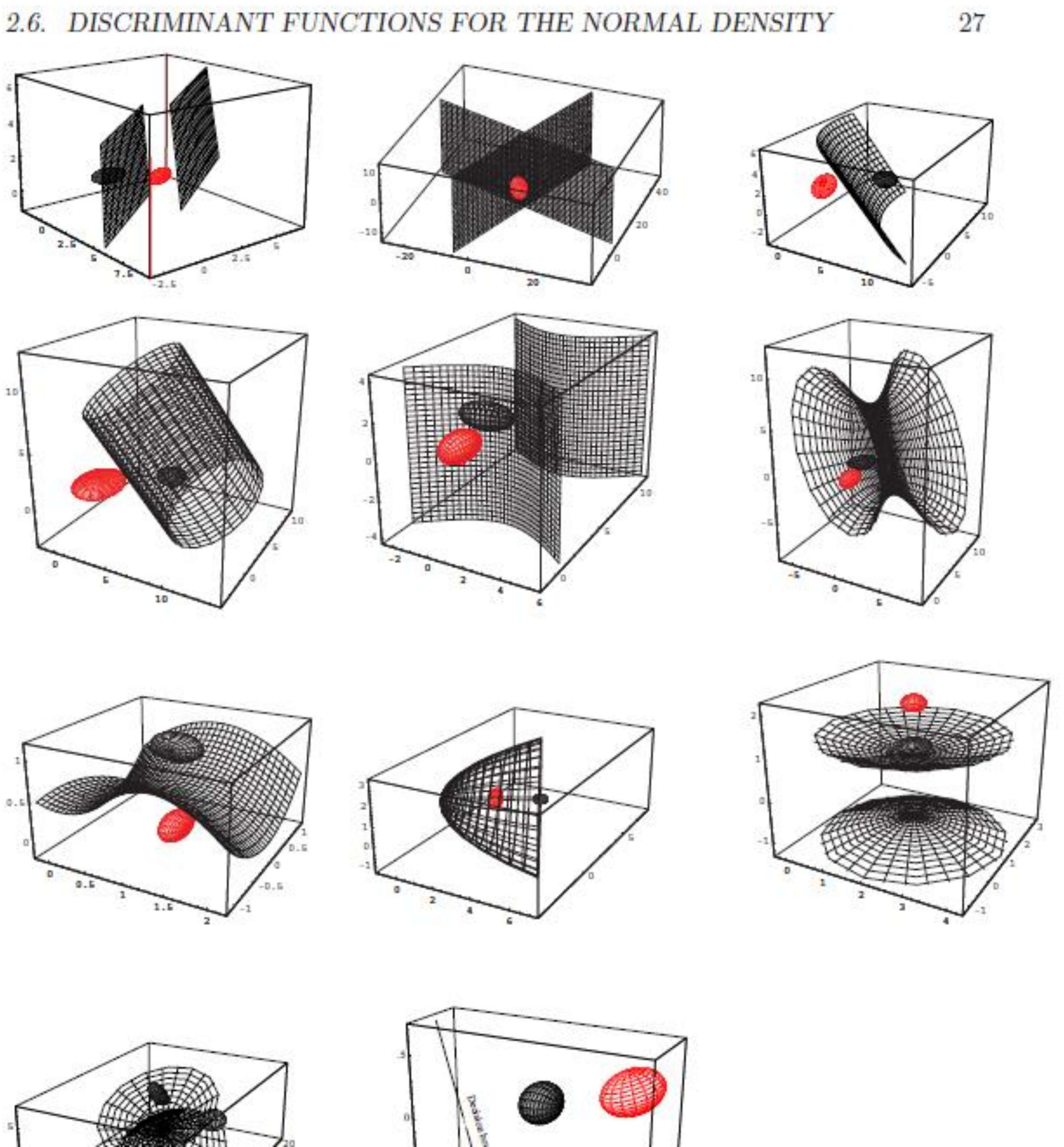
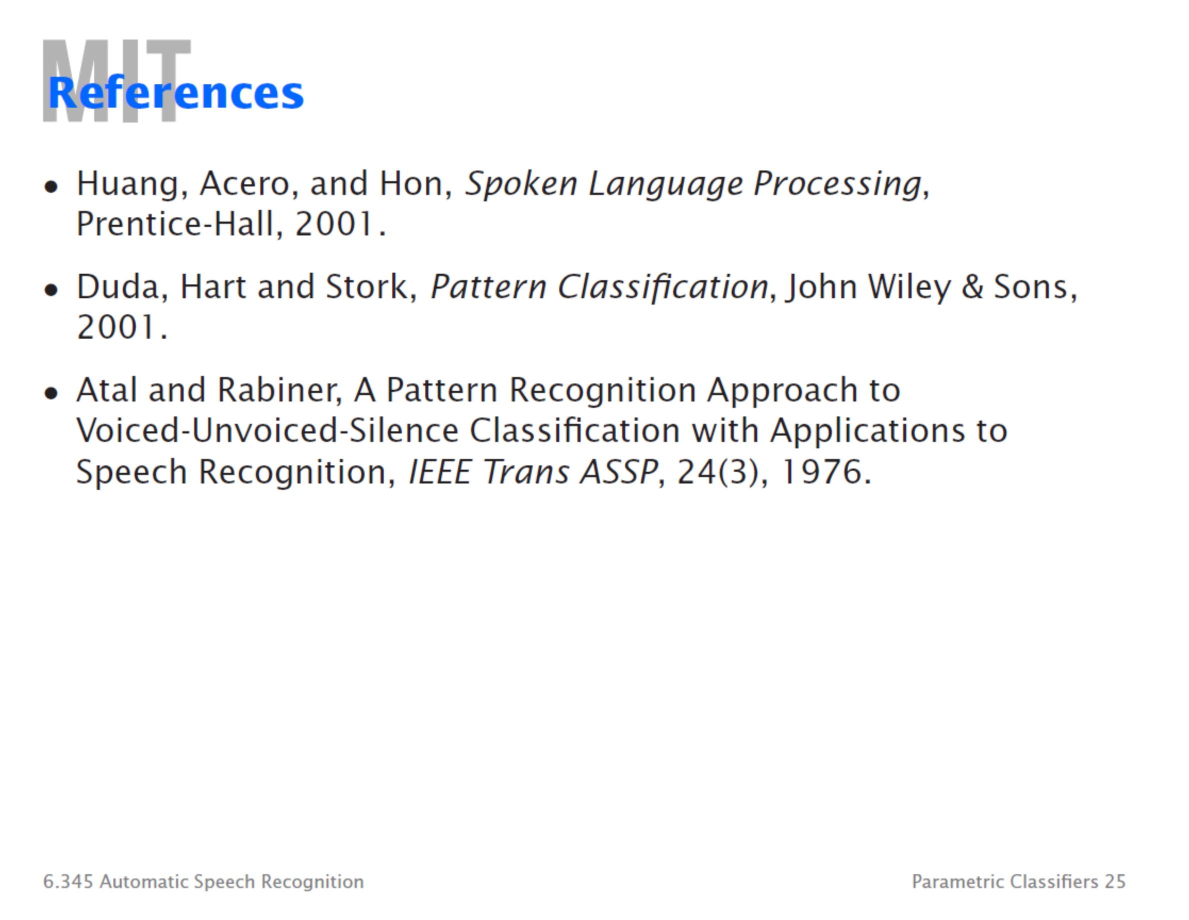


Figure 2.14: Arbitrary Gaussian distributions lead to Bayes decision boundaries that are general hyperquadrics. Conversely, given any hyperquadratic, one can find two Gaussian distributions whose Bayes decision boundary is that hyperquadric.



2.6. DISCRIMINANT FUNCTIONS FOR THE NORMAL DENSITY 27

Pattern Classification



References

- Huang, Acero, and Hon, *Spoken Language Processing*, Prentice-Hall, 2001.
- Duda, Hart and Stork, *Pattern Classification*, John Wiley & Sons, 2001.
- Atal and Rabiner, A Pattern Recognition Approach to Voiced-Unvoiced-Silence Classification with Applications to Speech Recognition, *IEEE Trans ASSP*, 24(3), 1976.

6.345 Automatic Speech Recognition

Parametric Classifiers 25