《模式识别》

第六章 贝叶斯决策理论

马锦华

https://cse.sysu.edu.cn/teacher/MaJinhua

SUN YAT-SEN University

课程目录（暂定）

- 第一章　课程简介与预备知识　6学时
- 第二章　特征提取与表示　6学时
- 第三章　主成分分析　3学时
- 第四章　归一化、判别分析、人脸识别　3学时
- 第五章　EM算法与聚类　3学时
- 第六章　贝叶斯决策理论　3学时
- 第七章　线性分类器与感知机　3学时
- 第八章　支持向量机　3学时
- 第九章　神经网络、正则项和优化方法　3学时
- 第十章　卷积神经网络及经典框架　3学时
- 第十一章　循环神经网络　3学时
- 第十二章　Transformer　3学时
- 第十三章　自监督与半监督学习　3学时
- 第十四章　开放世界模式识别　6学时

Bayesian Decision Theory

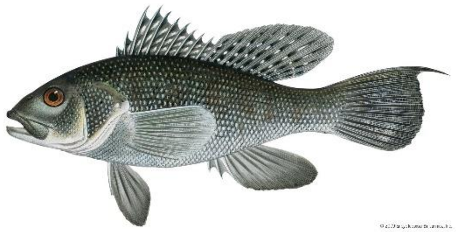
Sections 2.1-2.10 (Duda et al.)

- A statistical approach for designing pattern classification systems.

- Quantifies trade-offs between various **classification decisions** by using probability and the costs associated with such decisions.

- Fundamental to this approach is the Bayes rule.

Terminology



- State of nature *ω:*
  - e.g., ω_1 for sea bass, ω_2 for salmon

- Prior probability *$P(\omega)$:*
  - e.g., $P(\omega_1)$ and $P(\omega_2)$ reflect our prior knowledge of how likely is to get a sea bass or a salmon <u>before</u> the fish is actually caught.
- Features x and probability density p(x) *(evidence) :*
  - e.g., the probability density of some feature(s) x (e.g., lightness) <u>independently</u> of the class.

Terminology (cont'd)

- Conditional probability density $p(x/\omega_j)$ (*likelihood*):
  - e.g., the probability density of some feature(s) x (e.g., lightness) given that it belongs to class ω_j

- Conditional probability $P(\omega_j/x)$ *(posterior)* :
  - e.g., the probability that the fish belongs to class ω_j given x

Decision Rule Using Prior Probabilities Only

Decide ω_1 if $P(\omega_1) > P(\omega_2)$; otherwise **decide** ω_2

- Will be making the same decision at all times!
  - Favors the most likely class.
  - Optimum if no other information is available.

- What is the probability of error ?

$$P(error) = \begin{cases} P(\omega_1) \ \ if \ \ we\,decide\,\omega_2 \\ P(\omega_2) \ if \ \ we\,decide\,\omega_1 \end{cases}$$

or $\quad$ $P(error) = \min[P(\omega_1), P(\omega_2)]$

Decision Rule Using Conditional Probabilities

- Decide using the Bayes' rule:

$$\boxed{P(\omega_j / x) = \frac{p(x/\omega_j)P(\omega_j)}{p(x)} = \frac{likelihood \ \times \ prior}{evidence}}$$

where $p(x) = \sum_{j=1}^{2} p(x/\omega_j)P(\omega_j)$ (i.e., scale factor – ensures probs sum to 1)

Decide ω_1 if $P(\omega_1/x) > P(\omega_2/x)$; otherwise **decide** ω_2

or

Decide ω_1 if $p(x/\omega_1)P(\omega_1) > p(x/\omega_2)P(\omega_2)$; otherwise **decide** ω_2

or

Decide ω_1 if $\underbrace{p(x/\omega_1)/p(x/\omega_2)}_{\text{likelihood ratio}} > \underbrace{P(\omega_2)/P(\omega_1)}_{\text{threshold}}$; otherwise **decide** ω_2

Decision Rule Using Conditional Probabilities (cont'd)

$$P(\omega_1) = \frac{2}{3} \qquad P(\omega_2) = \frac{1}{3}$$

$p(x/\omega_j)$

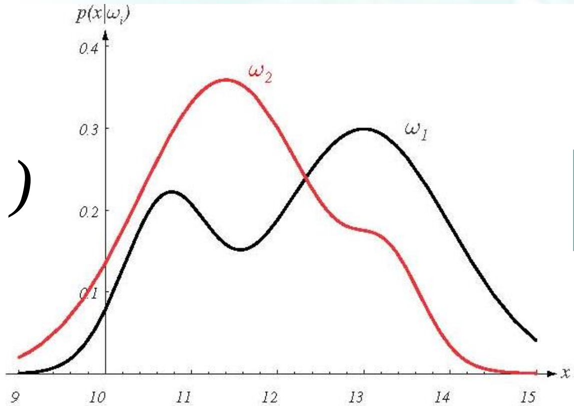


FIGURE 2.1. Hypothetical class-conditional probability density functions show the probability density of measuring a particular feature value x given the pattern is in category ω_i. If x represents the lightness of a fish, the two curves might describe the difference in lightness of populations of two types of fish. Density functions are normalized, and thus the area under each curve is 1.0. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons,

$P(\omega_j/x)$

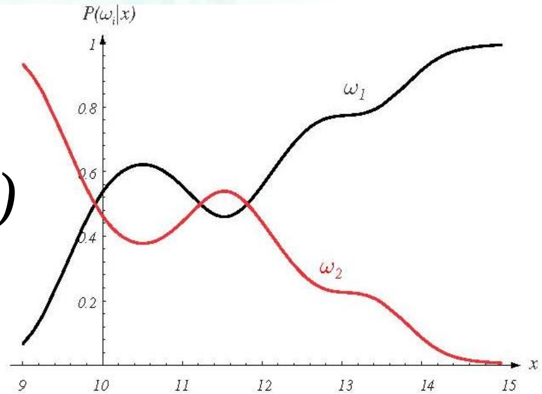


FIGURE 2.2. Posterior probabilities for the particular priors $P(\omega_1) = 2/3$ and $P(\omega_2) = 1/3$ for the class-conditional probability densities shown in Fig. 2.1. Thus in this case, given that a pattern is measured to have feature value $x = 14$, the probability it is in category ω_2 is roughly 0.08, and that it is in ω_1 is 0.92. At every x, the posteriors sum to 1.0. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Probability of Error

- What is the probability of error ?

$$P(error/x) = \begin{cases} P(\omega_1/x) \text{ if we decide } \omega_2 \\ P(\omega_2/x) \text{ if we decide } \omega_1 \end{cases}$$

or $P(error/x) = min[P(\omega_1/x), P(\omega_2/x)]$

- What is the average probability error?

$$P(error) = \int_{-\infty}^{\infty} P(error, x)dx = \int_{-\infty}^{\infty} P(error/x)p(x)dx$$

- Bayes rule is an optimum classification rule (i.e., it minimizes the average probability error).
  - **Warning**: this is true only under the assumption that $p(x/\omega_i)$ and $P(\omega_i)$ have been modelled/estimated correctly!

How is $p(x/\omega_i)$ estimated?

- Two competitive approaches:
  - Using histograms
  - Using models
- Each approach has its strengths and weaknesses.

Example (using histograms)

- Classify cars into two classes:
  - Classes: C_1 if price > \$50K, C_2 if price <= \$50K
  - Feature: x, the height of a car

- Use the Bayes' rule to compute the posterior probabilities:

$$P(C_i / x) = \frac{p(x / C_i) P(C_i)}{p(x)}$$

- We need to estimate $p(x/C_1)$, $p(x/C_2)$, $P(C_1)$, $P(C_2)$

Example (using histograms) **(cont'd)**

- Collect data
  - Ask drivers how much their car was and measure height.

- Determine prior probabilities $P(C_1)$, $P(C_2)$
  - e.g., 1209 samples: $\#C_1=221$ $\#C_2=988$

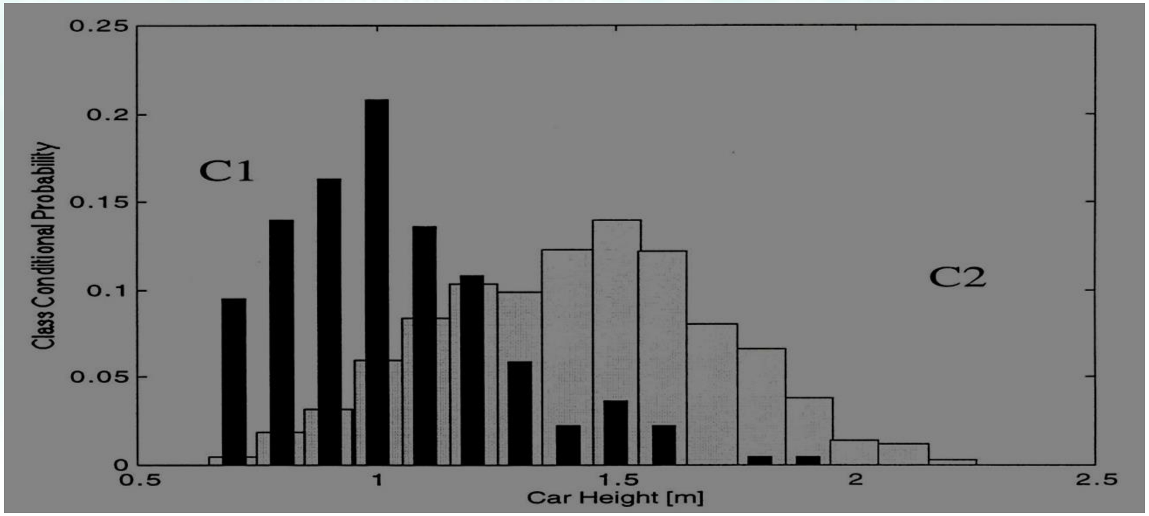
$$P(C_1) = \frac{221}{1209} = 0.183$$

$$P(C_2) = \frac{988}{1209} = 0.817$$

Example (using histograms) (cont'd)

- Determine class conditional probabilities (*likelihood*)
  - Discretize car height into bins and compute normalized histogram.

$$p(x / C_i)$$



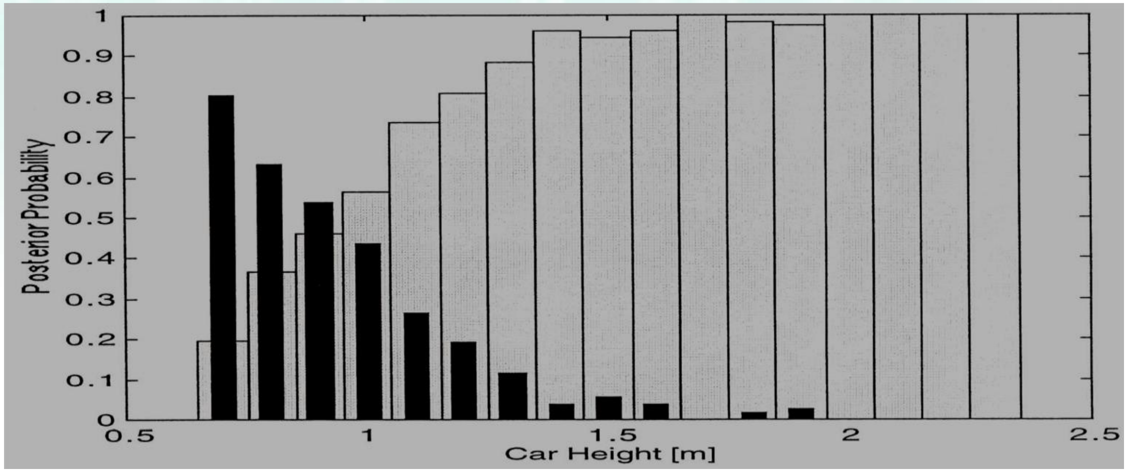
Example (using histograms) **(cont'd)**

- Calculate the posterior probability for each bin, e.g.:

$$P(C_1 / x = 1.0) = \frac{p(x = 1.0 / C_1)\,P(C_1)}{p(x = 1.0 / C_1)\,P(C_1) + p(x = 1.0 / C_2)\,P(C_2)} =$$

$$= \frac{0.2081 * 0.183}{0.2081 * 0.183 + 0.0597 * 0.817} = 0.438$$

$P(C_i / x)$



Example (using models)

Model each class using some pdf, e.g., a Gaussian (parametric)

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

$p(x/C_1)$ ~ $N(\mu_1,\sigma_1)$ μ_1,σ_1 are estimated from C_1 data

$p(x/C_2)$ ~ $N(\mu_2,\sigma_2)$ μ_2,σ_2 are estimated from C_2 data

Compute priors as before or maybe set $P(C_1) = P(C_2) = 0.5$

- Use Bayes rule to compute posterior probabilities:

$$P(C_i / x) = \frac{p(x / C_i) P(C_i)}{p(x)}$$

A More General Theory

- More than one features.
- More than two categories.
- Allow actions other than classification (e.g., rejection when classification is uncertain).
- Associate costs with different actions.
- Assume a more general error function (i.e., conditional risk) to perform classification using **probability** and **costs**.

Terminology

- Features form a vector $\mathbf{x} \in R^d$
- A set of *c* categories $\omega_1, \omega_2, \ldots, \omega_c$
- A finite set of *l* actions $\alpha_1, \alpha_2, \ldots, \alpha_l$ (typically $l \geq c$)
  - e.g., α_i: decide ω_i $(1 \leq i \leq c)$, α_{c+1}: reject
- A *loss* function $\lambda(\alpha_i / \omega_j) = \lambda_{ij}$
  - i.e., the cost associated with taking action α_i when the correct classification category is ω_j
- Conditional risk $R(\alpha_i / \mathbf{x})$ – expected loss of taking action α_i given $\mathbf{x}$

Classification will be performed by minimizing $R(\alpha_i / \mathbf{x})$ instead of maximizing $P(\omega_i / \mathbf{x})$

Conditional Risk $R(\alpha_i /\mathbf{x})$

- The conditional risk $R(\alpha_i /\mathbf{x})$ is defined as the expected loss of taking action α_i given **x**:

$$R(a_i / \mathbf{x}) = \sum_{j=1}^{c} \lambda(a_i / \omega_j) P(\omega_j / \mathbf{x})$$

where $$P(\omega_j / \mathbf{x}) = \frac{p(\mathbf{x} / \omega_j) P(\omega_j)}{p(\mathbf{x})}$$

Overall Risk

- The overall risk is the expected loss associated with $\alpha(\mathbf{x})$:

$$R = \int R(a(\mathbf{x}) / \mathbf{x}) \, p(\mathbf{x}) d\mathbf{x}$$

where $\alpha(\mathbf{x})$ is the decision rule which determines which action $\alpha_1, \alpha_2, \ldots, \alpha_l$ to take for any **x**.

- How should we minimize R?

Decision Rule Using Conditional Risk

- R can be **minimized** by minimizing $R(\alpha_i/\mathbf{x})$:

  (i) Computing $R(\alpha_i/\mathbf{x})$ for every α_i given an $\mathbf{x}$

  (ii) Choosing the action α_i with the **minimum** conditional risk $R(\alpha_i/\mathbf{x})$

- The resulting minimum R^* is called *Bayes risk* and is the best performance that can be achieved:

$$R^* = \min R$$

Example: Two-category classification

- Define
  - α_1: decide ω_1
  - $\alpha_{2:}$ decide ω_2
  - $\lambda_{ij} = \lambda(\alpha_i / \omega_j)$ (e.g., $\lambda_{11} = \lambda_{22} = 0$, $\lambda_{12} = 10$, $\lambda_{21} = 2$)

- The conditional risk associated with each action is:

$$R(a_i / \mathbf{x}) = \sum_{j=1}^{c} \lambda(a_i / \omega_j) P(\omega_j / \mathbf{x})$$

$$R(a_1/\mathbf{x}) = \lambda_{11} P(\omega_1/\mathbf{x}) + \lambda_{12} P(\omega_2/\mathbf{x})$$
$$R(a_2/\mathbf{x}) = \lambda_{21} P(\omega_1/\mathbf{x}) + \lambda_{22} P(\omega_2/\mathbf{x})$$

Example: Two-category classification (cont'd)

- Minimum risk decision rule:

Decide ω_1 if $R(a_1/\mathbf{x}) < R(a_2/\mathbf{x})$; otherwise decide ω_2

or Decide ω_1 if $(\lambda_{21} - \lambda_{11})P(\omega_1/\mathbf{x}) > (\lambda_{12} - \lambda_{22})P(\omega_2/\mathbf{x})$; otherwise decide ω_2

or Decide ω_1 if $\underbrace{\frac{p(\mathbf{x}/\omega_1)}{p(\mathbf{x}/\omega_2)}}_{\text{likelihood ratio}} > \underbrace{\frac{(\lambda_{12} - \lambda_{22})}{(\lambda_{21} - \lambda_{11})} \frac{P(\omega_2)}{P(\omega_1)}}_{\text{threshold}}$; otherwise decide ω_2

Special Case:
Zero-One Loss Function

- Assign the same loss (cost) to all errors:

$$\lambda(a_i/\omega_j) = \begin{cases} 0 & i = j \\ 1 & i \neq j \end{cases}$$

- The conditional risk is given by:

$$R(a_i/\mathbf{x}) = \sum_{j=1}^{c} \lambda(a_i/\omega_j) P(\omega_j/\mathbf{x}) = \sum_{i \neq j} P(\omega_j/\mathbf{x}) = 1 - P(\omega_i/\mathbf{x})$$

Special Case: Zero-One Loss Function (cont'd)

- In this case, the decision rule becomes:

$$\textbf{Decide } \omega_1 \text{ if } R(a_1/\mathbf{x}) < R(a_2/\mathbf{x})\text{; otherwise decide } \omega_2$$

$$R(a_i/\mathbf{x}) = 1 - P(\omega_i/\mathbf{x})$$

or $$\textbf{Decide } \omega_1 \text{ if } 1 - P(\omega_1/\mathbf{x}) < 1 - P(\omega_2/\mathbf{x})\text{; otherwise decide } \omega_2$$

or $$\textbf{Decide } \omega_1 \text{ if } P(\omega_1/\mathbf{x}) > P(\omega_2/\mathbf{x})\text{; otherwise decide } \omega_2$$

Same as in the case with no costs!

- The overall risk in this case is the average probability error which is minimized by the Bayes rule!

Effect of using a loss function

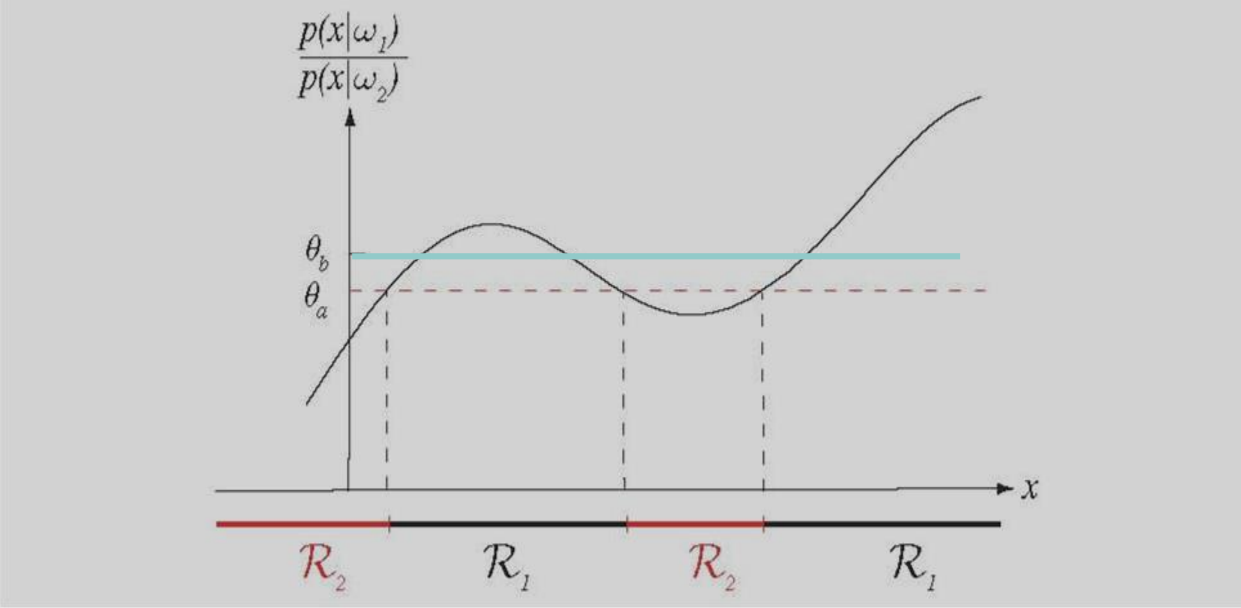
Assuming a zero-one loss function λ_{ij} :

Decide ω_1 if $p(x/\omega_1)/p(x/\omega_2) > P(\omega_2)/P(\omega_1)$ otherwise **decide** ω_2

$$\theta_a = P(\omega_2)/P(\omega_1)$$

Assuming a general loss function λ_{ij} :

Decide ω_1 if $\dfrac{p(\mathbf{x}/\omega_1)}{p(\mathbf{x}/\omega_2)} > \dfrac{(\lambda_{12}-\lambda_{22})}{(\lambda_{21}-\lambda_{11})}\dfrac{P(\omega_2)}{P(\omega_1)}$; otherwise decide ω_2



(decision regions)

$$\theta_b = \frac{P(\omega_2)(\lambda_{12}-\lambda_{22})}{P(\omega_1)(\lambda_{21}-\lambda_{11})}$$

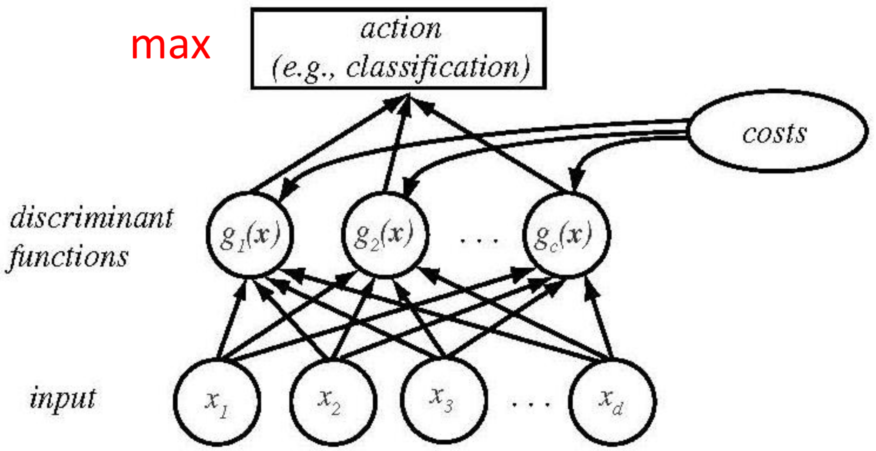
Discriminant Functions

- A classifier can also be represented by a set of discriminant functions, one for each class:

$$g_i(\mathbf{x}),\ i = 1, \ldots, c$$

- An input **x** is assigned to class ω_i if:

$$g_i(\mathbf{x}) > g_j(\mathbf{x}) \text{ for all } j \neq i$$

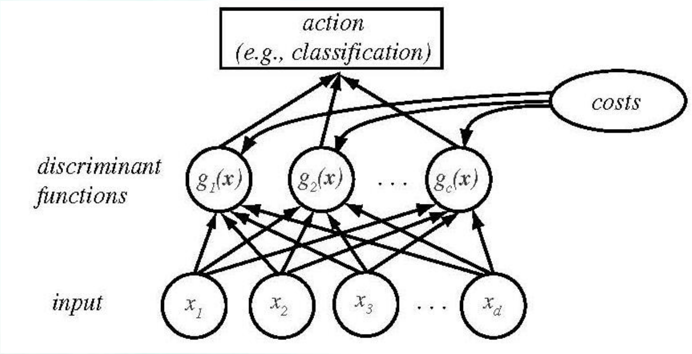


Examples of Discriminants

- Assuming a zero-one loss function:

$$g_i(\mathbf{x}) = P(\omega_i / \mathbf{x})$$

$$g_i(\mathbf{x}) = p(\mathbf{x} / \omega_i)P(\omega_i)$$



- Assuming a general loss function:

$$g_i(\mathbf{x}) = -R(\alpha_i / \mathbf{x})$$

Examples of Discriminants (cont'd)

- Replacing $g_i(\mathbf{x})$ with $f(g_i(\mathbf{x}))$, where $f()$ is monotonically increasing, will yield the same classification results!

$$g_i(\mathbf{x}) = p(\mathbf{x}/\omega_i)P(\omega_i)$$

take ln()

$$\boxed{g_i(\mathbf{x}) = \ln p(\mathbf{x}/\omega_i) + \ln P(\omega_i)}$$

We'll use this formulation extensively!

Case of two categories

- More common to use a single discriminant function (*dichotomizer*，二分器) instead of two:

$$g(\mathbf{x}) = g_1(\mathbf{x}) - g_2(\mathbf{x})$$

Decide ω_1 if $g(\mathbf{x}) > 0$; otherwise decide ω_2

<u>Examples</u>:

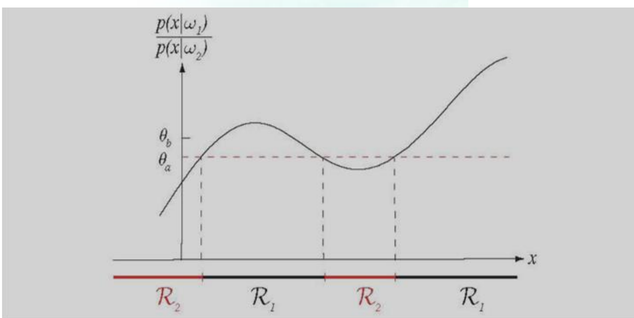
$$g(\mathbf{x}) = P(\omega_1 / \mathbf{x}) - P(\omega_2 / \mathbf{x})$$

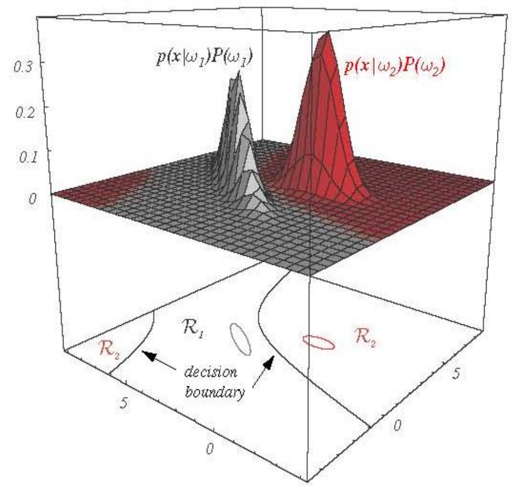
$$g(\mathbf{x}) = [\ln p(\mathbf{x} / \omega_1) + \ln P(\omega_1)] - [\ln p(\mathbf{x} / \omega_2) + \ln P(\omega_2)]$$

$$g(\mathbf{x}) = \ln \frac{p(\mathbf{x} / \omega_1)}{p(\mathbf{x} / \omega_2)} + \ln \frac{P(\omega_1)}{P(\omega_2)}$$

Decision Regions and Boundaries

- Discriminants divide the feature space into *decision regions* R_1, $R_{2, \ldots,}$ R_c, separated by *decision boundaries*.





How is the decision boundary defined?

$g_1(\mathbf{x})=g_2(\mathbf{x})$

- Next, let's examine the form of discriminants (and corresponding decision boundaries) when $p(\mathbf{x}/\omega_i)$ is modelled by a multivariate Gaussian density!

Log Refresher

Logarithmic Properties	
Product Rule	$\log_a(xy) = \log_a x + \log_a y$
Quotient Rule	$\log_a\left(\frac{x}{y}\right) = \log_a x - \log_a y$
Power Rule	$\log_a x^p = p\log_a x$
Change of Base Rule	$\log_a x = \frac{\log_b x}{\log_b a}$
Equality Rule	If $\log_a x = \log_a y$ then $x = y$

Discriminant Functions assuming a Multivariate Gaussian Density

- Let's consider the following discriminant function:

$$g_i(\mathbf{x}) = \ln p(\mathbf{x}/\omega_i) + \ln P(\omega_i) \qquad i = 1, \ldots, c$$

assuming that $p(\mathbf{x}/\omega_i) \sim N(\mu_i, \Sigma_i)$

$$N(\mu,\Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} exp[-\frac{1}{2}(\mathbf{x}-\mu)^t\Sigma^{-1}(\mathbf{x}-\mu)] \quad \mathbf{x} \in R^d$$

- In this case, the discriminant can be expressed as:

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\mu_\mathbf{i})^t\Sigma_i^{-1}(\mathbf{x}-\mu_\mathbf{i}) - \frac{d}{2}ln2\pi - \frac{1}{2}\ln|\Sigma_i| + \ln P(\omega_i)$$

Discriminant Function assuming Multivariate Gaussian Density (cont'd)

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\mu_{\mathbf{i}})^t \Sigma_i^{-1}(\mathbf{x}-\mu_{\mathbf{i}}) - \frac{d}{2} ln 2\pi - \frac{1}{2}\ln|\Sigma_i| + \ln P(\omega_i)$$

- The complexity of $g_i(\mathbf{x})$ depends on Σ_i which has $d(d+1)/2$ parameters in general (μ_i has d parameters).

- We will consider three different cases to better understand **simple** vs **complex** models:
  - Case 1: $\boldsymbol{\Sigma_i=\sigma^2 I}$ for each ω_i (one parameter total)
  - Case 2: $\boldsymbol{\Sigma_i= \Sigma}$ for each ω_i ($d(d+1)/2$ parameters total)
  - Case 3: $\boldsymbol{\Sigma_i}$**= arbitrary** for each ω_i ($cd(d+1)/2$ parameters total)

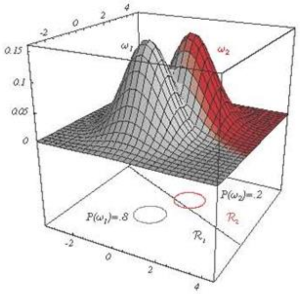
Case I

- $\Sigma_i = \sigma^2 I$ (each class is modeled by the same cov. matrix, diagonal with equal values)
  - Features are uncorrelated with the same variance.
  - Clusters have a spherical shape and the same size (centered at μ_i)
  - How could the discriminant be simplified in this case?

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \mu_\mathbf{i})^t \Sigma_i^{-1}(\mathbf{x} - \mu_\mathbf{i}) - \frac{d}{2} ln 2\pi - \frac{1}{2}\ln|\Sigma_i| + \ln P(\omega_i)$$

- If we disregard $\frac{d}{2} ln 2\pi$ and $\frac{1}{2}\ln|\Sigma_i|$ (constants):

$$\boxed{g_i(\mathbf{x}) = -\frac{\|\mathbf{x} - \mu_i\|^2}{2\sigma^2} + \ln P(\omega_i)}$$



where $\|\mathbf{x} - \mu_i\|^2 = (\mathbf{x} - \mu_\mathbf{i})^t(\mathbf{x} - \mu_\mathbf{i})$

  - This is a linear discriminant, let's see why!

Case I (cont'd)

$$g_i(\mathbf{x}) = -\frac{\|\mathbf{x} - \mu_i\|^2}{2\sigma^2} + \ln P(\omega_i)$$

- Expanding the above expression:

$$g_i(\mathbf{x}) = -\frac{1}{2\sigma^2}[\mathbf{x}^t\mathbf{x} - 2\mu_i^t\mathbf{x} + \mu_i^t\mu_i] + \ln P(\omega_i)$$

- Disregarding $\mathbf{x}^t\mathbf{x}$ (constant), we get a $\boxed{\text{linear discriminant:}}$

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

where $\mathbf{w}_i = \frac{1}{\sigma^2}\mu_i$, and $w_{i0} = -\frac{1}{2\sigma^2}\mu_i^t\mu_i + \ln P(\omega_i)$

- What is the form of the decision boundary in this case?

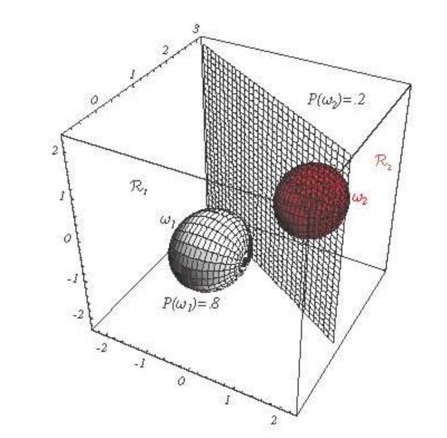
Let's set $g_1(\mathbf{x}) = g_2(\mathbf{x})$

Case I (cont'd)

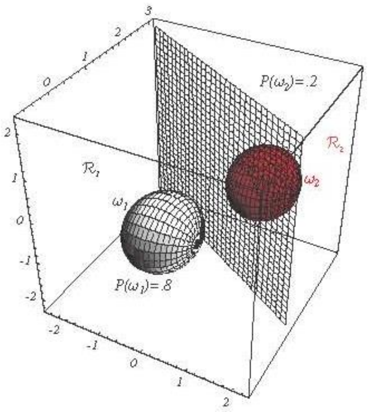
- Decision boundary is determined by $\boxed{\text{hyperplanes;}}$ setting $g_i(\mathbf{x}) = g_j(\mathbf{x})$:

$$\boxed{\mathbf{w}^t(\mathbf{x} - \mathbf{x_0}) = 0}$$

where $\mathbf{w} = \mu_i - \mu_j$, and $\mathbf{x}_0 = \frac{1}{2}(\mu_i + \mu_j) - \frac{\sigma^2}{\|\mu_i - \mu_j\|^2} \ln \frac{P(\omega_i)}{P(\omega_j)}(\mu_i - \mu_j)$



Case I (cont'd)

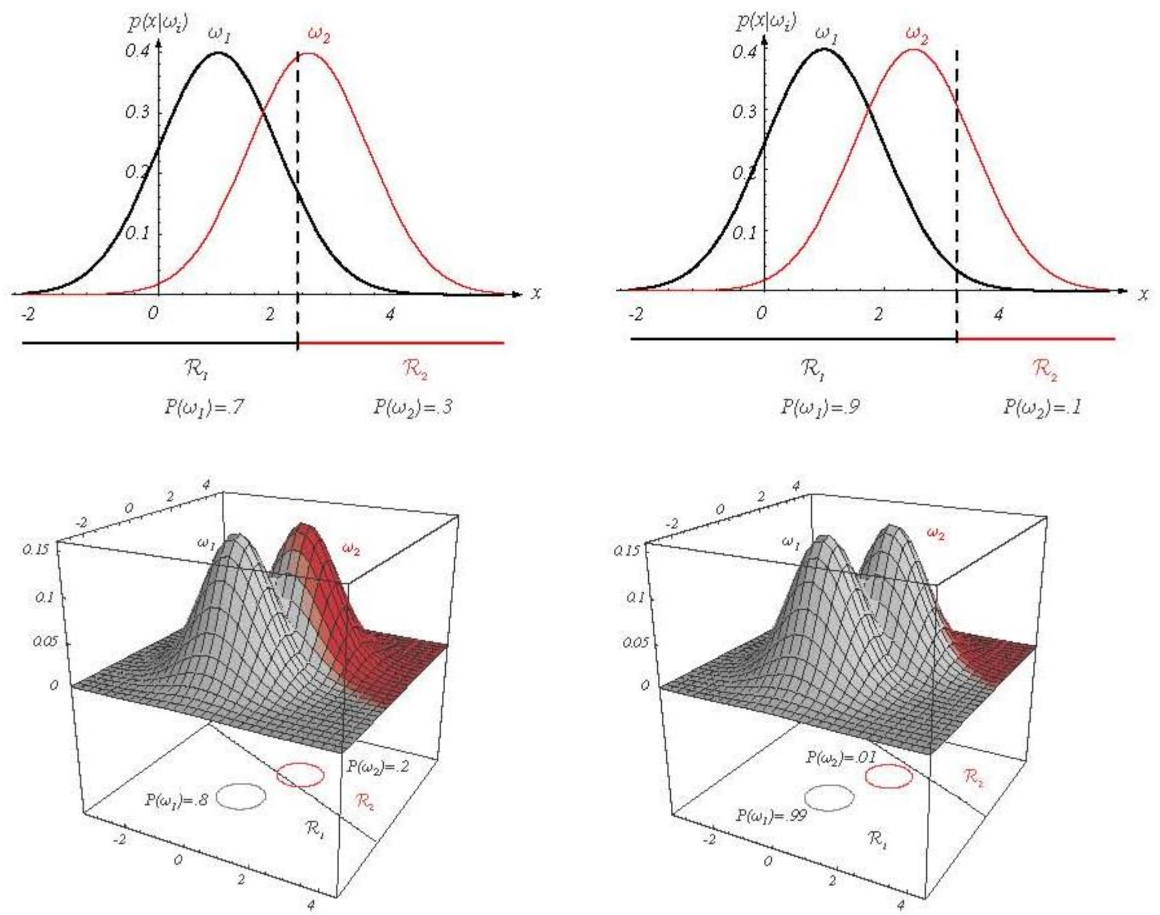


$$\mathbf{w}^t(\mathbf{x}-\mathbf{x_0})=0$$

$$\text{where } \mathbf{w} = \mu_i - \mu_j, \text{ and } \mathbf{x}_0 = \frac{1}{2}(\mu_i + \mu_j) - \frac{\sigma^2}{\|\mu_i - \mu_j\|^2} \ln \frac{P(\omega_i)}{P(\omega_j)}(\mu_i - \mu_j)$$

- Properties of decision boundary:
  - It passes through $\mathbf{x_0}$
  - It is **orthogonal** to the line connecting the two means.
  - What happens if σ is very small? x_0 is insensitive to $P(\omega_i)$ and $P(\omega_j)$
  - What happens when $P(\omega_i) = P(\omega_j)$? $\mathbf{x}_0 = \frac{1}{2}(\mu_i + \mu_j)$
  - What happens if $P(\omega_i) \neq P(\omega_j)$? $\mathbf{x_0}$ shifts away from the most likely category!

Case I (cont'd)



If $P(\omega_i) \neq P(\omega_j)$, then $\mathbf{x_0}$ shifts away from the most likely category.

Case I (cont'd)

- When $P(\omega_i)$ are all *equal*, then the discriminant can be further simplified:

$$g_i(\mathbf{x}) = -\frac{\|\mathbf{x} - \mu_i\|^2}{2\sigma^2} + \ln P(\omega_i) \Rightarrow \boxed{g_i(\mathbf{x}) = -\|\mathbf{x} - \boldsymbol{\mu}_i\|^2}$$

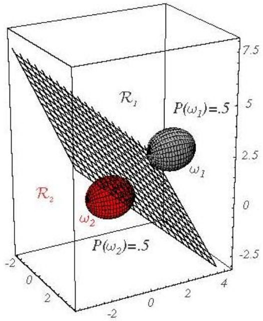
Euclidean distance

- This is known as the Euclidean distance classifier.

Case II

- $\Sigma_i = \Sigma$ (<u>each class </u>is modeled by the same cov. matrix, not necessarily diagonal)
  - Clusters are hyper ellipsoidal with same size (centered at μ_i)
  - How could the discriminant be simplified in this case?

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\mu_\mathbf{i})^t\Sigma_i^{-1}(\mathbf{x}-\mu_\mathbf{i}) - \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln|\Sigma_i| + \ln P(\omega_i)$$



- If we disregard $\frac{d}{2}\ln 2\pi$ and $\frac{1}{2}\ln|\Sigma_i|$ (constants):

$$\boxed{g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\mu_\mathbf{i})^t\Sigma^{-1}(\mathbf{x}-\mu_\mathbf{i}) + \ln P(\omega_i)}$$

  - This is also a linear discriminant, let's see why!

Case II (cont'd)

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x} - \mu_\mathbf{i})^t \Sigma^{-1}(\mathbf{x} - \mu_\mathbf{i}) + \ln P(\omega_i)$$

- Expanding the above expression and disregarding the quadratic term:

$$\boxed{\begin{array}{c} g_i(\mathbf{x}) = \mathbf{w}_i^t \mathbf{x} + w_{i0} \\ (\textit{linear discriminant}) \end{array}}$$

where $\mathbf{w}_i = \Sigma^{-1}\mu_i$, and $w_{i0} = -\frac{1}{2}\mu_i^t \Sigma^{-1}\mu_i + \ln P(\omega_i)$

- What is the form of the decision boundary in this case?

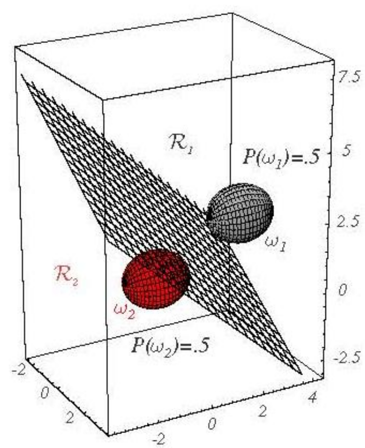
Let's set $g_1(\mathbf{x})=g_2(\mathbf{x})$

Case II (cont'd)

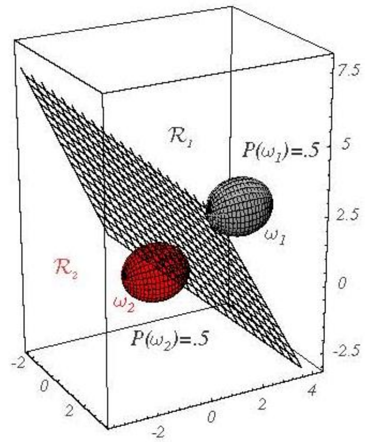
- Decision boundary is determined by hyperplanes; setting $g_i(\mathbf{x}) = g_j(\mathbf{x})$:

$$\boxed{\mathbf{w}^t(\mathbf{x} - \mathbf{x_0}) = 0}$$

where $\mathbf{w} = \Sigma^{-1}(\mu_i - \mu_j)$ and $\mathbf{x}_0 = \frac{1}{2}(\mu_i + \mu_j) - \frac{ln[P(\omega_i)/P(\omega_j)]}{(\mu_i - \mu_j)^t \Sigma^{-1}(\mu_i - \mu_j)}(\mu_i - \mu_j)$



Case II (cont'd)

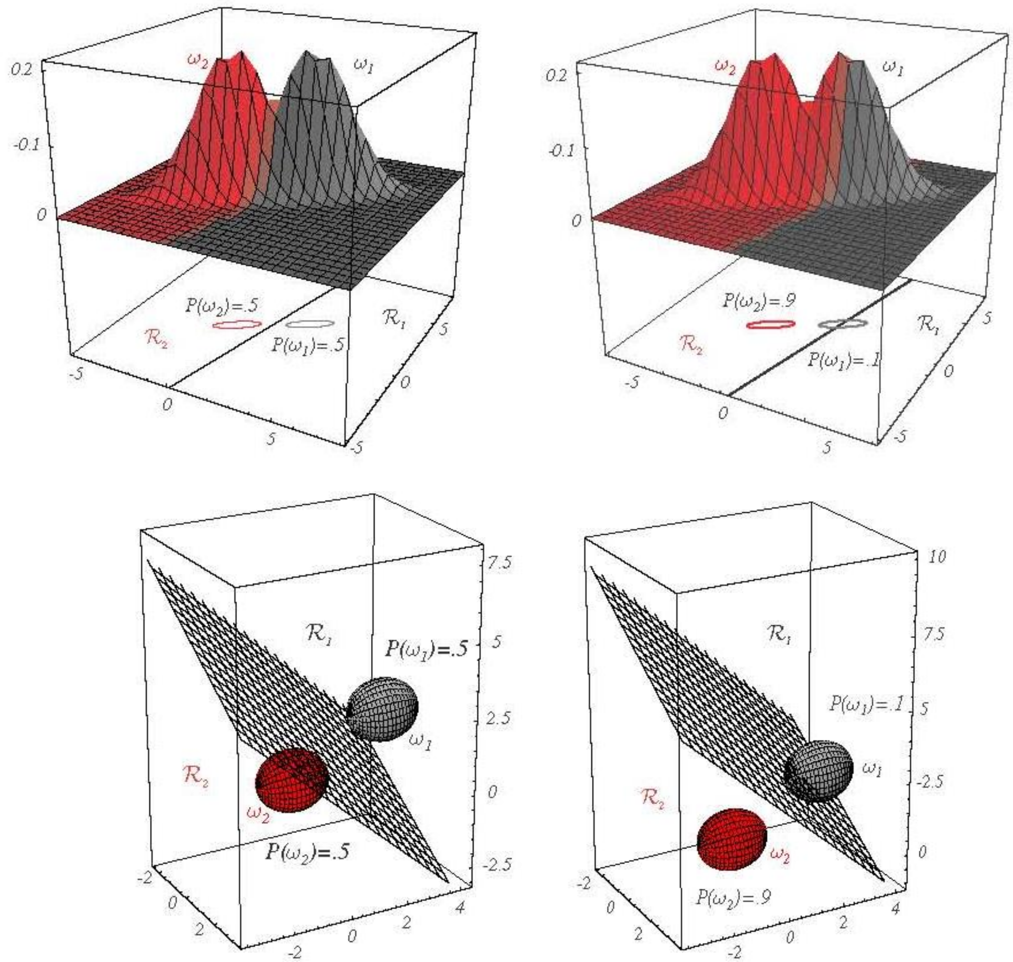


$$\mathbf{w}^t(\mathbf{x}-\mathbf{x_0})=0$$

$$\text{where } \mathbf{w}=\Sigma^{-1}(\mu_i-\mu_j) \text{ and } \mathbf{x}_0=\frac{1}{2}(\mu_i+\mu_j)-\frac{ln[P(\omega_i)/P(\omega_j)]}{(\mu_i-\mu_j)^t\Sigma^{-1}(\mu_i-\mu_j)}(\mu_i-\mu_j)$$

- Properties of hyperplane (decision boundary):
  - It passes through $\mathbf{x_0}$
  - It is not orthogonal to the line connecting the two means.
  - What happens when $P(\omega_i) = P(\omega_j)$? $\mathbf{x}_0=\frac{1}{2}(\mu_i+\mu_j)$
  - What happens if $P(\omega_i) \neq P(\omega_j)$? $\mathbf{x_0}$ shifts away from the most likely category.

Case II (cont'd)



If $P(\omega_i) \neq P(\omega_j)$, then $\mathbf{x_0}$ shifts away from the most likely category.

Case II (cont'd)

- When $P(\omega_i)$ are all equal, the discriminant can be further simplified:

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\mu_\mathbf{i})^t \Sigma^{-1}(\mathbf{x}-\mu_\mathbf{i}) + \ln P(\omega_i)$$

$$\Downarrow$$

$$\boxed{g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\mu_\mathbf{i})^t \Sigma^{-1}(\mathbf{x}-\mu_\mathbf{i})}$$

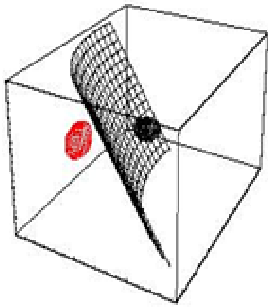
Mahalanobis distance

- This is known as the Mahalanobis distance classifier.

Case III

- Σ_i= arbitrary (each class has its own covariance matrix)
  - Clusters have different shapes and sizes (centered at μ_i)
  - How could the discriminant be simplified in this case?

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\mu_i)^t\Sigma_i^{-1}(\mathbf{x}-\mu_i) - \frac{d}{2}\,ln2\pi - \frac{1}{2}\ln|\Sigma_i| + \ln P(\omega_i)$$



- If we disregard $\frac{d}{2}\,ln2\pi$ (constant):

$$\boxed{g_i(x) = x^t W_i x + w_i^t x + w_{i0}}$$

(quadratic discriminant)

where $\mathbf{W}_i = -\frac{1}{2}\Sigma_i^{-1}$, $\mathbf{w}_i = \Sigma_i^{-1}\mu_i$, and $w_{i0} = -\frac{1}{2}\mu_i^t\Sigma_i^{-1}\mu_i - \frac{1}{2}\,ln|\Sigma_i| + \ln P(\omega_i)$

- What is the form of the decision boundary in this case?

Let's set $g_1(\mathbf{x})=g_2(\mathbf{x})$

Case III (cont'd)

超二次曲面

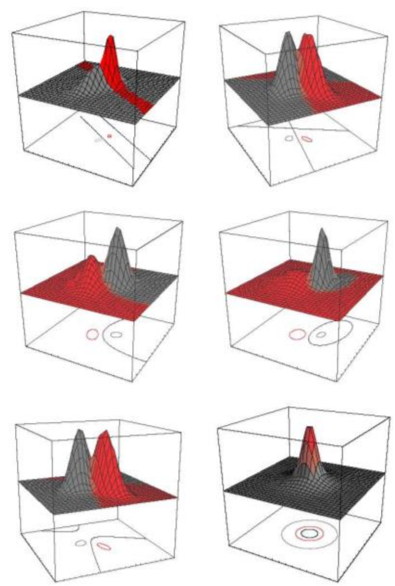
- Decision boundary is determined by hyperquadrics; setting $g_i(\mathbf{x}) = g_j(\mathbf{x})$

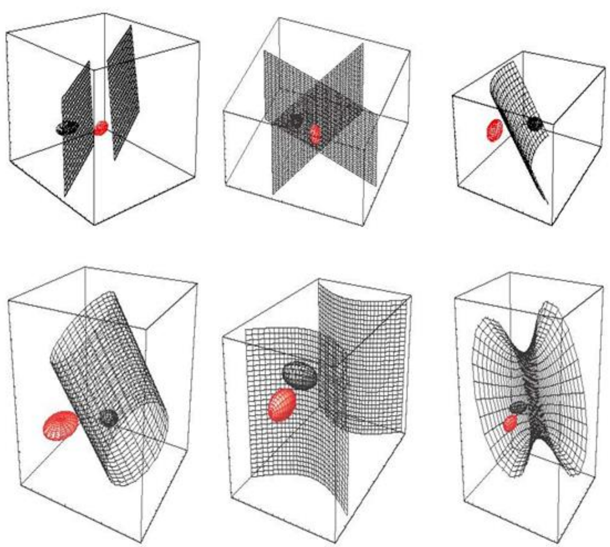
$$\mathbf{x}^t \mathbf{W}_1 \mathbf{x} + \mathbf{w}_1^t \mathbf{x} + w_{1,0} = \mathbf{x}^t \mathbf{W}_2 \mathbf{x} + \mathbf{w}_2^t \mathbf{x} + w_{2,0}$$

or

$$\mathbf{x}^t (\mathbf{W}_1 - \mathbf{W}_2)\mathbf{x} + (\mathbf{w}_1^t - \mathbf{w}_2^t)\mathbf{x} + (w_{1,0} - w_{2,0}) = 0$$

non-linear decision boundary





e.g., hyperplanes, hyperspheres, hyperellipsoids, hyperparaboloids etc.

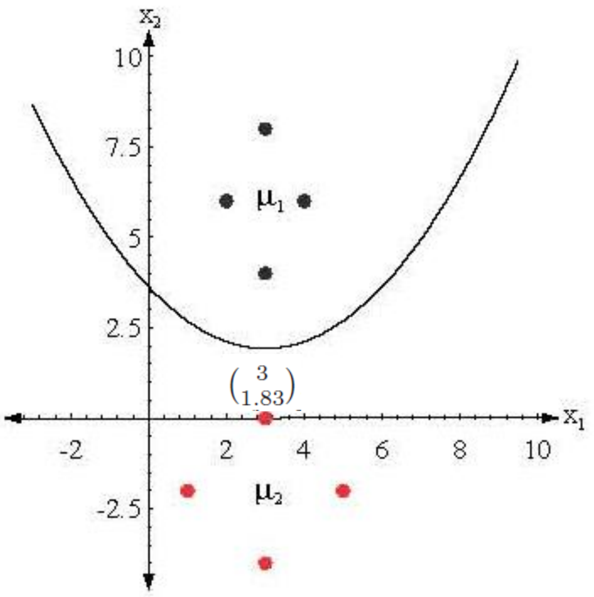
Example

Assume $P(\omega_1)=P(\omega_2)$

$$\boldsymbol{\mu}_1 = \begin{bmatrix} 3 \\ 6 \end{bmatrix}; \quad \Sigma_1 = \begin{pmatrix} 1/2 & 0 \\ 0 & 2 \end{pmatrix} \text{ and } \boldsymbol{\mu}_2 = \begin{bmatrix} 3 \\ -2 \end{bmatrix}; \quad \Sigma_2 = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}.$$

What case is this? Case III

$$x_2 = 3.514 - 1.125x_1 + 0.1875x_1^2.$$



Note that the decision boundary does not pass through the midpoint of μ_1,μ_2

Error Bounds

- Exact error calculations could be difficult – it is easier to estimate error bounds.

$$P(error) = \int P(error, \mathbf{x})\, d\mathbf{x} = \int P(error/\mathbf{x}) p(\mathbf{x})\, d\mathbf{x}$$

$$P(error/\mathbf{x}) = \begin{cases} P(\omega_1/\mathbf{x}) & \text{if we decide } \omega_2 \\ P(\omega_2/\mathbf{x}) & \text{if we decide } \omega_1 \end{cases}$$

or

min[P(ω_1/x), P(ω_2/x)]

$$P(error) = \int \min[P(\omega_1/\mathbf{x}), P(\omega_2/\mathbf{x})] p(\mathbf{x}) d\mathbf{x} =$$

$$\int \min[P(\omega_1/\mathbf{x}) p(\mathbf{x}), P(\omega_2/\mathbf{x}) p(\mathbf{x})] d\mathbf{x} =$$

$$\int \min[p(\mathbf{x}/\omega_1) P(\omega_1), p(\mathbf{x}/\omega_2) P(\omega_2)] d\mathbf{x}$$

Error Bounds

- Using the inequality:

$$min[a, b] \leq a^{\beta} b^{1-\beta}, \quad a, b \geq 0, 0 \leq \beta \leq 1$$

$$P(error) = \int \min[p(\mathbf{x}/\omega_1)P(\omega_1), p(\mathbf{x}/\omega_2)P(\omega_2)]d\mathbf{x} \leq$$

$$P^{\beta}(\omega_1)P^{1-\beta}(\omega_2) \int p^{\beta}(\mathbf{x}/\omega_1)\ p^{1-\beta}(\mathbf{x}/\omega_2)d\mathbf{x}$$

Can we compute the following integral?

$$\int p^{\beta}(\mathbf{x}/\omega_1)\ p^{1-\beta}(\mathbf{x}/\omega_2)d\mathbf{x}$$

Error Bounds (cont'd)

- It can be shown that if $p(\mathbf{x}/\omega_i)$ is Gaussian, then:

$$\int p^{\beta}(\mathbf{x}/\omega_1)\, p^{1-\beta}(\mathbf{x}/\omega_2) d\mathbf{x} = e^{-\kappa(\beta)}$$

where
$$k(\beta) = \frac{\beta(1-\beta)}{2}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^t[\beta\boldsymbol{\Sigma}_1 + (1-\beta)\boldsymbol{\Sigma}_2]^{-1}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1) + \frac{1}{2}\ln\frac{|\beta\boldsymbol{\Sigma}_1 + (1-\beta)\boldsymbol{\Sigma}_2|}{|\boldsymbol{\Sigma}_1|^{\beta}|\boldsymbol{\Sigma}_2|^{1-\beta}}.$$

determinant

So: $$\boxed{P(error) \leq P^{\beta}(\omega_1)P^{1-\beta}(\omega_2)e^{-k(\beta)}}$$

Chernoff Error Bound

- Can be obtained by minimizing $P^{\beta}(\omega_1)P^{1-\beta}(\omega_2)e^{-k(\beta)}$
  – This is a 1-D optimization problem, regardless to the dimensionality of the class conditional densities $p(\mathbf{x}/\omega_i)$.

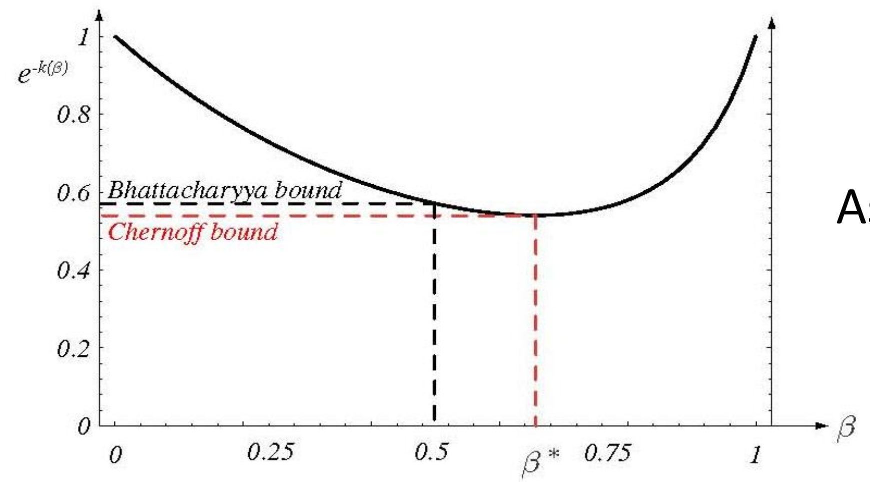


FIGURE 2.18. The Chernoff error bound is never looser than the Bhattacharyya bound. For this example, the Chernoff bound happens to be at $\beta^* = 0.66$, and is slightly tighter than the Bhattacharyya bound ($\beta = 0.5$). From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Assuming $P(\omega_1)=P(\omega_2)$

Bhattacharyya Error Bound

- Can be obtained by simply setting β=0.5
  - Easier to compute but typically looser.

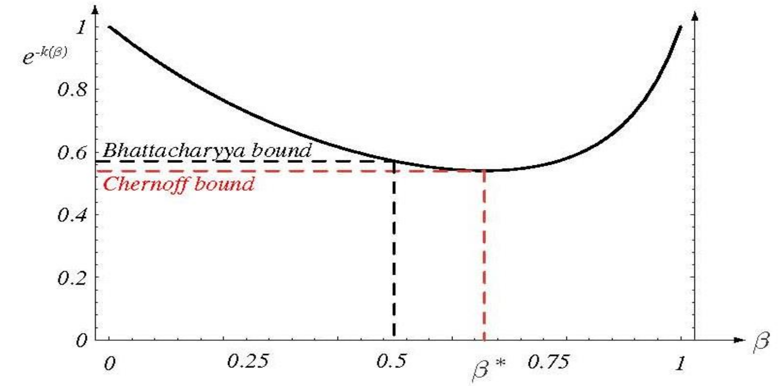


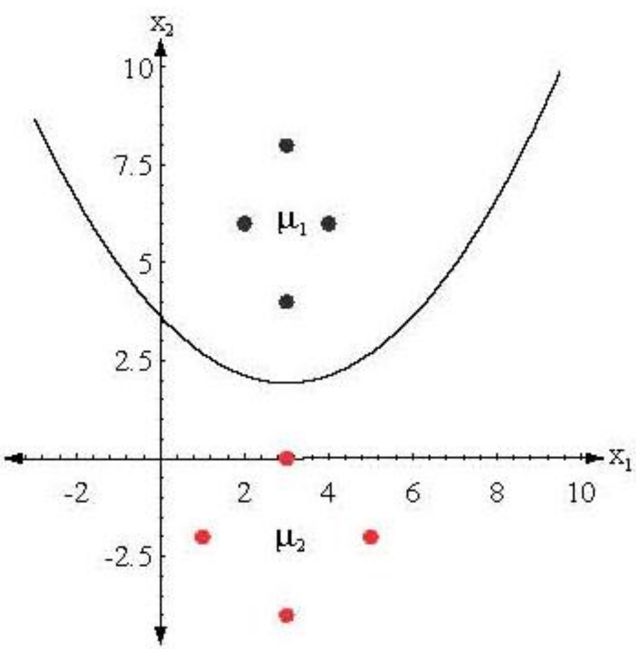
FIGURE 2.18. The Chernoff error bound is never looser than the Bhattacharyya bound. For this example, the Chernoff bound happens to be at $\beta^* = 0.66$, and is slightly tighter than the Bhattacharyya bound ($\beta = 0.5$). From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

- **Warning**: both bounds are reliable only if $p(\mathbf{x}/\omega_i)$ is Gaussian!

Example (cont'd)

$$k(\beta) = \frac{\beta(1-\beta)}{2}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^t[\beta\boldsymbol{\Sigma}_1 + (1-\beta)\boldsymbol{\Sigma}_2]^{-1}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1) + \frac{1}{2}\ln\frac{|\beta\boldsymbol{\Sigma}_1 + (1-\beta)\boldsymbol{\Sigma}_2|}{|\boldsymbol{\Sigma}_1|^{\beta}|\boldsymbol{\Sigma}_2|^{1-\beta}}.$$

$P(\omega_1)=P(\omega_2)=0.5$



$$\boldsymbol{\mu}_1 = \begin{bmatrix} 3 \\ 6 \end{bmatrix}; \quad \boldsymbol{\Sigma}_1 = \begin{pmatrix} 1/2 & 0 \\ 0 & 2 \end{pmatrix}$$

$$\boldsymbol{\mu}_2 = \begin{bmatrix} 3 \\ -2 \end{bmatrix}; \quad \boldsymbol{\Sigma}_2 = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}.$$

Bhattacharyya error:

$k(0.5)=4.06$

$$P(error) \le P^{\beta}(\omega_1)P^{1-\beta}(\omega_2)e^{-k(\beta)}$$

$$P(error) \le 0.0087$$

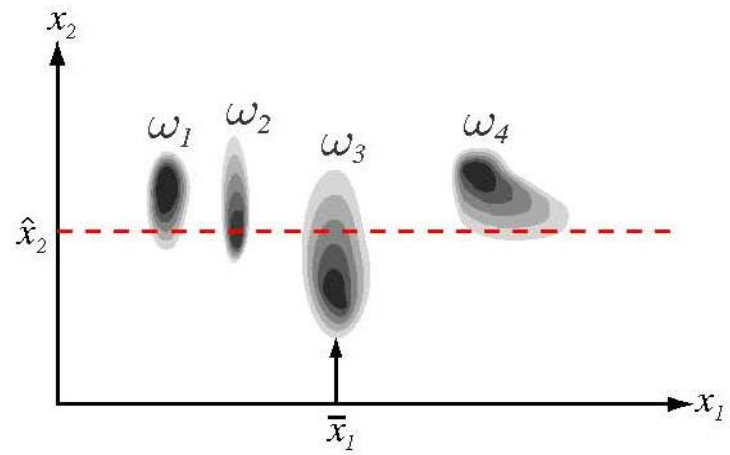
Bayes Decision Theory: Case of Discrete Features

- Replace $\int p(\mathbf{x}/\omega_j)d\mathbf{x}$ with $\sum_{\mathbf{x}} P(\mathbf{x}/\omega_j)$
- See section 2.9 for details

Missing Features

- Suppose **x**=(x_1,x_2) is a test vector where x_1 is missing and x_2 = $\hat{x}_2$; how should we classify it?

Example:



  - If we set x_1 equal to the average value, we will classify **x** as ω_3
  - But $p(\hat{x}_2 / \omega_2)$ is larger; should we classify **x** as ω_2 ?

Marginalize Posterior Probability

- Suppose **x**=[$\mathbf{x}_g$, $\mathbf{x}_b$] ($\mathbf{x}_g$: good features, $\mathbf{x}_b$: bad features)
- Compute posterior probability using good features only:

$$P(\omega_i/\mathbf{x}_g) = \frac{p(\omega_i, \mathbf{x}_g)}{p(\mathbf{x}_g)} = \frac{\int p(\omega_i, \mathbf{x}_g, \mathbf{x}_b) d\mathbf{x}_b}{p(\mathbf{x}_g)} =$$

$$\frac{\int P(\omega_i/\mathbf{x}_g, \mathbf{x}_b) p(\mathbf{x}_g, \mathbf{x}_b) d\mathbf{x}_b}{p(\mathbf{x}_g)} = \frac{\int P(\omega_i/\mathbf{x}_g, \mathbf{x}_b) p(\mathbf{x}) d\mathbf{x}_b}{\int p(\mathbf{x}) d\mathbf{x}_b}$$

Marginalize over "bad" (missing) features optimized by Expectation-Maximization (EM).

$$\boxed{\textbf{Decide } \omega_1 \text{ if } P(\omega_1/\mathbf{x}_g) > P(\omega_2/\mathbf{x}_g)\text{; otherwise decide } \omega_2}$$

Compound Bayesian Decision Theory

- Sequential decision

  (1) Decide as each pattern (e.g., fish) emerges.

- Compound decision

  (1) Wait for *n* patterns (e.g., fish) to emerge.

  (2) Make all *n* decisions jointly.

– Could improve performance when consecutive states of nature ($\omega(1)$, $\omega(2)$, …, $\omega(n)$) are **not** statistically independent!

Compound Bayesian Decision Theory (cont'd)

- $\mathbf{X}=(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ are **n** observed vectors.
- $\mathbf{\Omega}=(\omega(1), \omega(2), \ldots, \omega(n))$ denotes the **n** states of nature.
  - $\omega(i)$ can take one of c values $\omega_1, \omega_2, \ldots, \omega_c$
- $P(\mathbf{\Omega})$ is the prior probability of the **n** states
- $p(\mathbf{X}/\mathbf{\Omega})$ is the conditional probability density (likelihood).

Compound Bayesian Decision Theory (cont'd)

- We can compute P(**Ω/X**) using the Bayes Rule:

$$P(\mathbf{\Omega} / \mathbf{X}) = \frac{p(\mathbf{X} / \mathbf{\Omega}) P(\mathbf{\Omega})}{p(\mathbf{X})}$$

- The following assumption is not acceptable:

$$p(\mathbf{\Omega}) = \prod_{i=1}^{n} P(\omega(i))$$

i.e., consecutive states of nature may **not** be statistically independent!

  – Difficult to compute $p(\mathbf{\Omega})$ with c^n possible $\mathbf{\Omega}$
  – Possible solution: use Markov Model to speed up

- The following assumption might be acceptable:

$$p(\mathbf{X} / \mathbf{\Omega}) = \prod_{i=1}^{n} p(\mathbf{x}_i / \omega(i))$$

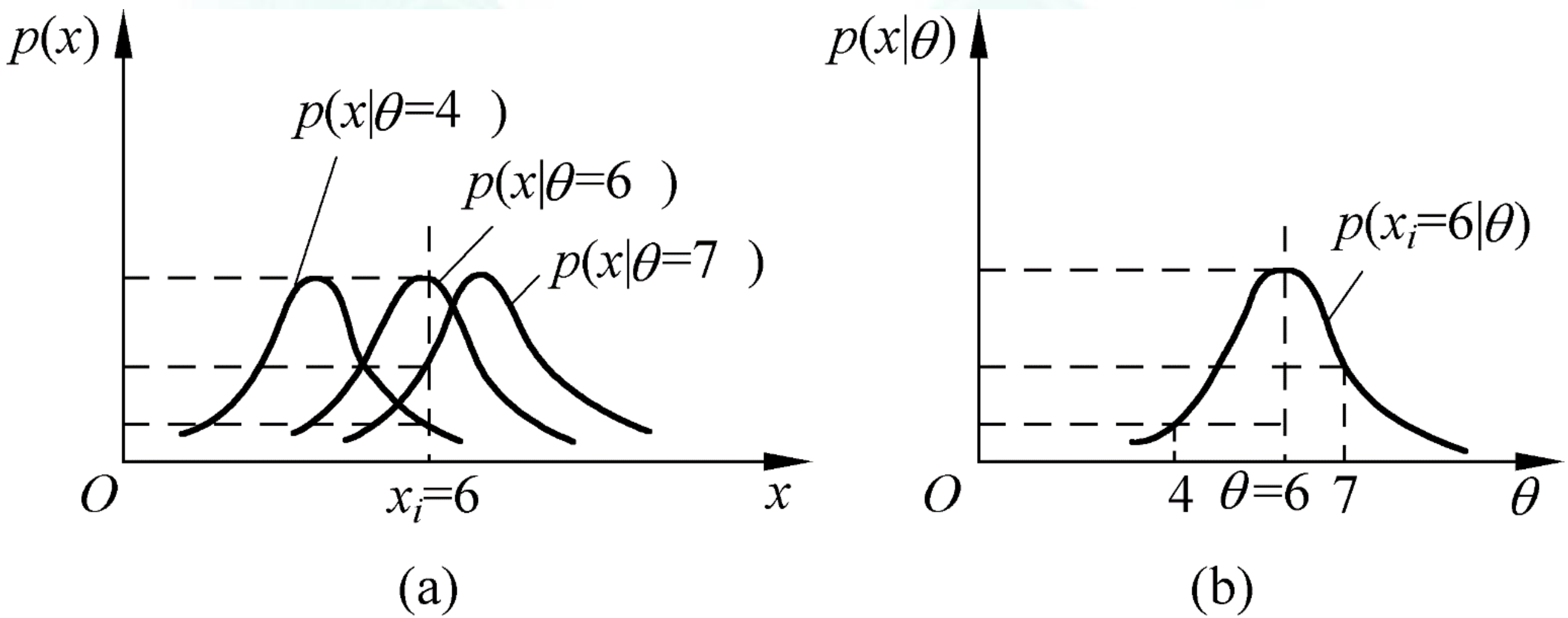
如何表示/估计概率密度

（吴建鑫《模式识别》第8章）

- <u>参数估计</u>
  - 点估计point estimation
  - 贝叶斯估计Bayesian estimation
- 非参数估计
  - 直方图估计
  - KDE

最大似然估计的基本思想

- 样本集最可能来自哪个参数



以高斯分布为例

- 假设$x \sim N(\mu, \sigma^2)$，从数据$D = \{x_1, \ldots, x_n\}$估计
  - 数据独立同分布i.i.d. (independently identically distributed)
- 参数记为$\boldsymbol{\theta}$，这里$\boldsymbol{\theta} = (\mu, \sigma)$，如何估计？形式化？
- 一种直觉：如果有两个不同的参数$\boldsymbol{\theta}_1$和$\boldsymbol{\theta}_2$
  - 假设$\boldsymbol{\theta}$是参数的真实值, 似然（likelihood）函数是
    $p(D|\boldsymbol{\theta}) = \prod_i p(x_i|\boldsymbol{\theta}) = \prod_i \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(x-\mu)^2}{2\sigma^2})$
  - 若$p(D|\boldsymbol{\theta}_1) > p(D|\boldsymbol{\theta}_2)$，该选择哪个？

易混淆的表示法notation

- 目前$\boldsymbol{\theta}$不是随机变量，所以$p(D|\boldsymbol{\theta})$不是条件分布
  - D固定，$\boldsymbol{\theta}$是变量，$p(D|\boldsymbol{\theta})$是$\boldsymbol{\theta}$的函数，不是一个PDF！
  - $p(x_i|\boldsymbol{\theta})$是一个PDF，因为$\boldsymbol{\theta}$不是随机变量，这不是一个条件分布，只是习惯上这么写，表明这个分布依赖于参数$\boldsymbol{\theta}$的值，x_i是PDF的变量
- 较好的表示法：定义似然函数likelihood function
  - $\ell(\boldsymbol{\theta}) = p(D|\boldsymbol{\theta}) = \prod_i p(x_i|\boldsymbol{\theta})$ （或者$\boldsymbol{x}_i$）
- 为了方便，定义对数似然函数log-likelihood function
  - $\ell\ell(\boldsymbol{\theta}) = \ln p(D|\boldsymbol{\theta}) = \sum_i \ln p(x_i|\boldsymbol{\theta})$

最大似然估计

- Maximum likelihood estimation, MLE

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, \ell\ell(\boldsymbol{\theta})$$

- 高斯分布的最大似然估计
  - 参数为$(\boldsymbol{\mu}, \Sigma)$，数据为$D = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_n\}$
  - 练习：通过对$\ell\ell(\boldsymbol{\theta})$求导发现最佳的参数值，可以查表

$$\boldsymbol{\mu}^* = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i$$

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

最大后验估计及其他

- Maximium a posteriori estimation, MAP
  - $\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, \ell(\boldsymbol{\theta})\, p(\boldsymbol{\theta})$
  - 将参数$\boldsymbol{\theta}$自身不同取值的可能性$p(\boldsymbol{\theta})$（参数的先验概率）考虑进来
- 与MLE的关系
  - 假设我们对$\boldsymbol{\theta}$一无所知，那么应该怎样设定$p(\boldsymbol{\theta})$？
  - noninformative prior时，MLE等价于MAP
    - 若$\boldsymbol{\theta}$是离散的随机变量，离散的均匀分布， $p(\theta) = \frac{1}{N}$
    - 若$\boldsymbol{\theta}$是有限区间$[a, b]$的连续随机变量，$p(\theta) = \frac{1}{b-a}$
    - 若$\boldsymbol{\theta}$是$(-\infty, +\infty)$上的连续随机变量，？
    - 假设$p(\theta) = \text{const}$，称为improper prior

参数估计的一些性质

- 样例越多，估计越准！
- 渐进性质asymptotic property：研究$n \to \infty$时的性质，如
  - 一致性consistency：随样本容量增大收敛到参数真值的估计量
- 其他性质如
  - 无偏估计unbiased estimate：指估计量的期望和被估计量的真值相等
- 进一步阅读：关于一致和无偏

贝叶斯参数估计

- 点估计point estimation
  - MLE：视$\boldsymbol{\theta}$为固定的参数，假设存在一个最佳的参数（或参数的真实值是存在的），目的是找到这个值
  - MAP：将$p(\boldsymbol{\theta})$的影响代入MLE中，仍然假设存在最优的参数
- 在贝叶斯观点中，$\boldsymbol{\theta}$是一个分布/随机变量，所以估计应该是估计一个分布，而不是一个值（点）！
  - $p(\boldsymbol{\theta}|D)$：这是贝叶斯参数估计的输出，是一个完整的分布，而不是一个点

高斯分布参数的贝叶斯估计

- 参数$\boldsymbol{\theta}$的先验分布$p(\boldsymbol{\theta})$，数据$D = \{x_1, \dots, x_n\}$，估计$p(\boldsymbol{\theta}|D)$。这里假设单变量，只估计μ，方差σ已知
  - 第一步：设定$p(\mu)$的参数形式：$p(\mu) = N(\mu_0, \sigma_0^2)$，目前假设参数$\mu_0, \sigma_0^2$已知
  - 第二步：贝叶斯定理和独立性得到$p(\mu|D) = \frac{p(D|\mu)p(\mu)}{\int p(D|\mu)p(\mu)d\mu} = \alpha p(D|\mu)p(\mu) = \alpha \prod_{i=1}^{n} p(x_i|\mu)p(\mu)$
  - 第三步，应用高斯分布的性质，进一步得到其解析形式
    - 注意这里所有$p(\cdot)$都是合法的密度函数

解的形式

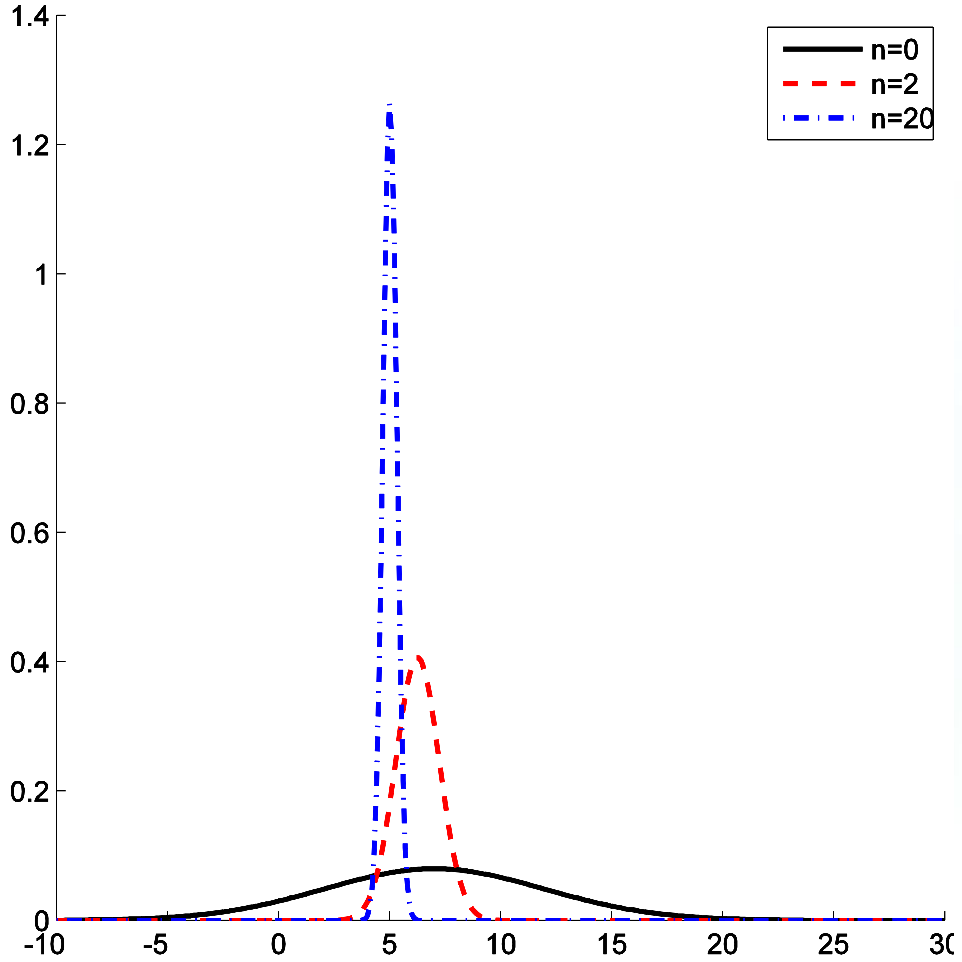
$$p(\mu|D) = N(\mu_n, \sigma_n^2)$$

- 均值为$\mu_n = \frac{\sigma^2}{n\sigma_0^2+\sigma^2}\mu_0 + \frac{n\sigma_0^2}{n\sigma_0^2+\sigma^2}\mu_{\mathrm{ML}}$
  - 其中μ_{ML}为MLE的估计值，即$\mu_{\mathrm{ML}} = \frac{1}{n}\sum_{i=1}^{n} x_i$
- 方差为σ_n^2，其值由如下公式确定：$\frac{1}{\sigma_n^2} = \frac{1}{\sigma_0^2} + \frac{n}{\sigma^2}$，或者为了便于记忆

$$(\sigma_n^2)^{-1} = (\sigma_0^2)^{-1} + n(\sigma^2)^{-1}$$

- 先验和数据的综合!

Bayes估计的例子



贝叶斯的进一步讨论

- 共轭先验conjugate prior
  - 若$p(\boldsymbol{x}|\boldsymbol{\theta})$，存在先验$p(\boldsymbol{\theta})$，使得$p(\boldsymbol{\theta}|D)$和$p(\boldsymbol{\theta})$有相同的函数形式，从而简化推导和计算
  - 如高斯分布的共轭先验分布仍然是高斯分布
- 优缺点：
  - 理论上非常完备，数学上很优美
  - 推导困难（怎样求任意分布的共轭？怎样用于决策？μ_0的prior）、计算量极大（需要很多积分）
  - 在数据较多时，学习效果常不如直接用discriminant function

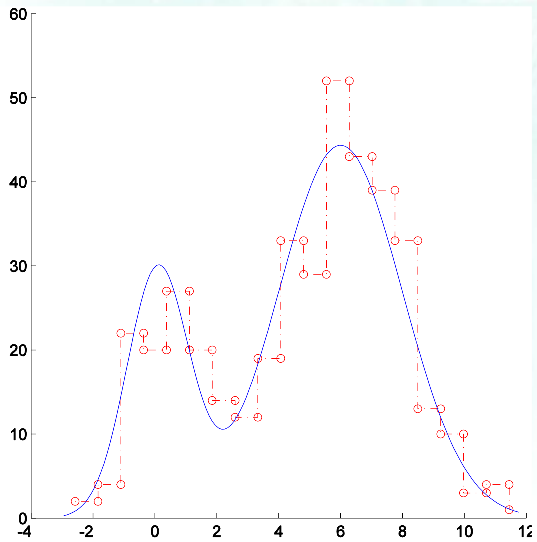
如何表示/估计概率密度

（吴建鑫《模式识别》第8章）

- 参数估计
  - 点估计point estimation
  - 贝叶斯估计Bayesian estimation
- <u>非参数估计</u>
  - 直方图估计
  - KDE

非参数估计

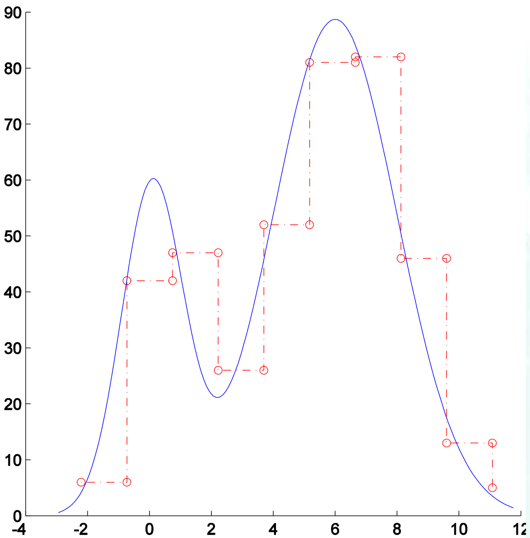
- 常用的参数形式基本都是单模single modal的，不足以描述复杂的数据分布：即应该直接以训练数据自身来估计分布
  - 例如直方图histogram，基于计数counting



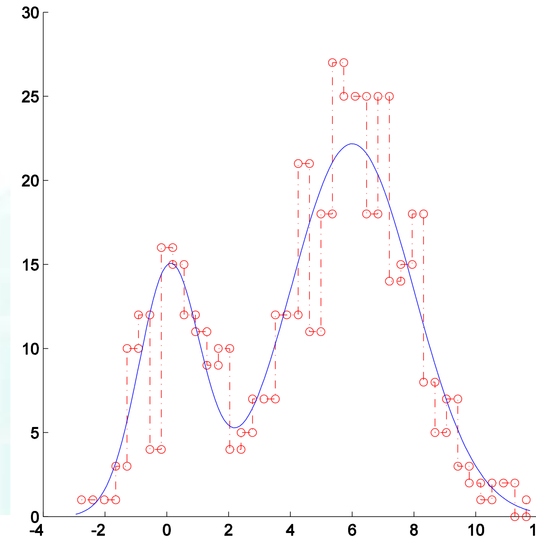
有很多问题：

- 多维怎么办？
- 怎么确定bin的个数？
- 连续？
- 需要保存数据吗？

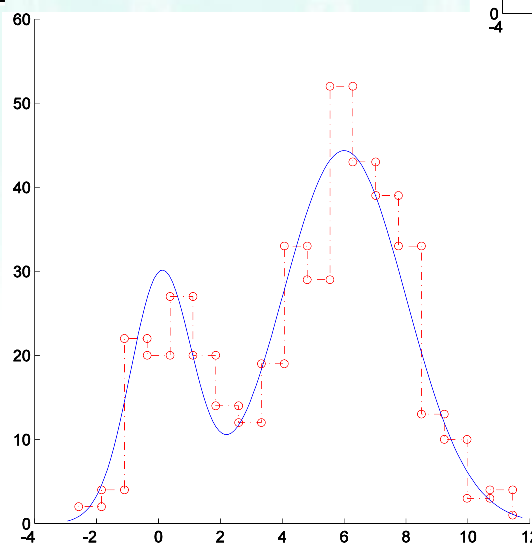
Bin个数（宽度）的影响



10个bin
（欠拟合）



40个bin
（过拟合）



20个bin

维数灾难

- Curse of dimensionality
  - 以直方图为例，需要保存的参数是什么？
  - 如果每维n个bin，那么d维应该保存多少个bin的参数？
  - 如果$n = 4, d = 100$，那么应该保存多少个bin的参数？
  - $4^{100} = 2^{200} \approx 10^{60}$！那么，需要多少样例来学习？
    - $1G = 10^9$
- 不仅局限于直方图、非参数估计，在参数估计、以及很多其他统计学习方法中都是如此

Kernel Density Estimation (KDE)

- 举例：Parzen window（一维，使用高斯核）

$$p(x) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{(2\pi h^2)^{\frac{1}{2}}} \exp\left(-\frac{|x - x_i|^2}{2h^2}\right)$$

问题：

- 连续吗？
- 多维：多个维度乘积（独立性假设）
- 需要保存数据吗？
  - 存储和计算实际代价大
  - 无穷多的参数
- 怎么确定h？