KECE470 Pattern Recognition

Chapter 2. Classifiers Based on Bayes Decision Theory

Chang-Su Kim

Many slides are modified from Serigos Theodoridis's own notes.

Classification Problem

- There are M classes: $\omega_1, \ldots, \omega_M$
- Given a pattern with feature vector $\mathbf{x}$, classify it into one of the classes

BAYESIAN CLASSIFICATION

Bayesian Classification Rule

- Classify $\mathbf{x}$ into ω_i if (1)

$$P(\omega_i|\mathbf{x}) > P(\omega_j|\mathbf{x})$$

for all j

Bayesian Classification Rule

- Classify $\mathbf{x}$ into ω_{i^*} where (2)

$$i^* = \arg\max_i P(\omega_i|\mathbf{x})$$

  - $P(\omega_i)$: *a priori* probability
  - $P(\omega_i|\mathbf{x})$: *a posteriori* probability
  - $P(\mathbf{x}|\omega_i)$: likelihood of ω_i with respect to $\mathbf{x}$
  - Bayesian decision is also called **maximum *a posteriori* (MAP) decision**

Bayesian Classification Rule

- Bayes rule

$$P(\omega_i|\mathbf{x}) = \frac{P(\mathbf{x}|\omega_i)P(\omega_i)}{P(\mathbf{x})} = \frac{P(\mathbf{x}|\omega_i)P(\omega_i)}{\sum_j P(\mathbf{x}|\omega_j)P(\omega_j)}$$

- Classify $\mathbf{x}$ into ω_{i^*} where (3)

$$i^* = \arg\max_i P(\mathbf{x}|\omega_i)P(\omega_i)$$

- When all prior probabilities are identical, this becomes
  - Classify $\mathbf{x}$ into ω_{i^*} where

$$i^* = \arg\max_i P(\mathbf{x}|\omega_i)$$

  - This is the **maximum likelihood (ML) decision**

Bayesian Classification Rule

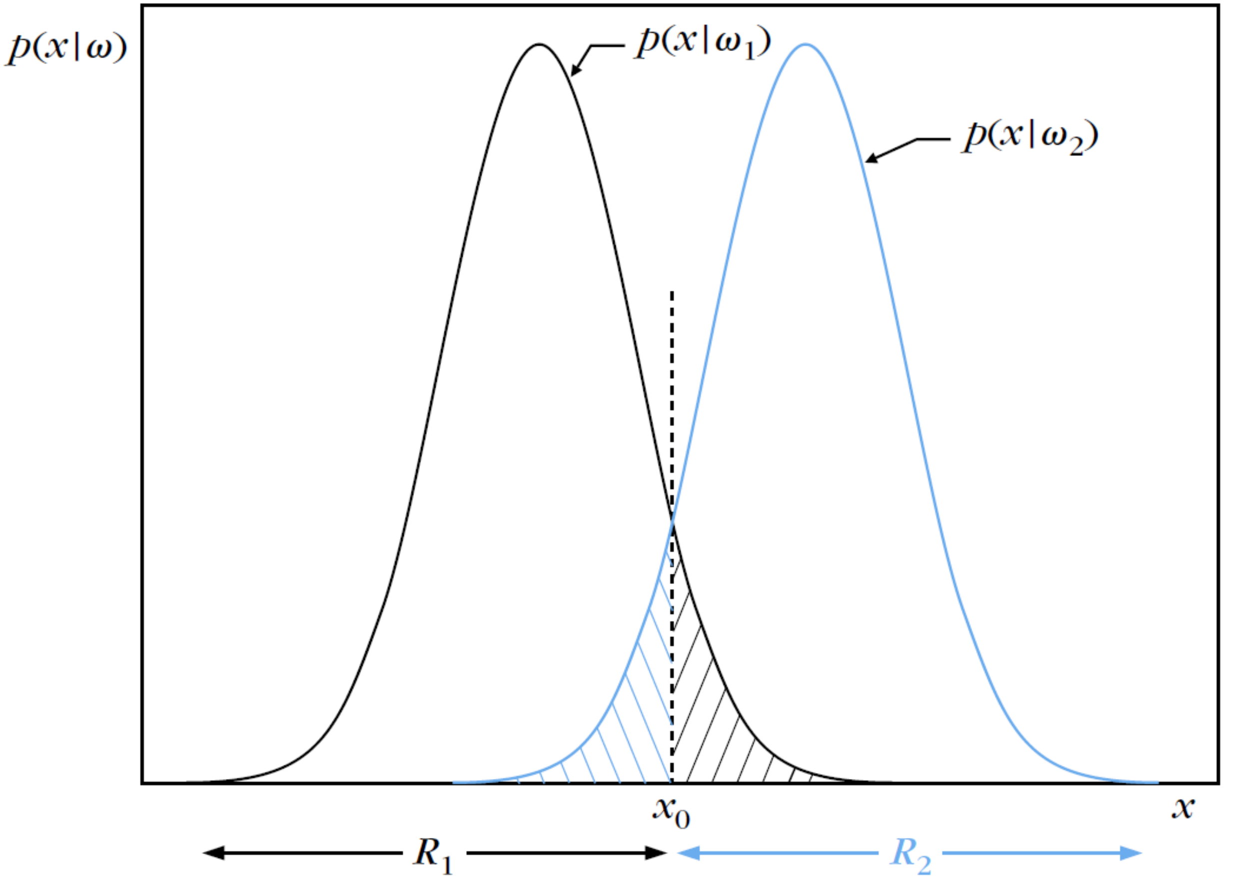


FIGURE 2.1

Example of the two regions R_1 and R_2 formed by the Bayesian classifier for the case of two equiprobable classes.

Bayesian classifier minimizes classification error probability

- Two-class problem
  - Classification error probability
    $$P_e = P(\mathbf{x} \in R_2, \omega_1) + P(\mathbf{x} \in R_1, \omega_2)$$
  - To minimize P_e,
    $$R_1 = \{\mathbf{x}: P(\omega_1|\mathbf{x}) > P(\omega_2|\mathbf{x})\}$$
    $$R_2 = \{\mathbf{x}: P(\omega_1|\mathbf{x}) < P(\omega_2|\mathbf{x})\}$$

- The Bayesian classifier is optimal in that it minimizes P_e

Minimizing Risk

- Medical doctor's problem
  - False negative is more critical than false positive
- Loss (or penalty term) λ_{ij}
  - The penalty for classifying a pattern in ω_i into ω_j
- Loss matrix $L = [\lambda_{ij}]$

Minimizing Risk

- Risk in two-class problem (assuming $\lambda_{ii} = 0$)

$$r = \lambda_{12}P(\omega_1)\int_{R_2} P(\mathbf{x}|\omega_1)d\mathbf{x} + \lambda_{21}P(\omega_2)\int_{R_1} P(\mathbf{x}|\omega_2)d\mathbf{x}$$

- Risk minimizing decision rule (further assuming $P(\omega_1) = P(\omega_2)$)
  - Assign x into ω_2 if

$$P(\mathbf{x}|\omega_2) > P(\mathbf{x}|\omega_1)\frac{\lambda_{12}}{\lambda_{21}}$$

  - This becomes identical with the Bayesian classifier if $\lambda_{12} = \lambda_{21}$

Example 2.1

In a two-class problem with a single feature x the pdfs are Gaussians with variance $\sigma^2 = 1/2$ for both classes and mean values 0 and 1, respectively, that is,

$$p(x|\omega_1) = \frac{1}{\sqrt{\pi}} \exp(-x^2)$$

$$p(x|\omega_2) = \frac{1}{\sqrt{\pi}} \exp(-(x-1)^2)$$

If $P(\omega_1) = P(\omega_2) = 1/2$, compute the threshold value x_0 (a) for minimum error probability and (b) for minimum risk if the loss matrix is

$$L = \begin{bmatrix} 0 & 0.5 \\ 1.0 & 0 \end{bmatrix}$$

Taking into account the shape of the Gaussian function graph (Appendix A), the threshold for the minimum probability case will be

$$x_0 : \exp(-x^2) = \exp(-(x-1)^2)$$

Taking the logarithm of both sides, we end up with $x_0 = 1/2$. In the minimum risk case we get

$$x_0 : \exp(-x^2) = 2\exp(-(x-1)^2)$$

or $x_0 = (1 - \ln 2)/2 < 1/2$; that is, the threshold moves to the left of $1/2$. If the two classes are not equiprobable, then it is easily verified that if $P(\omega_1) > (<) P(\omega_2)$ the threshold moves to the right (left). That is, we expand the region in which we decide in favor of the most probable class, since it is better to make fewer errors for the most probable class.

Discriminant Functions and Decision Surfaces

- If R_i, R_j are contiguous, they are separated by a **decision surface**

$$P(\omega_i|\mathbf{x}) - P(\omega_j|\mathbf{x}) = 0$$

- Equivalently, the decision surface is given by

$$g_i(\mathbf{x}) - g_j(\mathbf{x}) = 0$$

where $g_i(\mathbf{x}) \equiv f\big(P(\omega_i|\mathbf{x})\big)$ is a **discriminant function** and f is monotonically increasing

Bayesian Classification for Normal Distributions

- Multivariate Gaussian PDF

$$P(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{l}{2}}|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

where $\boldsymbol{\mu} = E[\mathbf{x}]$ is the mean vector
$\Sigma = E[(\mathbf{x}-\boldsymbol{\mu})(\mathbf{x}-\boldsymbol{\mu})^T]$ is the covariance matrix

Bayesian Classification for Normal Distributions

- Multivariate Gaussian PDF

$$\Sigma = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}$$

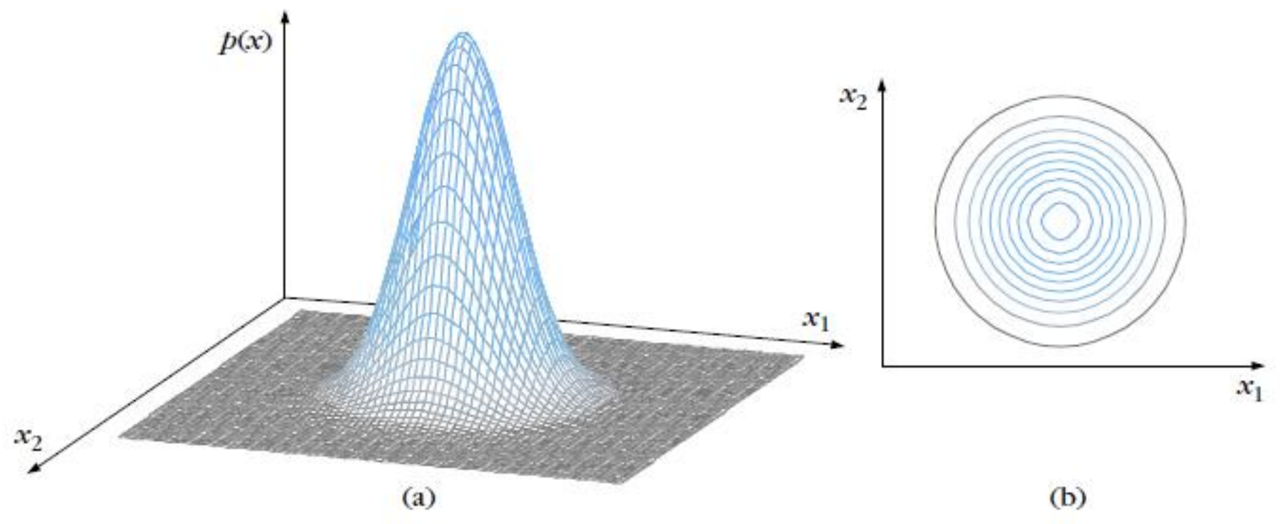


FIGURE 2.3

Bayesian Classification for Normal Distributions

- Multivariate Gaussian PDF

$$\Sigma = \begin{bmatrix} 15 & 0 \\ 0 & 3 \end{bmatrix}$$

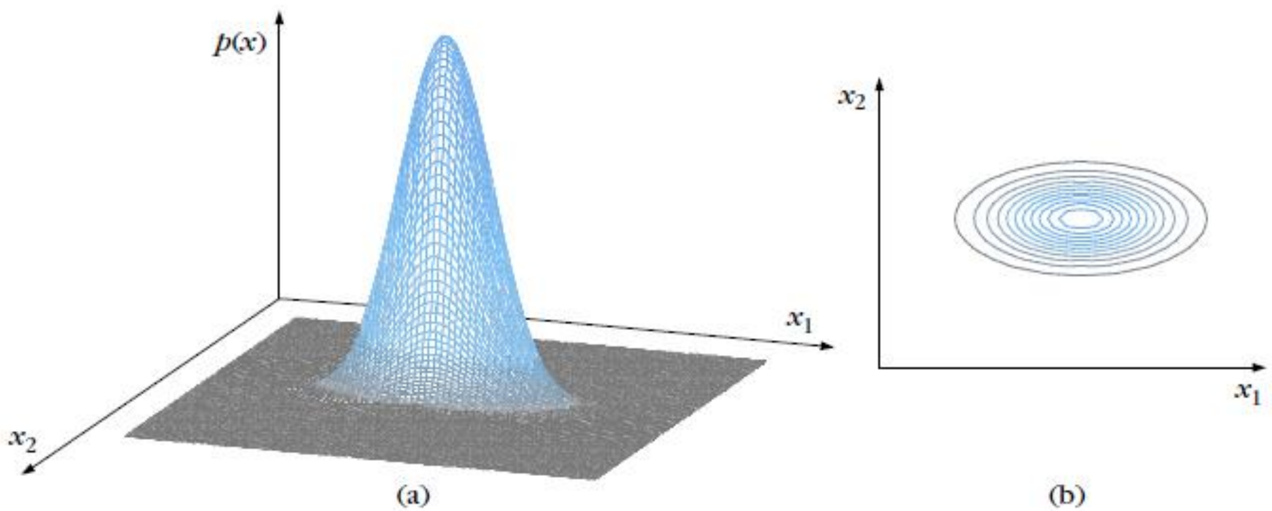


FIGURE 2.4

Bayesian Classification for Normal Distributions

- Multivariate Gaussian PDF

$$\Sigma = \begin{bmatrix} 3 & 0 \\ 0 & 15 \end{bmatrix}$$

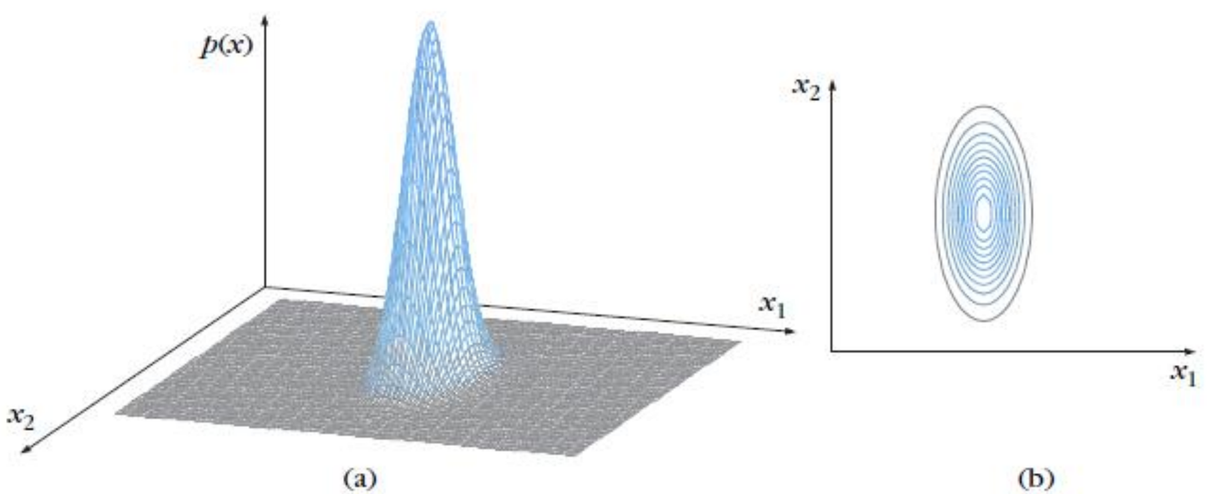


FIGURE 2.5

Bayesian Classification for Normal Distributions

- Multivariate Gaussian PDF

$$\Sigma = \begin{bmatrix} 15 & 6 \\ 6 & 3 \end{bmatrix}$$

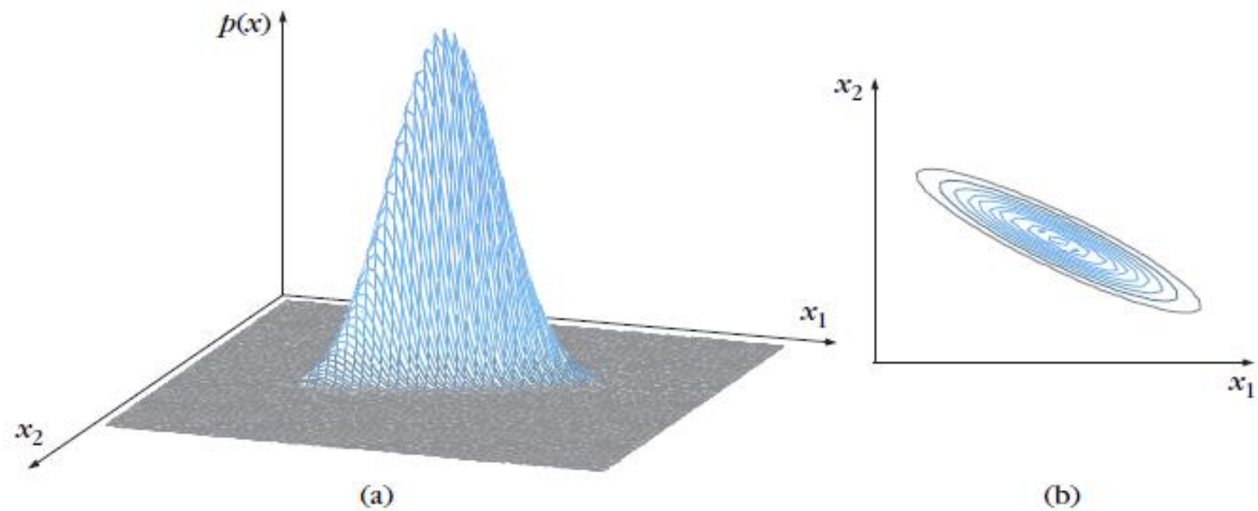


FIGURE 2.6

Bayesian Classification for Normal Distributions

- Discriminant function

$$g_i(\mathbf{x}) = \log P(\mathbf{x}|\omega_i)P(\omega_i)$$
$$= -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \Sigma_i^{-1}(\mathbf{x} - \boldsymbol{\mu}_i) + C_i$$

- Thus, decision surfaces are quadrics (ellipsoids, parabolas, hyperbolas, and pairs of lines)

Bayesian Classification for Normal Distributions

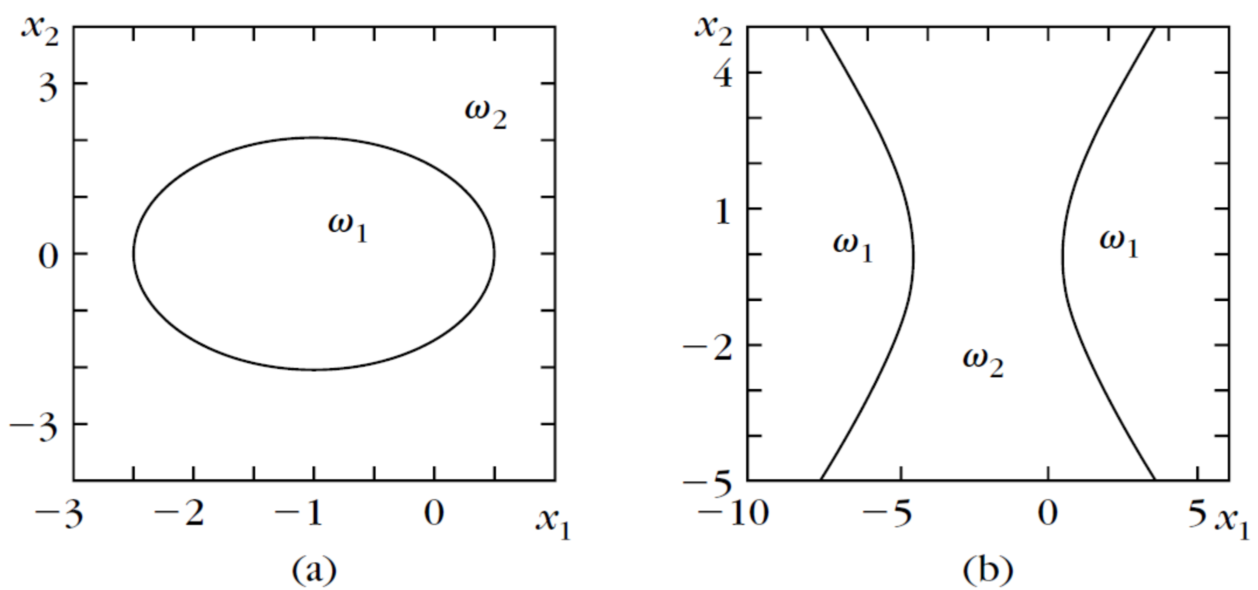


FIGURE 2.7

Examples of quadric decision curves. Playing with the covariance matrices of the Gaussian functions, different decision curves result, that is, ellipsoids, parabolas, hyperbolas, pairs of lines.

Bayesian Classification for Normal Distributions

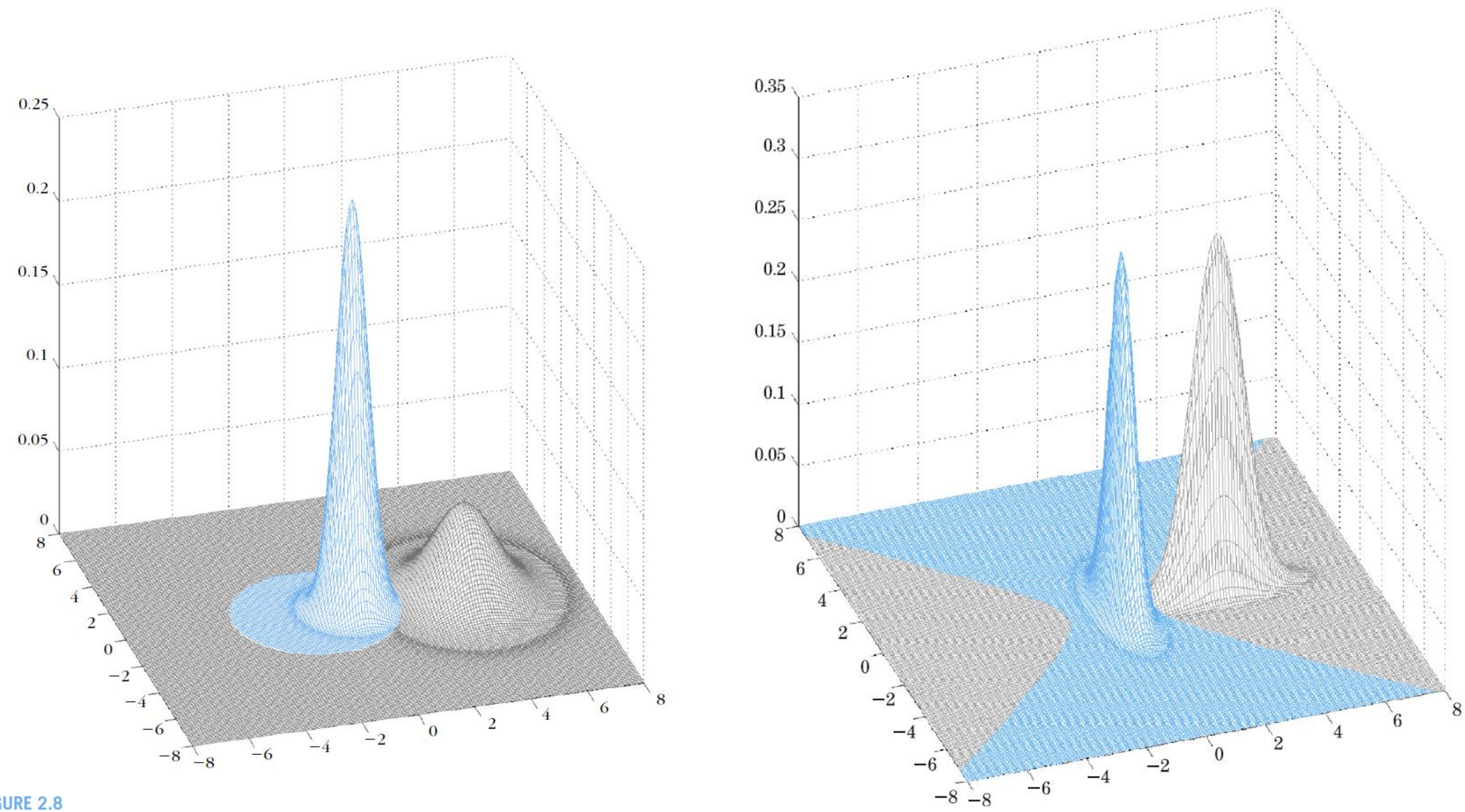


FIGURE 2.8

Special Case I: $\Sigma_i = \sigma^2 \mathrm{I}$

- Decision hyperplane

$$g_{ij}(\mathbf{x}) = \mathbf{w}^T(\mathbf{x} - \mathbf{x}_0) = 0$$

  - $\mathbf{w}^T = \boldsymbol{\mu}_i - \boldsymbol{\mu}_j$
  - $\mathbf{x}_0 = \frac{1}{2}(\boldsymbol{\mu}_i + \boldsymbol{\mu}_j) - \sigma^2 \ln\left(\frac{P(\omega_i)}{P(\omega_j)}\right) \frac{\boldsymbol{\mu}_i - \boldsymbol{\mu}_j}{\|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|^2}$

Special Case I: $\Sigma_i = \sigma^2 \mathrm{I}$

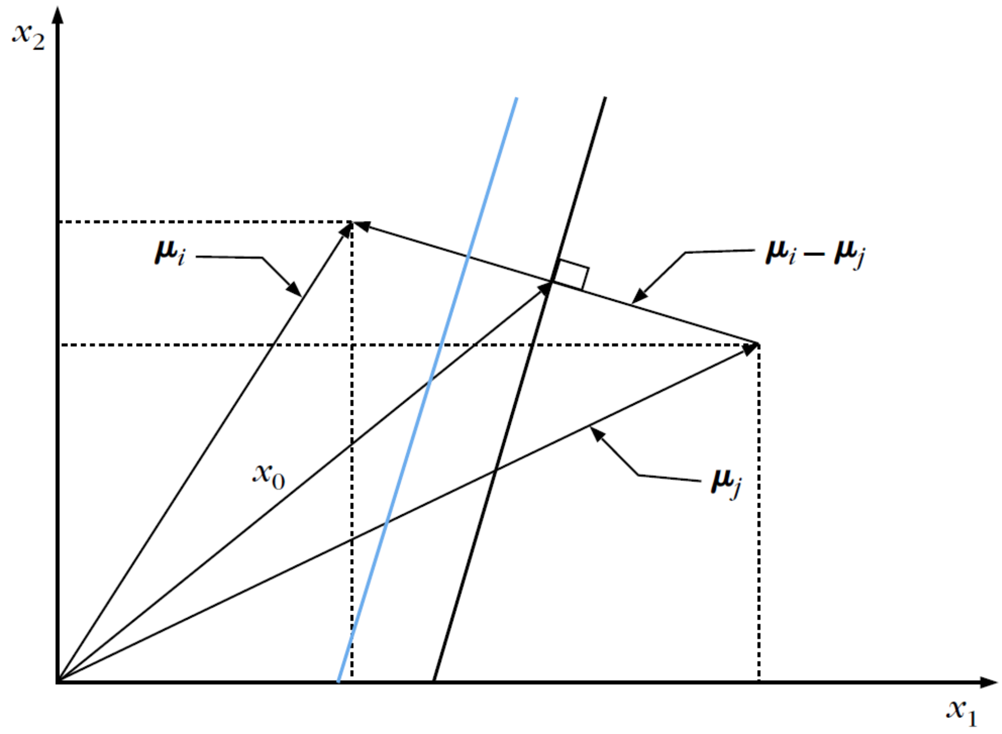


FIGURE 2.10

Decision lines for normally distributed vectors with $\Sigma = \sigma^2 I$. The black line corresponds to the case of $P(\omega_j) = P(\omega_i)$ and it passes through the middle point of the line segment joining the mean values of the two classes. The red line corresponds to the case of $P(\omega_j) > P(\omega_i)$ and it is closer to $\boldsymbol{\mu}_i$, leaving more "room" to the more probable of the two classes. If we had assumed $P(\omega_j) < P(\omega_i)$, the decision line would have moved closer to $\boldsymbol{\mu}_j$.

Special Case II: $\Sigma_i = \Sigma$

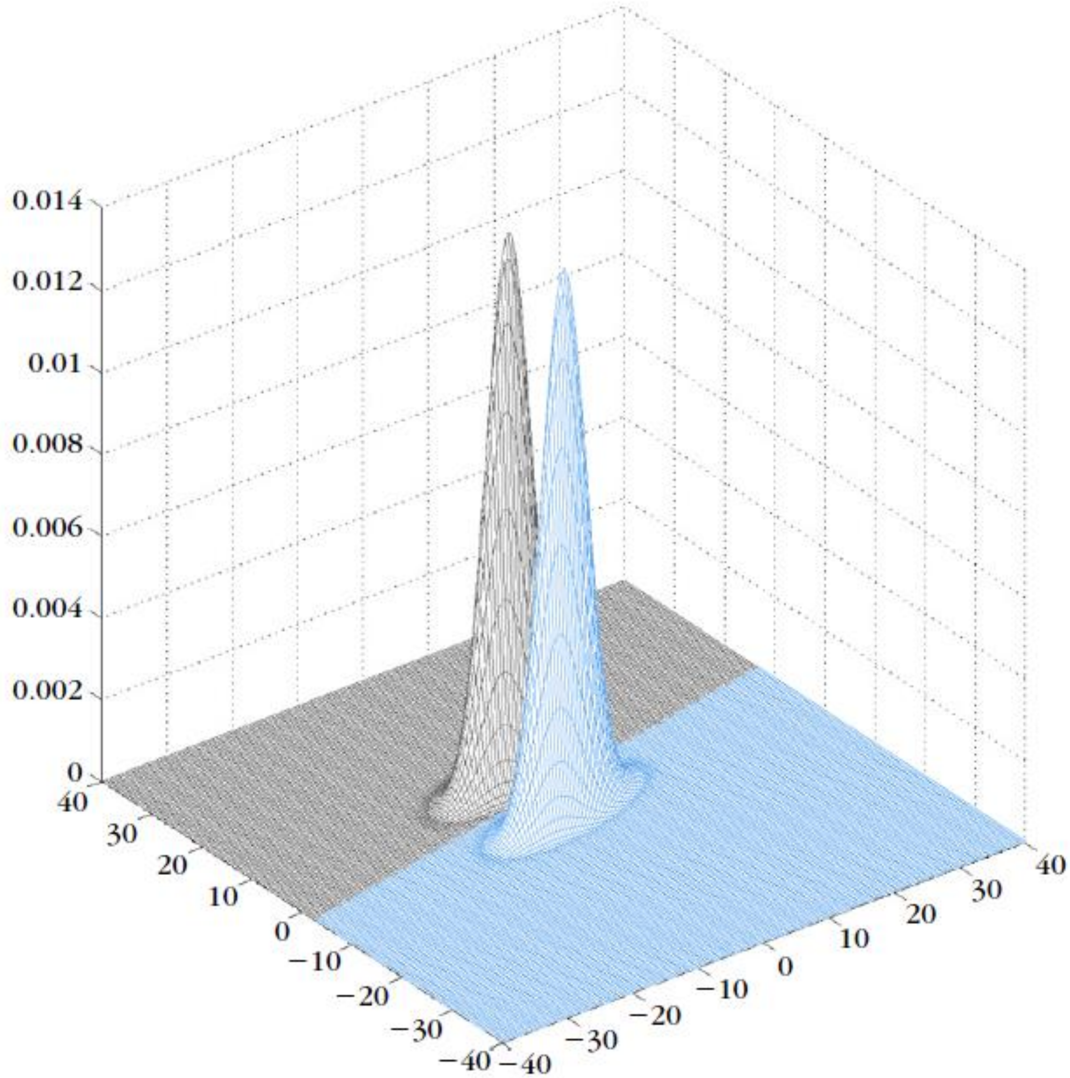
- Decision hyperplane

$$g_{ij}(\mathbf{x}) = \mathbf{w}^T(\mathbf{x} - \mathbf{x}_0) = 0$$

  - $\mathbf{w}^T = \Sigma^{-1}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)$
  - $\mathbf{x}_0 = \frac{1}{2}(\boldsymbol{\mu}_i + \boldsymbol{\mu}_j) - \sigma^2 \ln\left(\frac{P(\omega_i)}{P(\omega_j)}\right) \frac{\boldsymbol{\mu}_i - \boldsymbol{\mu}_j}{(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)}$

Assignment #1. Prove this.

Special Case II: $\Sigma_i = \Sigma$



Minimum Distance Classifiers

- Assuming equiprobable classes, maximize

$$g_i(x) = -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \Sigma_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)$$

  - $\Sigma_i = \sigma^2 \mathrm{I}$: minimize the Euclidean distance

$$\|\mathbf{x} - \boldsymbol{\mu}_i\|$$

  - $\Sigma_i = \Sigma$: minimize the **Mahalanobis distance**

$$\left((\mathbf{x} - \boldsymbol{\mu}_i)^T \Sigma_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)\right)^{\frac{1}{2}}$$

Minimum Distance Classifiers

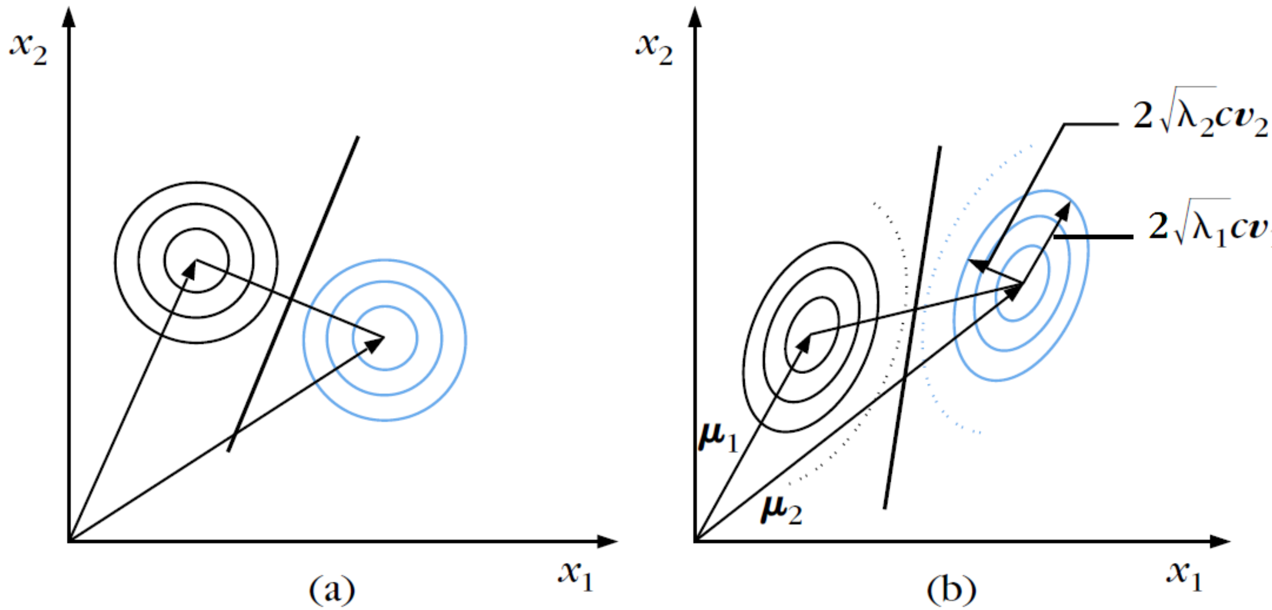


FIGURE 2.13

Curves of (a) equal Euclidean distance and (b) equal Mahalanobis distance from the mean points of each class. In the two-dimensional space, they are circles in the case of Euclidean distance and ellipses in the case of Mahalanobis distance. Observe that in the latter case the decision line is no longer orthogonal to the line segment joining the mean values. It turns according to the shape of the ellipses.

Minimum Distance Classifiers

Example 2.2

In a two-class, two-dimensional classification task, the feature vectors are generated by two normal distributions sharing the same covariance matrix

$$\Sigma = \begin{bmatrix} 1.1 & 0.3 \\ 0.3 & 1.9 \end{bmatrix}$$

and the mean vectors are $\boldsymbol{\mu}_1 = [0, 0]^T$, $\boldsymbol{\mu}_2 = [3, 3]^T$, respectively.

(a) Classify the vector $[1.0, 2.2]^T$ according to the Bayesian classifier.

It suffices to compute the Mahalanobis distance of $[1.0, 2.2]^T$ from the two mean vectors. Thus,

$$d_m^2(\boldsymbol{\mu}_1, \boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{\mu}_1)^T \Sigma^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_1)$$

$$= [1.0, 2.2] \begin{bmatrix} 0.95 & -0.15 \\ -0.15 & 0.55 \end{bmatrix} \begin{bmatrix} 1.0 \\ 2.2 \end{bmatrix} = 2.952$$

Similarly,

$$d_m^2(\boldsymbol{\mu}_2, \boldsymbol{x}) = [-2.0, -0.8] \begin{bmatrix} 0.95 & -0.15 \\ -0.15 & 0.55 \end{bmatrix} \begin{bmatrix} -2.0 \\ -0.8 \end{bmatrix} = 3.672 \qquad (2.54)$$

Thus, the vector is assigned to the class with mean vector $[0, 0]^T$. *Notice that the given vector* $[1.0, 2.2]^T$ *is closer to* $[3, 3]^T$ *with respect to the Euclidean distance.*

Minimum Distance Classifiers

(b) Compute the principal axes of the ellipse centered at $[0, 0]^T$ that corresponds to a constant Mahalanobis distance $d_m = \sqrt{2.952}$ from the center.

To this end, we first calculate the eigenvalues of Σ.

$$\det\left(\begin{bmatrix} 1.1-\lambda & 0.3 \\ 0.3 & 1.9-\lambda \end{bmatrix}\right) = \lambda^2 - 3\lambda + 2 = 0$$

or $\lambda_1 = 1$ and $\lambda_2 = 2$. To compute the eigenvectors we substitute these values into the equation

$$(\Sigma - \lambda I)\boldsymbol{v} = \mathbf{0}$$

and we obtain the unit norm eigenvectors

$$\boldsymbol{v}_1 = \begin{bmatrix} \frac{3}{\sqrt{10}} \\ -\frac{1}{\sqrt{10}} \end{bmatrix}, \quad \boldsymbol{v}_2 = \begin{bmatrix} \frac{1}{\sqrt{10}} \\ \frac{3}{\sqrt{10}} \end{bmatrix}$$

It can easily be seen that they are mutually orthogonal. The principal axes of the ellipse are parallel to $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ and have lengths 3.436 and 4.859, respectively.

PARAMETRIC ESTIMATION OF UNKNOWN PDF

ML Parameter Estimation

- Let $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ be independent feature vectors, and $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$
- Assume that feature vectors have the PDF $P(\mathbf{x}|\boldsymbol{\theta})$ with unknown parameters $\boldsymbol{\theta}$
- $P(X|\boldsymbol{\theta}) \equiv P(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N|\boldsymbol{\theta}) = \prod_{k=1}^{N} P(\mathbf{x}_k|\boldsymbol{\theta})$
- $\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} = \arg\max_{\boldsymbol{\theta}} \prod_{k=1}^{N} P(\mathbf{x}_k|\boldsymbol{\theta})$
- A necessary condition
  - $L(\boldsymbol{\theta}) \equiv \ln P(X|\boldsymbol{\theta}) = \sum_{k=1}^{N} \ln P(\mathbf{x}_k|\boldsymbol{\theta})$
  - $\frac{\partial L(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_{k=1}^{N} \frac{1}{P(\mathbf{x}_k;\boldsymbol{\theta})} \frac{\partial P(\mathbf{x}_k;\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \mathbf{0}$

ML Parameter Estimation

- Properties of the ML estimate
  - Asymptotically unbiased

$$\lim_{N\to\infty} E\left[\widehat{\boldsymbol{\theta}}_{\mathrm{ML}}\right] = \boldsymbol{\theta}_{\mathrm{true}}$$

  - Asymptotically consistent

$$\lim_{N\to\infty} E\left[\left\|\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} - \boldsymbol{\theta}_{\mathrm{true}}\right\|^2\right] = 0$$

ML Parameter Estimation

- Example 2.3
  - Assume that N data points $x_1, x_2, \dots, x_N$ have been generated by a 1D Gaussian PDF of a known mean μ but of a unknown variance. Derive the ML estimate of the variance.

ML Parameter Estimation

- Example 2.4
  - Assume that N data points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ have been generated by a Gaussian PDF of a known covariance matrix Σ but of a unknown mean vector. Derive the ML estimate of the mean vector.

MAP Parameter Estimation

- $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$
- $P(\boldsymbol{\theta}|X) = \frac{P(\boldsymbol{\theta})P(X|\boldsymbol{\theta})}{P(X)}$
- $\widehat{\boldsymbol{\theta}}_{\mathrm{MAP}} = \arg\max_{\boldsymbol{\theta}} P(\boldsymbol{\theta}|X)$

MAP Parameter Estimation

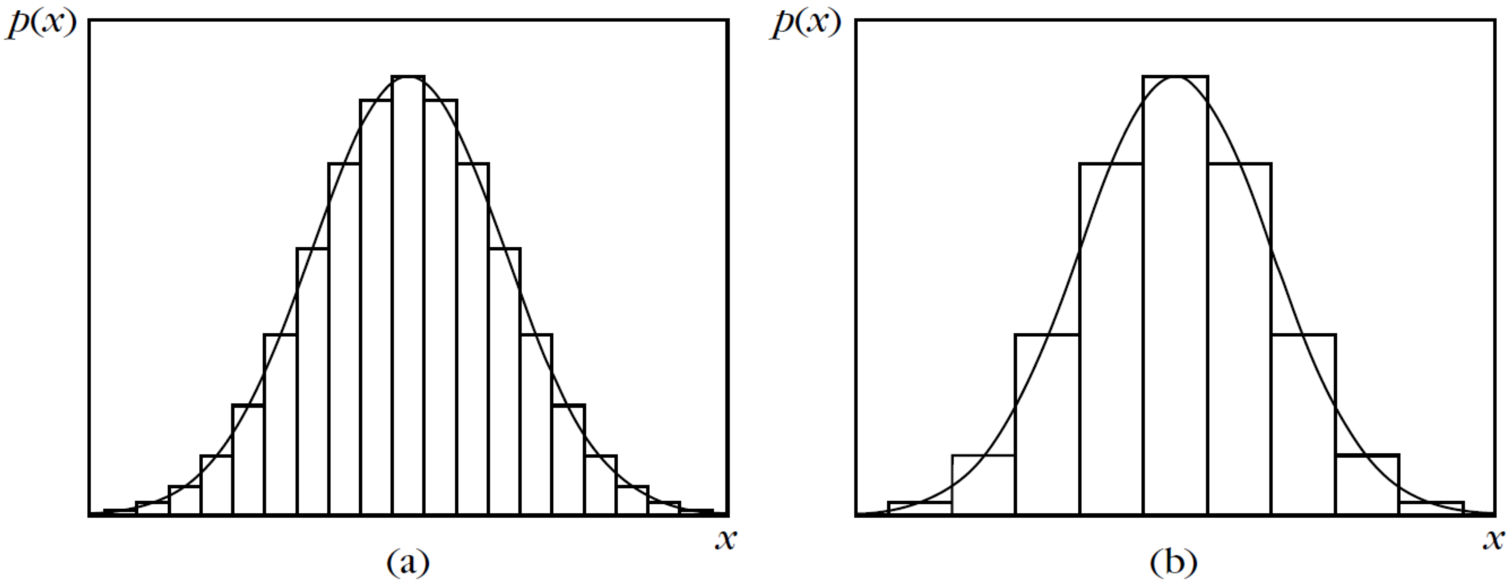
- Example 2.5
  - Assume that N data points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ have been generated by a Gaussian PDF of a known covariance matrix Σ but of a unknown mean vector $\boldsymbol{\mu}$. It is, however, known that

$$P(\boldsymbol{\mu}) = \frac{1}{(2\pi)^{l/2}\sigma_\mu^l} \exp(-\frac{1}{2} \frac{\|\boldsymbol{\mu} - \boldsymbol{\mu}_0\|^2}{\sigma_\mu^2})$$

Derive the MAP estimate of the mean vector.

NONPARAMETRIC ESTIMATION OF UNKNOWN PDF

Nonparametric Estimation



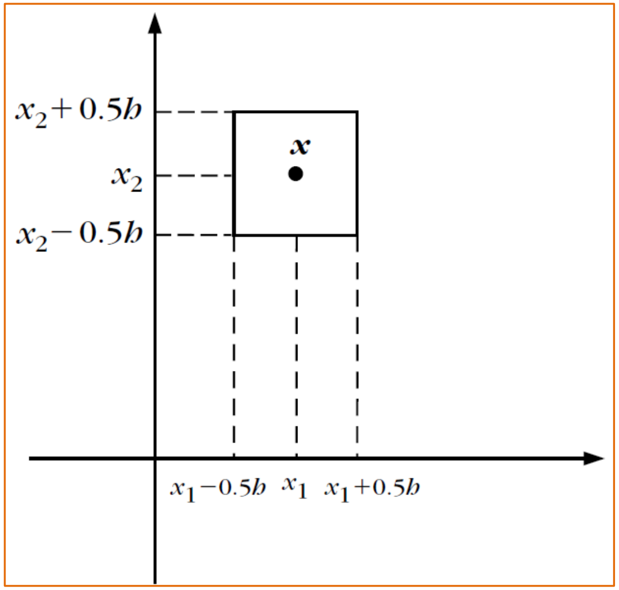
- PDF approximation by the histogram method with (a) small and (b) large intervals (bins)
- $\hat{P}(x) = \frac{1}{h}\frac{k_N}{N}$

Nonparametric Estimation

- $\hat{P}(x)$ converges to the true $P(x)$ if
  - $h \to 0$
  - $k_N \to \infty$
  - $\frac{k_N}{N} \to 0$

Parzen Windows

- An example (histogram method)
  - $\varphi(\mathbf{x}) = \begin{cases} 1 & |x_i| \leq \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$
  - $\hat{P}(\mathbf{x}) = \frac{1}{h^l}\frac{1}{N}\sum_{i=1}^{N}\varphi\left(\frac{\mathbf{x}_i - \mathbf{x}}{h}\right)$



- In general, we use **kernels** or **Parzen windows** $\varphi(\mathbf{x})$ such that
  - $\varphi(\mathbf{x}) > \mathbf{0}$
  - $\int_{\mathbf{x}} \varphi(\mathbf{x})\, d\mathbf{x} = 1$

Parzen Windows

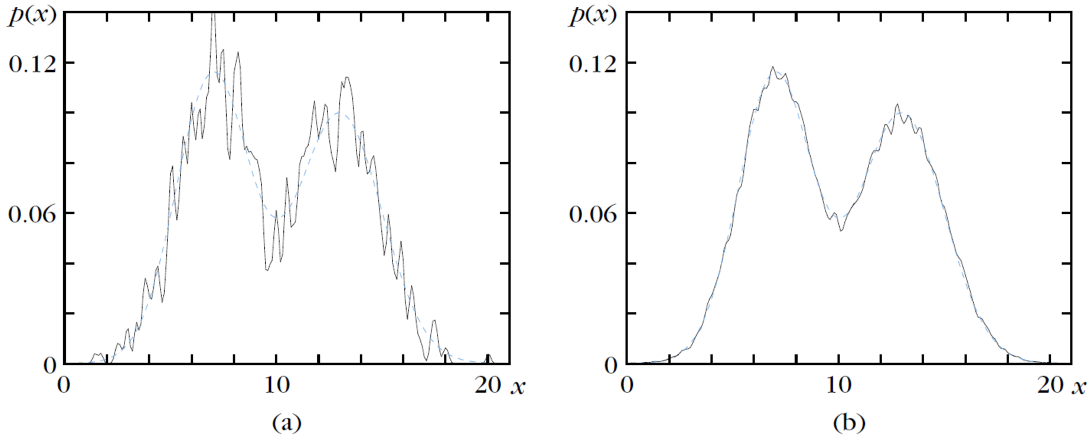


FIGURE 2.20

Approximation (full-black line) of a pdf (dotted-red line) via Parzen windows, using Gaussian kernels with (a) $h = 0.1$ and 1,000 training samples and (b) $h = 0.1$ and 20,000 samples. Observe

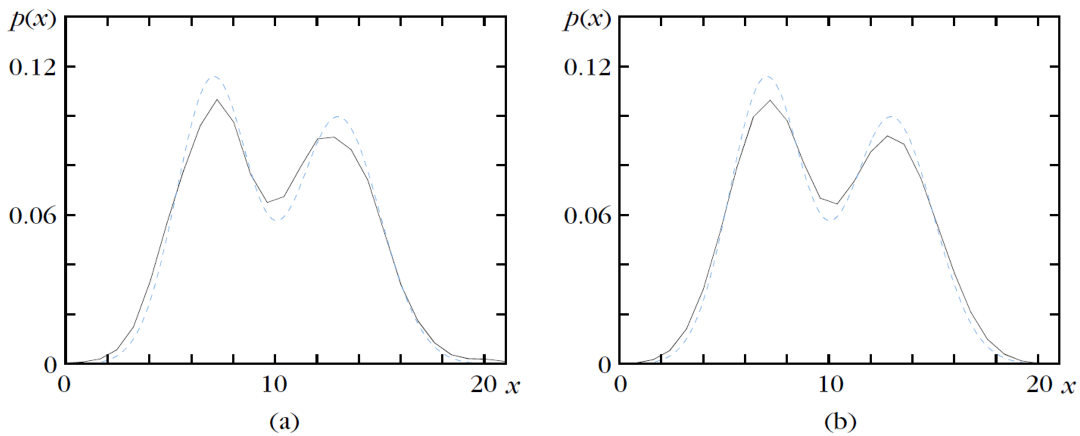


FIGURE 2.21

Approximation (full-black line) of a pdf (dotted-red line) via Parzen windows, using Gaussian kernels with (a) $h = 0.8$ and 1,000 training samples and (b) $h = 0.8$ and 20,000 samples. Observe

- Unbiased only if $h \rightarrow 0$
- Smaller h ⇒
  - More accurate
  - Less smooth
  - Higher variance
  - Less consistent
- Bigger N
  - Smoother
  - Smaller variance
  - More consistent

Parzen Windows

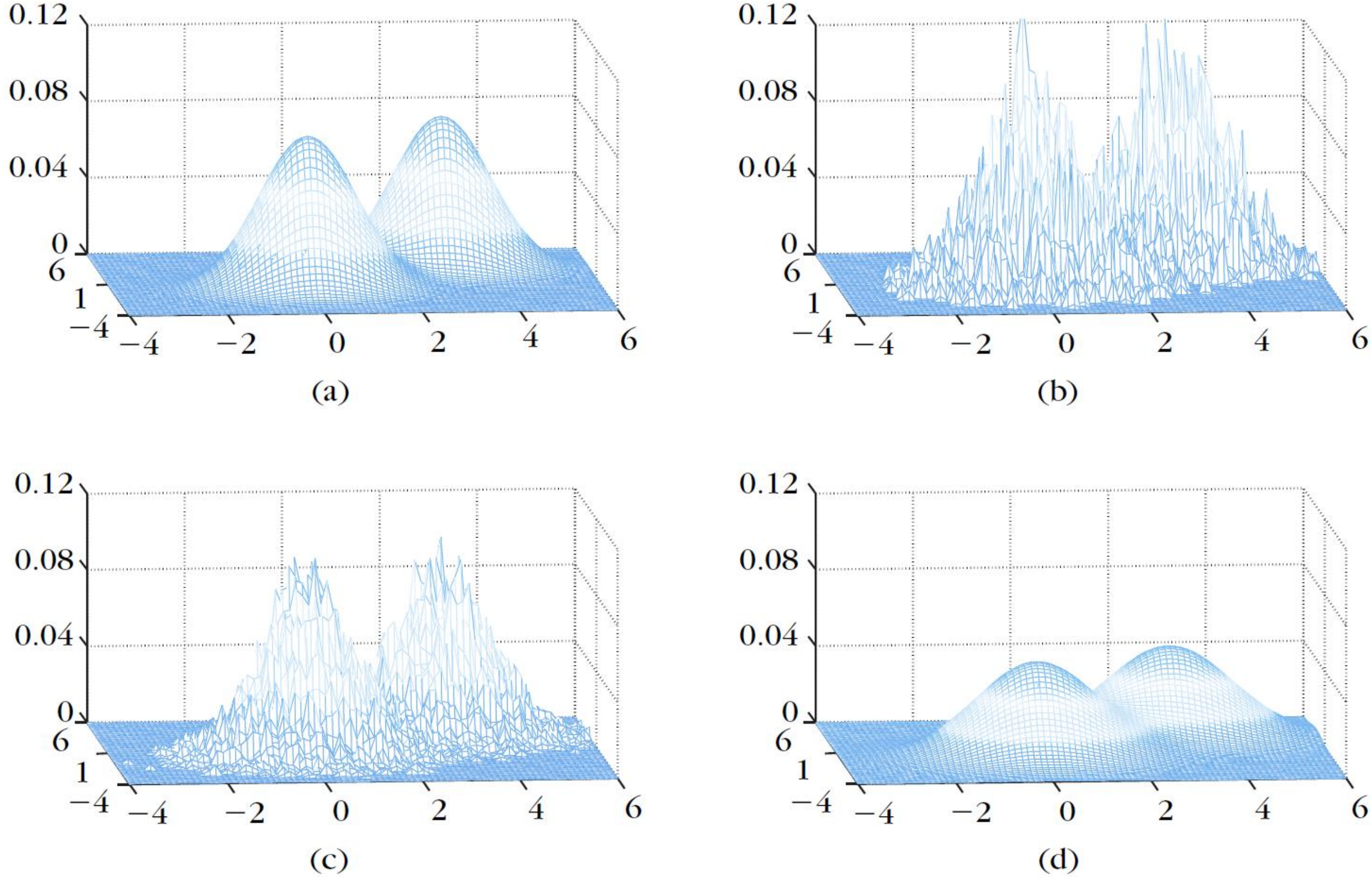


FIGURE 2.22

Approximation of a two-dimensional pdf, shown in (a), via Parzen windows, using two-dimensional Gaussian kernels with (b) $h = 0.05$ and $N = 1000$ samples, (c) $h = 0.05$ and $N = 20000$ samples and (d) $h = 0.8$ and $N = 20000$ samples. Large values of h lead to smooth

Curse of dimensionality

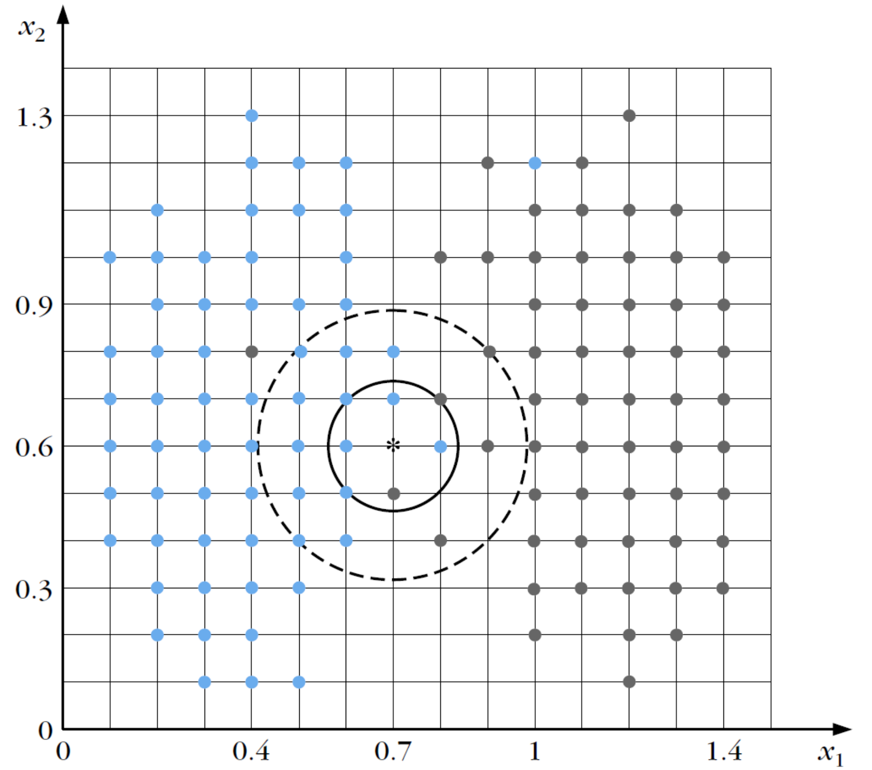
- For a large l, the number of data, N, for reliable PDF estimation becomes impractically high.

k-NN Density Estimation

Example 2.9

The points shown in Figure 2.24 belong to either of two equiprobable classes. Black points belong to class ω_1 and red points belong to class ω_2. For the needs of the example we assume that all points are located at the nodes of a grid. We are given the point denoted by a "star", with coordinates $(0.7, 0.6)$, which is to be classified in one of the two classes. The Bayesian (minimum error probability) classifier and the k-nearest neighbor density estimation technique, for $k = 5$, will be employed.

$$\hat{P}(x) = \frac{k}{NV(x)}$$



ETC

Naive-Bayes Classifier

- Independence Assumption

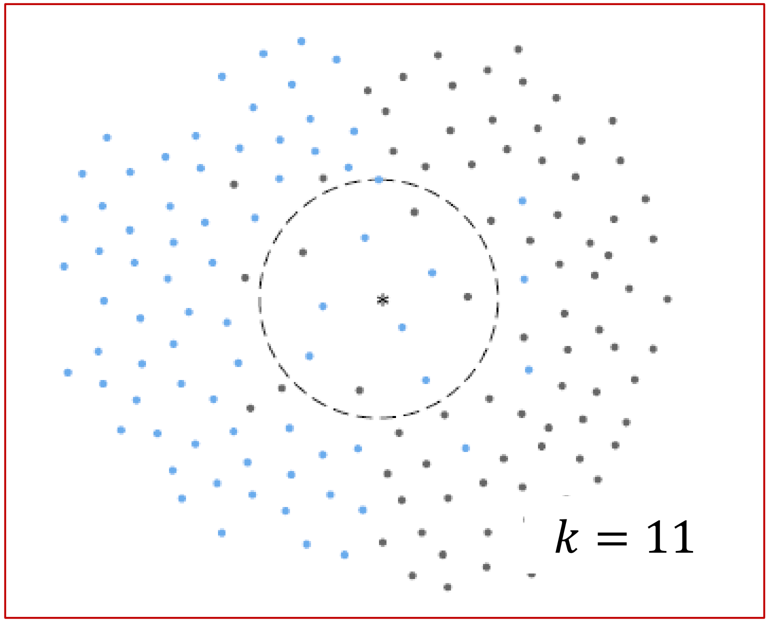
$$P(\mathbf{x}|\omega_i) = \prod_{j=1}^{l} P(x_j|\omega_i)$$

- Bayesian Classification

$$\omega^* = \arg\max_{\omega_i} \prod_{j=1}^{l} P(x_j|\omega_i)$$

k-NN Classification

1. Out of the N training vectors, identify the k nearest neighbors.
2. Inspect these k vectors to determine the number of vectors k_j in the class ω_j.
3. Assign x to ω_i if $k_i > k_j$, $\forall j \neq i$

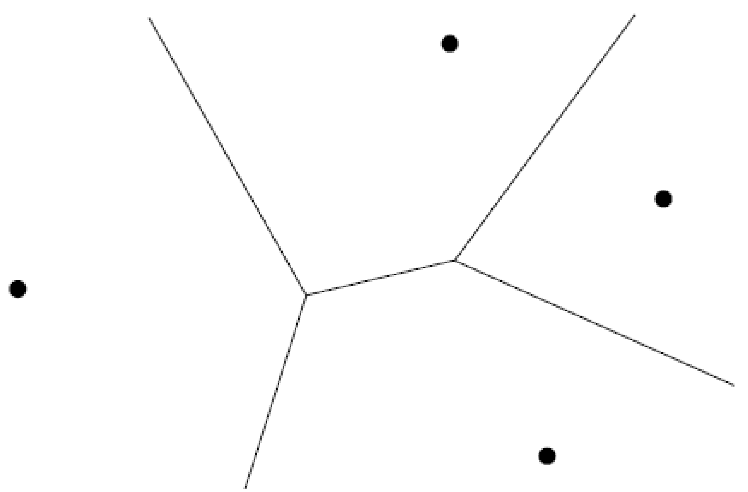


k-NN Classification

- The simplest version $k = 1$ is known as the **NN rule**.
- It can be shown that, as $N \to \infty$,

$$P_B \leq P_{NN} \leq 2P_B$$

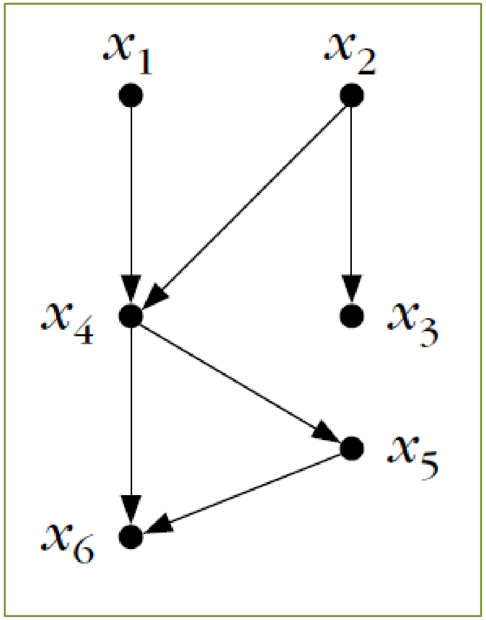
- **Voronoi tessellation**



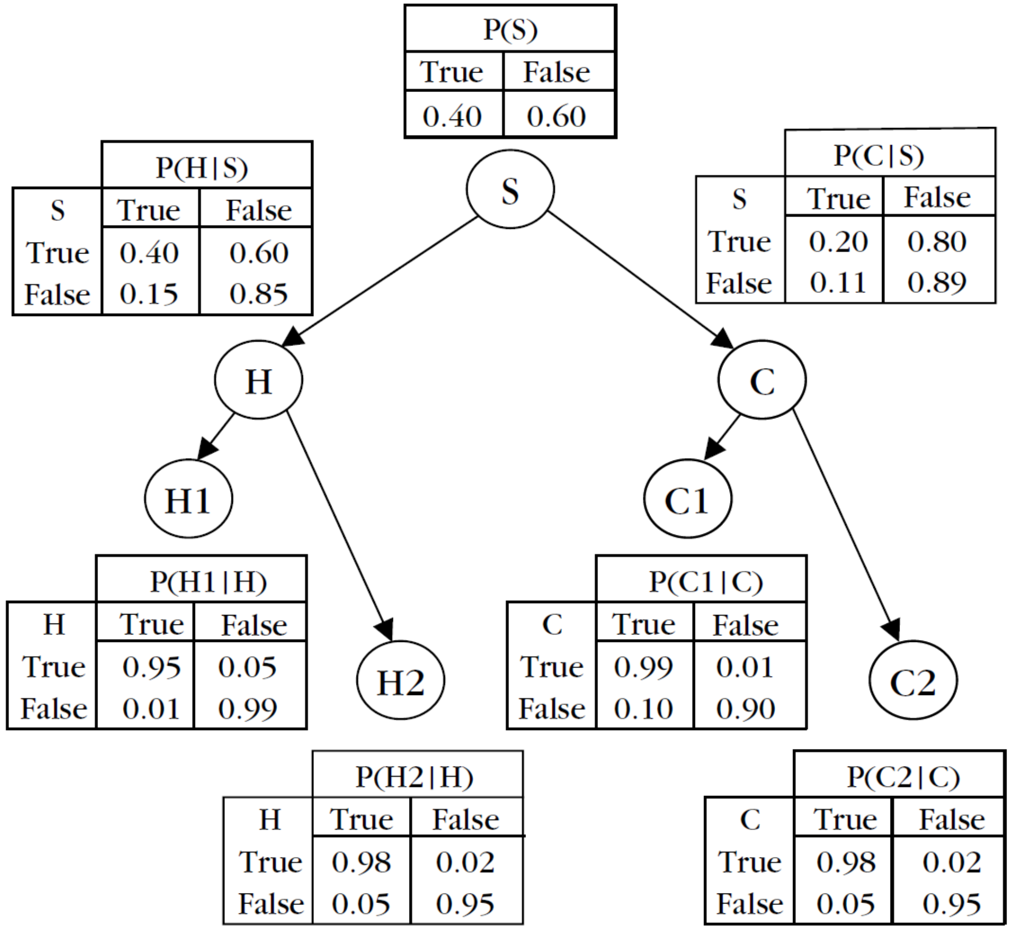
$$R_i = \{\mathbf{x}: d(\mathbf{x}, \mathbf{x}_i) < d(\mathbf{x}, \mathbf{x}_j), i \neq j\}$$

Bayesian Networks

- Bayesian network
  - directed acyclic graph (DAG)
  - Each node is associated with a conditional probability, $P(x_i|A_i)$
    - x_i: the corresponding feature
    - A_i: the set of its parents
  - x_i is conditionally independent of any combination of its non-descendants, given its parents



Bayesian Networks



P(S)	
True	False
0.40	0.60

	P(H|S)	
S	True	False
True	0.40	0.60
False	0.15	0.85

	P(C|S)	
S	True	False
True	0.20	0.80
False	0.11	0.89

	P(H1|H)	
H	True	False
True	0.95	0.05
False	0.01	0.99

	P(C1|C)	
C	True	False
True	0.99	0.01
False	0.10	0.90

	P(H2|H)	
H	True	False
True	0.98	0.02
False	0.05	0.95

	P(C2|C)	
C	True	False
True	0.98	0.02
False	0.05	0.95

FIGURE 2.28

Bayesian network modeling conditional dependencies for an example concerning smokers (S), tendencies to develop cancer (C), and heart disease (H), together with variables corresponding to heart (H1, H2) and cancer (C1, C2) medical tests.

Bayesian Networks

- Compute
  a. $P(z_1|x_1) = P(z = 1|x = 1)$
  b. $P(w_0|x_1)$
  c. $P(x_0|w_1)$

$P(x1) = 0.60$ $\quad P(y1|x1) = 0.40$ $\quad P(z1|y1) = 0.25$ $\quad P(w1|z1) = 0.45$

$P(y1|x0) = 0.30$ $\quad P(z1|y0) = 0.60$ $\quad P(w1|z0) = 0.30$