EE 583
PATTERN RECOGNITION

Statistical Pattern Recognition

- Bayes Decision Theory
- Supervised Learning
- Linear Discriminant Functions
- Unsupervised Learning

METU EE583 Lecture Notes by A.Aydin ALATAN © 2014

Bayes Decision Theory

- Fundamental statistical approach to PR
- Assumptions:
  - Decision problem is probabilistic
  - All relevant probability values are known
- The decision rules are *optimal* in the sense that it either minimizes average probability of error or overall risk.

Example : Bayes Decision (1/2)

- Classification problem of apple and peach by *color*
- Assume initial observation probabilities are <u>not</u> equal,
  i.e., assume P(apple) > P(peach)

- If you do not have a chance to *see* the fruit,
  ➔ every time decide/predict as apple!

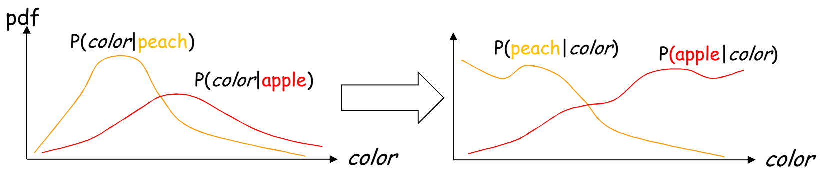
- If you are able to observe the color of the fruit,
  - <u>Question</u> : P(apple|*color*)=? , P(peach|*color*)= ?
    Intuitively, choose the class with higher conditional probability.
  - How to find these probabilities?

  Try using Bayes Rule :

  P(apple|*color*) = p(*color*|apple)*P(apple) / p(*color*)
  P(peach|*color*) = p(*color*|peach)*P(peach) / p(*color*)

METU EE583 Lecture Notes by A.Aydin ALATAN © 2014

Example : Bayes Decision (2/2)



<u>Bayes Decision Rule</u> :

If P(apple|*color*) > P(peach| *color*), then choose *apple*

p(*color*|apple)/p(*color*|peach) > P(peach)/P(apple)

Likelihood ratio — Constant

- Note that the *evidence* P(*color*) is only necessary for normalization purposes; it does not affect the decision rule

Bayes Decision Theory (General)[1/4]

- Generalize Bayes Decision Theory by
  - allowing to use multi features
  - allowing to use more that two states
  - allowing actions rather than choosing states
  - introducing a loss function rather than probability of error

$\vec{x}$: feature vector (d x 1)

$\Omega = \{\omega_1, \cdots, \omega_s\}$: states (classes)

$A = \{\alpha_1, \cdots, \alpha_q\}$: actions (allows possibility of rejection)

$\lambda(\alpha_i \mid \omega_j)$: *loss* for taking action i for state j

A posteriori probability : $$P(\omega_j \mid \vec{x}) = \frac{p(\vec{x} \mid \omega_j) P(\omega_j)}{p(\vec{x})}$$

$$p(\vec{x}) = \sum_{j=1}^{s} p(\vec{x} \mid \omega_j) P(\omega_j)$$

Bayes Decision Theory (General) [2/4]

Minimum Risk Classifier

We observe $\boldsymbol{x}$, then should take one of the actions i, α_i

Bayes decision rule should minimize the overall risk R:

$$R = \int R(\alpha(\vec{x}) \mid \vec{x}) \, p(\vec{x}) \, d\vec{x}$$

where expected loss (*conditional risk*) by taking action i :

$$R(\alpha_i \mid \vec{x}) = \sum_{j=1}^{s} \lambda(\alpha_i \mid \omega_j) P(\omega_j \mid \vec{x})$$

$\lambda(\alpha_i \mid \omega_j)$: *loss* for taking action i for state j

Rule : Compute *conditional risk* for every action and select the action with minimum conditional risk.

$$\min \{R(\alpha_i \mid \vec{x})\} \Rightarrow \min \{R\}$$

Bayes Decision Theory (General) [3/4]

Minimum Risk Classifier : Two Category Case

Assume there are only 2 classes

$$R(\alpha_1 \mid \vec{x}) = \lambda(\alpha_1 \mid \omega_1) P(\omega_1 \mid \vec{x}) + \lambda(\alpha_1 \mid \omega_2) P(\omega_2 \mid \vec{x})$$
$$R(\alpha_2 \mid \vec{x}) = \lambda(\alpha_2 \mid \omega_1) P(\omega_1 \mid \vec{x}) + \lambda(\alpha_2 \mid \omega_2) P(\omega_2 \mid \vec{x})$$

$\lambda(\alpha_i \mid \omega_j)$: *loss* for taking action i for state j

Take action-1, α_1 (α_1 : decide on class-1), if $R(\alpha_1|x) < R(\alpha_2|x)$

$$(\lambda(\alpha_1 \mid \omega_2) - \lambda(\alpha_2 \mid \omega_2)) P(\omega_2 \mid \vec{x}) < (\lambda(\alpha_2 \mid \omega_1) - \lambda(\alpha_1 \mid \omega_1)) P(\omega_1 \mid \vec{x})$$

$$\Rightarrow \frac{P(\vec{x} \mid \omega_1)}{P(\vec{x} \mid \omega_2)} > \frac{\lambda_{12} - \lambda_{22}}{\lambda_{21} - \lambda_{11}} \frac{P(\omega_2)}{P(\omega_1)}$$

Likelihood ratio (a function of x)

Constant

Bayes Decision Theory (General) [4/4]

Minimum Error-Rate Classifier :

- Special case for *Minimum Risk Classifier*
    - Correct actions : zero loss ; wrong actions : equal unit loss
- If errors are to be avoided, decision rule should minimize average probability of error, i.e. error-rate

→ Loss function : $\lambda(\alpha_i \mid \omega_j) = \begin{cases} 0 & i = j \\ 1 & i \neq j \end{cases}$

$$R(\alpha_i \mid \vec{x}) = \sum_{j=1}^{s} \lambda(\alpha_i \mid \omega_j) P(\omega_j \mid \vec{x}) = \sum_{j \neq i} P(\omega_j \mid \vec{x}) = 1 - P(\omega_i \mid \vec{x})$$

<u>Rule</u> : Maximize posteriori probability (in order to minimize risk, i.e. average probability of error)

$$Decide \ on \ \omega_i, \ if \ P(\omega_i \mid \vec{x}) > P(\omega_j \mid \vec{x}) \ for \ all \ i \neq j$$

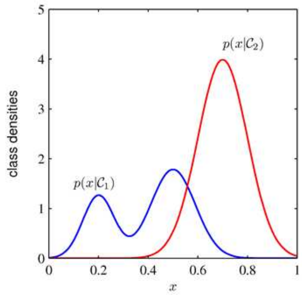
$$\text{For a 2-class problem} \quad \Rightarrow \frac{P(\vec{x} \mid \omega_1)}{P(\vec{x} \mid \omega_2)} > \frac{P(\omega_2)}{P(\omega_1)}$$

Minimizing Classification Error

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

- **Q** : After observing x, what is *probability of error*, if we decide on one of the 2 classes? (Assume 2 class problem)
- ***A*** : Probability of obtaining the "other" class

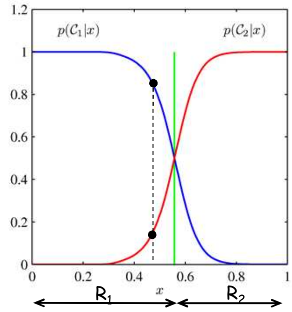
$$P(error \mid x) = \begin{cases} P(C_1 \mid x) & if \ \ decide \ \ C_2 \\ P(C_2 \mid x) & if \ \ decide \ \ C_1 \end{cases}$$



Bayes Decision Rule

decide C_1 *if* $P(C_1 \mid x) > P(C_2 \mid x)$

decide C_2 *if* $P(C_2 \mid x) > P(C_1 \mid x)$



$$P(error \mid x) = \min \{ P(C_1 \mid x), P(C_2 \mid x) \}$$

Minimizing Classification Error

$$P(error \mid x) = \min \{ P(C_1 \mid x), P(C_2 \mid x) \}$$

- The average probability of error will be smaller, since $P(error|x)$ is forced to be minimum by Bayes decision rule for every x.
- Another way to show minimum error :

$$P(error) = P(C_1)P(x \in R_2 \mid C_1) + P(C_2)P(x \in R_1 \mid C_2)$$

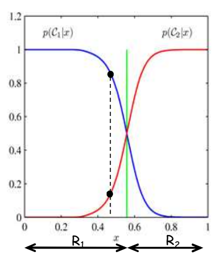
$$= P(C_1)\int_{R2} P(x \mid C_1)\, dx + P(C_2)\int_{R1} P(x \mid C_2)\, dx$$

$$= \int_{R2} P(C_1 \mid x)\ p(x)dx + \int_{R1} P(C_2 \mid x) p(x)\, dx$$

Since

$$P(C_1) = \int_{R1} P(C_1 \mid x)\ p(x)dx + \int_{R2} P(C_1 \mid x) p(x)\, dx$$

$$\Rightarrow \quad P(error) = P(C_1) - \int_{R1} \big(P(C_1 \mid x) - P(C_2 \mid x)\big) p(x)dx$$



Bayes Decision Rule

decide C_1 *if* $P(C_1 \mid x) > P(C_2 \mid x)$

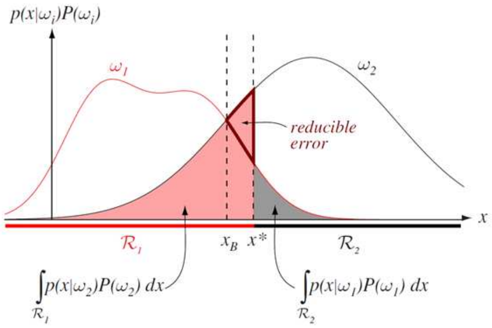
decide C_2 *if* $P(C_2 \mid x) > P(C_1 \mid x)$

Minimizing Classification Error

Remember the relation for $P(error)$ for classes ω_1 and ω_2

$$P(error) = P(x \in R_2 \mid \omega_1)P(\omega_1) + P(x \in R_1 \mid \omega_2)P(\omega_2)$$

$$= \int_{R2} p(x \mid \omega_1)\, P(\omega_1) dx + \int_{R1} p(x \mid \omega_2) P(\omega_2)\, dx$$



Moving from x^* to x_B, the error probability should decrease

METU EE583 Lecture Notes by A.Aydin ALATAN © 2014

Receiver Operating Characteristic (ROC)

For a 2-class problem, ω_1 & ω_2 , x is measured with noise

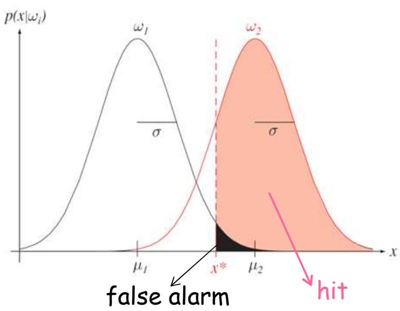
Let x^* denote a detection threshold of the classifier

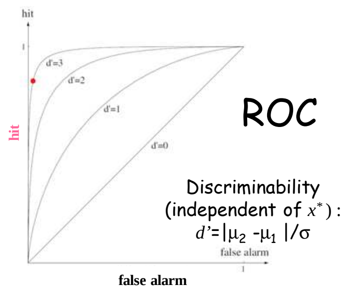
$P(x > x^* \mid x \in \omega_2) : \text{hit}$ $\quad$ $P(x > x^* \mid x \in \omega_1) : \text{false alarm}$

$P(x < x^* \mid x \in \omega_2) : \text{miss}$ $\quad$ $P(x < x^* \mid x \in \omega_1) : \text{correct rejection}$

These probabilities can be estimated experimentally

Change x^* and determine hit and false alarm ➔ ROC





Discriminant Functions & Bayes Classifier

Discriminant function is <u>one of the ways</u> to obtain a pattern classifier.

A classifier based on a discriminant function assigns a feature, x, to class-i, if $g_i(\vec{x}) > g_j(\vec{x}) \quad \textit{for all } j \neq i$

Bayes classifiers can be represented by this approach:

$$g_i(\vec{x}) = -R(\alpha_i \mid \vec{x}) \quad \text{: Minimum conditional risk}$$

$$g_i(\vec{x}) = P(\omega_i \mid \vec{x}) \quad \text{: Minimum error-rate}$$

Selection of a discriminant function is <u>not unique</u>

For minimum error-rate classifier:

$$g_i(\vec{x}) = P(\omega_i \mid \vec{x}) \ \text{or}$$

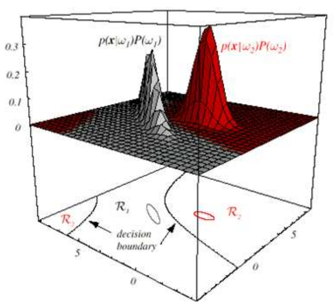
$$g_i(\vec{x}) = p(\vec{x} \mid \omega_i) P(\omega_i) \ \text{or}$$

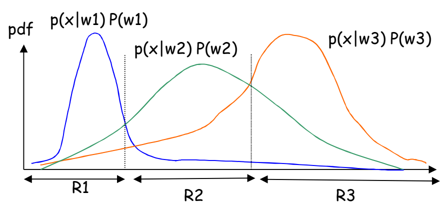
$$g_i(\vec{x}) = \ln p(\vec{x} \mid \omega_i) + \ln P(\omega_i)$$

Discriminant Functions & Bayes Classifier

Discriminant functions might be in different forms, but the effect of the decision rules is the same :

Decision boundaries are obtained



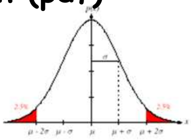


The relation determining the <u>decision boundary</u> between class-i and class-j : $g_i(\vec{x}) = g_j(\vec{x})$

Discriminant Functions for Normal Probability Density (1/7)

Normal (Gaussian) Probability Density Function (pdf)

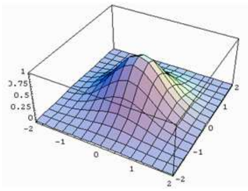
Univariate Normal Density : $p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$

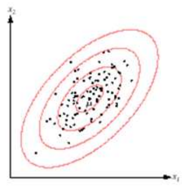


$\mu \equiv E\{x\}$: mean value, $\sigma^2 \equiv E\{(x-\mu)^2\}$: variance

Multivariate Normal Density : $p(\vec{x}) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} e^{-\frac{1}{2}(\vec{x}-\vec{\mu})^t \Sigma^{-1} (\vec{x}-\vec{\mu})}$

$\Sigma \equiv E\{(\vec{x}-\vec{\mu})(\vec{x}-\vec{\mu})^T\}$: Covariance matrix, Σ, determines "shape" of Gaussian curve

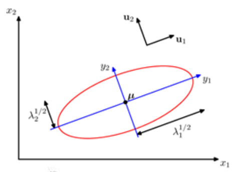




$(\vec{x}-\vec{\mu})^t \Sigma^{-1} (\vec{x}-\vec{\mu})$ is called Mahalanobis distance

Discriminant Functions for Normal Probability Density (1/7)

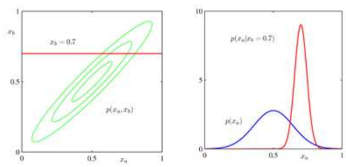
Properties of Normal pdf



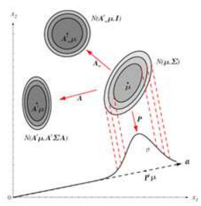
- Eigenvalues and eigenvectors of $\Sigma \equiv E\{(\vec{x}-\vec{\mu})(\vec{x}-\vec{\mu})^T\}$

  $\Sigma \vec{u}_i = \lambda_i \vec{u}_i$

- Marginal pdf of a multivariate normal distribution is also Gaussian



- Linear transforms also yield Gaussians



$\vec{y} = A^T \vec{x}, \quad pdf(\vec{x}) = N(\vec{\mu}, \Sigma) \Rightarrow pdf(\vec{y}) = N(A^T \vec{\mu}, A^T \Sigma A)$

$y = a^t \vec{x}, \quad pdf(\vec{x}) = N(\vec{\mu}, \Sigma) \Rightarrow pdf(y) = N(\mu, \sigma)$

It is possible to transform an arbitrary shaped covariance matrix into obtain a circular one

Discriminant Functions for Normal Probability Density (2/7)

For minimum-error-rate classification, one can choose discriminant function as :

$$g_i(\vec{x}) = \log p(\vec{x} \mid \omega_i) + \log P(\omega_i)$$

For multivariate normal conditional density, discriminant function is :

$$p(\vec{x} \mid \omega_i) = \frac{1}{(2\pi)^{d/2} |\Sigma_i|^{1/2}} e^{-\frac{1}{2}(\vec{x}-\vec{\mu}_i)^t \Sigma_i^{-1} (\vec{x}-\vec{\mu}_i)}$$

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

Discriminant Functions for Normal Probability Density (3/7)

<u>Case 1 :</u> $\Sigma_i = \sigma^2 I$ (independence, equal σ)

$$g_i(\vec{x}) = -\frac{\|\vec{x}-\vec{\mu}_i\|^2}{2\sigma^2} + \log P(\omega_i) \Rightarrow g_i(\vec{x}) = -\frac{1}{2\sigma^2}\left[\vec{x}^t\vec{x} - 2\vec{\mu}_i^t\vec{x} + \vec{\mu}_i^t\vec{\mu}_i\right] + \log P(\omega_i)$$

$$g_i(\vec{x}) = \vec{w}_i^t\vec{x} + w_{i0} \quad \text{where} \quad \vec{w}_i = \frac{1}{\sigma^2}\vec{\mu}_i, \; w_{i0} = -\frac{1}{2\sigma^2}\vec{\mu}_i^t\vec{\mu}_i + \log P(\omega_i)$$

(note: $g_i(x)$ is a linear function ➔ linear discriminant function)

Decision boundary :

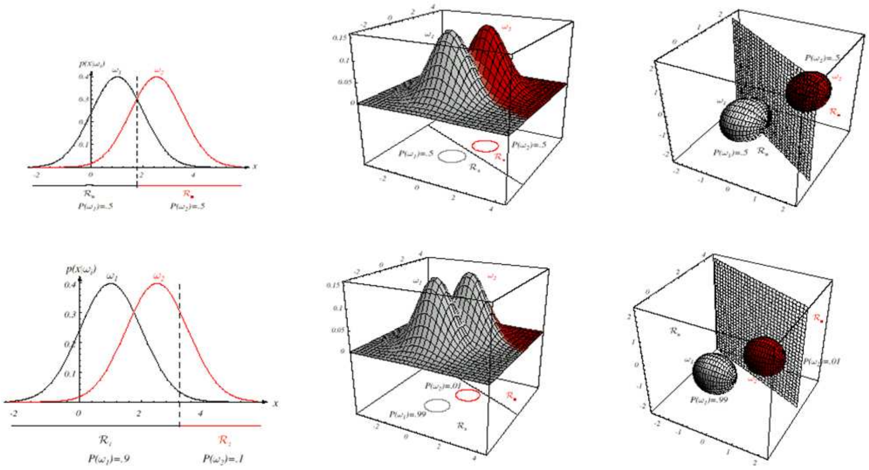
$$g_i(\vec{x}) = g_j(\vec{x}) \Rightarrow (\mu_i - \mu_j)^t(\vec{x} - \vec{x}_0) = 0$$

a hyperplane thru x_0

$$\text{where} \quad \vec{x}_0 = \frac{1}{2}(\mu_i + \mu_j) - \frac{\sigma^2}{\|\mu_i - \mu_j\|^2}\log\frac{P(\omega_i)}{P(\omega_j)}(\mu_i - \mu_j)$$

Discriminant Functions for Normal Probability Density (4/7)

<u>Case 1 :</u> $\Sigma_i = \sigma^2 I$ (independence, equal σ)



Discriminant Functions for Normal Probability Density (5/7)

<u>Case 2 :</u> $\Sigma_i = \Sigma$ (arbitrary & identical Σ)

$$g_i(\vec{x}) = -\frac{1}{2}\left[(\vec{x}-\vec{\mu}_i)^t\,\Sigma^{-1}(\vec{x}-\vec{\mu}_i)\right] + \log P(\omega_i)$$

$$g_i(\vec{x}) = \vec{w}_i^t\vec{x} + w_{i0} \quad where \quad \vec{w}_i = \Sigma^{-1}\vec{\mu}_i\,, w_{i0} = -\frac{1}{2\sigma^2}\vec{\mu}_i^{\,t}\,\Sigma^{-1}\vec{\mu}_i + \log P(\omega_i)$$

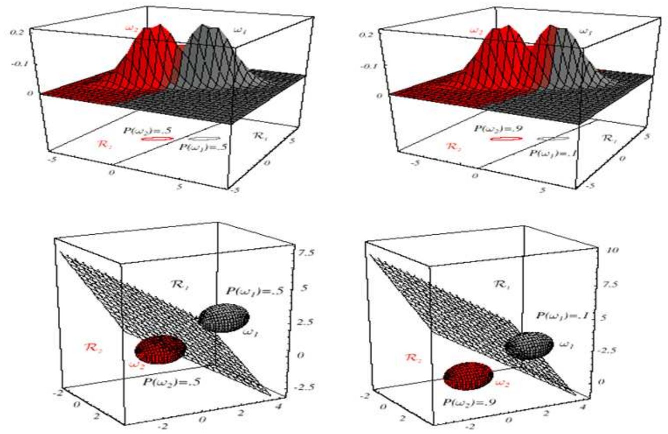
Decision boundary :

$$g_i(\vec{x}) = g_j(\vec{x}) \quad \Rightarrow \quad (\mu_i - \mu_j)^t(\Sigma^{-1})^t(\vec{x}-\vec{x}_0) = 0$$

A hyperplane thru x_0

Discriminant Functions for Normal Probability Density (6/7)

<u>Case 2 :</u> $\Sigma_i = \Sigma$ (arbitrary & identical Σ)



Discriminant Functions for Normal Probability Density (7/7)

<u>Case 3 :</u> Σ_i (arbitrary Σ_i)

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

Decision boundary is a hyperquadric