Optimum Statistical Classifiers

Bayes Classifiers

Computer Assisted Image Analysis, Spring 2009

Criterion Function

Make the decision that yields the lowest probability of committing an error.

\[ p(w_i|x) \] - Probability that pattern \( x \) comes from class \( w_i \).

\( L_{ij} \) - The cost (loss) when putting \( x \) in \( w_j \) when it belongs to \( w_i \).

These are combined to form the conditional average risk (or loss)

\[ r_j(x) = \sum_{k=1}^{W} L_{kj} p(w_k|x). \quad (1) \]

Assign \( x \) to the class \( w_j \) that gives the smallest risk. The classifier that minimises the total average loss is called the Bayes classifier. Thus the Bayes classifier assigns an unknown pattern \( x \) to class \( w_i \) if \( r_i(x) \leq r_j(x) \) for \( j = 1, 2, ..., W; j \neq i \), where \( W \) is the number of classes.

Use of Bayes Rule

From statistics we have Bayes rule

\[ p(A|B) = p(A) \cdot \frac{p(B|A)}{p(B)}. \quad (2) \]

We use Equation 2 to rewrite Equation 1 as

\[ r_j(x) = \frac{1}{p(x)} \sum_{k=1}^{W} L_{kj} p(x|w_k) P(w_k), \quad (3) \]

where \( p(x|w_k) \) is the probability density function of class \( w_k \) and \( P(w_k) \) is the probability of class \( w_k \) occurring. \( p(x) \) is the same for all classes and will thus not affect the order of the different loss values.
Simplification 1: Assume 0-1 Loss Function

The loss may vary between different decisions. In medicine false negatives (a patient with a disease is classified as healthy) are often more costly than false positives (a healthy patient is classified as having a disease). However, if we assume that all errors are equally costly we get the loss function

$$L_{ij} = 1 - \delta_{ij}, \quad (4)$$

where $\delta_{ij}$ is the Dirac function (1 if $i = j$, otherwise 0). From this assumption we can rewrite our conditional risk function as

$$r_j(x) = \sum_{k=1}^{N} (1 - \delta_{kj})p(x|w_k)P(w_k) = p(x) - p(x|w_j)P(w_j). \quad (5)$$

Since $p(x)$ is the same for all classes, it will not affect the order of the loss values and can therefore be eliminated. We see that Bayes classifier for a 0-1 loss function (Equation 5) is nothing more than computing the discriminant function

$$d_j(x) = p(x|w_j)P(w_j), \quad (6)$$

where $x$ should be assigned to the class whose decision function yields the largest value. Even though we have handled the step of assigning costs to errors, we still need to know the probability $P(w_j)$ and probability density function $p(x|w_j)$ for each class.

Simplification 2: Assume Gaussian Probability Density Functions

We assume that the probability density function $p(x|w_j)$ is a Gaussian for all classes. Then when using only one feature, $x$, we get

$$p(x|w_j) = \frac{1}{\sqrt{2\pi\sigma_j}} \cdot e^{-\frac{(x-m_j)^2}{2\sigma_j^2}} \quad (7)$$

where $m_j$ is the mean of class $w_j$, and $\sigma_j$ is the standard deviation of class $w_j$. Generalised to $n$ features, we get

$$p(x|w_j) = \frac{1}{(2\pi)^{n/2}|C_j|^{1/2}} \cdot e^{-\frac{1}{2}(x-m_j)^T C_j^{-1}(x-m_j)} \quad (8)$$

where $m_j$ now is the mean vector of class $w_j$

$$m_j = E_j(x) \approx \frac{1}{N_j} \sum_{x \in w_j} x, \quad (9)$$
and \( C_j \) is the covariance matrix for class \( w_j \)

\[
C_j = E_j \{(x - m_j)(x - m_j)^T\} \approx \frac{1}{N_j} \sum_{x \in w_j} xx^T - m_jm_j^T
\]

where \( N_j \) is the number of pattern vectors from class \( w_j \). In Figure 1 probability density functions for two 1D pattern classes are shown. The point \( x_0 \) shown is the decision boundary if the two classes are equally likely to occur.

![Probability density functions of two 1D pattern classes.](image)

Taking the natural logarithm on \( d_j(x) \) will not affect the order of the loss values since the logarithm is monotonically increasing. We can therefore use the discriminant function in Equation 6 on the form

\[
d_j(x) = \ln(p(x|w_j)P(w_j)) = -\frac{n}{2} \ln 2\pi - \frac{1}{2} \ln |C_j| - \frac{1}{2} [(x - m_j)^T C_j^{-1} (x - m_j)] + \ln P(w_j)
\]

and since the first term is the same for all classes, and therefore does not affect the order, we can use the form

\[
d_j(x) = \ln P(w_j) - \frac{1}{2} \ln |C_j| - \frac{1}{2} [(x - m_j)^T C_j^{-1} (x - m_j)]
\]

which represents the Bayes decision functions for Gaussian pattern classes under the condition of a 0-1 loss function. The decision functions are hyperquadratics (quadratic functions in n-dimensional space).
Simplification 3: Assume Equal Probability

If we assume that all classes are equally likely to occur we can disregard the term \( \ln P(w_j) \) in Equation 12 and instead use the form

\[
d_j(x) = - \frac{1}{2} \ln |C_j| - \frac{1}{2} [(x - m_j)^T C_j^{-1} (x - m_j)]
\]

which is the discriminant function for the **Maximum Likelihood** classifier. The discriminant creates a *hyperquadric* decision function.

**Example:** When using a 2D pattern vector the covariance matrices will be on the form

\[
C_1 = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \quad C_2 = \begin{bmatrix} E & F \\ G & H \end{bmatrix},
\]

where \( A, B, \ldots, H \) are constants.

This is an optimal classifier if

- we have a 0-1 loss function
- the pattern classes are Gaussian
- all classes are equally likely to occur

Simplification 4: Assume Uncorrelated Patterns

We can simplify the discrimination process further by assuming that the patterns are uncorrelated.

**Example:** When using a 2D pattern vector the covariance matrices will be on the form

\[
C_1 = \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix} \quad C_2 = \begin{bmatrix} C & 0 \\ 0 & D \end{bmatrix},
\]

where \( A, B, C, \) and \( D \) are constants.

Simplification 5: Assume Same Covariance

If we assume that all classes have the same covariance we get the discriminant function

\[
d_j(x) = -(x - m_j)^T C^{-1} (x - m_j)
\]

which will result in linear decision functions with hyperplanes as decision boundaries. The discriminant function is also called the Mahalanobis distance.
**Example:** When using a 2D pattern vector the covariance matrices will be on the form

\[ C_1 = \begin{bmatrix} A & B \\ C & D \end{bmatrix} = C_2 = C, \]

where \( A, B, C, \) and \( D \) are constants.

\[ \square \]

**Simplification 6: Assume Independent Patterns And Same Variance**

We assume that we have independent patterns, and same variance for all patterns \( C_1 = C_2 = I \). This together with Equation 14 yields

\[ d_j(x) = (x - m_j)(x - m_j)^T \]

which is the same as minimising the Euclidean norm \( ||x - m_j||^2 \) (choosing the nearest class).

An simpler form for computation can be obtained by simplifying Equation 15 as

\[ d_j(x) = (x - m_j)(x - m_j)^T = (x^T x - m_j^T x - x^T m_j + m_j^T m_j) \]

which is the same as maximising

\[ (m_j^T (x - \frac{1}{2} m_j) + (x - \frac{1}{2} m_j)^T m_j). \]

Since Equation 17 consists of two equivalent terms, it is sufficient to maximise one of them. The discriminant function then becomes

\[ d_j(x) = x^T m_j - \frac{1}{2} m_j^T m_j \]

which is the discriminant function for the **Minimum Distance** classifier. The discriminant creates a hyperplane or linear decision function.

This is an optimal classifier if

- we have a 0-1 loss function
- the pattern classes are Gaussian
- all classes are equally likely to occur
- all classes have no covariance, and equal variance