

第 3 章 模式相似性测度

本章要点:

- ☑ 模式相似性测度的基本概念
- ☑ 距离测度分类法

3.1 模式相似性测度的基本概念

模式识别最基本的研究问题是样品与样品之间或类与类之间相似性测度问题。判断样品之间的相似性常采用近邻准则,即将待分类样品与标准模板进行比较,看跟哪个模板匹配程度更高些,从而确定待测试样品的分类。近邻法则在原理上属于模板匹配。它将训练样品集中的每个样品都作为模板,用测试样品与每个模板做比较,看与哪个模板最相似(即为近邻),就将最近似的模板的类别作为自己的类别。计算模式相似性测度有欧式距离、马氏距离、夹角余弦距离、Tanimoto 测度等多种距离算法。依照近邻准则进行分类通常有两种计算方法,一种方法是通过与样本库所有样品特征分别做相似性测度,找出最接近的样品,取该样品所属类别作为待测样品的类别。另一种方法是与样本库中不同类别的中心或重心做相似性测度,找出最接近类的中心,以该类作为待测样品的类别。例如,A 类有 10 个训练样品,因此有 10 个模板;B 类有 8 个训练样品,就有 8 个模板。一种方法是:任何一个待测试样品在分类时都与这 18 个模板算一算相似性测度,如果最相似的那个近邻是 B 类中的一个,就确定待测试样品为 B 类;否则为 A 类。另一种方法是:分别求出 A 类和 B 类的中心,待测试样品分别与这两个中心做相似性测度,与哪个类的中心最接近,就将待测样品归为该类。

原理上说近邻法是最简单的。但是近邻法有一个明显的缺点就是计算量大,储存量大,要储存的模板很多,当每个测试样品要对每个模板计算一次相似性测度时,所需的计算时间相对其他方法多一些。

1. 样品与样品之间的距离

设有两个样品 $\mathbf{X}_i$ 、 $\mathbf{X}_j$ 的特征向量分别为

$$\mathbf{X}_i = \begin{pmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{in} \end{pmatrix} = (x_{i1}, x_{i2}, \cdots, x_{in})^T, \quad \mathbf{X}_j = \begin{pmatrix} x_{j1} \\ x_{j2} \\ \vdots \\ x_{jn} \end{pmatrix} = (x_{j1}, x_{j2}, \cdots, x_{jn})^T$$

样品间的距离示意图如图 3-1 所示。这两个样品可能在同一个类中,见图 3-1(a);也可能在不同的类中,见图 3-1(b)。因此,可以计算同一个类内样品与样品之间的距离,也可以计算不同类样品与样品之间的距离。

样品与样品间的距离计算有五种方法,分别是欧氏距离法、马氏距离法、夹角余弦距离法、

二值夹角余弦法和具有二值特征的 Tanimoto 测度法,样品间的距离计算公式见表 3-1。

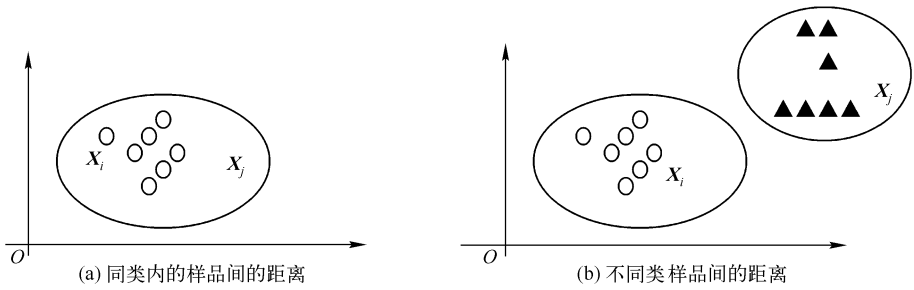


图 3-1 样品间的距离示意图

表 3-1 样品间的距离计算公式

计算距离方法	样品间距离计算公式	说 明
欧氏距离法	$D_{ij}^2 = (X_i - X_j)^T (X_i - X_j) = \ X_i - X_j\ ^2$ $= \sum_{k=1}^n (x_{ik} - x_{jk})^2$	D_{ij} 越小,则两个样品距离越近,就越相似
马氏距离法	$D_{ij}^2 = (X_i - X_j)^T S^{-1} (X_i - X_j)$ $S = \frac{1}{N-1} \sum_{i=1}^N (X_i - \bar{X})(X_i - \bar{X})^T$ $\bar{X} = \frac{1}{N} \sum_{i=1}^N X_i$	D_{ij} 越小,则两个样品距离越近,就越相似
夹角余弦距离法	$S(X_i, X_j) = \cos\theta = \frac{X_i^T X_j}{\ X_i\ \cdot \ X_j\ }$	S 值越大,则相似度越大
二值夹角余弦法	$S(X_i, X_j) = \cos\theta = \frac{X_i^T X_j}{\sqrt{(X_i^T X_i)(X_j^T X_j)}}$	要求 X_i, X_j 向量的各个特征都以二值(0 或 1)表示, S 越大越相似
具有二值特征的 Tanimoto 测度法	$S(X_i, X_j) = \frac{X_i^T X_j}{X_i^T X_i + X_j^T X_j - X_i^T X_j}$	要求 X_i, X_j 向量的各个特征都以二值(0 或 1)表示, S 越大越相似

2. 样品与类之间的距离

样品与类之间的距离如图 3-2 所示。 ω 代表某类样品的集合, ω 中有 N 个样品, X 是某一个待测样品。

样品与类之间距离的计算方法有两种。

① 计算该样品到 ω 类内各个样品之间的距离,将这些距离求和,取平均值作为样品与类之间的距离。样品与类之间的距离可描述为

$$\overline{D^2(X, \omega)} = \frac{1}{N} \sum_{i=1}^N D^2(X, X_i^{(\omega)}) = \frac{1}{N} \sum_{i=1}^N \sum_{k=1}^n |x_k - x_{ik}^{(\omega)}|^2 \tag{3-1}$$

② 计算 ω 类的中心点 $M^{(\omega)}$,以 ω 中的所有样品特征的平均值作为类中心,然后计算待测样品 X 到 ω 的中心点 $M^{(\omega)}$ 的距离。

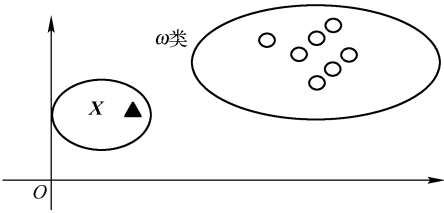


图 3-2 样品与类之间的距离

$$D^2(\boldsymbol{X}, \omega) = D^2(\boldsymbol{X}, \boldsymbol{M}^{(\omega)}) = \sum_{k=1}^n |x_k - m_k^{(\omega)}|^2 \tag{3-2}$$

本书实例均采用式(3-2)作为样品与类之间的距离计算公式。

3. 类内距离

类内距离是指同一个类内任意样品之间距离之和的平均值。 ω 类内的距离如图 3-3 所示，

类内点集 $\{\boldsymbol{X}_i, i = 1, 2, \cdots, N\}$ ，各点之间的内部距离平方为 $\overline{D^2(\{\boldsymbol{X}_i\}, \{\boldsymbol{X}_j\})}$ ， $(i, j = 1, 2, \cdots, N, i \neq j)$ ，从集内一固定点 $\boldsymbol{X}_i$ 到所有其他的 $N - 1$ 个点 $\boldsymbol{X}_j$ 之间的距离平方为 $\overline{D^2(\boldsymbol{X}_i, \{\boldsymbol{X}_j\})} = \frac{1}{N - 1} \sum_{\substack{j=1 \\ j \neq i}}^N \sum_{k=1}^n (x_{ik} - x_{jk})^2$ 。同样的道理，取 ω 内所有点的平均距离表示其类内距离：

$$\overline{D^2(\{\boldsymbol{X}_i\}, \{\boldsymbol{X}_j\})} = \frac{1}{N} \sum_{i=1}^N \left[\frac{1}{N - 1} \sum_{j=1}^N \sum_{k=1}^n (x_{ik} - x_{jk})^2 \right] = \frac{1}{N(N - 1)} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \sum_{k=1}^n (x_{ik} - x_{jk})^2 \tag{3-3}$$

4. 类与类之间的距离

设有两个类 ω_i, ω_j ，类间距离如图 3-4 所示，计算类与类之间的距离有多种方法，例如最短距离法、最长距离法、重心法和平均距离法等，类间的距离计算见表 3-2。

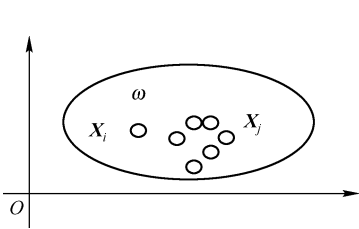


图 3-3 ω 类内的距离

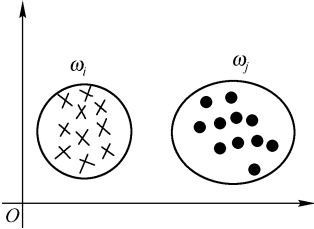


图 3-4 类间距离

表 3-2 类间的距离计算

距离方法	距离定义	说 明
最短距离法	规定两个类间相距最近的两个点之间的距离， 为两类的距离	$D_{i,j} = \min(d_{ij})$ $d_{ij} = \ \boldsymbol{X}_i - \boldsymbol{X}_j \ , \boldsymbol{X}_i \in \omega_i, \boldsymbol{X}_j \in \omega_j$
最长距离法	规定两个类间相距最远的两个点之间的距离， 为两类的距离	$D_{ij} = \max(d_{ij})$ $d_{ij} = \ \boldsymbol{X}_i - \boldsymbol{X}_j \ , \boldsymbol{X}_i \in \omega_i, \boldsymbol{X}_j \in \omega_j$
重心法	将各类中所有样品的平均值作为类的重心，将 两类的重心间的距离作为两类的距离	$D_{i,j} = \ \overline{\boldsymbol{X}^{(\omega_i)}} - \overline{\boldsymbol{X}^{(\omega_j)}} \ $ $\overline{\boldsymbol{X}^{(\omega_i)}} = \frac{1}{N_i} \sum_{\boldsymbol{X} \in \omega_i} \boldsymbol{X} \quad \overline{\boldsymbol{X}^{(\omega_j)}} = \frac{1}{N_j} \sum_{\boldsymbol{X} \in \omega_j} \boldsymbol{X}$ N_i, N_j 分别是 ω_i, ω_j 类中样品的个数
平均距离法	计算两类之间所有样品的距离，求和，取距离 的平均值作为两类间的距离	$D_{i,j} = \frac{1}{N_i N_j} \sum_{\substack{\boldsymbol{X}_i \in \omega_i \\ \boldsymbol{X}_j \in \omega_j}} \ \boldsymbol{X}_i - \boldsymbol{X}_j \ $

3.2 距离测度分类法

3.2.1 模板匹配法

1. 理论基础

最简单的识别方法就是模板匹配法,即把未知样品和一个标准样品模板相比,看它们是否相同或相似。下面讨论两类别和多类别的情况。

(1) 两类别

设有两个标准样品模板为 A 和 B,其特征向量为 n 维特征: $\mathbf{X}_A = (x_{A1}, x_{A2}, \dots, x_{An})^T$ 和 $\mathbf{X}_B = (x_{B1}, x_{B2}, \dots, x_{Bn})^T$ 。任何一个待识别的样品 $\mathbf{X}$,其特征向量为 $\mathbf{X} = (x_1, x_2, \dots, x_n)^T$,那么,它是 A 还是 B 呢?

用模板匹配法来识别,若 $\mathbf{X} = \mathbf{X}_A$,则该样品为 A;若 $\mathbf{X} = \mathbf{X}_B$,则该样品为 B。怎样知道 $\mathbf{X} = \mathbf{X}_A$,还是 $\mathbf{X} = \mathbf{X}_B$ 呢? 最简单的识别方法就是利用距离来判别。如果 $\mathbf{X}$ 距离 $\mathbf{X}_A$ 比距离 $\mathbf{X}_B$ 近,则 $\mathbf{X}$ 属于 $\mathbf{X}_A$,否则属于 $\mathbf{X}_B$ 。这就是最小距离判别法。

任意两点 $\mathbf{X}$ 、 $\mathbf{Y}$ 之间的距离:

$$d(\mathbf{X}, \mathbf{Y}) = \left[\sum_{i=1}^n (x_i - y_i)^2 \right]^{\frac{1}{2}} \quad (3-4)$$

距离远近可作为判据,构成距离分类器,其判别法则为

$$\begin{cases} d(\mathbf{X}, \mathbf{X}_A) < d(\mathbf{X}, \mathbf{X}_B) \Rightarrow \mathbf{X} \in A \\ d(\mathbf{X}, \mathbf{X}_A) > d(\mathbf{X}, \mathbf{X}_B) \Rightarrow \mathbf{X} \in B \end{cases}$$

(2) 多类别

设有 M 个类别: $\omega_1, \omega_2, \dots, \omega_M$ 。每类由若干向量表示,如 ω_i 类,有

$$\mathbf{X}_i = \begin{pmatrix} x_{i1} \\ x_{i2} \\ x_{i3} \\ \vdots \\ x_{in} \end{pmatrix}$$

对于任意被识别的样品 $\mathbf{X}$,有

$$\mathbf{X} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix}$$

计算距离 $d(\mathbf{X}_i, \mathbf{X})$,若存在某一个 i ,使

$$d(\mathbf{X}_i, \mathbf{X}) < d(\mathbf{X}_j, \mathbf{X}), j = 1, 2, \dots, M, i \neq j$$

即到某一个样品最近,则 $\mathbf{X} \in \omega_i$ 。

具体判别时, $\mathbf{X}$ 、 $\mathbf{Y}$ 两点距离可以用 $\|\mathbf{X} - \mathbf{Y}\|^2$ 表示, 即

$$\begin{aligned} d(\mathbf{X}, \mathbf{X}_i) &= \|\mathbf{X} - \mathbf{X}_i\|^2 = (\mathbf{X} - \mathbf{X}_i)^T (\mathbf{X} - \mathbf{X}_i) \\ &= \mathbf{X}^T \mathbf{X} - \mathbf{X}^T \mathbf{X}_i - \mathbf{X}_i^T \mathbf{X} + \mathbf{X}_i^T \mathbf{X}_i \\ &= \mathbf{X}^T \mathbf{X} - (\mathbf{X}^T \mathbf{X}_i + \mathbf{X}_i^T \mathbf{X} - \mathbf{X}_i^T \mathbf{X}_i) \end{aligned} \quad (3-5)$$

式中的 $\mathbf{X}^T \mathbf{X}_i + \mathbf{X}_i^T \mathbf{X} - \mathbf{X}_i^T \mathbf{X}_i$ 为特征的线性函数, 可作为判别函数

$$d_i(\mathbf{X}) = \mathbf{X}^T \mathbf{X}_i + \mathbf{X}_i^T \mathbf{X} - \mathbf{X}_i^T \mathbf{X}_i \quad (3-6)$$

若 $d(\mathbf{X}, \mathbf{X}_i) = \min d_i(\mathbf{X})$, 则 $\mathbf{X} \in \omega_i$ 。这就是多类问题的最小距离分类法。

2. 实现步骤

① 待测样品 $\mathbf{X}$ 与训练集里每个样品 $\mathbf{X}_i$ 的距离为 $d(\mathbf{X}, \mathbf{X}_i) = \|\mathbf{X} - \mathbf{X}_i\|^2$ 。

② 循环计算待测样品和训练集中各已知样品之间的距离, 找出距离待测样品最近的已知样品, 该已知样品的类别就是待测样品的类别。

3. 编程代码

```
%%%
% 函数名称: neartemptlet()
% 参数: sample, 待识别样品特征
% 返回值: y, 待识别样品所属类别
% 函数功能: 按照模板匹配法计算待测样品与样品库中的样品相似度
%%
function y = neartemptlet( sample );

clc;

load templet pattern;% 加载样品库
d = 0;% 距离
min = [ inf, 0 ];
for i = 1:10
    for j = 1:pattern(i). num
        % 计算待测样品与样品库样品间的最小距离
        d = sqrt( sum( ( pattern(i). feature( :, j) - sample' ). ^2 ) );
        % 求最小距离及其类号
        if min(1) > d
            min(1) = d;
            min(2) = i-1;
        end
    end
end

end

% 输出类别
y = min(2);
```

4. 效果图

模板匹配法运行效果如图 3-5 所示。



(a) 待测样品



(b) 分类结果

图 3-5 模板匹配法运行效果

3.2.2 基于 PCA 的模板匹配法

在使用基于 PCA 的模板匹配法之前,先对特征进行主成分分析。按照一定的贡献值,提取前 m 个主分量,用较低维数的特征进行分类。

1. 实现步骤

- ① 选取各类全体样品组成矩阵 $\mathbf{X}_{n \times N}$,待测样品为 $\mathbf{X}_{n \times 1}$ 。
- ② 计算 $\mathbf{X}_{n \times N}$ 的协方差矩阵 $\mathbf{S}_{n \times n}$ 。
- ③ 计算 $\mathbf{S}_{n \times n}$ 的特征值 $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$ 和特征向量 $\mathbf{C}_{n \times n}$ 。
- ④ 根据一定的贡献率,选取 $\mathbf{C}_{n \times n}$ 的前 m 列,构成 $\mathbf{C}_{n \times m}$ 。
- ⑤ 计算样品库样品主成分 $\mathbf{X}_{m \times N} = \mathbf{C}_{n \times m}^T \mathbf{X}_{n \times N}$ 和样品主成分 $\mathbf{X}_{m \times 1} = \mathbf{C}_{n \times m}^T \mathbf{X}_{n \times 1}$ 。
- ⑥ 采用模板匹配法进行多类别分类。

2. 编程代码

```
%%%%%%%%%%
% 函数名称:pcaneartemplet( )
% 参数:sample,待识别样品特征
% 返回值: y,待识别样品所属类别
% 函数功能:按照基于 PCA 的模板匹配法计算待测样品与样品库中的样品相似度
%%%%%%%%%%
function y = pcaneartemplet( sample );
    clc;
    load templet pattern;
    % 对样品和样品库进行主成分分析
    [pcapat,pcasamp] = pcapro( sample );
    temp = 0;
```

```

for i = 1:10
    pattern(i).feature = pcapat( :,temp + 1:temp + pattern(i).num);
    temp = temp + pattern(i).num;
end
d = 0;% 距离
min = [ inf,0];
for i = 1:10
    for j = 1:pattern(i).num
        % 计算待测样品与样品库样品间的最小距离
        d = sqrt( sum( ( pattern(i).feature( :,j)-pcasamp).^2 ));
        % 求最小距离及其类号
        if min(1) > d
            min(1) = d;
            min(2) = i-1;
        end
    end
end
% 输出类别
y = min(2);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 函数名称:pcapro(
% 参数:sample,待识别样品特征
% 返回值: y1,样品库样品经主成分分析后的主分量矩阵;y2,待识别样品经主成分分析后的
% 主分量向量
% 函数功能:对样品库和待测样品用主成分分析法进行降维
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [y1,y2] = pcapro( sample)
    load templet pattern;% 加载样品库
    mixedsig = [];
    sum1 = 0;
    % 将所有类别的所有样品合并到 mixedsig
    for i = 1:10
        sum1 = sum1 + pattern(i).num;
        mixedsig = [ mixedsig pattern(i).feature];
    end
    [ Dim,NumofSampl] = size( mixedsig);% Dim 为特征数,NumofSampl 为样品总个数
    dsig_cov = cov( mixedsig');% 求 mixedsig 的协方差矩阵
    % 利用 pcacov( ) 函数求得从大到小排好序的协方差矩阵的特征值 latent 和相应的特征向量 pc
    [ pc,latent,tsquare] = pcacov( dsig_cov);
    temp = 0;con = 0;m = 0;
    % 根据贡献率取舍特征向量
    sum2 = sum( latent);
    for i = 1:25

```

```

if(con < 0.9)
    temp = temp + latent(i);
    con = temp/sum2;
    m = m + 1;
else
    break;
end
end
pc(:,m+1:25) = [];
% 求待测样品主成分
x = sample * pc;
% 求样品库主成分
y = mixedsig' * pc;
y1 = y';
y2 = x';

```

3. 效果图

基于 PCA 的模板匹配法运行效果如图 3-6 所示。



(a) 待测样品



(b) 分类结果

图 3-6 基于 PCA 的模板匹配法运行效果

3.2.3 马氏距离分类

1. 理论基础

设有 M 个类别: $\omega_1, \omega_2, \dots, \omega_M$ 。如 ω_i 类, 每类有 N_i 个样品, 可表示为 $\mathbf{X}^{(\omega_i)} = (\mathbf{X}_1^{(\omega_i)}, \mathbf{X}_2^{(\omega_i)}, \mathbf{X}_3^{(\omega_i)}, \dots, \mathbf{X}_{N_i}^{(\omega_i)})^T$ 。对于任意待识别的样品 $\mathbf{X} = (x_1, x_2, x_3, \dots, x_n)$, 计算该样品到各类中心的马氏距离 $d^2(\mathbf{X}, \omega_i) = (\mathbf{X} - \overline{\mathbf{X}^{(\omega_i)}})^T \mathbf{S}^{-1} (\mathbf{X} - \overline{\mathbf{X}^{(\omega_i)}})$, 其中 $\overline{\mathbf{X}^{(\omega_i)}}$ 为第 i 类的类中心, $\mathbf{S}$ 为全体样本的协方差矩阵。比较 $\mathbf{X}$ 到各类的距离若满足下式

$$d(\mathbf{X}, \omega_i) < d(\mathbf{X}, \omega_j) \quad j = 1, 2, \dots, M, i \neq j$$

则 $\mathbf{X}$ 到 ω_i 类最近, 则 $\mathbf{X} \in \omega_i$ 。

2. 实现步骤

① 待测样品 X 与训练集里每个类中心的距离采用马氏距离算法计算,计算公式为

$$d^2(X, \omega_i) = (X - \overline{X^{\omega_i}})^T S^{-1} (X - \overline{X^{\omega_i}})$$

② 循环计算待测样品到各类中心的马氏距离,找出距离最小的类别作为该待测样品的类别。

3. 编程代码

```
%%%%%%%%%%
% 函数名称:mahalanobis()
% 参数:sample,待识别样品特征
% 返回值: y,待识别样品所属类别
% 函数功能:按照马氏距离算法计算待测样品与样品库中的样品相似度
%%%%%%%%%
function y = mahalanobis( sample );
   clc;
    load templet pattern;% 加载样品库
    pdata = [ ];
    c = 0;
    % 求协方差矩阵
    for i = 1:10
        for j = 1:pattern(i). num
            c = c + 1;
            pdata( :,c) = pattern(i). feature( :,j);
        end
    end
    % 求特征间的协方差矩阵及其逆矩阵
    s _ cov = cov( pdata' );
    s _ inv = inv( s _ cov );
    d = 0;% 距离
    p = [ ];% 各类别的代表点
    dmin = [ inf,0];
    % 求各类别中值点
    for i = 1:10
        temp = mean( pattern( i). feature' );
        p( :,i) = temp';
    end
    for i = 1:10
        % 计算待测样品与样品库样品间的马氏距离
        d = ( sample' - p( :,i) )' * s _ inv * ( sample' - p( :,i) );
        % 求最小距离及其类号
        if dmin(1) > d
```

```
dmin(1) = d;  
dmin(2) = i - 1;  
end  
end  
% 输出类别  
y = dmin(2);
```

4. 效果图

使用马氏距离算法识别效果如图 3-7 所示。



图 3-7 使用马氏距离算法识别效果

本章小结

本章着重介绍了样品与样品、样品与类、类与类之间的相似性测度计算方法,介绍了模板匹配法、基于 PCA 的模板匹配法、马氏距离分类法等各种距离法的原理与实现步骤。采用距离法分类是本书所介绍的各种分类方法中最简单的一种,分类效果也非常理想。

习题 3

1. 简述模板匹配法的基本原理。
2. 用模板匹配法编程实现英文字符的识别。