深圳大学研究生课程：模式识别理论与方法

课程作业实验报告

实验名称：最小马氏距离分类器

实验编号：Proj02-02

签　　名：

姓　　名：夏荣杰

学　　号：2170269107

截止提交日期：2018 年 4 月 6 日

摘要：马氏距离表示数据的协方差距离，是一种有效的计算两个未知样本集的相似度的方法。与欧氏距离不同的是，它考虑到各种特性之间的联系。本实验首先计算三类给定样本中每一类样本的均值矢量和协方差矩阵（假设每一类均为正态分布），计算马氏距离函数，用不同的颜色在一个图上分别画出这三个类在马氏距离分别为 D=1、2、3 时的图形。接着通过给出的样本数据，设计一个最小马氏距离分类器并对测试点进行分类，然后将其与最小欧式距离分类器进行比较，实验结果表明，当协方差矩阵为单位矩阵时，最小马氏距离分类器与最小欧式距离分类器等价。此外，本实验根据实验样本设计贝叶斯分类器对测试点进行分类，并与最小马氏距离分类器进行比较，比较两种分类器的分类效果，实验表明，距离分类器方法简单，容易实现，贝叶斯分类器考虑先验概率和似然概率，实现了最小错误率意义上的优化，但必须知道先验概率。

一、 背景技术 或 基本原理

1. 估计样本的均值和协方差

假设服从多元高斯分布的样本集合 X 中有 N 个样本，第 k 个样本为 x(k)∈X，则样本集合 X 的均值 m、协方差 S 的估计值由下面的公式给出：

$$m=\frac{1}{N}\sum_{k=1}^{N}x\left(k\right),S=\frac{1}{N}\sum_{k=1}^{N}\left(x\left(k\right)-m\right)\left(x\left(k\right)-m\right)^{T} \quad (1)$$

2. 马氏距离（Mahalanobis Distance）

考虑一个由 d 维向量 x 组成的样本集合 X，若该集合中的样本 x 服从均值为 m，协方差为 S 的多元高斯分布：

$$P\left(x\right)=\frac{1}{\left(2\pi\right)^{d/2}\sqrt{\left|S\right|}}e^{-\frac{1}{2}\left(x-m\right)^{T}S^{-1}\left(x-m\right)} \quad (2)$$

则

$$D\left(x,m\right)=\sqrt{\left(x-m\right)^{T}S^{-1}\left(x-m\right)} \quad (3)$$

称 $D\left(x,m\right)$ 为向量 x 到样本集合 X 的马氏距离（Mahalanobis Distance）。

3. 最小马氏距离分类器

考虑 c 类样本类 $\omega_j, j=1,2,\cdots,c$，均服从均值为 $m\left(j\right)$ 与协方差矩阵为 $S\left(j\right)$ 的多元高斯分布。令分类器函数为：

$$g_j\left(x\right)=-D\left(x,m\left(j\right)\right)=-\sqrt{\left(x-m\left(j\right)\right)^{T}S^{-1}\left(j\right)\left(x-m\left(j\right)\right)}, j=1,2,\cdots,c \quad (4)$$

则

$$x\in\omega_i,\ \text{if}\ \ i=\arg\max_{j}\left\{g_j\left(x\right)\middle|j=1,2,\cdots,c\right\} \quad (5)$$

称上述为最小马氏距离分类器。

4. 欧氏距离（Euclidean distance）

欧氏距离是一个通常采用的距离算法，考虑一个由向量 x 组成的样本集合 X，若该集合中的样本 x 的均值为 m。

则

$$d=\sqrt{\left(x-m\right)^{T}\left(x-m\right)} \quad (6)$$

称 d 为向量 x 到样本集合 X 的欧氏距离（Euclidean distance）。

5. 最小欧氏距离分类器

考虑 c 类样本类 $\omega_j, j=1,2,\cdots,c$，令分类器函数为：

$$g_j\left(x\right)=-d\left(x,m\left(j\right)\right)=-\sqrt{\left(x-m\left(j\right)\right)^{T}\left(x-m\left(j\right)\right)}, j=1,2,\cdots,c \quad (7)$$

则

$$x\in\omega_i,\ \text{if}\ \ i=\arg\max_{j}\left\{g_j\left(x\right)\middle|j=1,2,\cdots,c\right\} \quad (8)$$

称上述为最小欧氏距离分类器。

6. 最小错误率贝叶斯分类器

考虑 c 类样本$\omega_j, j=1,2,\cdots,c$，令分类器函数为：

$$g_j\left(x\right)=P\left(x\,|\,\omega_j\right)P\left(\omega_j\right)\ j=1,2,\cdots,c \tag{9}$$

则

$$x\in\omega_j, if\quad i=\arg\max_j\left\{g_j\left(x\right)\,|\,j=1,2,\cdots,c\right\} \tag{10}$$

称上述为最小错误率贝叶斯分类器。

二、 实验方法 或 算法流程步骤

实验用到的方法和步骤如下：

1. 最小马氏距离分类器

① 利用公式（1）计算三类样本每一类样本的均值矢量和协方差矩阵；

② 根据公式（4）和（5）设计最小马氏距离分类器；

③ 利用公式（3）分别计算 4 个待测试点到三个样本中心的马氏距离；

④ 利用设计的最小马氏距离分类器对 4 个待测试点进行分类。

2. 最小欧氏距离分类器

① 根据公式（7）和（8）设计最小欧氏距离分类器；

② 利用公式（6）分别计算 4 个待测试点到三个样本中心的欧氏距离；

③ 利用设计的最小欧氏距离分类器对 4 个待测试点进行分类。

3. 最小错误率贝叶斯分类器

① 根据公式（7）和（8）设计最小错误率贝叶斯分类器；

④ 利用 matlab 中的函数 mvnpdf()实现 4 个待测试点的条件概率密度计算；

② 利用公式（9）分别计算 4 个待测试点的分类器函数值；

③ 利用设计的最小错误率贝叶斯分类器对 4 个待测试点进行分类。

三、 实验结果

1. 实验(a)结果如下：

三类样本中的各 10 个样本点，假设每一类均为正态分布。用不同的颜色在一个图上分别画出这三个类在马氏距离分别为 D=1、2、3 时的图形，如图 1-1、图 1-2 和图 1-3 所示。

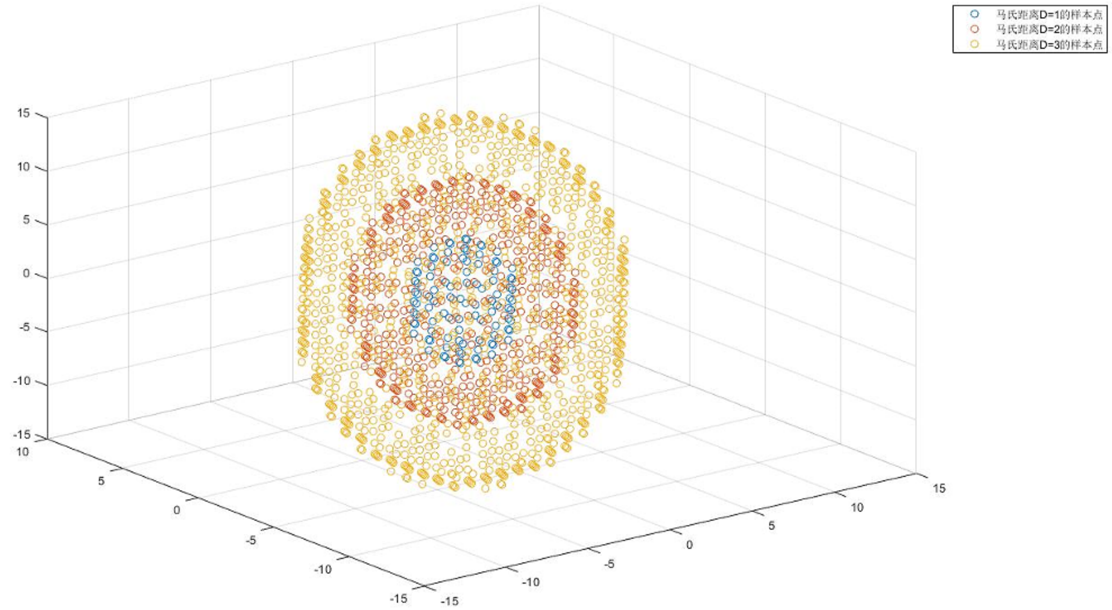

图 1-1. 第一类样本在马氏距离分别为 D=1、2、3 时的图形

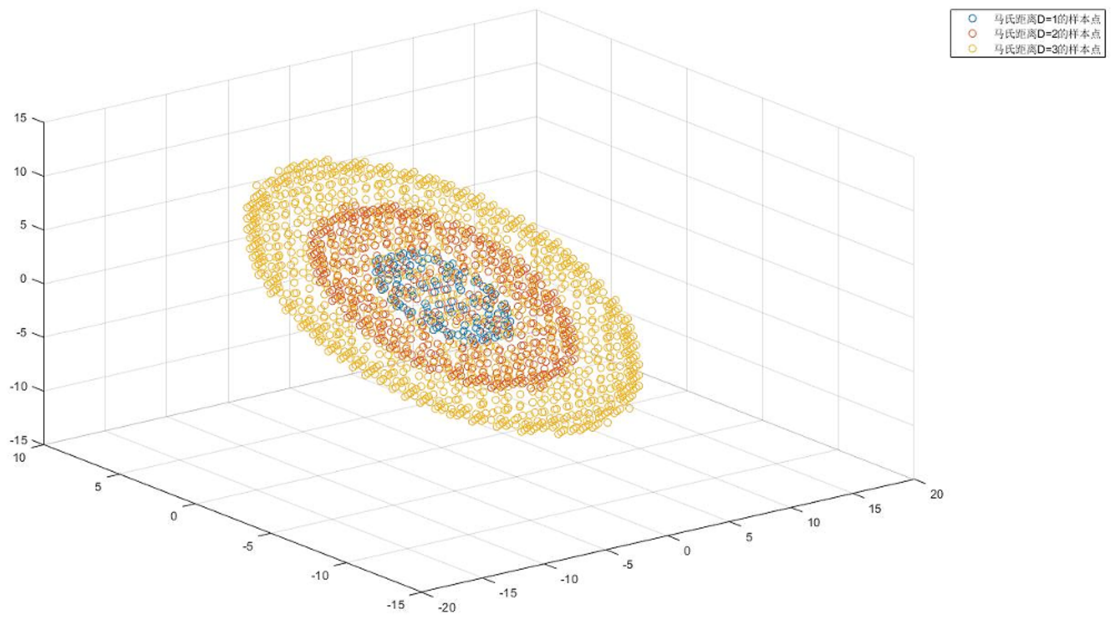

图 1-2. 第二类样本在马氏距离分别为 D=1、2、3 时的图形

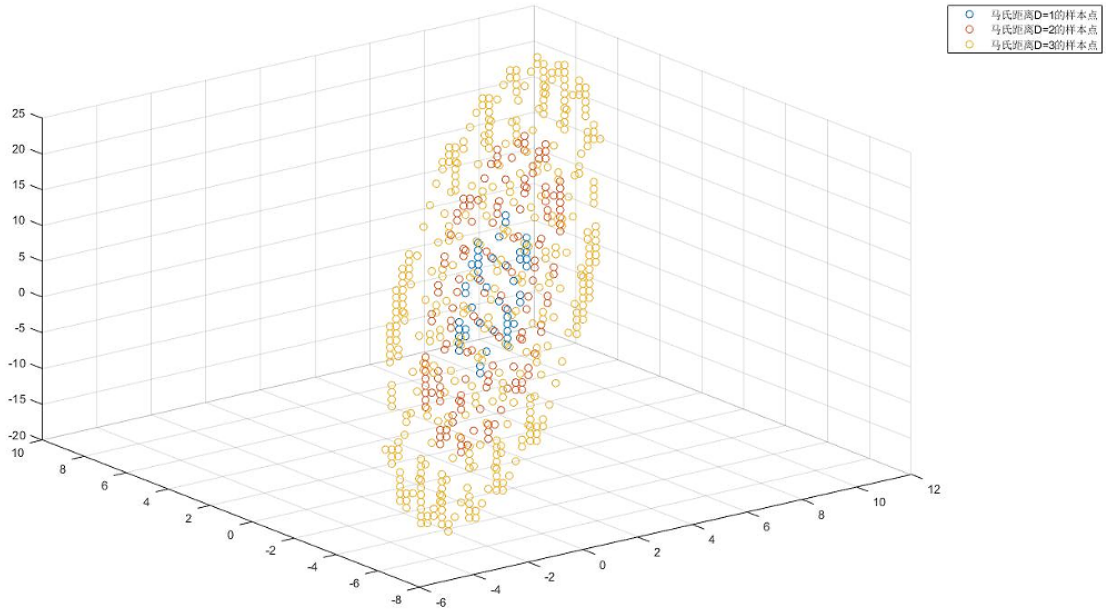

图 1-3. 第三类样本在马氏距离分别为 D=1、2、3 时的图形

由图 1-1、图 1-2 和图 1-3 可知，当马氏距离为一个定值时，它对应的样本点分布图是一个类似椭球的三维几何形状。

2. 实验(b)结果如下：

对测试点：$(1, 2,1)^t$，$(5,3, 2)^t$，$(0,0,0)^t$，$(1,0,0)^t$，分别计算这些点到各个类的马氏距离，如表 1 所示；并用设计好的最小马氏距离分类器对它们进行分类，分类结果如图 2 所示。

表 1. 测试点到各个类的马氏距离

马氏距离	第一类	第二类	第三类
test1	1.0699	**0.8829**	2.8194
test2	1.6414	1.8505	**0.6820**
test3	0.5165	**0.2680**	2.3627
test4	0.5136	**0.4875**	1.5414

```
--利用最小马氏距离分类器对测试样本点进行分类--
---------------测试点1：第2类----------------
---------------测试点2：第3类----------------
---------------测试点3：第2类----------------
---------------测试点4：第2类----------------
```

图 2. 最小马氏距离分类器对测试点的分类结果

对比表 2 和图 2，表 2 中每个测试点的马氏距离最小所对应的类别即为图 2 中经过最小马氏距离分类器的分类结果。

3. 实验(c)结果如下：

计算测试点到各个类中心的欧氏距离，如表 2 所示；使用最小欧氏距离分类器对测试点进行分类，分类结果如图 3 所示。

表 2. 测试点到各个类的欧氏距离

欧氏距离	第一类	第二类	第三类
test1	4.3872	3.3760	**3.0062**
test2	7.7329	7.0696	**2.0153**
test3	1.9594	**0.9597**	4.4122
test4	2.3915	**1.7341**	3.5639

```
--利用最小欧式距离分类器对测试样本点进行分类--
--------------测试点1： 第3类---------------
--------------测试点2： 第3类---------------
--------------测试点3： 第2类---------------
--------------测试点4： 第2类---------------
```

图 3. 最小欧氏距离分类器对测试点的分类结果

对比表 3 和图 3，表 3 中每个测试点的欧氏距离最小所对应的类别即为图 3 中经过最小欧氏距离分类器的分类结果。

4. 实验(d)结果如下：

使用最小错误率贝叶斯分类器对测试点进行分类，分类结果如图 4 所示。

```
--利用最小错误率贝叶斯分类器对测试样本点进行分类--
--------------测试点1： 第2类---------------
--------------测试点2： 第3类---------------
--------------测试点3： 第1类---------------
--------------测试点4： 第1类---------------
```

图 4. 最小错误率贝叶斯分类器对测试点的分类结果

5. 使用不同的分类器的分类结果对比如下：

对比最小马氏距离分类器、最小欧式距离分类器和最小错误率贝叶斯分类器的分类结果如图 5 所示。

	测试样本点1	测试样本点2	测试样本点3	测试样本点4
马氏分类器	2	3	2	2
欧式分类器	3	3	2	2
贝叶斯分类器	2	3	1	1

图 5. 三种分类器对测试点的分类结果对比

四、 讨论与分析

1. 三个样本类在马氏距离分别为 D=1、2、3 时的图形

由图 1-1、图 1-2 和图 1-3 可知，当马氏距离为一个定值时，它对应的样本点分布图是一个类似椭球的三维几何形状。因为由样本的分布函数：

$$P(x)=\frac{1}{(2\pi)^{d/2}\sqrt{|S|}}e^{-\frac{1}{2}(x-m)^T S^{-1}(x-m)}$$

可以看出，当马氏距离为 D（定值）时，该三元高斯分布的概率函数值为一定值 $P=\frac{1}{(2\pi)^{d/2}\sqrt{|S|}}e^{-\frac{1}{2}\sqrt{D}}$，因此，使得该分布函数值为定值 P 时的自变量 x 即为马氏距离为 D 时的样本，在图形上表现为三元高斯分布图形，即三维椭球体，这些椭球体的体积决定了均值附近的样本的离散程度，对于一给定的维数，样本的离散程度直接随协方差矩阵而变化。

2. 最小欧氏距离分类器和最小马氏距离分类器的分类结果对比

实验结果中的图 3 为最小欧氏距离分类器对 4 个测试点的分类结果; 图 2 为最小马氏距离分类器对 4 个测试点的分类结果。对比如下：

分类器 \ 测试点	$(1,2,1)^T$	$(5,3,2)^T$	$(0,0,0)^T$	$(1,0,0)^T$
最小欧氏距离分类器	3	3	2	2
最小马氏距离分类器	2	3	2	2

由上表可知，使用最小欧氏距离分类器和最小马式距离分类器的分类结果不同，这是由于对比两者的计算公式，公式（6）和公式（3），马氏距离较欧式距离多了一个协方差矩阵 S，这个协方差矩阵表示该类别中各个样本之间的关系，添加与否产生不同的分类结果。

若该协方差矩阵为单位矩阵时，马氏距离就变成了欧式距离，其物理意义表示样本中不同特征之间相互独立，在这种情况下，马氏距离分类器等价于欧氏距离分类器。经实验证明确实如此，如下表所示：

分类器 \ 测试点	$(1,2,1)^T$	$(5,3,2)^T$	$(0,0,0)^T$	$(1,0,0)^T$
最小欧氏距离分类器	3	3	2	2
最小马氏距离分类器（S 为单位矩阵）	3	3	2	2

3. 最小错误率贝叶斯分类器和最小马氏距离分类器的分类结果对比

实验结果中的图 4 为最小错误率贝叶斯分类器对 4 个测试点的分类结果; 图 2 为最小马氏距离分类器对 4 个测试点的分类结果。对比如下：

分类器 \ 测试点	$(1,2,1)^T$	$(5,3,2)^T$	$(0,0,0)^T$	$(1,0,0)^T$
最小错误率贝叶斯分类器	2	3	1	1
最小马氏距离分类器	2	3	2	2

由上表可知，使用最小错误率贝叶斯分类器和最小马式距离分类器的分类结果不同，这是由于两者采用的判别函数不一样，最小马式距离分类器使用的是计算向量 x 到每一个类均值向量的马氏距离，将 x 归于离它最近的均值所属的类。而最小错误率贝叶斯分类器使用的是计算 x 的后验概率，选择具有最大后验概率的类作为该对象所属的类。由此可以看出在先验知识的指导下，分类结果产生了改变，但使用贝叶斯分类器必须知道先验概率。

综上所述，实验中使用的最小马氏距离分类器和最小欧式距离分类器都属于距离分类器，而贝叶斯分类器属于概率分类器。欧式距离仅依赖测试点与样本类中心的欧式距离进行分类；马氏距离稍作改进，考虑了样本类中各个样本之间的关系，引入样本的协方差矩阵。可以看出，距离分类器方法简单，概念直观，容易实现。然而，距离分类器没有考虑统计概率和先验知识，所以适用范围有限。贝叶斯分类器引入了先验概率和似然概率，利用贝叶斯公式计算出其后验概率，即该对象属于某一类的概率，选择具有最大后验概率的类作为该对象所属的类。贝叶斯分类器适用范围较广，是一种常用的分类器，但必须知道先验概率。

附录.

```
%%---------------Proj02-02：最小马氏距离分类器-------------%%
%% --计算样本数据均值和协方差矩阵，并分别使用最小马氏距离分类器、最小欧式距离分类器和最小错误率贝叶斯分类器对待测试点进行分类--%%
clear; clc;
%%计算每一类样本的均值矢量和协方差矩阵
N = 10;
c = 3;
%%第一类
w1 = [-5.01 -8.12 -3.68; -5.43 -3.48 -3.54; 1.08 -5.52 1.66; 0.86 -3.78 -4.11; -2.67 0.63 7.39;
    4.94 3.29 2.08; -2.51 2.09 -2.59; -2.25 -2.13 -6.94; 5.56 2.86 -2.26; 1.03 -3.33 4.33];
%%第二类
w2 = [-0.91 -0.18 -0.05; 1.30 -.206 -3.53; -7.75 -4.54 -0.95; -5.47 0.50 3.92; 6.14 5.72 -4.85;
    3.60 1.26 4.36; 5.37 -4.63 -3.65; 7.18 1.46 -6.66; -7.39 1.17 6.30; -7.50 -6.32 -0.31];
%%第三类
w3 = [5.35 2.26 8.13; 5.12 3.22 -2.66; -1.34 -5.31 -9.87; 4.48 3.42 5.19; 7.11 2.39 9.21;
    7.17 4.33 -0.98; 5.75 3.97 6.65; 0.77 0.27 2.41; 0.90 -0.43 -8.71; 3.52 -0.36 6.43];
```

```
%%计算均值
m1 = mean(w1)';
m2 = mean(w2)';
m3 = mean(w3)';
%%计算协方差矩阵
S1 = Cov(w1, m1, N);
S2 = Cov(w2, m2, N);
S3 = Cov(w3, m3, N);
%%作图
figure(1); draw(m1, S1);
figure(2); draw(m2, S2);
figure(3); draw(m3, S3);
%%利用最小马氏距离分类器对测试点进行分类
test1 = [1 2 1]; test2 = [5 3 2]; test3 = [0 0 0]; test4 = [1 0 0];
[g_max1, pre_m1] = Ma_cla(test1, m1, m2, m3, S1, S2, S3, c);
[g_max2, pre_m2] = Ma_cla(test2, m1, m2, m3, S1, S2, S3, c);
[g_max3, pre_m3] = Ma_cla(test3, m1, m2, m3, S1, S2, S3, c);
[g_max4, pre_m4] = Ma_cla(test4, m1, m2, m3, S1, S2, S3, c);
fprintf('--利用最小马氏距离分类器对测试点进行分类--\n');
fprintf('--------------测试点 1: 第%d 类---------------\n', pre_m1);
fprintf('--------------测试点 2: 第%d 类---------------\n', pre_m2);
fprintf('--------------测试点 3: 第%d 类---------------\n', pre_m3);
fprintf('--------------测试点 4: 第%d 类---------------\n\n', pre_m4);
%%利用最小欧式距离分类器对测试点进行分类
[h_max1, pre_e1] = Eu_cla(test1, m1, m2, m3, c);
[h_max2, pre_e2] = Eu_cla(test2, m1, m2, m3, c);
[h_max3, pre_e3] = Eu_cla(test3, m1, m2, m3, c);
[h_max4, pre_e4] = Eu_cla(test4, m1, m2, m3, c);
fprintf('--利用最小欧式距离分类器对测试点进行分类--\n');
fprintf('--------------测试点 1: 第%d 类---------------\n', pre_e1);
fprintf('--------------测试点 2: 第%d 类---------------\n', pre_e2);
fprintf('--------------测试点 3: 第%d 类---------------\n', pre_e3);
fprintf('--------------测试点 4: 第%d 类---------------\n\n', pre_e4);
%%利用最小错误率贝叶斯分类器对测试点进行分类
p_w1 = 1/3; p_w2 = 1/3; p_w3 = 1/3; %先验概率
[f_max1, pre_b1] = Bayes_cla(test1, m1, m2, m3, S1, S2, S3, p_w1, p_w2,
p_w3);
[f_max2, pre_b2] = Bayes_cla(test2, m1, m2, m3, S1, S2, S3, p_w1, p_w2,
p_w3);
[f_max3, pre_b3] = Bayes_cla(test3, m1, m2, m3, S1, S2, S3, p_w1, p_w2,
p_w3);
[f_max4, pre_b4] = Bayes_cla(test4, m1, m2, m3, S1, S2, S3, p_w1, p_w2,
p_w3);
fprintf('--利用最小错误率贝叶斯分类器对测试点进行分类--\n');
```

```
fprintf('--------------测试点 1: 第%d 类---------------\n', pre_b1);
fprintf('--------------测试点 2: 第%d 类---------------\n', pre_b2);
fprintf('--------------测试点 3: 第%d 类---------------\n', pre_b3);
fprintf('--------------测试点 4: 第%d 类---------------\n', pre_b4);

%% -----------将数据生成表格数据--------------- %%
data_row1 = [pre_m1, pre_m2, pre_m3, pre_m4];
data_row2 = [pre_e1, pre_e2, pre_e3, pre_e4];
data_row3 = [pre_b1, pre_b2, pre_b3, pre_b4];
data=[data_row1; data_row2; data_row2];
%%生成表格行列名称，m 行 n 列
str1='测试点';str2='纵轴';
m=3; n=4;
column_name = strcat(str1,num2str((1:n)'));
row_name = {'马氏分类器', '欧式分类器', '贝叶斯分类器'}';
%%表格作图
set(figure(4),'position',[200 200 450 150]);
uitable(gcf,'Data',data,'Position',[20 20 400
100],'Columnname',column_name,'Rowname',row_name);

%% ---------------------子函数------------------------- %%
%%协方差矩阵计算函数
function S = Cov(x, m, N) %%x 为矩阵，m 为向量，N 为样本数目（标量）
S = zeros(size(m, 1));
for i = 1: N
    A = (1 / N) .* ((x(i, :)' - m) * (x(i, :)' - m)');
    S = A + S;
end
end

%%计算马氏距离
function D = Ma_dis(x, m, S)
    D = sqrt((x' - m)' / S * (x' - m));
end

%%马氏距离分布图
function draw(m, S)
    t = 25;
    x = m(1) - t : 1 : m(1) + t;
    y = m(2) - t : 1 : m(2) + t;
    z = m(3) - t : 1 : m(3) + t;
    [X, Y, Z]  = meshgrid(x, y, z);
    %color = ['r', 'g', 'b'];
    for D = 1: 3
```

```
        point = [];
        for j = 1: size(X(:))
            if abs(Ma_dis([X(j), Y(j), Z(j)], m, S) - D) < 0.1
                point = [point, [X(j); Y(j); Z(j)]];
            end
        end
        scatter3(point(1, :), point(2, :), point(3, :));
        hold on;

    end
    legend('马氏距离 D=1 的样本点', '马氏距离 D=2 的样本点', '马氏距离 D=3 的样本
点');
end

%%最小马氏距离分类器
function [g_max, pre_m] = Ma_cla(x, m1, m2, m3, S1, S2, S3, c)
m = [m1, m2, m3];
S = [S1, S2, S3];
g = zeros(c, 1);
for i = 1: c
    g(i) = - Ma_dis(x, m(:, i), S(:, 3*i-2:3*i));
end
[g_max, pre_m] = max(g);
end

%%计算欧式距离
function E = Eu_dis(x, m)
    E = sqrt(sum((x' - m) .* (x' - m)));
end

%%最小欧式距离分类器
function [h_max, pre_e] = Eu_cla(x, m1, m2, m3, c)
m = [m1, m2, m3];
g = zeros(c, 1);
for i = 1: c
    g(i) = - Eu_dis(x, m(:, i));
end
[h_max, pre_e] = max(g);
end

%%最小错误率贝叶斯分类器
function [f_max, pre_b] = Bayes_cla(x, m1, m2, m3, S1, S2, S3, p_w1, p_w2,
p_w3)
p1 = mvnpdf(x, m1', S1); %条件概率
```

```
p2 = mvnpdf(x, m2', S2);
p3 = mvnpdf(x, m3', S3);
g1 = p1 .* p_w1; %分类器函数
g2 = p2 .* p_w2;
g3 = p3 .* p_w3;
[f_max, pre_b] = max([g1; g2; g3]);
end
```