## 贝叶斯分类器的综合设计

**15191011 段易通**

1. **背景介绍**

不同于线性分类器，生成分类器将每一个样本数据视为随机的特征向量通过其分布和密度函数来对其进行分类。为了得到分类结果，应该计算出样本数据的似然函数，得出样本最有可能属于哪一个类，贝叶斯规则为我们提供了一种从密度函数和相关信息计算给定样本的类别可能性的方法，其出发点是利用概率的不同分类决策与相应的决策代价之间的定量折中。

1. **算法设计**
2. **贝叶斯原理**

假设一直先验概率，也知道类条件概率密度，，**X**给定，则后验概率为（贝叶斯公式）

对于决定集，条件风险为

因此，最小的贝叶斯风险决定为

1. **样本数据**

根据题目要求的不同，共有两组样本数据，第一组为实验中给定的两类一维样本数据和决定风险矩阵；第二组采用了MATLAB随机函数生成三类二维数据样本，并自定义了决定风险矩阵。具体见附录。

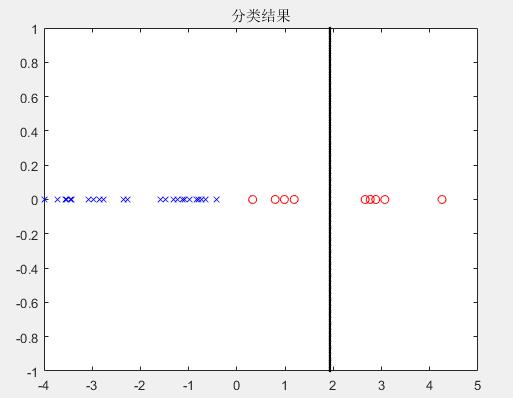
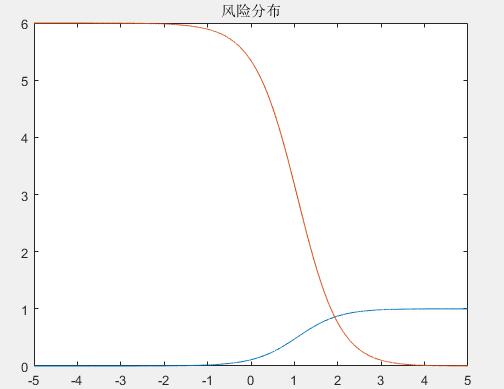
1. **求解思想**

题中给的数据点服从高斯分布，因此先通过概率统计的知识，利用公式

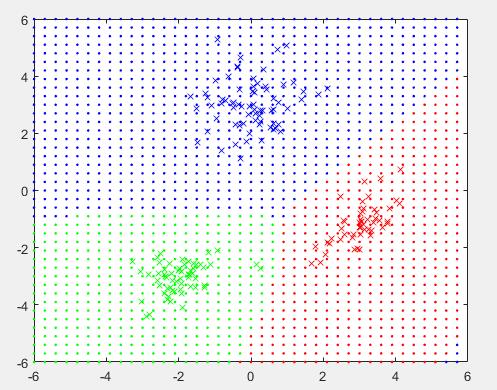
得到样本数据的估计均值和方差，再由此通过高斯分布概率公式得到条件概率

则可通过贝叶斯公式得到后验概率，从而得到条件风险函数，在根据函数的大小就可以判定分类区域。

1. **结果显示**



**一维数据**



**二维数据**

**附录：MATLAB代码**

1. **一维**

clc;

clear;

close all;

action\_risk = [0,1;6,0];

data1 = [-3.9847,-3.5549,-1.2401,-0.9780,-0.7932,-2.8531,-2.7605,-3.7287,-3.5414,...

-2.2692,-3.4549,-3.0752,-3.9934,-0.9780,-1.5799,-1.4885,-0.7431,-0.4221,...

-1.1186,-2.3462,-1.0826,-3.4196,-1.3193,-0.8367,-0.6579,-2.9683];

data2 = [2.8792,0.7932,1.1882,3.0682,4.2532,0.3271,0.9846,2.7648,2.6588];

P1\_prior = 0.9;

P2\_prior = 0.1;

mu1 = mean(data1);

sigma1 = (data1-mu1)\*(data1-mu1)'/length(data1);

mu2 = mean(data2);

sigma2 = (data2-mu2)\*(data2-mu2)'/length(data2);

x = -5:0.001:5;

P1\_conditional = 1/((2\*pi)^0.5\*sigma1)\*exp(-0.5\*((x-mu1)/sigma1).^2);

P2\_conditional = 1/((2\*pi)^0.5\*sigma2)\*exp(-0.5\*((x-mu2)/sigma2).^2);

P1\_posterior = (P1\_conditional\*P1\_prior)./(P1\_conditional\*P1\_prior+P2\_conditional\*P2\_prior);

P2\_posterior = (P2\_conditional\*P2\_prior)./(P1\_conditional\*P1\_prior+P2\_conditional\*P2\_prior);

P\_posterior(1,:) = P1\_posterior;

P\_posterior(2,:) = P2\_posterior;

R = action\_risk\*P\_posterior;

len = length(R(1,:));

for i = 1:len-1

if((R(1,i)>R(2,i))&&(R(1,i+1)<=R(2,i+1)))

boundary = x(i+1);

elseif((R(1,i)<R(2,i))&&(R(1,i+1)>=R(2,i+1)))

boundary = x(i+1);

end

end

figure;

plot(x,R(1,:),x,R(2,:));

title('风险分布');

figure;

plot(data1,0,'bx');

hold on;

plot(data2,0,'ro');

hold on;

plot(boundary,-1:0.01:1,'k.');

title('分类结果');

1. **二维**

clc;

clear;

close all;

num1 = 50; mu1\_prior=[3,-1]; sigma1\_prior=[0.3,0.2;0.2,0.4];

num2 = 60; mu2\_prior=[-2,-3]; sigma2\_prior=[0.4,0.1;0.1,0.3];

num3 = 70; mu3\_prior=[0,3]; sigma3\_prior=[0.6,0;0,0.7];

action\_risk = [0,2,1;1,0,2;2,1,0];

data1 = mvnrnd(mu1\_prior,sigma1\_prior,num1);

data2 = mvnrnd(mu2\_prior,sigma2\_prior,num2);

data3 = mvnrnd(mu3\_prior,sigma3\_prior,num3);

mu1 = mean(data1);

mu2 = mean(data2);

mu3 = mean(data3);

sigma1 = (data1-mu1)'\*(data1-mu1)/length(data1);

sigma2 = (data2-mu2)'\*(data2-mu2)/length(data2);

sigma3 = (data3-mu3)'\*(data3-mu3)/length(data3);

P\_prior = [0.3,0.3,0.4];

x = -6:0.3:6;

len = length(x);

X = zeros(len\*len,2);

for i = 1:len

x0 = x(i)\*ones(len,1);

X((len-1)\*i+1:(len-1)\*i+len,1)=x;

X((len-1)\*i+1:(len-1)\*i+len,2)=x0;

end

P1\_conditional = 1/((2\*pi)^1.5\*det(sigma1))\*exp(-0.5\*(X-mu1)\*sigma1^-1\*(X-mu1)');

P2\_conditional = 1/((2\*pi)^1.5\*det(sigma2))\*exp(-0.5\*(X-mu2)\*sigma2^-1\*(X-mu2)');

P3\_conditional = 1/((2\*pi)^1.5\*det(sigma3))\*exp(-0.5\*(X-mu3)\*sigma3^-1\*(X-mu3)');

PX = P1\_conditional\*P\_prior(1)+P2\_conditional\*P\_prior(2)+P3\_conditional\*P\_prior(3);

P1\_posterior = (P1\_conditional\*P\_prior(1))./(PX);

P2\_posterior = (P2\_conditional\*P\_prior(2))./(PX);

P3\_posterior = (P3\_conditional\*P\_prior(3))./(PX);

P\_posterior(1,:) = diag(P1\_posterior);

P\_posterior(2,:) = diag(P2\_posterior);

P\_posterior(3,:) = diag(P3\_posterior);

R = action\_risk\*P\_posterior;

[Rmin,index] = min(R);

figure;

for i = 1:num1

plot(data1(i,1),data1(i,2),'rx');

hold on;

end

for i = 1:num2

plot(data2(i,1),data2(i,2),'gx');

hold on;

end

for i = 1:num3

plot(data3(i,1),data3(i,2),'bx');

hold on;

end

for i = 1:len\*len-1

switch index(i)

case 1

plot(X(i,1),X(i,2),'r.');

hold on;

case 2

plot(X(i,1),X(i,2),'g.');

hold on;

case 3

plot(X(i,1),X(i,2),'b.');

hold on;

end

end