模式识别上机实验 4. 线性判别函数(一)

专业：<u>信息与计算科学</u> 学号：<u>20131910023</u> 姓名：<u>金洋</u>

假设一个产品有两个参数来衡量它是否合格，我们假设两个参数分别为：

参数 A	参数 B	是否合格
2.95	6.63	合格
2.53	7.79	合格
3.57	5.65	合格
3.16	5.47	合格
2.58	4.46	不合格
2.16	6.22	不合格
3.27	3.52	不合格

实验数据来源：http://people.revoledu.com/kardi/tutorial/LDA/Numerical%20Example.html

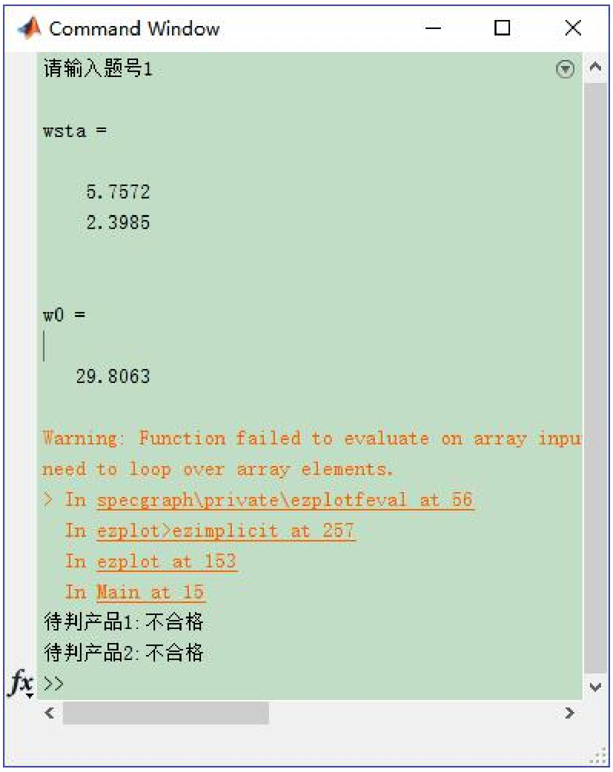
试用 Fisher 判别和感知器判别两种方法求出产品合格与否的判别函数及判别准则，**作出样本**及**其判别边界的图像**并就参数为（2.81，5.46）和（2.90,5.34）的某两个产品判别其所属的类别。

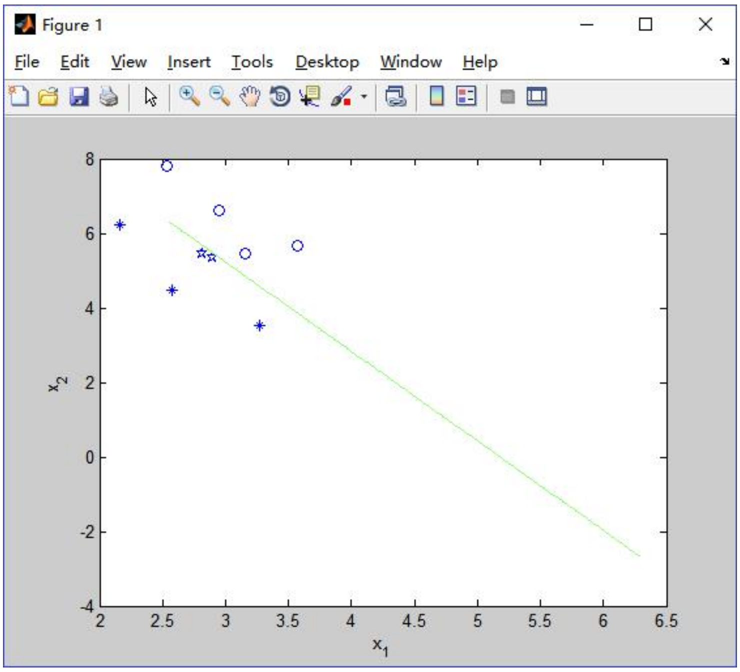
实验判别结果：

①Fisher 判别

决策面函数 $g(x)=5.7572x_1+2.3985x_2-29.8063=0$

判别准则 $\begin{cases} g(x)>0, x\in\omega_1 \\ g(x)\le 0, x\in\omega_2 \end{cases}$



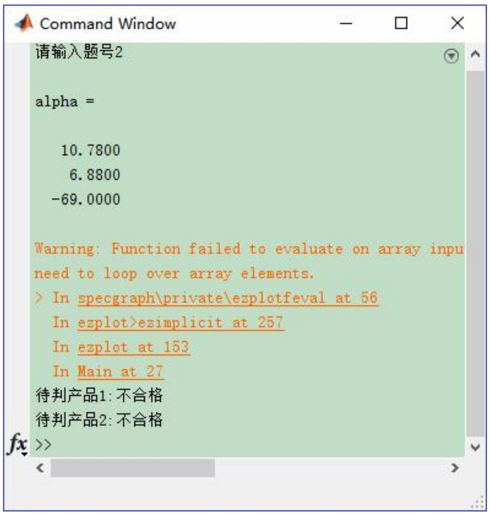


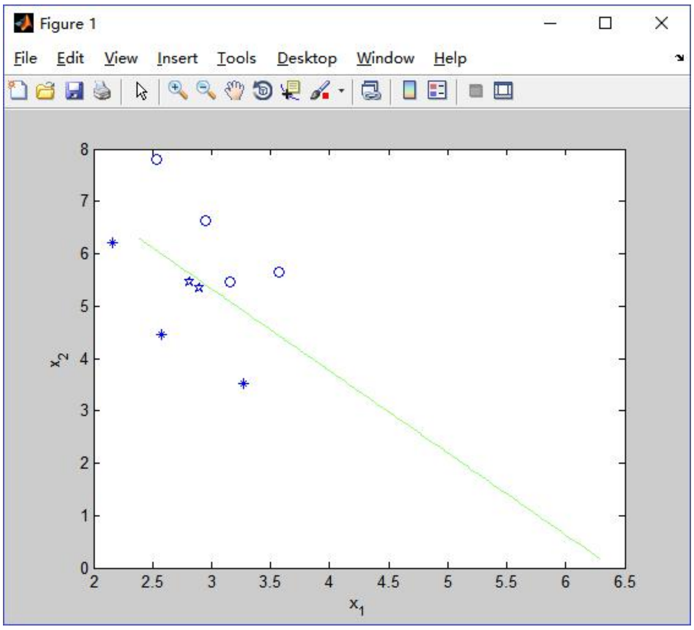
②感知器判别

决策面函数 $g(x)=10.78x_1+6.88x_2-69=0$

判别准则 $\begin{cases} g(x)>0, x\in\omega_1 \\ g(x)\le 0, x\in\omega_2 \end{cases}$

类别中去。





一、算法介绍：

线性判别函数法假定判别函数 $g(x)$ 是 x 的线性函数，即 $g(x)=w^T x+w_0$，对于 c 类问题，可以定义 c 个判别函数，

$$g_i(x)=w_i^T x+w_{i0}, i=1,\ldots,c;$$

我们要用样本去估计 w_i 和 w_{i0}，并把未知样本归类到具有最大判别函数值的类别中去。

（1）Fisher 线性判别

Fisher 线性判别法把 d 维空间的样本投影到一条直线上，形成一维空间，即

把维数压缩到一维。如何根据实际情况找到这条最好的、最易于分类的投影线。这就是 Fisher 法要解决的基本问题。

我们希望投影后，在一维 Y 空间里各类样本尽可能分得开些，即希望两类均值之差越大越好；同时希望样本内部尽量密集，即希望类内离散度越小越好。因此我们可以定义 Fisher 准则函数(相关符号定义见课本)为：

$$J_F(w)=\frac{(\widetilde{m}_1-\widetilde{m}_2)^2}{\widetilde{S}_1^{\ 2}+\widetilde{S}_2^{\ 2}} \tag{1}$$

显然应该寻找使 $J_F(w)$ 的分子尽可能大，分母尽可能小，也就是使 $J_F(w)$ 尽可能大的 w 作为投影方向。

将(1)式化为 w 的显函数，并采用梯度下降法，忽略比例因子，可得最佳投影方向 $w^*=S_\omega^{-1}(m_1-m_2)$。

$w^*=S_\omega^{-1}(m_1-m_2)$ 就是使 Fisher 准则函数 $J_F(w)$ 取极大值时的解，也就是 d 维 X 空间到一维 Y 空间的最好的投影方向。有了 w^*，利用式 $y_n=w^Tx_n$，就可以把 d 维样本 x_n 投影到一维。

d 维分类问题已经转化成一维分类问题了。只要确定一个阈值 y_0，将投影点 y_n 与 y_0 比较，便可做出决策。

对 y_0 的选择有如下几种方式

$$y_0^{(1)}=\frac{\widetilde{m}_1+\widetilde{m}_2}{2}$$

$$y_0^{(2)}=\frac{N_1\widetilde{m}_1+N_2\widetilde{m}_2}{N_1+N_2}$$

这样对于任意给定的未知样本 x，只要计算它的投影点 $y=w^{*T}x$，再根据决策规则 $y \begin{matrix}>\\<\end{matrix} y_0 \rightarrow x\in\begin{cases}\omega_1\\ \omega_2\end{cases}$，就可判定 x 属于什么类别。

（2）感知器准则函数

该算法需要先对样本进行增广和规范化。

设有一组样本 $y_1, y_2, \cdots y_N$ ，其中 y_n 是规范化增广样本向量，我们的目的是找一个解向量 α^* ，使得 $\alpha^T y_n > 0, n = 1,2,\cdots, N$

显然，对于线性可分解情况，问题才有解。为此这里首先考虑处理线性可分解问题的算法。构造一个准则函数：

$$J_P(\alpha) = \sum_{y \in Y^k} (-\alpha^T y)$$

式中 Y^k 是被权向量 α 错分类的样本集。

当且仅当 Y^k 为空集时， $J_P^*(\alpha) = \min J_P(\alpha) = 0$ 这时将不存在错分样本，这时的 α 就是我们要寻找的解向量。

采用梯度下降法求解：

$$\nabla J_P(\alpha) = \frac{\partial J_P(\alpha)}{\partial \alpha} = \sum_{y \in Y^k} (-y) \tag{2}$$

梯度下降法的迭代公式为

$$\alpha(k+1) = \alpha(k) - \rho_k \nabla J \tag{3}$$

将(3)代入(2)得：

$$\alpha(k+1) = \alpha(k) + \rho_k \sum_{y \in Y^k} y$$

不失一般性，令 $\rho_k = 1$，经过简化后的梯度下降算法可以写成

$$\begin{cases} \alpha(1), \text{任意} \\ \alpha(k+1) = \alpha(k) + y^k \end{cases}$$

其中对任何 k 都有 $\alpha(k)^T y^k \le 0$.

二、实验过程：

1、程序源代码

① Main.m

```
clc,clear;
class1=[2.95	6.63;2.53	7.79;3.57	5.65;3.16	5.47]';
class2=[2.58	4.46;2.16	6.22;3.27	3.52]';
in=input('请输入题号');
plot(class1(1,:),class1(2,:),'o')
hold on
```

```
plot(class2(1,:),class2(2,:),'*')
ncl=[2.81 5.46;2.90 5.34]';
plot(ncl(1,:),ncl(2,:),'p')


if (in==1)
    [ wsta,w0 ] = Fisher( class1,class2)
    f =@(x1,x2)   wsta'*[x1;x2]-w0;
    ezplot(f);title('');
    for i=1:size(ncl,2)
        fprintf('待判产品%d:',i);
        if wsta'*ncl(:,i)>w0
            fprintf('合格\n');
        else
            fprintf('不合格\n');
        end
    end
else
    alpha=Percept( class1,class2)
    f =@(x1,x2) alpha(1:2)'*[x1;x2]+alpha(3);
    ezplot(f);title('');
    for i=1:size(ncl,2)
        fprintf('待判产品%d:',i);
        if alpha(1:2)'*ncl(:,i)+alpha(3)>0
            fprintf('合格\n');
        else
            fprintf('不合格\n');
        end
    end
end
hold off
```

② Fisher.m

```
function [ wsta,w0 ] = Fisher( class1,class2)
m1=mean(class1,2);
m2=mean(class2,2);
S1=0;S2=0;
for i=1:size(class1,2)
   S1=S1+(class1(:,i)-m1)*(class1(:,i)-m1)';
end
```

```
for i=1:size(class2,2)
    S2=S2+(class2(:,i)-m2)*(class2(:,i)-m2)';
end
Sw=S1+S2;
Sb=(m1-m2)*(m1-m2)';

wsta=inv(Sw)*(m1-m2);

for i=1:size(class1,2)
    y1(i)= wsta'*class1(:,i);
end
for i=1:size(class2,2)
    y2(i)= wsta'*class2(:,i);
end
m1_=wsta'*m1;
m2_=wsta'*m2;
w0=(m1_+m2_)/2;


end
```

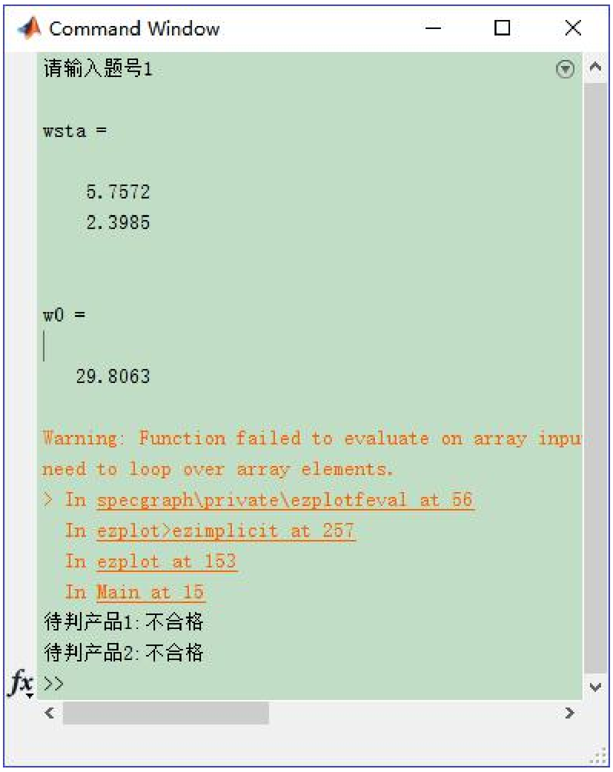
③ Percept.m

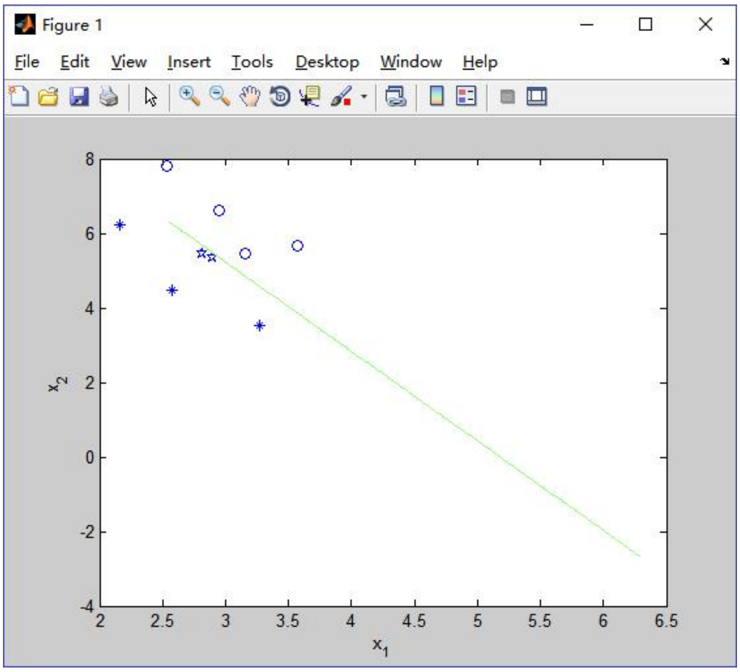
```
function alpha=Percept( class1,class2)%感知准则函数
class1(3,:)=ones(1,size(class1,2));
class2(3,:)=ones(1,size(class2,2));
class2=-class2;
y=[class1,class2];


flag=0;
alpha=ones(3,1);
while (~flag)
    flag=1;
    for i=1:size(y,2)
        if (alpha'*y(:,i)<=0)
            alpha= alpha+y(:,i);
            flag=0;
        end
    end
```

```
end

end
```

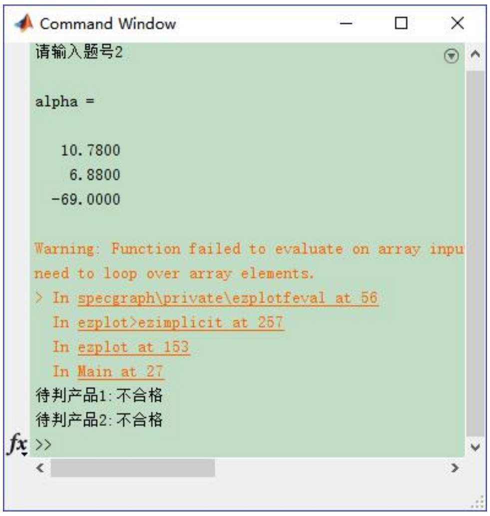
2、运行结果

①Fisher 判别





②感知器判别

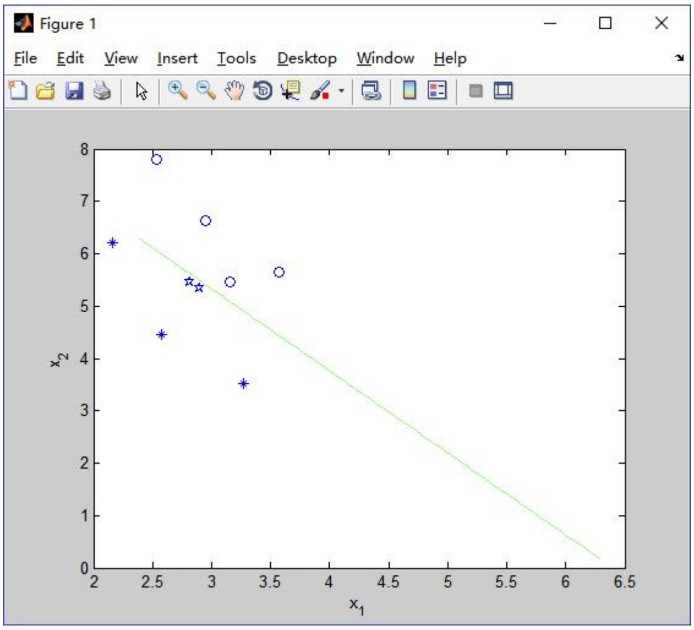

Command Window

请输入题号2

alpha =

   10.7800
    6.8800
  -69.0000

Warning: Function failed to evaluate on array inpu
need to loop over array elements.
> In specgraph\private\ezplotfeval at 56
  In ezplot>ezimplicit at 257
  In ezplot at 153
  In Main at 27
待判产品1:不合格
待判产品2:不合格
>>



三、实验结果

①Fisher 判别

决策面函数 $g(x)=5.7572x_1+2.3985x_2-29.8063=0$

判别准则 $\begin{cases} g(x)>0, x\in\omega_1 \\ g(x)\le 0, x\in\omega_2 \end{cases}$

两待判产品均不合格。

②感知器判别

决策面函数 $g(x)=10.78x_1+6.88x_2-69=0$

判别准则 $\begin{cases} g(x)>0, x\in\omega_1 \\ g(x)\le 0, x\in\omega_2 \end{cases}$

两待判产品均不合格。