模式识别基础 HW2

贾成君 2014011552 自46

1. 证明
2. 对w0进行求导，可得

则当导数为0时，最优，可得.

1. 同（1）可得



 其中证毕。

1. 因为 ,可得 ,证毕
2. 解：

使用老师提供的logistics回归函数和自行编写的Fisher线性判别函数进行了50次测试，得到两者的正确率（正确的测试样本数目/总的测试样本数目[总的测试样本为210]）如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 测试 | 1 | 2 | 3 | … | 平均 |
| Logistics回归正确率 | 0.9190 | 0.9714 | 0.9571 |  | 0.9512 |
| Fisher回归正确率 | 0.9667 | 0.9667 | 0.9619 |  | 0.9591 |

可见，两种方案的预测效果都不错，其中Fisher判别可能更加适合。

其中关于Fisher的线性判别函数进行说明——

Fisher判别使用的方法为 ，而的选取根据公式使用贝叶斯决策（参考教材的说明），当先验概率取样本中出现的概率时，预测结果很差；当假设两种类别的概率相等时，预测效果最好。

代码如下：

%HW2

allerr **=** zeros**(**50**,**4**);**

**for** m **=** 1**:**10

allerr**(**m**,:)** **=** main**();**

**end**

**function** er **=** main**()**

load**(**'Breast\_Cancer\_Wisconsin\_data.txt'**);**

InData **=** Breast\_Cancer\_Wisconsin\_data**;**

XAll **=** InData**(:,**2**:**10**);**

YAll **=** **(**InData**(:,**11**)-**2**)** **./**2**;**

nAll **=** size**(**XAll**,**1**);**

nTest **=** uint32**(**0.7**\***nAll**);**

Choice **=** randperm**(**nAll**);**

XTrain **=** XAll**(**Choice**(**1**:**nTest**),:);**

YTrain **=** YAll**(**Choice**(**1**:**nTest**),:);**

XTest **=** XAll**(**Choice**(** **(**nTest**+**1**):**nAll**),:);**

YTest **=** YAll**(**Choice**(** **(**nTest**+**1**):**nAll**),:);**

%% logisticRegress

w0 **=** rand**(**size**(**XTrain**,**2**)+**1**,**1**);**

weight **=** logisticRegressionWeights**(** XTrain**,**YTrain**,**w0**,**1000**,**0.1**);**

res **=** logisticRegressionClassify**(**XTest**,**weight**);**

**[**temp1**,**per1**]** **=** ShowErr**(**YTest**,**res**);**

%% Fisher

**[**Fisherw**,**Fisherw0**]** **=** Fisher**(**XTrain**,**YTrain**);**

res2 **=** FisherClassify**(**XTest**,**Fisherw**,**Fisherw0**);**

**[**temp2**,**per2**]=** ShowErr**(**YTest**,**res2**);**

er **=** **[**per1 per2 temp1 temp2**];**

**end**

%% Show the errors

**function** **[**err**,**percentage**]** **=** ShowErr**(**yReal**,**yCal**)**

errors **=** abs**(**yReal**-**yCal**);**

err **=** sum**(**errors**);**

percentage **=** 1**-**err**/**size**(**yReal**,**1**);**

disp**(**'err='**),**disp**(**err**);**

disp**(**'percentage='**),**disp**(**percentage**);**

**end**

%% Fisher Calculate Function

**function** result **=** FisherClassify**(** xTest**,** w**,** w0 **)**

res **=** w**'\***xTest**'+**w0**;**

result **=** **(**res**<**0**)';**

**end**

%% Fisher Function to Calculate the w and w0 (with inv(S\_w) )

**function** **[**w**,**w0**]** **=** Fisher**(**xTest**,**yTest**)**

% ybar=w'\*x+w0, ybar>0 ==> x<-w1,y=0

%

logic0 **=** **(**yTest **==** 0 **);**

logic1 **=** **(**yTest **==** 1 **);**

x0 **=** xTest**(**logic0**,:);**

x1 **=** xTest**(**logic1**,:);**

m0 **=** mean**(**x0**);**

m1 **=** mean**(**x1**);**

nFeature **=** size**(**m0**',**1**);**

Sw **=** zeros**(**nFeature**,**nFeature**);**

**for** m **=** 1**:**size**(**x0**,**1**)**

temp **=** x0**(**m**,:)** **-**m0**;**

Sw **=** Sw**+**temp**'** **\*** temp**;**

**end**

**for** m **=** 1**:**size**(**x1**,**1**)**

temp **=** x1**(**m**,:)-**m1**;**

Sw **=** Sw**+**temp**'** **\*** temp**;**

**end**

w **=** Sw**\(**m0**-**m1**)';**

w0 **=** **-**1**/**2**\*(**m0**+**m1**)\***w**;**%-log(size(x1,1)/size(x0,1));

**end**