

# Lecture4编程作业

Fisher线性判别代码见: [fisher.py](#)

## 1.Fisher线性判别核心算法

```
def fishertest(train_data, train_label):
    datadim=len(train_data[0])
    data1=train_data[np.where(train_label==1)]
    data2=train_data[np.where(train_label==-1)]
    #数据的均值
    u1 = np.mean(data1, axis=0)
    u2 = np.mean(data2, axis=0)
    cov1=np.zeros((datadim,datadim))
    cov2=np.zeros((datadim,datadim))
    #求数据的协方差矩阵
    for i in range(len(data1)):
        cov1+=np.dot(np.array([data1[i]-u1]).T,np.array([data1[i]-u1]))
    for i in range(len(data2)):
        cov2+=np.dot(np.array([data2[i]-u2]).T,np.array([data2[i]-u2]))
    Sw=cov1+cov2
    Swinv=np.linalg.pinv(Sw)
    w=np.dot(Swinv,np.array([(u1-u2)]).T)
    w=w.T[0]
    s=np.dot(w.T,np.array([(u1+u2)/2]).T)
    #类内总离差阵
    #求逆
    #最佳投影向量
    #分类阈值
    return w, s
```

## 2.

### (1) 数据集

产生两个都具有200个二维向量的数据集  $X_1$  和  $X_2$ 。数据集  $X_1$  的样本来自均值向量  $m_1 = [-5, 0]^T$ , 协方差矩阵  $s_1 = I$  的正态分布, 属于“+1”类, 数据集  $X_2$  的样本来自均值向量  $m_2 = [0, 5]^T$ , 协方差矩阵  $s_2 = I$  的正态分布, 属于“-1”类, 其中  $I$  是一个2\*2的单位矩阵。其中的数据中80%用于训练, 20%用于测试。

### (2) 训练集和测试集上, 两种算法的分类正确率

Fisher线性判别:  $Accuracy_{(in)} = 1.0, Accuracy_{(out)} = 1.0$

### (3) 画出数据集、最佳投影向量和分类阈值

关于作图, 训练集的数据标签为“+1”和“-1”类分别对应红色和蓝色的o点; 测试集的数据标签为“+1”和“-1”分别对应紫红色和青色的x点

绿色的直线为最佳投影向量

黄色的直线与绿色的直线的交点, 即为样本在投影向量上的投影分类阈值

