**机器学习实验报告**

**（二）**

****

姓名：崔玉峰

学号：201600301079

班级：2016级4班

1. **实验目的**

1.考虑不同维数下的高斯概率密度模型。

(a)编写程序,对表格中的类的中的3个特征x。,分别求解最大似然估计和。

(b)修改程序,处理二维数据的情形p(x)~N(μ,E).然后处理对表格中的类的中的

任意两个特征的组合(共3种可能)。

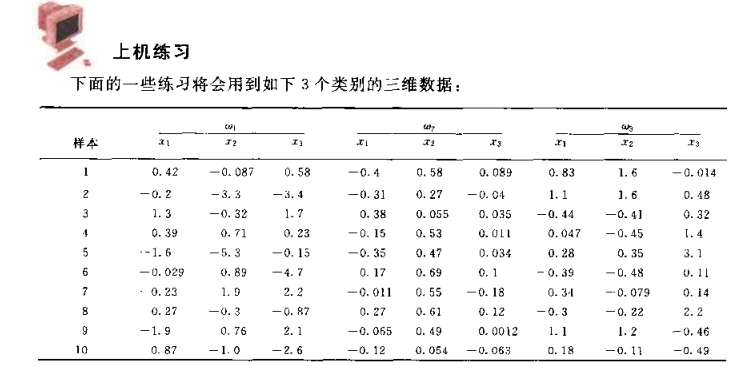
(c)修改程序,处理三維數据的情形p(x)~N(μ,E).然后处理对表格中的类的中3

个特征的组合。

(d)假设这个三维高斯模型是可分离的，即= diag(,,),写一个程序估计类别 w中的均值和协方差矩阵中的3个参数。

(e)比较前4种方式计算出来的每一个特征的均值p的异同。并加以解释。

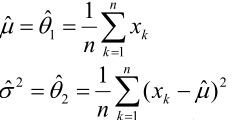
(f)比较前4种方式计算出来的每一一个特征的方差可的异同。井加以解释。



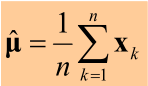
1. **实验思路**
2. **题目a,b,c的求解**

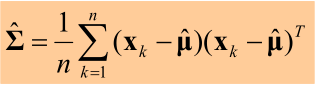
极大似然估计，：

在一维空间下可写为：



在多维空间下：





多维空间下的协方差求解公式同样适用于一维空间下方差求解，所以可以编写统一的求解方法进行求解。

根据这个公式就可以编写代码用极大似然估计求解题目a,b,c，

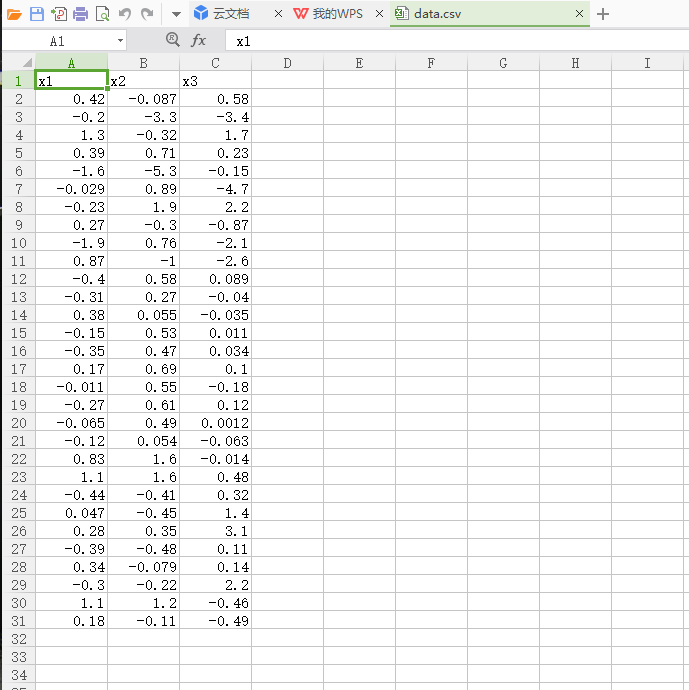
1. **题目d的求解：**

题目d给出了另一个求解协方差矩阵的公式，通过对样本的每一维求解出方差，然后组成对角矩阵当作协方差矩阵。

1. **题目e, f的求解：**

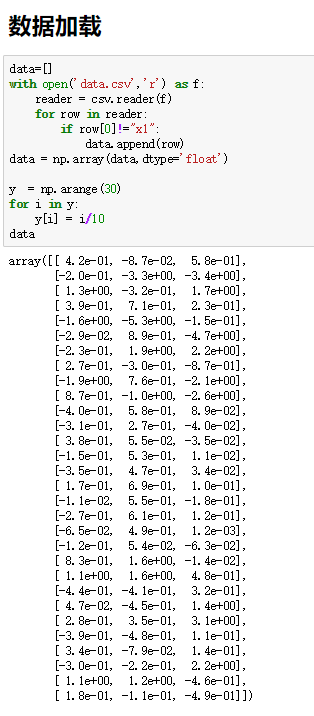
需要观察结果做成分析。

1. **具体实现**
2. **实验环境：**
3. 编程语言：Python
4. 软件环境 ：Jupyter Notebook
5. 硬件环境 ： PC
6. **实验准备：**
7. 将上机实验所用到的数据，手动录入成csv文件方便程序编程使用，可以方便通过程序读入读出。



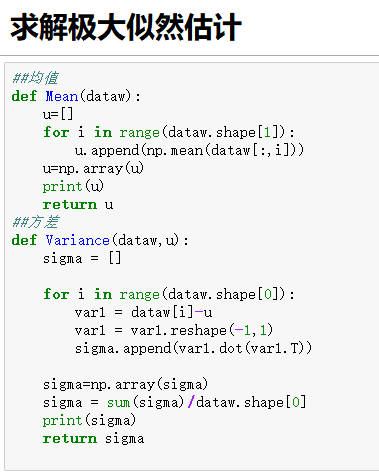
1. 准备Numpy库：方便进行矩阵的运算操作。
2. **实验实现：**
3. **数据加载：**

将数据30条样本读入，每个样本三维（x1,x2,x3）,前十个样本属于类，中间十个样本属于类，最后十个样本属于类，将这三十个样本存入一个数组data中，并用一个数组y存储每个样本的类别。



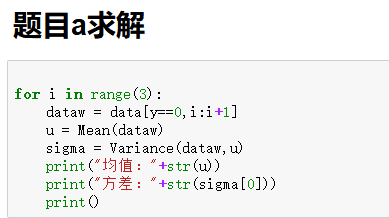
1. **用极大似然估计求解：**

根据公式所描述的编写统一的公式，可以求解任意维度下的均值和方差(协方差)，然后调用方法即可求解

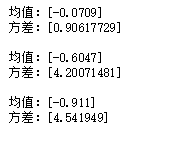


1. **题目a求解**

对样本的三个特征，调用函数分别求均值和方差：

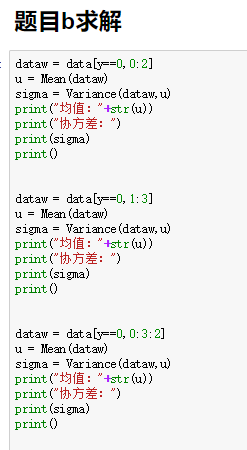


**运行结果：**

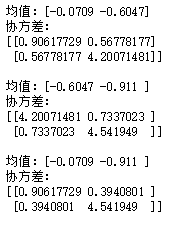


1. **题目b求解**

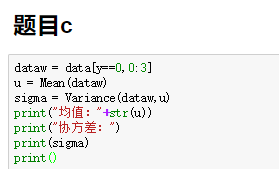
对样本的三个特征中任取两个组成二维数据，调用函数分别求均值和方差：[x1,x2],[x2,x3][x1,x3]



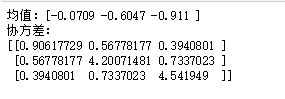
**运行结果：**



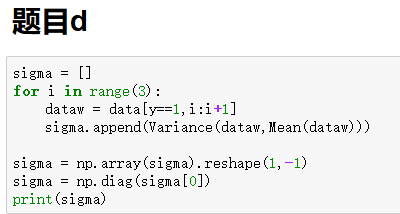
1. **题目c求解**



**运行结果：**



1. **题目d求解:**



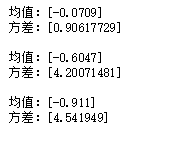
**运行结果：**

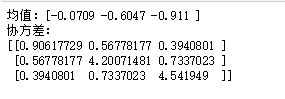
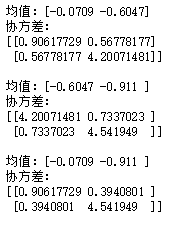


1. **题目e求解:**

通过观察实验结果，前四种方式计算出来的均值，在对应维度上是相同，不会因为维度增加而改变。

以特征为例，一维情况，二维情况，三维情况下计算的结果均为-0.0249.





1. **题目f求解:**

不同维度下计算出的方差不同，n维下的数据计算出的协方差矩阵是n\*n维的，每一个特征下的方差也是n维的。所以不同维度数据计算出的方差不同。

但是通过观察结果可以看到，协方差矩阵在对角线上的值，与在每个特征之下都是相同的。以特征为例，一维情况，二维情况，三维情况下计算的方差在第一行第一列的值都是0.91058409.

1. **实验总结：**

实验的代码，会以ipynb文件的形式上传可以随时进行运行和查看。

实验完成了上机实验的全部题目，并且测试通过，通过实验基本掌握了对于极大似然估计的实现和推导，不同维度下求解极大似然估计出的均值方差之间的异同。极大似然估计是重要的参数估计算法运用广泛。在开始进行实验时进展缓慢，对样本数，样本维数，样本类别数等最基本的概念也搞不清楚，在不断查询资料和回顾老师所讲的知识的过程中进步了很多，最后在弄懂了公式后编写代码，测试结果的速度也就变快，最终完成了整个实验，收获了很多。

1. **实验代码：**

import numpy as np

import csv

import math

data=[]

with open('data.csv','r') as f:

reader = csv.reader(f)

for row in reader:

if row[0]!="x1":

data.append(row)

data = np.array(data,dtype='float')

y = np.arange(30)

for i in y:

y[i] = i/10

##均值

def Mean(dataw):

u=[]

for i in range(dataw.shape[1]):

u.append(np.mean(dataw[:,i]))

u=np.array(u)

return u

##方差

def Variance(dataw,u):

sigma = []

for i in range(dataw.shape[0]):

var1 = dataw[i]-u

var1 = var1.reshape(-1,1)

sigma.append(var1.dot(var1.T))

sigma=np.array(sigma)

sigma = sum(sigma)/dataw.shape[0]

return sigma

for i in range(3):

dataw = data[y==0,i:i+1]

u = Mean(dataw)

sigma = Variance(dataw,u)

print("均值："+str(u))

print("方差："+str(sigma[0]))

print()

dataw = data[y==0,0:2]

u = Mean(dataw)

sigma = Variance(dataw,u)

print("均值："+str(u))

print("协方差："+str(sigma))

print()

dataw = data[y==0,1:3]

u = Mean(dataw)

sigma = Variance(dataw,u)

print("均值："+str(u))

print("协方差："+str(sigma))

print()

dataw = data[y==0,0:3:2]

u = Mean(dataw)

sigma = Variance(dataw,u)

print("均值："+str(u))

print("协方差："+str(sigma))

print()

dataw = data[y==0,0:3]

u = Mean(dataw)

sigma = Variance(dataw,u)

print("均值："+str(u))

print("协方差："+str(sigma))

print()

sigma = []

for i in range(3):

dataw = data[y==1,i:i+1]

sigma.append(Variance(dataw,Mean(dataw)))

sigma = np.array(sigma).reshape(1,-1)

sigma = np.diag(sigma[0])

print(sigma)

sigma = []

for i in range(3):

dataw = data[y==0,i]

sigma.append(Variance(dataw,np.mean(dataw)))

sigma = np.array(sigma).reshape(1,-1)

sigma = np.diag(sigma[0])

print(sigma)

1. **参考文献**

*[美]RichardO.Duda PeterE.Hart DavidG.Stork 著 模式分类 第二版*