



**《数据分析》实验报告**

**学 院**：

**班 级**：

**学 号**：

**姓 名**：

**指导教师**：

教 务 处

2022 年 4 月

题目

1. **正态总体的Bayesian决策分析**

**某医院利用心电图检测来对人群进行划分，数据见表. “g=1”表示健康人，“g=2”表示主动脉硬化患者，“g=3”表示冠心病患者，X1, X2表示测得的心电图中表明心脏功能的两项不相关的指标. 某受试者心电图该两项指标的数据分别为380.20，9.08. 设先验概率按比例分配，进行Bayes判别，判定其归属.**

**提示：首先按照*α=*5%的对三个正态总体的协方差矩阵是否相等进行显著性检验**

**表 24人心电图数据**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **编号** | ***X*1** | ***X*2** | **g** | **编号** | ***X*1** | ***X*2** | **g** |
| **1**  **2**  **3**  **4**  **5**  **6**  **7**  **8**  **9**  **10**  **11**  **12** | **261.01**  **185.39**  **249.58**  **137.13**  **231.34**  **231.38**  **260.25**  **259.51**  **273.84**  **303.59**  **231.03**  **308.90** | **7.36**  **5.99**  **6.11**  **4.35**  **8.79**  **8.53**  **10.02**  **9.79**  **8.79**  **8.53**  **6.15**  **8.49** | **1**  **1**  **1**  **1**  **1**  **1**  **1**  **1**  **1**  **1**  **1**  **2** | **13**  **14**  **15**  **16**  **17**  **18**  **19**  **20**  **21**  **22**  **23**  **24** | **258.69**  **355.54**  **476.69**  **316.12**  **274.57**  **409.42**  **330.34**  **331.47**  **352.50**  **347.31**  **189.59**  **380.20** | **7.16**  **9.43**  **11.32**  **8.17**  **9.67**  **10.49**  **9.61**  **13.72**  **11.00**  **11.19**  **5.46**  **9.08** | **2**  **2**  **2**  **2**  **2**  **2**  **3**  **3**  **3**  **3**  **3**  **待判** |

问题分析：

1.根据g列分类出三类人

2.根据三类人数据计算三个正态总体的协方差矩阵

3.判断协方差矩阵是否相等，显著性检验

4. 设先验概率按比例分配，进行Bayes判别，判定其归属

代码：

import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

from math import log

from numpy.linalg import \*

from scipy.stats.distributions import chi2

data1 = pd.read\_excel("心电图.xlsx")

A = data1[0:-1]

A

B = np.array(data1[-1:])[0][1:-1]

B

g1 = A[0:11]

g2 = A[11:18]

g3 = A[18:]

n = 23

k = 3

p = 2

n1 = 11

n2 = 7

n3 = 5

f = p\*(p+1)\*(k-1)/2

d = (2\*pow(p, 2)+3\*p-1)\*(1/(n-1)+1/(n2-1)+1/(n3-1)-1/(n-k))/(6\*(p+1)\*(k-1))

p1 = n1/n

p2 = n2/n

p3 = n3/n

m1 = np.mean(g1)

m2 = np.mean(g2)

m3 = np.mean(g3)

s1 = np.cov(np.array(g1["X1"]),np.array(g1["X2"]))

s2 = np.cov(np.array(g2["X1"]),np.array(g2["X2"]))

s3 = np.cov(np.array(g3["X1"]),np.array(g3["X2"]))

s = ((n1-1)\*s1+(n2-1)\*s2+(n3-1)\*s3)/(n-k)

M = (n-k)\*log(det(s))-((n1-1)\*log(det(s1))+(n2-1)\*log(det(s2))+(n3-1)\*log(det(s3)))

T = (1-d)\*M

C = chi2.ppf(0.95, f)

if T < C:

print("协方差相等")

else:

print("协方差不相等")

x = B

x

m1 = [m1.loc["X1"],m1.loc["X2"]]

m2 = [m2.loc["X1"],m2.loc["X2"]]

m3 = [m3.loc["X1"],m3.loc["X2"]]

tp1 = m1\*inv(s)\*x-m1\*inv(s)\*np.array(m1).reshape(1,2)/2+log(p1)

tp2 = m2\*inv(s)\*x-m2\*inv(s)\*np.array(m2).reshape(1,2)/2+log(p2)

tp3 = m3\*inv(s)\*x-m3\*inv(s)\*np.array(m3).reshape(1,2)/2+log(p3)

tps = np.array([tp1, tp2, tp3])

tps

i = 1

for tp in tps:

if tp.all() == np.max(tps.all(), axis = 0):

print(i)

i += 1

结果与分析：





实验过程记录：

比较顺畅，通过python将数学公式表达出来进行运算，锻炼了我各类python函数的应用能力。