**广州大学学生实验报告**

**开课学院及实验室：**计算机科学与网络工程学院软件实验室 **2020年 1 月1日**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **学院** | **计算机科学与网络工程学院** | **年级/专业/班** | **软件183** | **姓名** | 黄伟峰 | **学号** | 1806300016 |
| **实验课程名称** | **机器学习与数据挖掘实验** | | | | | **成绩** |  |
| **实验项目名称** | 数据统计与可视化 | | | | | **指导老师** | 彭伟龙 |

**实验二**

**一、实验题目**

基于实验一中清洗后的数据练习统计和视化操作，100个同学（样本），每个同学有11门课程的成绩（11维的向量）；那么构成了一个100x11的数据矩阵。以你擅长的语言C/C++/Java/Python/Matlab，编程计算：

1. 请以课程1成绩为x轴，体能成绩为y轴，画出散点图。
2. 以5分为间隔，画出课程1的成绩直方图。
3. 对每门成绩进行z-score归一化，得到归一化的数据矩阵。
4. 计算出100x100的相关矩阵，并可视化出混淆矩阵。（为避免歧义，这里“协相关矩阵”进一步细化更正为100x100的相关矩阵，100为学生样本数目，视实际情况而定）
5. 根据相关矩阵，找到距离每个样本最近的三个样本，得到100x3的矩阵（每一行为对应三个样本的ID）输出到txt文件中，以\t,\n间隔。

**二、作业环境（文件说明，函数说明，调用的函数库以及涉及哪些技术）**

### 实验环境

作业环境:Python 3.8

**实验导入的模块**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import random

import seaborn as sns

import copy

### 文件说明

AllDate.csv：存放实验一中的数据整理完成后的数据

exp2-5-detail.txt：实验二中题目5的矩阵结果

exp2-4.png:热点图（可视化关系矩阵）

**三、难题与解决**

**在可视化时因为参数不明导致结果十分不明显，一度以为是错误答案，通过多次调整参数进行修正和询问后解决。在写相关代码是因为逻辑不同一度停滞许久，最后还是通过寻找资料解决。**

1. **实验代码运行结果**

**#题四**

**PE\_list=[]**

**for i in range(len(df)):**

**if df.Constitution[i] == "bad":**

**j=random.randint(0,69)**

**PE\_list.append(j)**

**elif df.Constitution[i] == "general":**

**j = random.randint(70, 79)**

**PE\_list.append(j)**

**elif df.Constitution[i] == "good":**

**j = random.randint(80, 89)**

**PE\_list.append(j)**

**elif df.Constitution[i] == "excellent":**

**j = random.randint(90, 100)**

**PE\_list.append(j)**

**list1.append(PE\_list)**

**data1=np.mat(list1) #data1用来存放list1矩阵**

**data1\_mat=data1.T #data1转置就可以得到行表示学生，列表示成绩的矩阵**

**list\_avg=[] #存放每个学生的成绩平均值**

**for i in range(len(data1\_mat)):**

**sum=0 #一个学生的成绩和**

**for j in range(10):**

**sum+=data1\_mat[i,j]**

**list\_avg.append(sum/10.0)**

**# print(len(list\_avg))**

**list\_var=[] #每个学生成绩的标准差**

**for i in range(len(data1\_mat)):**

**sum=0 #一个学生的成绩-平均值然后平方，最后所有项求和**

**for j in range(10):**

**sum+=((data1\_mat[i,j]-list\_avg[i])\*\*2)**

**list\_var.append(np.sqrt(sum/9.0))**

**# print(list\_var)**

**cor\_mat=np.mat(np.zeros((len(data1\_mat),len(data1\_mat)))) #先生成一个以学生数量为阶的全0方阵，之后替换里面元素变为相关矩阵**

**# print(len(cor\_mat[0]))**

**def corvar(i,j):**

**#计算两个学生即两行的协方差**

**#运用公式cov（x，y）=E[（x-E（x)\*(y-E(y))]**

**stu1=[] #存放第i行的成绩-其对应平均值的列表**

**stu2=[] #存放第j行的成绩-其对应平均值的列表**

**for k in range(10):**

**stu1.append(data1\_mat[i,k]-list\_avg[i])**

**stu2.append(data1\_mat[j,k]-list\_avg[j])**

**sum=0 #stu1和stu2元素按照下标对应相乘后加起来**

**for k in range(10):**

**sum+=stu1[k]\*stu2[k]**

**return sum/9.0 #返回协方差**

**# t=corvar(0,1)**

**# cor\_mat[0,1]=t/(list\_var[0]\*list\_var[1])**

**for i in range(len(data1\_mat)):**

**for j in range(len(data1\_mat)):**

**t=corvar(i,j)**

**# print(t)**

**cor\_mat[i,j]=t/(list\_var[i]\*list\_var[j]) #cor\_mat[i,j]表示第i行第j行的相关系数**

**plt.figure(figsize=(20,20),dpi=80)**

**sns.heatmap(cor\_mat,vmin=0,vmax=0.6,linewidths=0.08,xticklabels=False,cmap='coolwarm') #可视化相关矩阵，用热点图**

**plt.savefig('exp2-4.png',dpi=100,bbox\_inches='tight')**

**plt.show()**

**#第五题**

**a=copy.deepcopy(cor\_mat)**

**maxlist=[]**

**id=[]**

**for i in range(len(a)):**

**p=[]**

**l=[]**

**b=np.argsort(a[i],axis=1)**

**p.append(a[i,b[0,len(a)-2]])**

**p.append(a[i, b[0, len(a) - 3]])**

**p.append(a[i, b[0, len(a) - 4]])**

**maxlist.append(p)**

**l.append(df.ID[b[0,len(a)-2]])**

**l.append(df.ID[b[0, len(a) - 3]])**

**l.append(df.ID[b[0, len(a) - 4]])**

**id.append(l)**

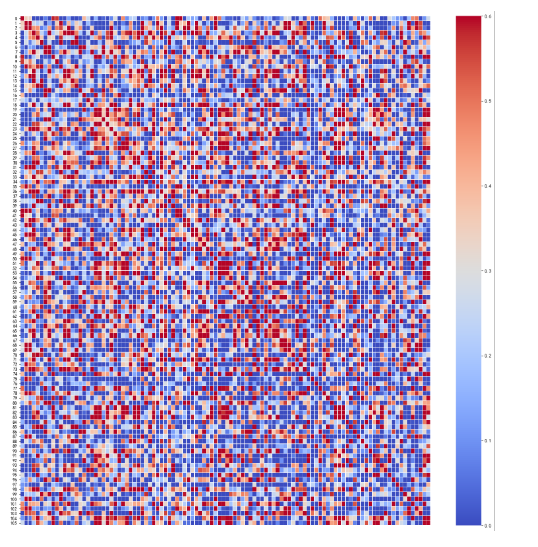
**id\_mat=np.mat(id)**

**dfid=pd.DataFrame(data=id\_mat)**

**print(dfid)**

**dfid.to\_csv('exp2-5-detail.txt',sep='\t',index=False,header=False)**

exp2-4.png



**五、实验总结**

本次实验中个人负责的是第四以及第五题，需要整合上述前面的数据，并形成相关矩阵，在网上查找相关信息和询问后，通过公式和库中提供的一些可视化工具画出相关系数的图形，根据脚标查找数据，故在此处对矩阵的每行数据进行前三次选择，找到最相关的三个同学，记录三人的脚标，再带入脚标至整体文件中查找三人的学号，再进行写入导出。对相邻的数据进行遍历从而形成相关数组。通过实验，加深了对相关矩阵的知识。