

**计算机与信息 学院实验报告**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实验课程： | Python数据分析 | | | | |
| 实验编号： | 实验六 | | | | |
| 实验名称： | numpy数组与矩阵的应用 | | | | |
| 实验人员： | 学号 | 18111207248 | | | |
| 姓名 | 吴钰 | | | |
| 班级 | 18计科创新班 | | | |
| 实验日期： | 2020.11.9 | | | | |
| 实验室： | 2060402 | | | | |
|  |  | | | | |
| 实验评价： |  | | | | |
| 实验成绩： | |  | 评价日期： |  |
|  | 指导教师： | |  | | |

# 一、实验目的

（1） 熟练使用numpy创建矩阵并完成矩阵运算。  
（2） 掌握线性代数模块的运算。

# 二、实验工具

Spyder(Python3.7)

# 三、实验要求

完成实验后请填写实验报告并上交。要求在实验报告中必须记录实验中遇到的问题及其问题解决方案。

# 四、实验内容

（1）生成两个2\*2矩阵，并计算矩阵乘积。

代码：

import numpy as np

#　设置随机种子，保证每次生成的随机数一样，可以不设置（去除下面一行代码，将所有的 rd 替换成 np.random 即可）

rd = np.random.RandomState(888)

matrix1 = rd.randint(1,1000,(2,2))

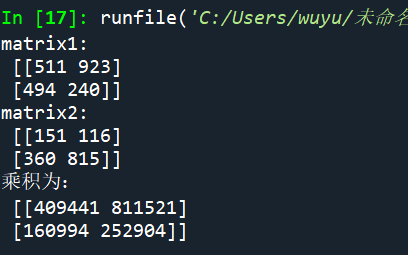
matrix2 = rd.randint(1,1000,(2,2))

c=np.dot(matrix1,matrix2)

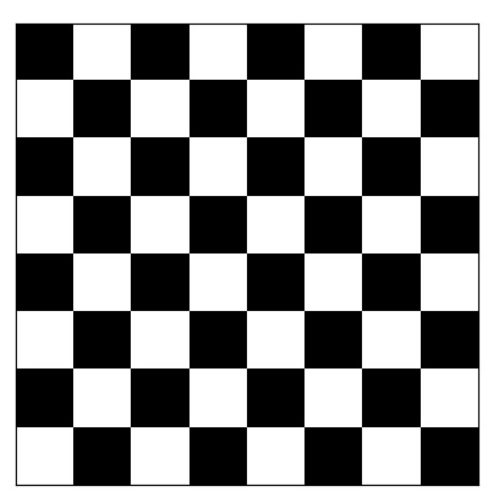
print('matrix1:\n',matrix1,end='\n')

print('matrix2:\n',matrix2,end='\n')

print('乘积为：\n',c)



（2）国际象棋棋盘是个正方形，由横纵向各8格、颜色一黑一白交错排列的64个小方格组成。创建国际象棋棋盘，填充8\*8矩阵。使用matplotlib绘制下方的棋盘图形，并将其保存到d盘，图片命名为自己的学号。



代码：

import numpy as np

from matplotlib import pyplot as plt

import matplotlib

l = 8

arr = np.zeros([l,l])

for i in range(l):

for j in range(l):

if((i+j)%2==0):arr[i][j]=1

color1 = (1,1,1)

color2 = (0,0,0)

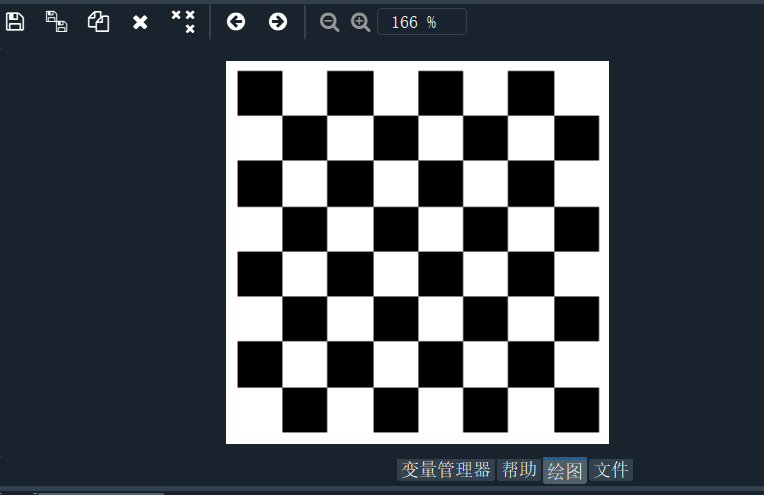
my = matplotlib.colors.LinearSegmentedColormap.from\_list('my', [color1, color2], 2)

cs = plt.imshow(arr, cmap=my)

plt.axis('off')

plt.savefig("D:\\18111207248.jpg")

plt.show()



（3）求矩阵A的伴随矩阵。其中，A为



代码：

import numpy as np

A=np.mat([[1,2,3],[4,5,6],[7,8,9]],dtype=int)

B=np.linalg.inv(A) # 返回自身的逆矩阵(np中的函数)

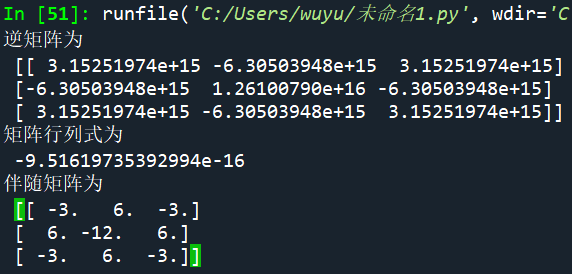
print('逆矩阵为\n',B,end='\n')

A\_abs=np.linalg.det(np.round(A)) #使用linalg.det求得方阵的行列式

print('矩阵行列式为\n',A\_abs,end='\n')

A\_bansui=B\*A\_abs # 求A矩阵的伴随矩阵

print('伴随矩阵为\n',A\_bansui,end='\n')



（4）请读取“朝阳医院2018年销售数据.csv”文件，统计1至7月份每个月份药品的销售数量总额与实际销售金额的总数，并将结果保存到“按月统计.txt”文件中。再请统计每种药品每个月的销售数量与金额，并将结果保存到“统计结果.csv”文件中。

import csv

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

reader = csv.reader(f)

print(type(reader))

for row in reader:

print(row)

代码：

from csv import reader, writer

ideal=[0 for i in range(7)]#销售数量

real=[0 for i in range(7)]

dict={}

flag=0

with open('朝阳医院2018年销售数据.csv', 'r', newline="") as csvfile:

reader = reader(csvfile) # 创建csv文件读对象

# print([row[5] for row in reader])

for row in reader:

if flag!=0:

temp=row[0].split('/')

ideal[int(temp[1])-1]+=float(row[3])

real[int(temp[1])-1]+=float(row[5])

if row[2] not in dict:

dict[row[2]]=[[0 for i in range(7)],[0 for i in range(7)]] #字典映射为二维数组

dict[row[2]][0][int(temp[1])-1]+=int(row[3])

dict[row[2]][1][int(temp[1])-1]+= float(row[5])

else:

dict[row[2]][0][int(temp[1])-1]+=int(row[3])

dict[row[2]][1][int(temp[1])-1] += float(row[5])

flag=1

with open('按月统计.txt', 'w',encoding='utf8') as fp1:

fp1.write('销售总额/实际销售总额\n')

for i in range(7):

fp1.write(str(i+1)+'月'+str(ideal[i])+'/'+str(real[i])+'\n')

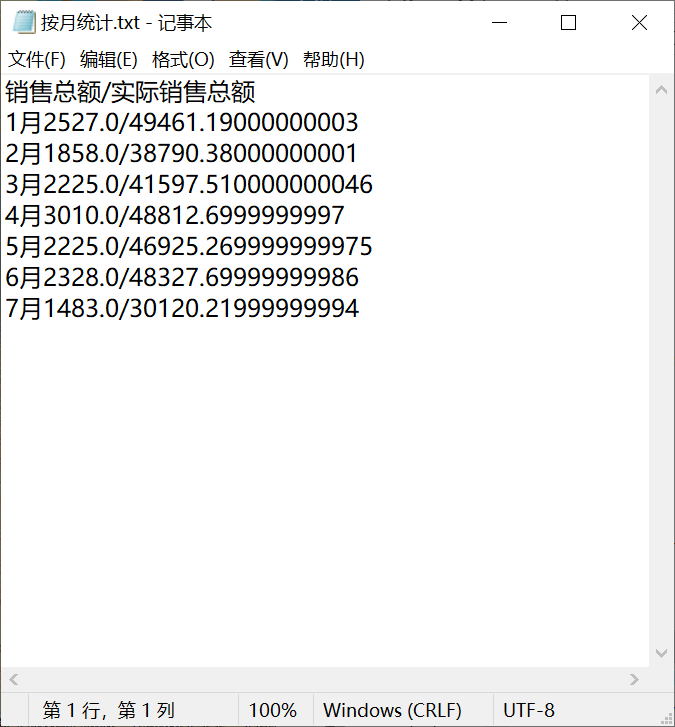
with open('统计结果.csv', 'w',newline="") as fp1:#newline=""解决写入空一行问题

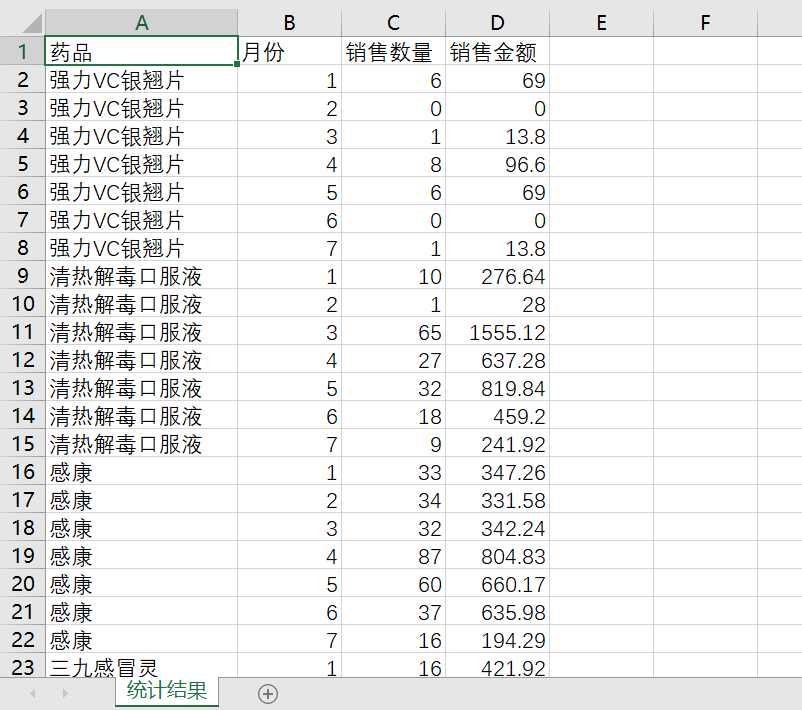
wr = writer(fp1) # 创建csv文件写对象

wr.writerow(['药品', '月份','销售数量', '销售金额'])

for i in dict:

for j in range(7):

wr.writerow([i,str(j+1),dict[i][0][j],dict[i][1][j]])



# 五、实验思考

NumPy 是 Python 语言的一个扩充程序库。支持高效的多数组与矩阵运算，此外也针对数组运算提供大量的数学函数库。NumPy 的科学计算十分高效，因此弥补了 Python 在运算效率上的不足

广播操作是 NumPy 非常重要的一个特点，它允许 NumPy 扩展矩阵间的运算。例如它会隐式地把一个数组的异常维度调整到与另一个算子相匹配的维度以实现维度兼容。所以将一个维度为 [3,2] 的矩阵与一个维度为 [3,1] 的矩阵相加是合法的，NumPy 会自动将第二个矩阵扩展到等同的维度。

为了定义两个形状是否是可兼容的，NumPy 从最后开始往前逐个比较它们的维度大小。在这个过程中，如果两者的对应维度相同，或者其一（或者全是）等于 1，则继续进行比较，直到最前面的维度。若不满足这两个条件，程序就会报错。。

第四题中，我在写统计结果.csv时，一开始写进去的都是乱码，文字很奇怪。后来我查找原因，发现得将with open('统计结果.csv', 'w',newline="" ,encoding='utf8')的encoding='utf8'去掉，就会变成正常的文字。