**实验二 文本数据处理**

**实验目的：**

1、 熟练掌握numpy的基本操作。

2、 掌握ndarray数组的使用，能够创建不同形状和取值的数组，并进行更简单的统计分析。

3、 熟悉通用函数的应用，并使用通用函数进行数组运算。

4、 掌握numpy进行文本数据处理的方法。

**实验内容：**

1. 使用numpy创建一个数组arr1，形状为10行10列，数值为大于0小于100的整型随机数；

import numpy as np

arr1 = np.random.randint(1, 100, size=(10, 10))

print(arr1)

1. 将arr1的形状、维数、元素个数、所占空间大小等属性打印出来。

print("形状: ", arr1.shape)

print("维数: ", arr1.ndim)

print("元素个数: ", arr1.size)

print("所占空间大小: ", arr1.nbytes, "bytes")

1. 将arr1中左上角5行5列的位置变成对角线为50，其余元素为1的数组，右下角5行5列变成对角线为40,30,20,10, 0，其余元素为1的数组。

# 修改左上角5行5列的元素

left\_top = np.multiply(np.eye(5), 50) + np.multiply(np.ones((5, 5), dtype=int), 1 - np.eye(5))

arr1[:5, :5] = left\_top

# 修改右下角5行5列的元素

right\_bottom = np.multiply(np.eye(5, dtype=int), [40, 30, 20, 10, 0]) + np.multiply(np.ones((5, 5), dtype=int), 1 - np.eye(5))

arr1[5:, 5:] = right\_bottom

print(arr1)

1. 使用numpy生成一个服从高斯分布的随机数数组arr2，数组形状为（10，10），均值为0，方差为1。

arr2 = np.random.normal(loc=0, scale=1, size=(10, 10))

print("arr2=",arr2)

1. 将arr2中的每行加上不同的数，其中第一行加上1，第二行加上2，第三行加上3，以此类推。

print("arr2=",arr2)

for i in range(len(arr2)):

    arr2[i,] += i+1

print("arr2\_1=",arr2)

1. 分别将arr1和arr2进行横向和纵向拼接成数组arr3和arr4，并将数组的数据类型转换成整型。

# 横向拼接

arr3 = np.concatenate((arr1, arr2), axis=1)

arr3 = arr3.astype(int)

# 纵向拼接

arr4 = np.concatenate((arr1, arr2), axis=0)

arr4 = arr4.astype(int)

print("arr3=",arr3)

print("arr4=",arr4)

1. 分别将arr3和arr4保存成arr.csv和arr.txt格式的文本文件，保存的格式为整型，数组元素之间的间隔符自己设定。

# 保存为csv文件

np.savetxt('arr3.csv', arr3, fmt='%d', delimiter=',')

# 保存为txt文件

np.savetxt('arr4.txt', arr4, fmt='%d', delimiter=',')

1. 将保存的arr.csv和arr.txt读取出来，分别保存在arr5和arr6中。

# 读取csv文件

arr5 = np.loadtxt('arr3.csv', dtype=int, delimiter=',')

# 读取txt文件

arr6 = np.loadtxt('arr4.txt', dtype=int, delimiter=',')

print("arr5=",arr5)

print("arr6=",arr6)

1. 对arr5按列进行升序排序，对arr6按行进行降序排序。

# 对arr5按列进行升序排序

arr5 = np.sort(arr5, axis=0)

print("arr5\_sort=",arr5)

# 对arr6按行进行降序排序

arr6 = np.flip(np.sort(arr6, axis=1), axis=0)

print("arr6\_sort=",arr6)

1. 已知《实验2数据.csv》为保存的班级成绩，全班共有50人，共有5门课的成绩，其中，每门课都有一个权重，分别为[0.1,0.2,0.2,0.15,0.35]，按要求完成如下分析：
   1. 将数据读取出来，并保存在data数组中；
   2. 在data最后添加一行，表示各科的平均分，同时在最右侧添加一列，表示每位同学的均分；
   3. 若需要从50名同学中，评选5位优秀学生，请设计一种合适的方法，评选出这5位同学（有不及格的同学不参与评选，计算权重分，通过权重分找出权重均分最高并且方差最小的五位）

import numpy as np

import csv

# a) 读取数据

filename = "实验2数据.csv"

data = []

with open(filename, "r") as f:

    for line in f.readlines():

        row = line.strip().split("@#")

        data.append(list(map(float, row)))

data = np.array(data)

# b) 计算各科平均分并添加到data中

average\_scores = np.mean(data, axis=0)

data = np.vstack([data, average\_scores])

# 计算每位同学的平均分并添加到data中

average\_student\_scores = np.mean(data, axis=1)

data = np.column\_stack([data, average\_student\_scores])

# 四舍五入，保留两位小数

data = np.round(data, 2)

# 保存数据到CSV文件

output\_filename = "成绩.csv"

with open(output\_filename, "w", newline="") as f:

    writer = csv.writer(f)

    for row in data:

        writer.writerow(row)

# c) 评选优秀学生

weights = np.array([0.1, 0.2, 0.2, 0.15, 0.35])

passing\_score = 60

# 计算权重分

weighted\_scores = np.dot(data[:-1, :-1], weights)

# 选择及格的同学

passing\_students = np.all(data[:-1, :-1] >= passing\_score, axis=1)

# 计算及格同学的权重分和方差

passing\_weighted\_scores = weighted\_scores[passing\_students]

var\_weighted\_scores = np.var(data[:-1, :-1][passing\_students], axis=1)

# 根据权重分和方差进行评选

evaluation\_scores = passing\_weighted\_scores - var\_weighted\_scores

outstanding\_students = np.argsort(-evaluation\_scores)[:5]

print("优秀学生的序号：", outstanding\_students)