云南大学旅游文化学院信息学院

实验报告

课程名称：数据分析程序语言设计　实验名称：实验 3 pandas 入门

序号 10　学号 20201200737　专业班级：计算机科学与技术一班

第 3 次实验

姓名 平德祥　成绩　代码行统计：共 50 行代码

一、实验目的

（1）熟练掌握 numpy，pandas 包中相关函数的计算与应用

（2）进一步熟悉行内函数、自定义行数等高阶知识点

二、实验环境

python3

三、实验原理

综合实验中用到的函数和点方法，复习填写在此部分

1.np.arange()

函数返回一个有终点和起点的固定步长的排列，如[1,2,3,4,5]，起点是 1，终点是 6，步长为 1。

参数个数情况： np.arange()函数分为一个参数，两个参数，三个参数三种情况

1）一个参数时，参数值为终点，起点取默认值 0，步长取默认值 1。

2）两个参数时，第一个参数为起点，第二个参数为终点，步长取默认值 1。

3）三个参数时，第一个参数为起点，第二个参数为终点，第三个参数为步长。其中步长支持小数

2. np.array(object, dtype=None, copy=True, order='K', subok=False, ndmin=0)

3. **numpy.reshape()** function shapes an array without changing the data of the array.

4.np.zeros():创建全零矩阵

5.np.linspace():序列生成器，避免精度丢失

四、实验步骤

Import numpy as np

Import pandas as pd

1、（1）建立一维数组 a1，内容为 2，4，6，8，10 。

```
a1 = np.arange(2, 11, 2)
print(a1)
```

```
[ 2  4  6  8 10]
```

（2）建立二维数组创建二维数组 a2，数组内容为$\begin{pmatrix}2 & 4\\3 & 5\end{pmatrix}$。

```
a2 = np.array([[2, 4], [3, 5]])
print(a2)
```

```
[[2 4]
 [3 5]]
```

（3）建立二维数组 a3，内容如下所示

array([[3, 4, 5, 6],
[3, 4, 5, 6],
[3, 4, 5, 6],
[3, 4, 5, 6],
[3, 4, 5, 6],
[3, 4, 5, 6],
[3, 4, 5, 6],
[3, 4, 5, 6]])

```
a3 = np.array(list(range(3, 7)) * 8).reshape(8, 4)
print(a3)
```

```
[[3 4 5 6]
 [3 4 5 6]
 [3 4 5 6]
 [3 4 5 6]
 [3 4 5 6]
 [3 4 5 6]
 [3 4 5 6]
 [3 4 5 6]]
```

（4）、创建二维数组 a4，数组内容为$\begin{pmatrix}4 & 5 & 6\\1 & 1 & 1\\0 & 0 & 0\end{pmatrix}$，数组类型为 int32。

```
a4 = np.zeros((3, 3), dtype=int)
a4[1: 2, :3] = 1
a4[0: 1, :3] = range(4, 7)
print(a4)
```

```
[[4 5 6]
 [1 1 1]
 [0 0 0]]
```

2、（1）创建数组 a=(-1,0,1,2,3)、数组 b=(1,1.5,2,2.5,3)；

```
a = np.arange(-1, 4)
b = np.arange(1, 3.5, 0.5)
print(a)
print(b)
```

```
[-1   0   1   2   3]
[1.   1.5 2.   2.5 3. ]
```

（2）计算：b-a，a/b，ba，（a+b)^2，(a+2)/(b-1)；

```
a = np.arange(-1, 4)
b = np.arange(1, 3.5, 0.5)
np.seterr(divide='ignore', invalid='ignore') //异常处理
print(f"b - a:{b - a}")
print(f"a / b:{a / b}")
print(f"b * a:{b * a}")
print(f"(a + b) ^ 2:{np.power((a + b), 2)}")
print(f"(a + 2) / (b - 1):{(a + 2) / (b - 1)}")
```

```
b - a:[2.  1.5 1.  0.5 0. ]
a / b:[-1.   0.   0.5  0.8  1. ]
b * a:[-1.  0.  2.  5.  9.]
(a + b) ^ 2:[ 0.    2.25  9.   20.25 36.  ]
(a + 2) / (b - 1):[       inf 4.         3.         2.66666667 2.5       ]
```

（3）计算：b 中各元素的正弦值；a 中各元素的余弦值,保留小数点后 3 位；a、b 中对应位置元素之和的自然对数，保留 2 位小数；

```
print("sin(b):", np.around(np.sin(b), decimals=3))
print("cos(a):", list(map("{:.3f}".format, np.cos(a))))
c = np.log(a + b)
c[np.isinf(c)] = np.nan   //处理 inf
print("ln(a+b):", list(map("{:.2f}".format, c)))
```

```
sin(b): [0.841 0.997 0.909 0.598 0.141]
cos(a): ['0.540', '1.000', '0.540', '-0.416', '-0.990']
ln(a+b): ['nan', '0.41', '1.10', '1.50', '1.79']
```

3. 完成如下操作：

（1）采用适当方法建立如下矩阵：

$$A=\begin{bmatrix}10 & 13 & 16 & 19\\ 11 & 14 & 17 & 20\\ 12 & 15 & 18 & 21\end{bmatrix} \quad B=\begin{bmatrix}0.1 & 0.2 & 0.3 & 0.4\\ 0.5 & 0.6 & 0.7 & 0.8\\ 0.9 & 1.0 & 1.1 & 1.2\end{bmatrix}$$

```
A = np.array([range(10, 20, 3), range(11, 21, 3), range(12, 22, 3)])
B = np.array([np.linspace(0.1, 0.4, num=4), np.linspace(0.5, 0.8, num=4),
np.linspace(0.9, 1.2, num=4)])
print(A)
print(B)
```

s

```
[[10 13 16 19]
 [11 14 17 20]
 [12 15 18 21]]
[[0.1 0.2 0.3 0.4]
 [0.5 0.6 0.7 0.8]
 [0.9 1.  1.1 1.2]]
```

（2）采用适当方法建立如下矩阵：

$$X1=\begin{bmatrix}2 & 2 & 2 & 2\\ 3 & 3 & 3 & 3\end{bmatrix} \quad X2=\begin{bmatrix}0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix} \quad X3=\begin{bmatrix}1 & 0.1\\ 2 & 0.2\\ 3 & 0.3\\ 4 & 0.4\end{bmatrix} \quad X4=\begin{bmatrix}5 & -3\\ 5 & -3\\ 5 & -3\\ 5 & -3\end{bmatrix}$$

```
X1 = np.array([[2, 2, 2, 2], [3, 3, 3, 3]])
X2 = np.zeros((2, 4), dtype=int)
X3 = np.array([[1, 0.1], [2, 0.2], [3, 0.3], [4, 0.4]])
X4 = np.array([[5, -3], [5, -3], [5, -3], [5, -3]])
print(X1)
print(X2)
print(X3)
print(X4)
```

```
[[2 2 2 2]
 [3 3 3 3]]
[[0 0 0 0]
 [0 0 0 0]]
[[1.  0.1]
 [2.  0.2]
 [3.  0.3]
 [4.  0.4]]
[[ 5 -3]
 [ 5 -3]
 [ 5 -3]
 [ 5 -3]]
```

4. 完成如下操作：

（1）读取 data.xlsx 中的数据，查看数据的情况，并将查看结果汇总于结论处。

代码：

```
data = pd.read_excel(r"E:\python\实验\实验 3data.xlsx")
na_check = data.describe(include='all').loc["count"]
print(na_check[na_check != len(data)])
print(data[data.年龄 < 18])
```

结论：

样本的数量 1200 ，有空值的样本数量 1，有误数据的样本数量 12

```
观点    1196
Name: count, dtype: object
     地区 性别 教育程度   观点  年龄   月收入   月支出
218   A  女    低  不支持  16  2508  2140
346   A  男    低  不支持  10  3172  1979
384   B  女    中   支持  16  1472  1498
433   D  男    低   支持  16  3673  1530
587   B  男    高   支持  15  1658  1955
803   C  女    中  不支持  16  2620  1816
811   B  女    高  不支持  13  3783  1618
875   B  男    高  不支持  12  3046  2176
934   D  男    中  不支持   6  2487  1939
951   B  男    中   支持  12  3654  2017
986   C  女    中   支持  12  3376  2045
991   B  男    高  不支持  16   879  2071
     地区 性别 教育程度   观点  年龄   月收入   月支出
```

（2）利用合适的数据填充空值，并查看填充结果。

```
data = data.fillna(data.观点.describe()["top"])
print(data)
```

```
1195  C  男    高  不支持  51  2268  1315
```

（3）查看数据中男性样本的数量，女性样本中观点表示同意的数量。

```
count1 = data['性别'] == '男'
count2 = (data['性别'] == '女')&(data['观点'] == '支持')
print(len(data[count1]))
print(len(data[count2]))
```

```
603
286
```

五、结果分析

熟悉了两个扩展程序库 numpy 和 pandas，了解了他们基本的点方法，numpy 提供了大量的数学方法，且运算速度很快，主要用于数组运算，而 pandas 主要用于数据分析，可归并，成型，选择，数据清洗等等