

上海大学

公共基础课 数据分析与智能计算 上机报告

作业：第三周

姓名：林艺璿

学号：18120189

学院：计算机工程与科学学院

日期：2020 年 4 月 21 日

第三章 数据汇总和统计

思考与练习 3.1

1. 统计量均值和中位数的区别是什么。如果某样本统计的均值和中位数存在较大差别，说明数据集具有什么特性？

均值：样本（一组数据）的算术平均值，反映数据的集中趋势；中位数：将样本数据从小到大顺序排列，如果样本容量为奇数，处在中间的数是中位数；否则处在最中间两个数的平均值是中位数。中位数的作用与均值类似，反应数据整体特征，但是不受最大、最小两个极端数值的影响。如果某样本统计的均值和中位数存在较大差别，说明数据集存在极端数值或极端数值数大或量多。

思考与练习 3.2

1. 创建并访问 Series 对象。

(1) 创建如下表的 Series 数据对象，其中 a-f 为索引；

a	b	c	d	e	f
30	25	27	41	25	34

(2) 增加数据 27，索引为 g；

(3) 修改索引 d 对应的值为 40；

(4) 查询值大于 27 的数据；

(5) 删除位置为 1-3 的数据。

【提示】位置 1-3 的索引列表，可以用 `series.index[1:3]` 来得到。

```
# 创建并访问Series对象
import pandas as pd
import numpy as np
from pandas import Series, DataFrame
s = Series([30, 25, 27, 41, 25, 34], index = ['a', 'b', 'c', 'd', 'e', 'f'])
print("创建Series数据对象")
print(s)
ns = Series([27], index = ['g'])
s = s.append(ns)
print("增加数据27，索引为g")
print(s)
s['d'] = 40
print("修改索引d对应的值为40")
print(s)
print("查询值大于27的数据")
print(s[s.values > 27])
s = s.drop(s.index[1:3])
print("删除位置为1-3的数据")
print(s)
```

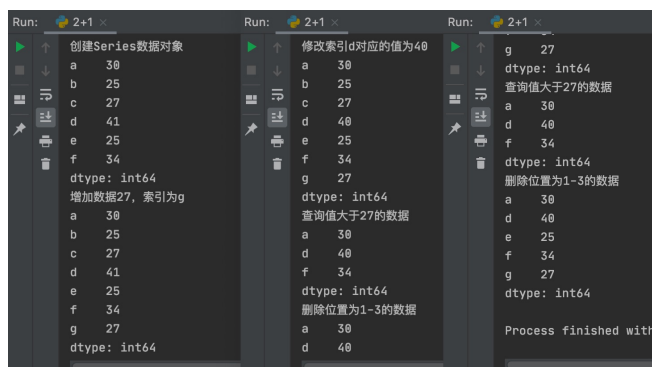


图 1: 思考 2-1 实验结果

2. 创建并访问 DataFrame 对象。

(1) 创建 3×3 DataFrame 数据对象：数据内容为 1-9；行索引为字符 a, b, c；列索引为字符串 one, two, three；

(2) 查询列索引为 two 和 three 两列数据；

(3) 查询第 0 行、第 2 行、第 0 列、第 2 列数据；

(4) 筛选第 1 列中值大于 2 的所有行数据，另存为 data1 对象；

(5) 为 data1 添加一列数据，列索引为 four，值都为 10；

(6) 将 data1 所有值大于 9 的数据修改为 8；

(7) 删除 data1 中第 0 行和第 1 行数据。

【提示】生成数据，使用 numpy 的 `arange()` 函数和 `reshape()` 函数；

使用 `data > 9` 生成布尔型的 DataFrame，用于整个 DataFrame 的数据过滤。

```
# 创建并访问 DataFrame 对象
import pandas as pd
import numpy as np
from pandas import Series, DataFrame
# 创建 3x3 DataFrame 数据对象：数据内容为 1-9；行索引为字符 a, b, c；列索引为字符串 one, two, three
d = np.random.randint(1, 10, size=(3, 3))
df = DataFrame(d, ['a', 'b', 'c'], ['one', 'two', 'three'])
print(df)
print("查询列索引为 two 和 three 两列数据")
print(df.loc[:, ['two', 'three']])
print("查询第 0 行、第 2 行、第 0 列、第 2 列数据")
print(df.iloc[[0, 2], [0, 2]])
i = df.iloc[:, 0] > 2
data1 = df.iloc[list(i)]
print("筛选第 1 列中值大于 2 的所有行数据，另存为 data1 对象")
print(data1)
data1['four'] = 10
print("为 data1 添加一列数据，列索引为 four，值都为 10")
print(data1)
```

```

n = data1 > 9
data1[n.reindex(index=data1.index, columns=data1.columns, fill_value=False)]
    = 8
print("将data1所有值大于9的数据修改为8")
print(data1)
print(data1.index)
data1.drop(data1.index[0:2], axis=0, inplace=True)
print("删除data1中第0行和第1行数据")
print(data1)

```

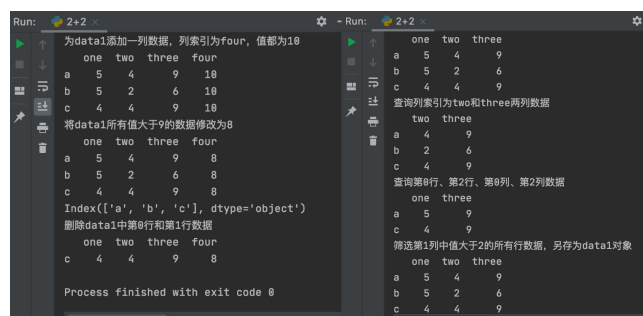


图 2: 思考 2-2 实验结果

思考与练习 3.3

1. 创建 50×7 的 DataFrame 对象，数据为 $[10,99]$ 之间的随机整数；columns 为字符 a-g；将 DataFrame 对象保存到 csv 文件中。

【提示】使用 NumPy 的随机生成函数 randint() 生成数据。

```

import numpy as np
import pandas as pd
from pandas import Series, DataFrame
d = np.random.randint(10, 99, size=(50, 7))
df = DataFrame(d, columns=['a', 'b', 'c', 'd', 'e', 'f', 'g'])
df.to_csv('3+1.csv', mode='w', header=True, index=False)

```

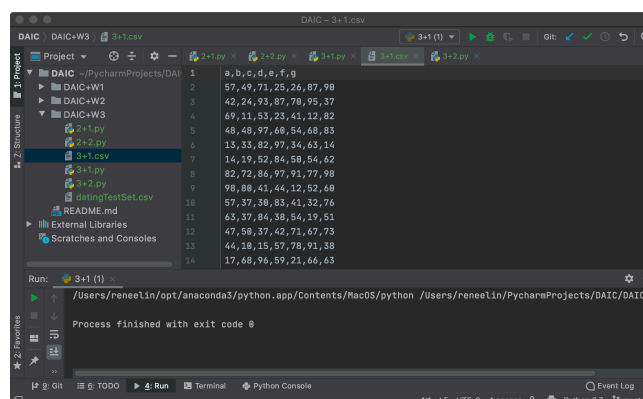


图 3: 思考 3-1 实验结果

2. 海伦一直使用在线交友网站寻找适合的约会对象，她将交友数据存放在 `datingTestSet.xls` 文件中。

- (1) 从文件中读取有效数据保存到 `Dataframe` 对象中，跳过所有文字解释行；
- (2) 列索引名设为 `['flymiles', 'videogame', 'icecream', 'type']`；
- (3) 显示读取到的前面 5 条数据；
- (4) 显示所有 `'type'` 为 `'largeDoses'` 的数据；
- (5) 将平均每周玩视频游戏时间超过 10 的数据都改成 10。

```
import numpy as np
import pandas as pd
from pandas import Series, DataFrame
# 从文件中读取有效数据保存到Dataframe对象中，跳过所有文字解释行
# 列索引名设为 ['flymiles', 'videogame', 'icecream', 'type']
cols = ['flymiles', 'videogame', 'icecream', 'type']
dating = pd.read_csv('datingTestSet.csv', skiprows=2, names=cols)
print("显示读取到的前面5条数据")
print(dating[:5])
t = dating['type'] == 'largeDoses'
print("显示所有'type'为'largeDoses'的数据")
print(dating[t])
# 将平均每周玩视频游戏时间超过10的数据都改成10
v = dating['videogame'] > 10
dating[v] = 10
```

```
Run: 3+2 x
/Users/reneelin/opt/anaconda3/python.app/Contents/Mac
显示读取到的前面5条数据
  flymiles  videogame  icecream  type
0  40920.0    8.326976  0.953952  largeDoses
1  14488.0    7.153469  1.673904  smallDoses
2  26052.0    1.441871  0.805124  didntLike
3  75136.0   13.147394  0.428964  didntLike
4  38344.0    1.669788  0.134296  didntLike
显示所有'type'为'largeDoses'的数据
  flymiles  videogame  icecream  type
0  40920.0    8.326976  0.953952  largeDoses
6  35948.0    6.830792  1.213192  largeDoses
7  42666.0   13.276369  0.543880  largeDoses
9  35483.0   12.273169  1.508053  largeDoses
19 28488.0   10.528555  1.304844  largeDoses
..      ...      ...      ...      ...
977 34143.0   13.609528  0.364240  largeDoses
990 27750.0    8.546741  0.128706  largeDoses
997 26575.0   10.650102  0.866627  largeDoses
998 48111.0    9.134528  0.728045  largeDoses
999 43757.0    7.882601  1.332446  largeDoses

[327 rows x 4 columns]

Process finished with exit code 0
```

图 4: 思考 3-2 实验结果