

**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： | 实验七 | | | | |
| 实验名称： | 使用pandas进行数据准备 | | | | |
| 实验人员： | 学号 | 18111207248 | | | |
| 姓名 | 吴钰 | | | |
| 班级 | 18计科创新班 | | | |
| 实验日期： | 2020.11.18 | | | | |
| 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） 熟练掌握pandas数据导入与导出操作。 （2） 熟练掌握Series和DataFrame的操作。

# 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. 读入餐饮数据订单详情表“meal\_order\_detail.xlsx”、订单信息表“meal\_order\_info.csv”、客户信息表“users.xlsx”，输出它们的形状；统计菜品的销量、菜品的平均售价和热销菜品等信息（说明：信息统计方法是根据菜品数据的数据类型分为数值型数据和类别型数据两种）； 剔除订单信息表和客户信息表中全为空值的列或者所有元素取值相同的列。

# 三张表中的数据特征说明见“数据特征说明.xlsx”

代码：

from pandas import read\_excel

from pandas import read\_csv

import pandas as pd

#将三个sheet整合为一个sheet

#第一步：调用pandas包

#第二步：读取数据

iris = pd.read\_excel('meal\_order\_detail.xlsx',None)#读入数据文件

keys = list(iris.keys())

#第三步：数据合并

iris\_concat = pd.DataFrame()

for i in keys:

iris1 = iris[i]

iris\_concat = pd.concat([iris\_concat,iris1])

iris\_concat.to\_excel('result.xlsx')#数据保存路径

df1=read\_excel('D:\大三（上）专业课\Python\实验七\meal\_order\_detail.xlsx',sheet\_name='meal\_order\_detail1')

print('meal\_order\_detail1',end='\n')

print(df1.shape)

df2=read\_excel('D:\大三（上）专业课\Python\实验七\meal\_order\_detail.xlsx',sheet\_name='meal\_order\_detail2')

print('meal\_order\_detail2',end='\n')

print(df2.shape)

df3=read\_excel('D:\大三（上）专业课\Python\实验七\meal\_order\_detail.xlsx',sheet\_name='meal\_order\_detail3')

print('meal\_order\_detail3',end='\n')

print(df3.shape)

rs=read\_excel('D://大三（上）专业课\Python//实验七//result.xlsx')

print('将meal\_order\_detail三个sheet整合',end='\n')

print(rs.shape)

df4=read\_csv('D:\大三（上）专业课\Python\实验七\meal\_order\_info.csv',sep=",")

print('meal\_order\_info.csv',end='\n')

print(df4.shape)

df5=read\_excel('D://大三（上）专业课\Python//实验七//users.xlsx')

print('users.xlsx',end='\n')

print(df5.shape)

print('菜品的销量：',end='\n')

dish\_count=rs['counts'].groupby(rs['dishes\_name']).sum()

dish\_count=dish\_count.sort\_values(ascending=False)

print(dish\_count)

print('\n')

print('菜品的平均售价：',end='\n')

#print('详情表里单价（counts）和数目（amounts）两列的描述性统计：\n',df1.loc[:,['counts','amounts']].describe())

dish\_avgprice=rs['amounts'].groupby(rs['dishes\_name']).sum()/rs['counts'].groupby(rs['dishes\_name']).sum()

dish\_avgprice=dish\_avgprice.sort\_values(ascending=False)

print(dish\_avgprice)

print('\n')

print('热销菜品排行榜：',end='\n')

print(rs['dishes\_name'].value\_counts())

print('\n')

print('哪个id点的菜最多：\n',end='\n')

print(rs['dishes\_name'].groupby(rs['emp\_id']).count().sort\_values(ascending=False).head(5))

print('\n')

print('哪个id花最多钱吃饭：\n',end='\n')

print(rs['amounts'].groupby(rs['emp\_id']).sum().sort\_values(ascending=False).head(5))

print('\n')

def dropNullStd(data):

beforerlen = data.shape[0]

beforeclen = data.shape[1]

colisNull = data.describe().loc['count'] == 0

print(type(colisNull))

print('beforerlen[0]:',beforerlen)

print('beforeclen[1]:',beforeclen)

for i in range(len(colisNull)):

if colisNull[i]:

data.drop(colisNull.index[i], axis = 1,inplace = True)

stdisZero = data.describe().loc['std'] == 0

for i in range(len(stdisZero)):

if stdisZero[i]:

data.drop(stdisZero.index[i], axis = 1, inplace = True)

afterclen = data.shape[1]

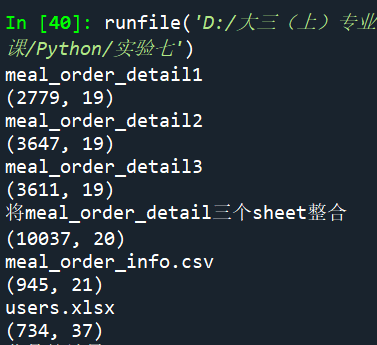
print('剔除的列的数目:',beforeclen-afterclen)

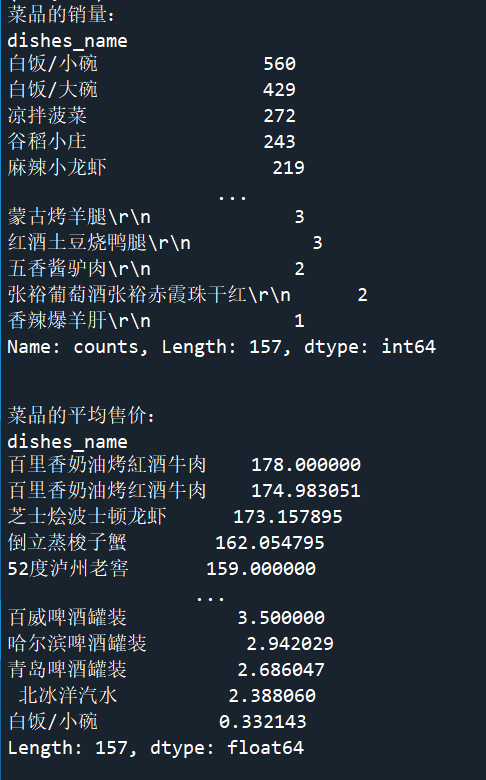
print('剔除后数据形状为:',data.shape)

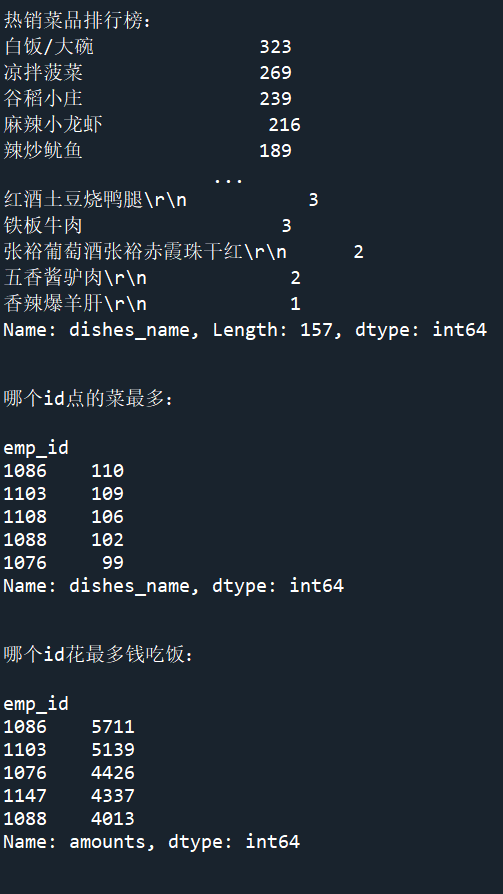
#剔除订单信息表和客户信息表中全为空值的列或者所有元素取值相同的列

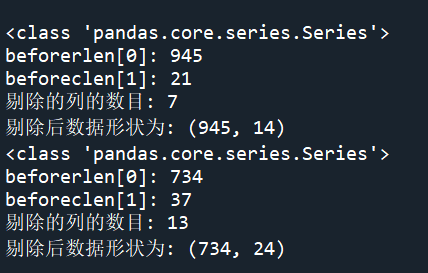
dropNullStd(df4)

dropNullStd(df5)









# 2. 读取mtcars数据集，输出该数据集的维度、大小等信息；使用describe方法对整个mtcars数据集进行描述性统计。

代码：

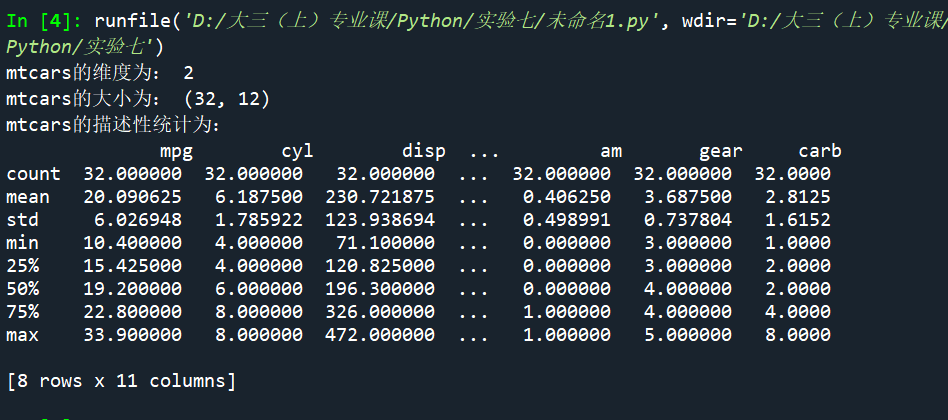
import pandas as pd

mtcars = pd.read\_csv('D:\大三（上）专业课\Python\实验七\mtcars.csv')

print('mtcars的维度为：',mtcars.ndim)

print('mtcars的大小为：',mtcars.shape)

print('mtcars的描述性统计为：\n',mtcars.describe())



# 五、实验思考

1、实验收获：

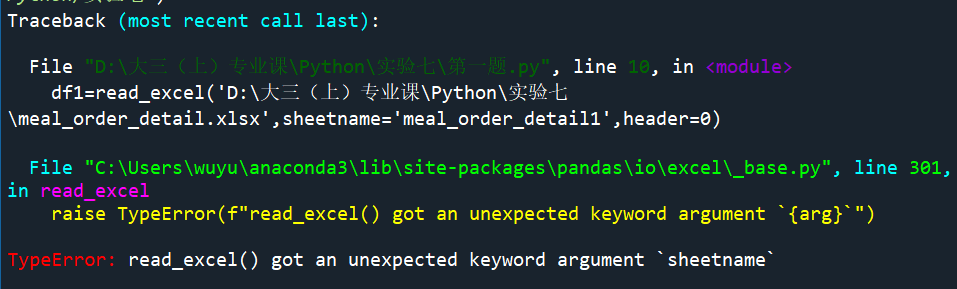
数据清洗要做的就是处理缺失数据以及清除无意义的信息。

垃圾数据，即使是通过最好的分析，也将产生错误的结果，并误导业务本身。结合相关的数学和统计学知识，去除连续型特征重复可以利用特征间的相似度将两个相似度为1的特征去除一个。在pandas中相似度的计算方法为corr，使用该方法计算相似度时，默认为“pearson”法 ，可以通过“method”参数调节，目前还支持“spearman”法和“kendall”法。但是通过相似度矩阵去重存在一个弊端，该方法只能对数值型重复特征去重，类别型特征之间无法通过计算相似系数来衡量相似度。除了使用相似度矩阵进行特征去重之外，可以通过DataFrame.equals的方法进行特征去重。

数据中的某个或某些特征的值是不完整的，这些值称为缺失值。

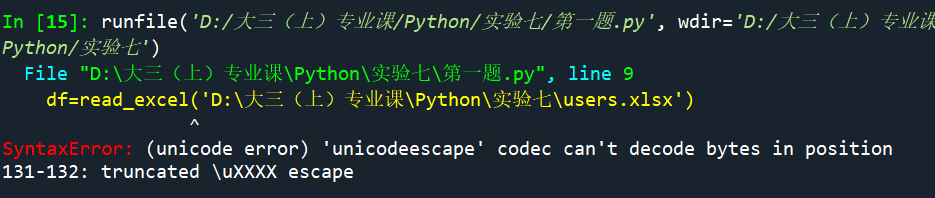
pandas提供了识别缺失值的方法isnull以及识别非缺失值的方法notnull，这两种方法在使用时返回的都是布尔值True和False。结合sum函数和isnull、notnull函数，可以检测数据中缺失值的分布以及数据中一共含有多少缺失值。isnull和notnull之间结果正好相反，因此使用其中任意一个都可以判断出数据中缺失值的位置。

2、在实验过程中遇到的问题：



解决方案：Anaconda 3.7版本中 'sheetname' 命令，已更新为 'sheet\_name'

遇到的问题：



解决方案：将’\’改为’//’就好了

原因：

window 读取文件可以用\，但是在字符串中\是被当作转义字符来使用，所以’d:\a.txt’会被转义成’d:\a.txt’这是正确路径，所以不会报错。而‘C:\Users\FrankYuan\Pictures\Camera Roll\WIN\_20161010\_08\_51\_57\_Pro.jpg ’中经过转义之后可能就找不到路径的资源了，例如\t可能就转义成tab键了。