运行代码：

import numpy as np  
import pandas as pd  
  
df1 = pd.read\_excel(**'./data/meal\_order\_detail1\_sql.xlsx'**)  
df2 = pd.read\_excel(**'./data/meal\_order\_detail2\_sql.xlsx'**)  
df3 = pd.read\_excel(**'./data/meal\_order\_detail3\_sql.xlsx'**)  
df = pd.concat([df1, df2, df3], axis=0, join=**'inner'**)  
print(**'三张订单详情表合并后的形状为：'**, df.shape)  
  
print(**"进行去重操作前订单详情表的形状为："**, df.shape)  
df = df.drop\_duplicates(subset=None, keep=**'first'**, inplace=False)  
print(**"样本去重后订单详情表的形状为："**, df.shape)  
  
  
*# 特征去重*def FeatureEquals(df):  
 *# 定义求取特征是否完全相同的矩阵的函数* dfEquals = pd.DataFrame([], columns=df.columns, index=df.columns)  
 for i in df.columns:  
 for j in df.columns:  
 dfEquals.loc[i, j] = df.loc[:, i].equals(df.loc[:, j])  
 return dfEquals  
  
  
detEquals = FeatureEquals(df)  
detEquals.to\_excel(**'data/detEquals\_js.xlsx'**)  
print(**'通过DataFrame.equals的方法得到特征矩阵并保存到文件detEquals\_js.xlsx中.'**)  
  
lenDet = detEquals.shape[0]  
dupCol = []  
for k in range(lenDet):  
 for l in range(k + 1, lenDet):  
 if detEquals.iloc[k, l] & (detEquals.columns[l] not in dupCol):  
 dupCol.append(detEquals.columns[l])  
print(**'根据特征矩阵确定需要删除的列有：'**, dupCol)  
df.drop(dupCol, axis=1, inplace=True)  
print(**'进行去重操作后订单详情表的形状为：'**, df.shape)  
  
print(**'detail每个特征缺失的数目为：**\n**'**, df.isnull().sum())  
df.dropna(axis=1, how=**'any'**, inplace=True)  
print(**'删除全部数据均为缺失的列后订单详情表的形状为：'**, df.shape)  
  
print(**'异常值处理前销售量最小值为：'**, df[**'counts'**].min())  
print(**'异常值处理前销售量最大值为：'**, df[**'counts'**].max())  
print(**'异常值处理前售价最小值为：'**, df[**'amounts'**].min())  
print(**'异常值处理前售价最大值为：'**, df[**'amounts'**].max())  
  
  
*# 异常值处理*def outRange(Ser):  
 QL = Ser.quantile(0.25) - 1.5 \* (Ser.quantile(0.75) - Ser.quantile(0.25))  
 QU = Ser.quantile(0.75) + 1.5 \* (Ser.quantile(0.75) - Ser.quantile(0.25))  
 Ser.loc[Ser > QU] = QU  
 Ser.loc[Ser < QL] = QL  
 return Ser  
  
  
df[**'counts'**] = outRange(df[**'counts'**].copy())  
df[**'amounts'**] = outRange(df[**'amounts'**].copy())  
print(**"异常值处理后销售量最大值为："**, df[**'counts'**].max())  
print(**"异常值处理后销售量最小值为："**, df[**'counts'**].min())  
print(**"异常值处理后售价最小值为："**, df[**'amounts'**].min())  
print(**"异常值处理后售价最大值为："**, df[**'amounts'**].max())  
  
*# 主键合并订单详情表、订单信息表和客户信息表*order = pd.read\_csv(**'data/meal\_order\_info.csv'**, sep=**','**, encoding=**'gb18030'**)  
user = pd.read\_excel(**'data/users\_info.xlsx'**)  
print(**'detail表大小：'**, df.shape)  
print(**'order表大小：'**, order.shape)  
print(**'user表大小：'**, user.shape)  
  
order.rename(columns={**"info\_id"**: **"order\_id"**}, inplace=True)  
merge1 = pd.merge(df, order, on=**"order\_id"**)  
merge1.drop(columns=**"emp\_id\_x"**, inplace=True)  
merge1.rename(columns={**"emp\_id\_y"**: **"emp\_id"**}, inplace=True)  
user.rename(columns={**"USER\_ID"**: **"emp\_id"**}, inplace=True)  
merge2 = pd.merge(merge1, user, on=**"emp\_id"**)  
print(**"三张表数据主键合并后的大小为："**, merge2.shape)  
print(**"合并后的宽表的列："**, merge2.columns)

运行结果：

C:\Users\congcong\AppData\Local\Programs\Python\Python37\python.exe "D:/我的坚果云/学校/北理珠/20-21-2/数据处理技术/04 使用pandas进行数据预处理-实训任务/第7次作业.py"

三张订单详情表合并后的形状为： (10037, 19)

进行去重操作前订单详情表的形状为： (10037, 19)

样本去重后订单详情表的形状为： (10037, 19)

通过DataFrame.equals的方法得到特征矩阵并保存到文件detEquals\_js.xlsx中.

根据特征矩阵确定需要删除的列有： ['parent\_class\_name', 'cost', 'discount\_amt', 'discount\_reason', 'kick\_back', 'add\_info', 'bar\_code', 'add\_inprice']

进行去重操作后订单详情表的形状为： (10037, 11)

detail每个特征缺失的数目为：

detail\_id 0

order\_id 0

dishes\_id 0

logicprn\_name 10037

dishes\_name 0

itemis\_add 0

counts 0

amounts 0

place\_order\_time 0

picture\_file 0

emp\_id 0

dtype: int64

删除全部数据均为缺失的列后订单详情表的形状为： (10037, 10)

异常值处理前销售量最小值为： 1

异常值处理前销售量最大值为： 10

异常值处理前售价最小值为： 1

异常值处理前售价最大值为： 178

异常值处理后销售量最大值为： 1.0

异常值处理后销售量最小值为： 1.0

异常值处理后售价最小值为： 1.0

异常值处理后售价最大值为： 102.5

detail表大小： (10037, 10)

order表大小： (945, 21)

user表大小： (758, 37)

三张表数据主键合并后的大小为： (14713, 65)

合并后的宽表的列： Index(['detail\_id', 'order\_id', 'dishes\_id', 'dishes\_name', 'itemis\_add',

'counts', 'amounts', 'place\_order\_time', 'picture\_file', 'emp\_id',

'number\_consumers', 'mode', 'dining\_table\_id', 'dining\_table\_name',

'expenditure', 'dishes\_count', 'accounts\_payable', 'use\_start\_time',

'check\_closed', 'lock\_time', 'cashier\_id', 'pc\_id', 'order\_number',

'org\_id', 'print\_doc\_bill\_num', 'lock\_table\_info', 'order\_status',

'phone', 'name', 'MYID', 'ACCOUNT', 'NAME', 'ORGANIZE\_ID',

'ORGANIZE\_NAME', 'DUTY\_ID', 'TITLE\_ID', 'PASSWORD', 'EMAIL', 'LANG',

'THEME', 'FIRST\_VISIT', 'PREVIOUS\_VISIT', 'LAST\_VISITS', 'LOGIN\_COUNT',

'ISEMPLOYEE', 'STATUS', 'IP', 'DESCRIPTION', 'QUESTION\_ID', 'ANSWER',

'ISONLINE', 'CREATED', 'LASTMOD', 'CREATER', 'MODIFYER', 'TEL', 'stuNo',

'qq', 'weixin', 'meal\_arithmetic\_id', 'arithmetic\_name', 'sex', 'poo',

'address', 'age'],

dtype='object')

进程已结束，退出代码为 0

