pandas 重复数据处理大全（附代码）

Original 东哥起飞 Python数据科学 2022-02-22 19:38

收录于话题
#数据清洗 2 #pandas 10 #pandas数据清洗 3

大家好，我是东哥。

继续更新 pandas 数据清洗，上一篇说到缺失值的处理。

链接：pandas 缺失数据处理大全（附代码）

顺兴趣可以关注这个话题pandas数据清洗，第一时间看到更新。

所有数据和代码可在我的 GitHub 获取：

```
https://github.com/xiaoyusmd/PythonDataScience
```

本次来介绍重复值处理的常用方法。

重复值处理主要涉及两个部分：一个是找出重复值，第二个是删除重复值，也就是根据自己设定的条件进行删除操作。

定位重复值

对于重复值，我们首先需要查看这些重复值是什么样的形式，然后确定删除的范围，而查询重复值需要用到 duplicated 函数。

duplicated 的返回值是布尔值，返回 True 和 False，默认情况下会按照一行的所有内容进行查重。

主要参数：

- subset：如果不按照全部内容查重，那么需要指定按照哪些列进行查重，比如按照姓名进行查重 subset=['name']，那么具有相同名字的人就只会保留一个，但很可能只是重名的原因，而并非真正同一个人，所以可以按照姓名和生日相同作查重，subset=['name','birthday']，同理还可以再添加列，这样就可以基本保证去重效果了。
- keep：用来确定要标记的重复值，可以设置为 first、last、False。
  - first：除第一次出现的重复值，其他都标记为True
  - last：除最后一次出现的重复值，其他都标记为True
  - False：所有重复值都标记为True

实例：

```
import pandas as pd
import numpy as np

data = {
    'user' : ['zszxz','zszxz','rose'],
    'price' : [100, 200, 300],
    'hobby' : ['reading','reading','hiking']
}
frame  = pd.DataFrame(data)
print(frame)
--------------------
    user  price    hobby
0  zszxz    100  reading
1  zszxz    200  reading
2   rose    300   hiking
--------------------

frame.duplicated()
----------
0    False
1    False
2    False
dtype: bool
----------
```

上面提到 duplicated 返回布尔值，所以如果要想输出这些重复值，还需要和查询的方法配合使用 df[df.duplicated()]，比如：

```
# 1：找user变量得这重复值
frame[frame.duplicated(subset=['user'])]
--------------------
  user price hobby
1 zszxz 200 reading
--------------------
```

上面按 user 一个变量进行查重，但没有设置 keep 参数，所以默认得选出除了第一个以外的其它重复值。

```
# 2：找user变量得这重复值，保留含变量重复
frame[frame.duplicated(subset=["user"], keep=False)]
--------------------
  user price hobby
0 zszxz 100 reading
1 zszxz 200 reading
--------------------
```

上面按 user 一个变量进行查重，并设置 keep 参数为 False，所以保留了全部的重复值。

```
# 3：找user和hobby变量得这重复值，得选出除最后一个重复值以外的其它重复值
frame[frame.duplicated(subset=['user','hobby'], keep='last')]
--------------------
  user price hobby
0 zszxz 100 reading
--------------------
```

上面按 user 和 hobby 两个变量进行查重，并设置 keep 参数为 last，所以筛选出了除最后一个重复值以外的其它重复值。

通过两个参数的设置就可以查看自己想要的重复值了，以此判断要删除哪个，保留哪个。

删除重复值

当确定好需要删除的重复值后，就进行进行删除的操作了。

删除重复值会用到 drop_duplicates 函数。

和 duplicated() 函数参数类似，主要有3个参数：

- subset：同 duplicated()，设置去重的字段
- keep：这里稍有不同，duplicated() 中是将除设置值以外重复值都返回 True，而这里是保留的意思，同样可以设置 first、last、False
  - first：保留第一次出现的重复行，删除其他重复行
  - last：保留最后一次出现的重复行，删除其他重复行
  - False：删除所有重复行
- inplace：布尔值，默认为 False，是否直接在原数据上删除重复项或删除重复项后返回副本。

实例：

1、全部去重

```
# 默认部分将删除，也原数据frame上生效
frame.drop_duplicates(inplace=True)
print(frame)
--------------------
    user  price    hobby
0  zszxz    100  reading
1  zszxz    200  reading
2   rose    300   hiking
--------------------
```

因为上面数据中没有全部重复的，因此没有可删除行。

2、指定列去重

```
# 默认base字段删除，也原数据frame上生效
frame.drop_duplicates(subset=['user'],inplace=True)
print(frame)
--------------------
    user  price    hobby
0  zszxz    100  reading
2   rose    300   hiking
--------------------
```

上面按 user 字段删除重复行，保留第一个重复行，因此第二行被删除了。但这里大家注意下，**执行删除重复行操作后，表的索引也会被删掉。**

如需要重置可以加上 reset_index()，设置 drop=True，用索引替代被打乱的索引。

```
frame.drop_duplicates(subset=['user'],inplace=True)
frame.reset_index(drop=True)
--------------------
    user  price    hobby
0  zszxz    100  reading
1   rose    300   hiking
--------------------
```

keep 默认为 first，下面手动设置为 last，只保留最后一个重复行。

```
# 按全部字段删除，也原数据frame上生效
frame.drop_duplicates(subset=['user','hobby'],keep='last',inplace=True)
print(frame)
--------------------
    user  price    hobby
1  zszxz    200  reading
2   rose    300   hiking
--------------------
```

keep 手动设置为 False，全部删除，这种一般很少用。

```
# 按全部字段删除，也原数据frame上生效
frame.drop_duplicates(subset=['user','hobby'],keep=False,inplace=True)
print(frame)
--------------------
    user  price    hobby
2   rose    300   hiking
--------------------
```

以上就是重复值相关的所有操作。

注意事项

在删除重复值时，要注意下删除的逻辑。

因为很多时候我们需要把这些离线的清洗操作在线上复现。

如果我们随机地删除重复行，没有明确的逻辑，那么对于这种随机性线上是无法复现的，即无法保证清洗后的数据一致性。

所以我们在删除重复行前，可以把重复列断字段进行排序处理。

比如上面例子中，如果要对 user 和 price 去重，那么比较严谨的做法是按照 user 和 price 进行排序。

```
frame.sort_values(by=['user','price'],ascending=True).reset_index(drop=True)
--------------------
  user price hobby
0 rose -300 hiking
1 zszxz 100 reading
2 zszxz 200 reading
--------------------
```

因为有了排序性，只要按这个逻辑它的顺序是固定的，而不是随机的，所以无论我们设置 keep 为 first 还是 last，都没有任何影响。

以上是本次分享，原创不易，欢迎点赞、在看支持。

推荐阅读

1. pandas100个骚操作
2. 机器学习原创系列
3. 数据科学干货下载

收录于话题 #pandas 10

上一篇
pandas 文本处理大全（附代码）

下一篇
10000 字的 pandas 核心操作知识大全！

Read more Modified on 2022/02/22