资源 | 23种Pandas核心操作，你需要过一遍吗？

作者：George Seif

机器之心编译

参与：思源

Pandas 是一个 Python 软件库，它提供了大量能使我们快速便捷地处理数据的函数和方法。一般而言，Pandas 是使 Python 成为强大而高效的数据分析环境的重要因素之一。在本文中，作者从基本数据集读写、数据处理和 DataFrame 操作三个角度展示了 23 个 Pandas 核心方法。

Pandas 是基于 NumPy 构建的库，在数据处理方面可以把它理解为 NumPy 加强版，同时 Pandas 也是一项开源项目。它基于 Cython，因此读取与处理数据非常快，并且还能轻松处理浮点数据中的缺失数据（表示为 NaN）以及非浮点数据。在本文中，基本数据集操作主要介绍了 CSV 与 Excel 的读写方法，基本数据处理主要介绍了缺失值及特征抽取，最后的 DataFrame 操作则主要介绍了函数和排序等方法。

基本数据集操作

（1）读取 CSV 格式的数据集

```
pd.DataFrame.from_csv(“csv_file”)
```

或者：

```
pd.read_csv(“csv_file”)
```

（2）读取 Excel 数据集

```
pd.read_excel("excel_file")
```

（3）将 DataFrame 直接写入 CSV 文件

如下采用逗号作为分隔符，且不带索引：

```
df.to_csv("data.csv",  sep=",",  index=False)
```

（4）基本的数据集特征信息

```
df.info()
```

（5）基本的数据集统计信息

```
print(df.describe())
```

(6) Print data frame in a table

将 DataFrame 输出到一张表：

```
print(tabulate(print_table,  headers=headers))
```

当「print_table」是一个列表，其中列表元素还是新的列表，「headers」为表头字符串组成的列表。

（7）列出所有列的名字

```
df.columns
```

基本数据处理

（8）删除缺失数据

```
df.dropna(axis=0,  how='any')
```

返回一个 DataFrame，其中删除了包含任何 NaN 值的给定轴，选择 how=「all」会删除所有元素都是 NaN 的给定轴。

（9）替换缺失数据

```
df.replace(to_replace=None,  value=None)
```

使用 value 值代替 DataFrame 中的 to_replace 值，其中 value 和 to_replace 都需要我们赋予不同的值。

（10）检查空值 NaN

```
pd.isnull(object)
```

检查缺失值，即数值数组中的 NaN 和目标数组中的 None/NaN。

（11）删除特征

```
df.drop('feature_variable_name',  axis=1)
```

axis 选择 0 表示行，选择表示列。

（12）将目标类型转换为浮点型

```
pd.to_numeric(df["feature_name"],  errors='coerce')
```

将目标类型转化为数值从而进一步执行计算，在这个案例中为字符串。

（13）将 DataFrame 转换为 NumPy 数组

```
df.as_matrix()
```

（14）取 DataFrame 的前面「n」行

```
df.head(n)
```

（15）通过特征名取数据

```
df.loc[feature_name]
```

DataFrame 操作

（16）对 DataFrame 使用函数

该函数将令 DataFrame 中「height」行的所有值乘上 2：

```
df["height"].apply(*lambda*  height:  2  *  height)
```

或：

```
def  multiply(x):
  return  x  *  2
df["height"].apply(multiply)
```

（17）重命名行

下面代码会重命名 DataFrame 的第三行为「size」：

```
df.rename(columns  =  {df.columns[2]:'size'},  inplace=True)
```

（18）取某一行的唯一实体

下面代码将取「name」行的唯一实体：

```
df["name"].unique()
```

（19）访问子 DataFrame

以下代码将从 DataFrame 中抽取选定了的行「name」和「size」：

```
new_df = df[["name", "size"]]
```

（20）总结数据信息

```
# Sum of values in a data frame

df.sum()

# Lowest value of a data frame

df.min()

# Highest value

df.max()

# Index of the lowest value

df.idxmin()

# Index of the highest value

df.idxmax()

# Statistical summary of the data frame, with quartiles, median, etc.

df.describe()
```

```
# Average values

df.mean()

# Median values

df.median()

# Correlation between columns

df.corr()

# To get these values for only one column, just select it like this#

df["size"].median()
```

（21）给数据排序

```
df.sort_values(ascending = False)
```

（22）布尔型索引

以下代码将过滤名为「size」的行，并仅显示值等于 5 的行：

```
df[df["size"] == 5]
```

（23）选定特定的值

以下代码将选定「size」列、第一行的值：

```
df.loc([0], ['size'])
```

原文链接：https://towardsdatascience.com/23-great-pandas-codes-for-data-scientists-cca5ed9d8a38