

1. 文本文件

```
> import numpy as np # 用 Numpy 导入数据
> import pandas as pd # 用 Pandas 导入数据
```

操作与读写 - 手动打开关闭

```
filename = 'huck_finn.txt'
# 以只读的方式读取文件
file = open(filename, mode='r')
text = file.read() # 读取文件内容
print(file.closed) # 查看文件是否已经关闭
file.close() # 关闭文件
print(text)
```

操作与读写 - 使用上下文管理器 with

```
with open('huck_finn.txt', 'r') as file:
    print(file.readline()) # 读取一行
    print(file.readline())
    print(file.readline())
```

2. 其他格式文件

2.1 Stata 文件

使用 pandas 读取

```
data = pd.read_stata('urbanpop.dta')
```

2.2 Matlab 文件

scipy 工具库读取

```
import scipy.io
filename = 'workspace.mat'
mat = scipy.io.loadmat(filename)
```

2.3 HDF5 文件

使用 h5py 工具库打开读取

```
import h5py
filename = 'H-H1_LOSC_4_v1-816.hdf5'
data = h5py.File(filename, 'r')
```

1.2 表格数据: 文本文件

用 Numpy 导入文本文件

单数据类型文件

```
filename = 'mnist.txt'
# 用于分割各列值的字符, 跳过前两行, 读取并使用第 1 列和第 3 列使用的数据类型
data = np.loadtxt(filename, delimiter=',', skiprows=2, usecols=[0,2], dtype=str)
```

多数据类型文件

```
filename = 'titanic.csv'
# 导入时查找列名
data = np.genfromtxt(filename, delimiter=',', names=True, dtype=None)
np.recfromcsv()
data_array = np.recfromcsv(filename) # 函数的 dtype 默认值为 None
```

用 Pandas 导入文本文件

```
filename = 'winequality-red.csv'
data = pd.read_csv(filename, nrows=5, header=None, \
    sep='\t', comment='#', \
    na_values=[""]) # 文件名, 读取的行数, 用哪一行做列名
# 分隔各列的字符, 分割注释的字符
# 读取时哪些值为 NA/NaN
```

读写文件

```
file = 'urbanpop.xlsx'
data = pd.ExcelFile(file)
df_sheet2 = data.parse('1960', skiprows=[0], names=['Country', 'AAM: War(2002)'])
df_sheet1 = data.parse(0, parse_cols=[0], skiprows=[0], names=['Country'])
```

使用 sheet_names 属性访问表单名称

data.sheet_names

读取成 Dataframe 格式

```
from sas7bdat import SAS7BDAT
with SAS7BDAT('urbanpop.sas7bdat') as file:
    df_sas = file.to_data_frame()
```

使用 pickle 工具库打开读取

```
import pickle
with open('pickled_fruit.pkl', 'rb') as file:
    pickled_data = pickle.load(file)
```

2.4 Excel 文件

2.5 SAS 文件

2.6 Pickled 文件

3. Array 与 Dataframe 数据

Numpy 数组

```
data_array.dtype # 查看数组元素的数据类型
data_array.shape # 查看数组维度
len(data_array) # 查看数组长度
```

Pandas 数据帧

```
df.head() # 返回数据帧的前几行, 默认为 5 行
df.tail() # 返回数据帧的后几行, 默认为 5 行
df.index # 查看数据帧的索引
df.columns # 查看数据帧的列名
df.info() # 查看数据帧各列的信息
# 将数据帧转换为 Numpy 数组
data_array = data.values
```

4. 字典数据

4.1 通过函数访问数据元素

```
print(mat.keys()) # 输出字典的键值 (Key)
for key in data.keys(): # 输出字典的键值 (Key)
    print(key)
meta
quality
strain
pickled_data.values() # 返回字典的值
print(mat.items()) # 返回由元组构成字典键值对列表
```

4.2 通过键访问数据

```
# 探索 HDF5 的结构
for key in data['meta'].keys():
    print(key)
# 提取某个键对应的值
print(data['meta']['Description'].value)
```

5. 数据库

5.1 关系型数据库

使用 sqlalchemy 库

```
from sqlalchemy import create_engine
engine = create_engine('sqlite:///Northwind.sqlite')
```

使用 table_names() 方法获取表名列表:

```
table_names = engine.table_names()
```

5.2 查询关系型数据库

执行 SQL 语句查询

```
con = engine.connect()
rs = con.execute("SELECT * FROM Orders")
df = pd.DataFrame(rs.fetchall())
df.columns = rs.keys()
con.close()
```

使用上下文管理器 with

```
with engine.connect() as con:
    rs = con.execute("SELECT OrderID FROM Orders")
    df = pd.DataFrame(rs.fetchmany(size=5))
    df.columns = rs.keys()
```

使用 Pandas 查询关系型数据库

```
df = pd.read_sql_query("SELECT * FROM Orders", engine)
```

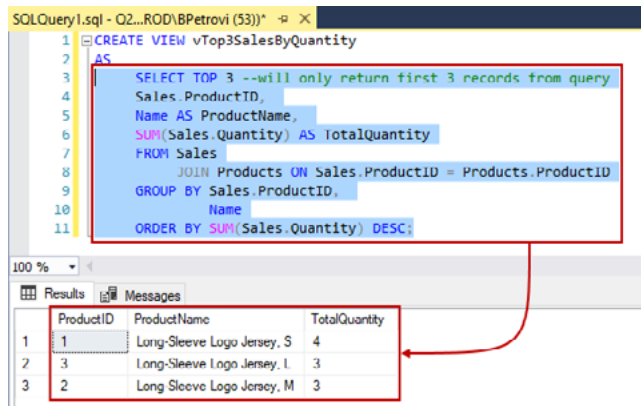
6. 文件系统与操作

6.1 魔法命令

```
!ls      # 列出目录里的子目录和文件夹
%cd ..   # 改变当前工作目录
%pwd     # 返回当前工作目录的路径
```

6.2 os 库

```
import os
path = "/usr/tmp"
wd = os.getcwd()      # 将当前工作目录存为字符串
os.listdir(wd)         # 将目录里的内容输出为列表
os.chdir(path)         # 改变当前的工作目录
os.rename("test1.txt", "test2.txt") # 重命名文件
os.remove("test1.txt") # 删除现有文件
os.mkdir("newdir")     # 新建文件夹
```



```
SQLQuery1.sql - Q2...ROD\BPetrovi (53)
1 CREATE VIEW vTop3SalesByQuantity
2 AS
3 SELECT TOP 3 --will only return first 3 records from query
4 Sales.ProductID,
5 Name AS ProductName,
6 SUM(Sales.Quantity) AS TotalQuantity
7 FROM Sales
8 JOIN Products ON Sales.ProductID = Products.ProductID
9 GROUP BY Sales.ProductID,
10 Name
11 ORDER BY SUM(Sales.Quantity) DESC;
```

ProductID	ProductName	TotalQuantity
1	Long-Sleeve Logo Jersey, S	4
3	Long-Sleeve Logo Jersey, L	3
2	Long Sleeve Logo Jersey, M	3



Importing Data 速查表

获取最新版 | <http://www.showmeai.tech/>

作者 | 韩信子 @ShowMeAI

设计 | 南乔 @ShowMeAI

参考 | DataCamp Cheatsheet

扫码回复“数据科学”

下载最新全套速查表

数据科学工具库速查表



NumPy 是 Python 数据科学计算的核心库，提供了高性能多维数组对象及处理数组的工具。使用以下语句导入 NumPy 库：

```
import numpy as np
```



SciPy 是基于 NumPy 创建的 Python 科学计算核心库，提供了众多数学算法与函数。



Pandas 是基于 NumPy 创建的 Python 库，为 Python 提供了易于使用的的数据结构和数据分析工具。使用以下语句导入：

```
import pandas as pd
```



Matplotlib 是 Python 的二维绘图库，用于生成符合出版质量或跨平台交互环境的各类图形。

```
import matplotlib.pyplot as plt
```



Seaborn 是基于 matplotlib 开发的高阶 Python 数据可视图库，用于绘制优雅、美观的统计图形。使用下列别名导入该库：

```
import matplotlib.pyplot as plt
```

```
import seaborn as sns
```



Bokeh 是 Python 的交互式可视图库，用于生成在浏览器里显示的大规模数据集高性能可视图。Bokeh 的中间层通用 **bokeh.plotting** 界面主要为两个组件：数据与图示例。

```
from bokeh.plotting import figure
```

```
from bokeh.io import output_file, show
```



PySpark 是 Spark 的 Python API，允许 Python 调用 Spark 编程模型。Spark SQL 是 Apache Spark 处理结构化数据模块。

AI 垂直领域工具库速查表



Scikit-learn 是开源的 Python 库，通过统一的界面实现机器学习、预处理、交叉验证及可视化算法。



Keras 是强大、易用的深度学习库，基于 Theano 和 TensorFlow 提供了高阶神经网络 API，用于开发和评估深度学习模型。



“TensorFlow™ is an open source software library for numerical computation using data flow graphs.” **TensorFlow** 是 Google 公司开发的机器学习架构，兼顾灵活性和扩展性，既适合用于工业生产也适合用于科学研究。



PyTorch 是 Facebook 团队 2017 年初发布的深度学习框架，有利于研究人员、爱好者、小规模项目等快速搞出原型。**PyTorch** 也是 Python 程序员最容易上手的深度学习框架。



Hugging Face 以开源的 NLP 预训练模型库 **Transformers** 而广为人知，目前 GitHub Star 已超过 54000+。**Transformers** 提供 100+ 种语言的 32 种预训练语言模型，简单，强大，高性能，是新手入门的不二选择。



OpenCV 是一个跨平台计算机视觉库，由 C 函数 /C++ 类构成，提供了 Python、MATLAB 等语言的接口。**OpenCV** 实现了图像处理和计算机视觉领域的很多通用算法。

编程语言速查表



SQL 是管理关系数据库的结构化查询语言，包括数据的增删查改等。作为数据分析的必备技能、岗位 JD 的重要关键词，SQL 是技术及相关岗位同学一定要掌握的语言。



Python 编程语言简洁快速、入门简单且功能强大，拥有丰富的第三方库，已经成为大数据和人工智能领域的主流编程语言。

More...

AI 知识技能速查表



Jupyter Notebook 交互式计算环境，支持运行 40+ 种编程语言，可以用来编写漂亮的交互式文档。这个教程把常用的基础功能讲解得很清楚，对新手非常友好。



正则表达式 非常强大，能匹配很多规则的文本，常用于文本提取和爬虫处理。这也是一门令人难以捉摸的语言，字母、数字和符号堆在一起，像极了“火星文”。

More...



ShowMeAI 速查表 (©2021)

获取最新版 | <http://www.showmeai.tech/>

作者 | 韩信子 @ShowMeAI

设计 | 南乔 @ShowMeAI

数据科学工具库速查表

扫码回复“数据科学”

获取最新全套速查表

AI 垂直领域工具库速查表

扫码回复“工具库”

获取最新全套速查表

编程语言速查表

扫码回复“编程语言”

获取最新全套速查表

AI 知识技能速查表

扫码回复“知识技能”

获取最新全套速查表