

## Python 数据分析入门秘籍



图 1 Python 入门秘籍

### 1 寻山问路

#### 1.1 开发工具<sup>[2]</sup>

##### 1.1.1 Pycharm<sup>[1]</sup>

Pycharm 是最好的 Python 代码编辑工具之一，没有什么学习成本，下载安装后可直接上手。PyCharm 用于一般 IDE 具备的功能，比如，调试、语法高亮、Project 管理、代码跳转、智能提示、自动完成、单元测试、版本控制。PyCharm 还提供了一些很好的功能用于 Django 开发，同时支持 Google App Engine，更酷的是，PyCharm 支持 IronPython！

主要特性：

（1）编码协助，其提供了一个带编码补全，代码片段，支持代码折叠和分割窗口的智能、可配置的编辑器，可帮助用户更快更轻松地完成编码工作

（2）项目代码导航，该 IDE 可帮助用户即时从一个文件导航至另一个，从一个方法至其申明或者用法甚至可以穿过类的层次。若用户学会使用其提供的快捷键的话甚至能更快

（3）代码分析，用户可使用其编码语法，错误高亮，智能检测以及一键式代码快速补全建议，使得编码更优化

（4）Python 重构，该功能用户便能在项目范围内轻松进行重命名，提取方法/超类，导入域/变量/常量，移动和前推/后退重构



图 2 Pycharm 界面图

- 下载地址: <https://www.jetbrains.com/pycharm/>
- 安装教程: <http://t.csdn.cn/bOFul>

### 1.1.2 Vscode<sup>[2]</sup>

VSCode 是一款更轻量级的代码编辑器, 需要额外安装 python 插件。界面截图如下:

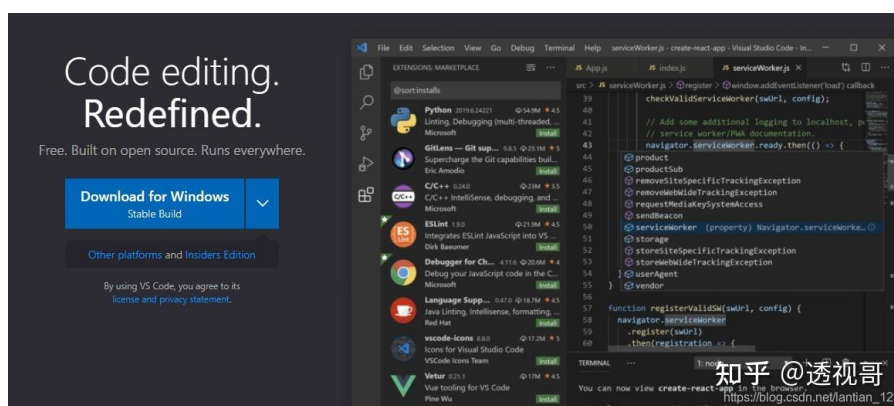


图 3 VsCode 界面图

- 下载地址: <https://code.visualstudio.com/>
- 安装教程: <http://t.csdn.cn/AbPeH>

### 1.1.3 Jupyter<sup>[2]</sup>

Jupyter Notebook 是一个以网页的形式打开, 可以直接在网页页面中编写代码和运行代码的工具, 代码的运行结果也会直接在代码块下显示的程序。更适合用在科学计算等场景中。界面截图如下:



图 4 jupyter 界面图

- 下载地址: <https://jupyter.org/install>
- 安装教程: <http://t.csdn.cn/D2Dd0>

### 1.1.4 IPython<sup>[1]</sup>

IPython 是 Python 的原生交互式 shell 的增强版，可以完成许多不同寻常的任务，比如帮助实现并行化计算；主要使用它提供的交互性帮助，比如代码着色、改进了的命令行回调、制表符完成、宏功能以及改进了的交互式帮助。主要特性：**HTML** 笔记本来连接浏览器、**Qt** 界面来显示、编辑和高亮，并行和分布式计算的组件。界面截图如下：

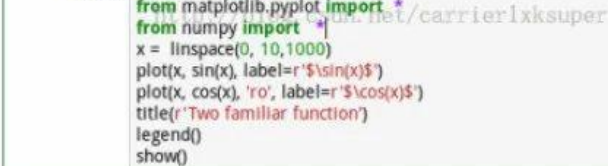


图 5 IPython 界面图

- 下载地址: <https://ipython.org/>
- 安装教程: <http://t.csdn.cn/wG4Ka>

## 1.2 自学网站<sup>[2]</sup>

- 菜鸟教程 <https://www.runoob.com/python3/python3-tutorial.html>
- 廖雪峰学 Python <https://www.liaoxuefeng.com/wiki/1016959663602400>
- Python 官方教程 <https://docs.python.org/zh-cn/3/tutorial/index.html>
- Python - 100 天从新手到大师 <https://github.com/jackfrued/Python-100-Days>

### 1.3 自学书籍<sup>[2]</sup>



图 6 书籍推荐

### 1.4 在线课程<sup>[2]</sup>

- B 站：小甲鱼零基础入门学 python
- B 站：Python 编程从入门到实践（第 2 版）视频版
- B 站：黑马程序员 Python 教程\_600 集 Python 从入门到精通教程（懂中文就能学会）
- B 站：花了 2 万多买的 Python 教程全套，现在分享给大家，入门到精通(Python 全栈开发教程)
- 慕课网：Python3 入门教程 2020 全新版
- 夜曲编程：小白编程入门第一堂课
- CSDN：Python 初级入门精讲

## 2 纸上谈兵

## python语言基础

- (1) Python3入门, 数据类型, 字符串
- (2) 判断/循环语句, 函数, 命名空间, 作用域
- (3) 类与对象, 继承, 多态
- (4) tkinter界面编程
- (5) 文件与异常, 数据处理简介
- (6) Pygame实战飞机大战, 2048

## python语言高级

- (1) Python常见第三方库与网络编程
- (2) Python正则表达式
- (3) 邮箱爬虫, 文件遍历, 金融数据爬虫, 多线程爬虫
- (4) Python线程、进程
- (5) Python MySQL数据库, 协程, jython

## python全栈工程师前端

- (1) HTML
- (2) HTML5
- (3) CSS
- (4) CSS3
- (5) 网页界面设计实战
- (6) JavaScript
- (7) jquery
- (8) jquery EasyUI, Mobile简介, photoshop
- (9) Bootstrap

## python全栈工程师后端

- (1) Django入门
- (2) Django高级
- (3) Django实战

## python全栈工程师后端高级

- (1) Flask开发原理
- (2) Flask开发项目实践
- (3) Tornado开发原理
- (4) Tornado开发项目实践

## Linux基础

- (1) 文件处理命令
- (2) 权限管理命令
- (3) 帮助命令
- (4) 文件搜索命令
- (5) 压缩解压命令
- (6) 命令使用技巧
- (7) VIM使用
- (8) 软件包管理
- (9) 用户和用户组管理
- (10) Linux Shell开发

## Linux运维自动化开发

- (1) Python开发Linux运维
- (2) Linux运维报警工具开发
- (3) Linux运维报警安全审计开发
- (4) Linux业务质量报表工具开发
- (5) Kali安全检测工具检测

## python数据分析

- (1) numpy数据处理
- (2) pandas数据分析
- (3) matplotlib数据可视化
- (4) scipy数据统计分析
- (5) python 金融数据分析

## python大数据

- (1) Hadoop HDFS
- (2) python Hadoop MapReduce
- (3) python Spark core
- (4) python Spark SQL
- (5) python Spark MLlib

## python机器学习

- (1) 机器学习基础知识简介
- (2) KNN算法
- (3) 线性回归
- (4) 逻辑斯蒂回归算法
- (5) 决策树算法
- (6) 朴素贝叶斯算法
- (7) 支持向量机
- (8) 聚类k-means算法

图 7 Python 学习流程

## 3 循序渐进（以数据分析为例）

### 3.1 Python 基础知识

- 思维导图（仅供参考）：<http://t.csdn.cn/OgSgu>

### 3.2 Python 数据分析

- 思维导图（仅供参考）：<http://t.csdn.cn/tjyeV>

## 4 一招制敌

## 6



```
# ----- 二分查找 -----
class Solution:
    def search(self, nums, target) -> int:
        left, right = 0, len(nums) - 1 # 初始化双指针 left, right 分别指向数组首、尾元素，代表查找区间为闭区间 [left, right]
        while left <= right:
            mid = (right - left) // 2 + left # 向下取整，确定中点m（考虑非整数问题）
            num = nums[mid]
            if num == target:
                return mid
            elif num > target:
                right = mid - 1
            else:
                left = mid + 1
        return -1

a = range(1, 10, 2)
print(a)
nums = [-1, 0, 3, 5, 9, 12]
print(len(nums))
print(nums[0])
target = 9
# 类的函数的调用
solution = Solution()
print(solution.search(nums, target))
```

BinarySearch

```
"D:\Program Files\Pycharm\PycharmProjects\pythonProject\venv\Scripts\python.exe" "D:/Program Files/Pycharm/PycharmProjects/
range(1, 10, 2)
6
-1
4
Process finished with exit code 0
```

图 9 实例演示 1：二分查找

## 4.2 实例演示 2：文本分类

```
76 start = time.time()
77 #模型训练, fit通常都是指模型的学习、训练过程
78 print('训练:')
79 model = estimator
80 model.fit(trainReview, trainLabel)
81 print(model)
82 #模型预测:
83 print('预测:')
84 pred_model = model.predict(testReview)
85 print(pred_model)
86 #算法评估
87 print('评估:')
88 score = metrics.accuracy_score(testLabel, pred_model)
89 matrix = metrics.confusion_matrix(testLabel, pred_model)
```

CSVtrain2

```
"D:\Program Files\Pycharm\PycharmProjects\pythonProject\venv\Scripts\python.exe" "D:/Program Files/Pycharm/PycharmProjects/
训练:
KNeighborsClassifier()
预测:
[1 1 1 ... 0 1 0]
评估:
>>>准确率
0.5929203539823009

>>>混淆矩阵
[[470 576]
 [344 870]]

>>>召回率
precision    recall  f1-score   support

0         0.58      0.45      0.51      1046
1         0.60      0.72      0.65      1214

accuracy          0.59
macro avg         0.59      0.58      0.58      2260
weighted avg      0.59      0.59      0.59      2260

>>>算法消耗时间为: 36.561806201934814 秒
```

图 10 实例演示 2：文本分类

### 4.3 实例演示 3：红酒品质分类

数据来源: kaggle

读取 winequality-red.csv 的数据, 并对各个指标进行初步的探索。

```
[2]: import numpy as np
import pandas as pd
import seaborn as sns
from sklearn import svm
from sklearn import model_selection
import matplotlib.pyplot as plt
import warnings
warnings.simplefilter(action="ignore", category=FutureWarning)
```

导入库后, 进行数据的读取。

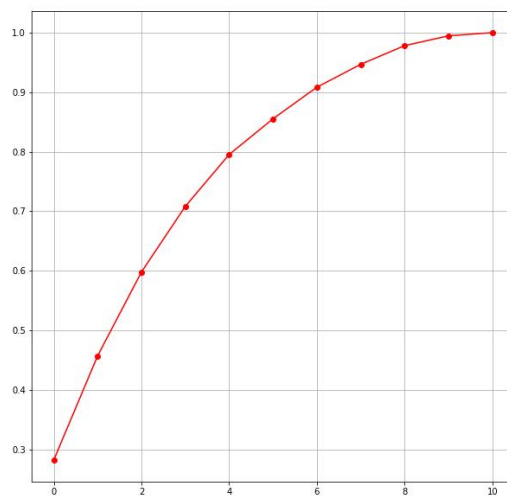
```
[3]: data=pd.read_csv('winequality-red.csv')
```

```
[4]: data.head()
```

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8	5
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8	6
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5

再看看汇总的信息:

图 11 实例演示 3：红酒品质分类-图 1



根据这次的降维结果, 我们可以选择6个维度进行分析:

```
25]: from sklearn.model_selection import train_test_split
pca_new = PCA(n_components=6)
x_new = pca_new.fit_transform(x)
x_train, x_test, y_train, y_test = train_test_split(x_new, y, test_size = 0.25)
lin_svc = SVC()
lin_svc.fit(x_train, y_train)
print(lin_svc.score(x_train, y_train))
0.9799833194328608
```

图 12 实例演示 3：红酒品质分类-图 2

### 4.4 实例演示 4：航空公司客户价值分析



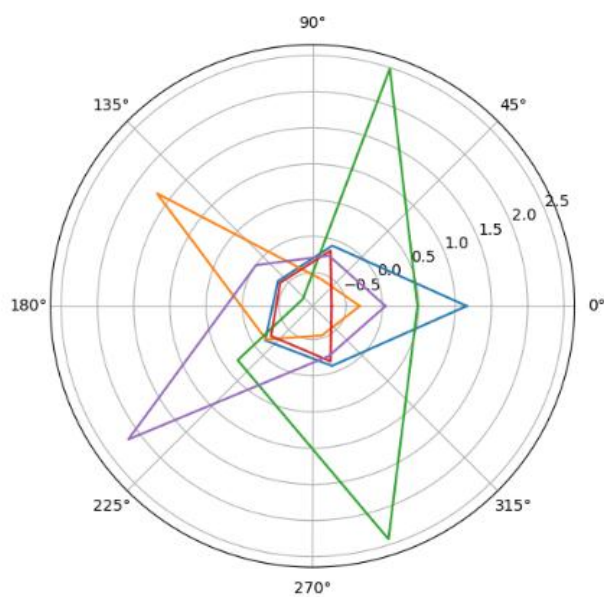


图 13 实例演示 4：航空公司客户价值分析

Reference:

- [1] [https://blog.csdn.net/xx\\_nm98/article/details/124714488](https://blog.csdn.net/xx_nm98/article/details/124714488)
- [2] <https://www.zhihu.com/question/451604793/answer/2632479142>