



Python环境

数据分析与数据挖掘



认识Python

- Python是一种解释型、面向对象、动态数据类型的高级程序设计语言。
- Python由Guido van Rossum于1989年底发明，第一个公开发行人版发行于1991年。
- Python是纯粹的自由软件，源代码和解释器CPython遵循 GPL(GNU General Public License)协议
- Python具有丰富和强大的库。它常被昵称为胶水语言，能够把用其他语言制作的各种模块（尤其是C/C++）很轻松地联结在一起。常见的一种应用情形是，使用Python快速生成程序的原型，然后对其中有特别要求的部分，用更合适的语言改写，比如3D游戏中的图形渲染模块，性能要求特别高，就可以用C/C++重写，而后封装为Python可以调用的扩展类库。

Python优点有哪些

- Python是一种开源的、解析性的，面向对象的编程语言
- Python使用一种优雅的语法，可读性强
- Python具有丰富的库，可以处理各种工作
- Python支持类和多层继承等的面向对象编程技术
- Python可运行在多种计算机平台和操作系统中，如Unix、Windows、MacOS、Ubuntu、OS/2等等

为什么选择Python进行数据分析

- 在众多解释型语言中，Python最大的特点是拥有一个巨大而活跃的科学计算社区。进入21世纪以来，在行业应用和学术研究中采用python进行科学计算的势头越来越猛。
- 近年来，由于Python有不断改良的库（主要是pandas），使其成为数据处理任务的一大代替方案，结合其在通用编程方面的强大实力，完全可以只是用Python这一种语言去构建以数据为中心的应用程序。
- 作为一个科学计算平台，Python的成功源于能够轻松的集成C、C++以及Fortran代码。大部分现代计算机环境都利用了一些Fortran和C库来是西安线性代数、优选、积分、快速傅里叶变换以及其他诸如此类的算法。

1.1 Python环境搭建

- Python2.X和Python3.X
 - Python的3.0版本，常被称为Python 3000，或简称Py3k。相对于Python的早期版本，这是一个较大的升级。
 - 为了不带入过多的累赘，Python 3.0在设计的时候没有考虑向下相容，许多针对早期Python版本设计的程式都无法在Python 3.0上正常执行。
 - 大多数第三方库都正在努力地相容Python 3.0版本。即使无法立即使用Python 3.0，也建议编写相容Python 3.0版本的程式，然后使用Python 2.6, Python 2.7来执行。

In summary : Python 2.x is legacy, Python 3.x is the present and future of the language

1.2 Python安装

- Unix & Linux平台安装Python

- 打开WEB浏览器访问<http://www.python.org/download/>
- 选择适用于Unix/Linux的源码压缩包。
- 下载及解压压缩包。
- 如果你需要自定义一些选项修改Modules/Setup
- 执行 `./configure` 脚本
- `make`
- `make install`

- Window平台安装Python

- 打开WEB浏览器访问<http://www.python.org/download/>
- 在下载列表中选择Window平台安装包，包格式为：python-XYZ.msi 文件，XYZ 为你要安装的版本号。
- 要使用安装程序 python-XYZ.msi，Windows系统必须支持Microsoft Installer 2.0搭配使用。只要保存安装文件到本地计算机，然后运行它，看看你的机器支持MSI。Windows XP和更高版本已经有MSI，很多老机器也可以安装MSI。
- 下载后，双击下载包，进入Python安装向导，安装非常简单，你只需要使用默认的设置一直点击“下一步”直到安装完成即可。

集成开发环境选择

- **PyCharm**

- PyCharm 是由 JetBrains 打造的一款 Python IDE，支持 macOS、Windows、Linux 系统。
- PyCharm 功能：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 下载地址：
<https://www.jetbrains.com/pycharm/download/>

Pycharm软件界面

