

python™

Beautiful is better than ugly.  
Explicit is better than implicit. Simple  
is better than complex. Complex is better  
than complicated. Flat is better than  
nested. Sparse is better than dense.  
Readability counts. Special cases aren't  
special enough to break the rules.

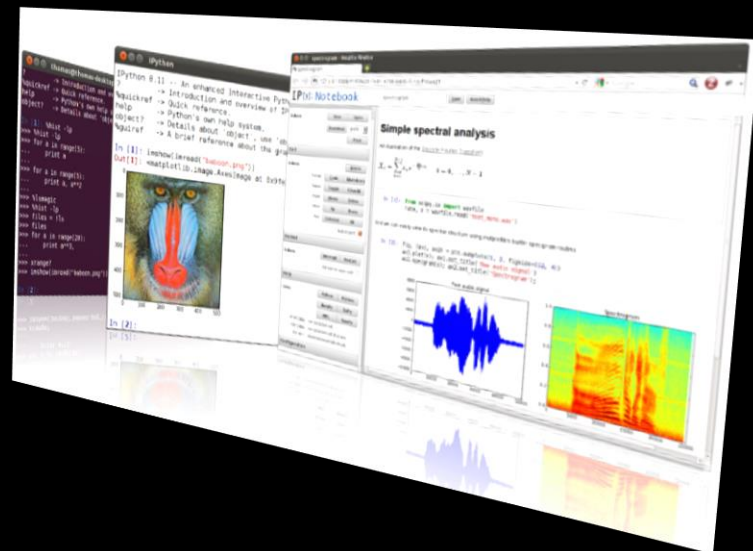
Although practicality beats purity. Errors should never  
pass silently. Unless explicitly silenced. In the face of  
ambiguity, refuse the temptation to guess. There should be one  
and preferably only one -- obvious way to do it. Although that  
way may not be the best, it is better than having several ways.  
Now is better than never. Although change is inevitable, to change  
is not easy, particularly if it's already there. It's better to have  
something that works than something that doesn't work. If the  
implementation is hard to explain, it's a bad idea. If the  
implementation is easy to explain, it may be a good idea.  
Namespaces are one honking great idea -- let's do more  
of those!

Although practicality beats purity. Errors should never  
pass silently. Unless explicitly silenced. In the face of  
ambiguity, refuse the temptation to guess. There should be one  
and preferably only one -- obvious way to do it. Although that  
way may not be the best, it is better than having several ways.  
Now is better than never. Although change is inevitable, to change  
is not easy, particularly if it's already there. It's better to have  
something that works than something that doesn't work. If the  
implementation is hard to explain, it's a bad idea. If the  
implementation is easy to explain, it may be a good idea.  
Namespaces are one honking great idea -- let's do more  
of those!

# Python 数据分析

梁斌

2016年10月30日



# 什么是“数据”？

- 结构化数据（structured data），如：
  - 多维数组（矩阵）
  - 表格型数据，其中各列可能是不同的类型（字符串、数值、日期等）。
  - 等等
- 原始数据
  - 可以被转化为更加适合分析和建模的结构化形式，即结构化数据。
  - 或者将原始数据的特征提取为某种结构化形式，例如，一组新闻文章可以被转化为词频表，从而用于情感分析。

# 工欲善其事必先利其“器”

- 电子表格软件
  - Microsoft Excel
  - OpenOffice Spreadsheet 等
- Python
  - 近几年非常流行的脚本（**scripting**）语言，用于编写简短而“粗糙”的小程序（即脚本）。
  - 拥有一个巨大而活跃的科学计算（**scientific computing**）社区。
  - 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。
  - 在**数据分析和交互**、**探索性计算**以及**数据可视化**等方面，Python将不可避免地接近于其他开源和商业的领域特定编程语言/工具，如R、Matlab等。

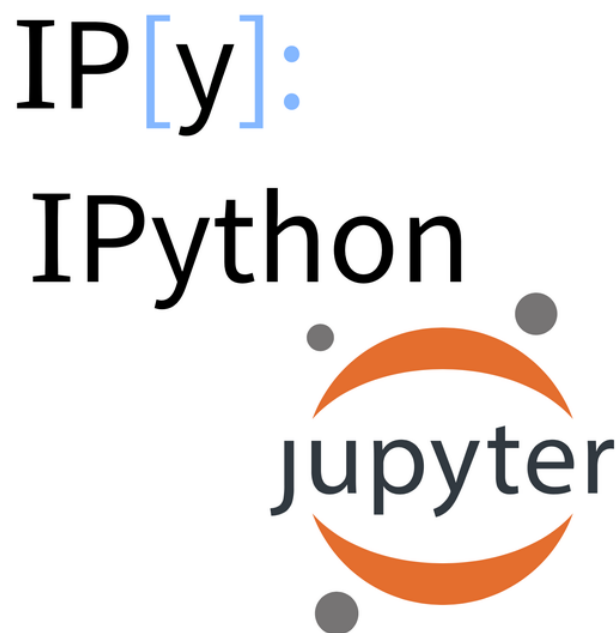
# Python 特点

- Python有不断改良的库（主要是pandas）。
- 在通用编程方面有强大实力。
- 完全可以只使用Python这一种语言去构建以数据为中心的应用程序。
- Python不仅适用于研究和原型构建，同时也使用于构建生态系统。
- Python是一种解释型编程语言，比用编译型语言编写的代码运行慢。
- 不适合高并发、多线程的应用程序。



# 编程环境

- PyCharm
- Eclipse + PyDev
- IPython
- Jupyter notebook



# 重要的Python库

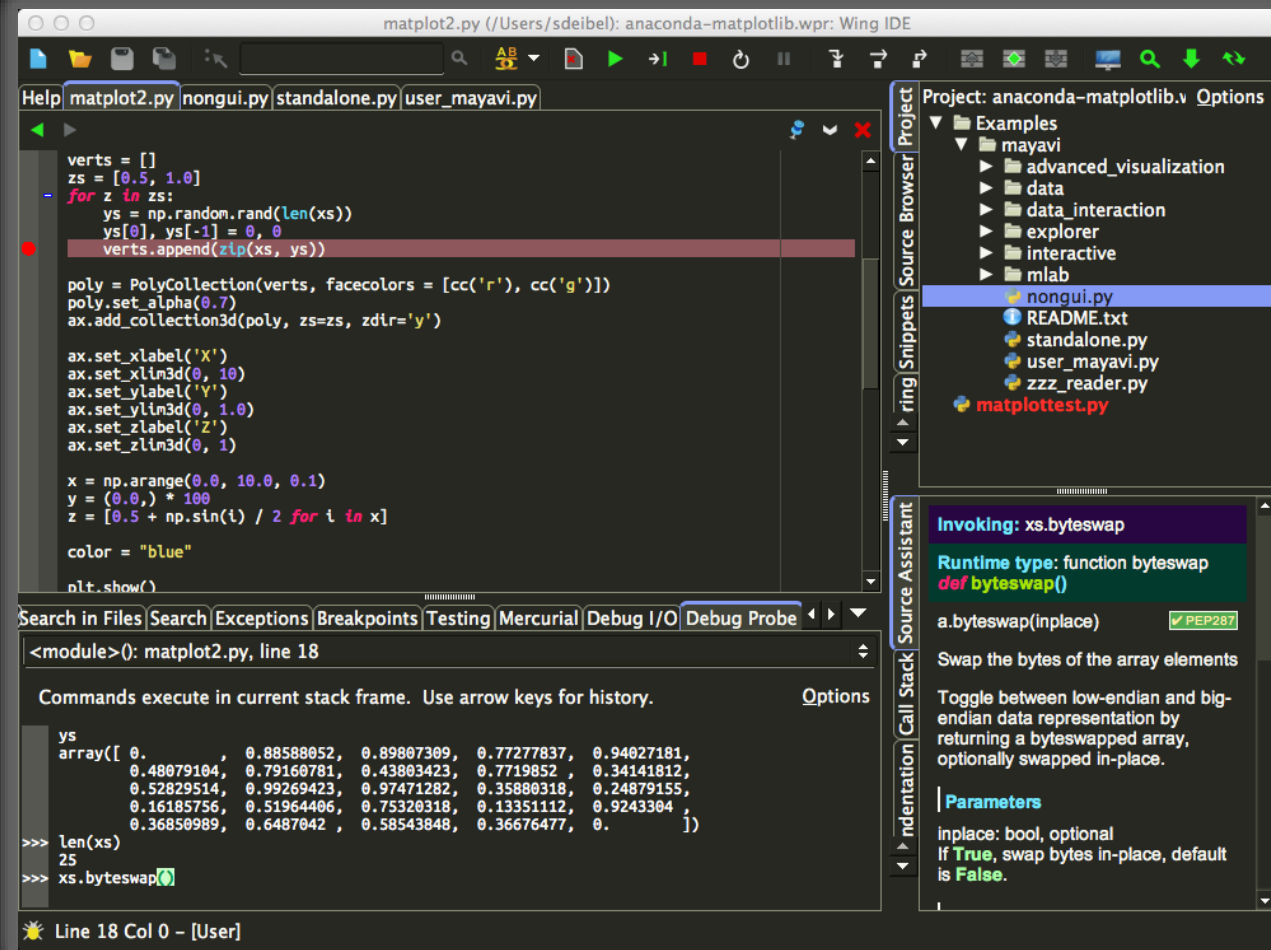
- **Numpy** ( Numerical Python ) , Python科学计算的基础包。
- SciPy ( Scientific Python ) , 一款方便、易于使用、转为科学和工程设计的Python工具包。
- **Matplotlib**, Python著名的绘图库。
- **Pandas** ( Python Data Analysis Library ) , 基于Numpy构建的含有更高级数据结构和工具的数据分析包。
- Scikit Learn, 基于Python的机器学习模块。

# 案例讲解


- 1880-2014年间全美婴儿姓名分析
- Kaggle US Baby Names项目提供了一份从1880年到2014年的婴儿名字数据。
- 利用这个数据集可以进行：
  - 计算指定姓名的年度比例
  - 计算某个姓名的相对排名
  - 计算各年度最流行的姓名，以及增长或减少最快的姓名
  - 分析姓名趋势：长度、元音、辅音等

# 代码讲解

- 数据概述
- 美剧对婴儿姓名的影响







*Any Questions?*