

## <<Python科学计算>>

### 图书基本信息

书名：<<Python科学计算>>

13位ISBN编号：9787302273608

10位ISBN编号：730227360X

出版时间：2012-1

出版时间：清华大学出版社

作者：张若愚

页数：621

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## &lt;&lt;Python科学计算&gt;&gt;

## 前言

前言 Python是一种面向对象的、动态的程序设计语言，具有非常简洁而清晰的语法，既可以用于快速开发程序脚本，也可以用于开发大规模的软件，特别适合于完成各种高层任务。

随着NumPy、SciPy、matplotlib、ETS 等众多程序库的开发，Python越来越适合于做科学计算。与科学计算领域最流行的商业软件MATLAB相比，Python是一门真正的通用程序设计语言，比MATLAB所采用的脚本语言的应用范围更广泛，有更多程序库的支持，适用于Windows和Linux等多种平台，完全免费并且开放源码。

虽然MATLAB中的某些高级功能目前还无法替代，但是对于基础性、前瞻性的科研工作和应用系统的开发，完全可以用Python来完成。

本书介绍如何用Python开发科学计算的应用程序，除了介绍数值计算之外，还着重介绍了如何制作交互式二维、三维图像，如何设计精巧的程序界面，如何与C语言编写的高速计算程序结合，如何编写声音、图像处理算法等内容。

由于Python的相关资源非常多，本书不可能全部涉及，相信读者在掌握本书所介绍的一些相关知识之后，只要充分利用互联网的搜索功能，就一定能够很快地找到合适的Python解决方案。

此外，由于绝大多数Python资源都开放源代码，因此读者将会很容易地对感兴趣的内容进行深度挖掘和研究。

本书适合于工科高年级本科生、研究生、工程技术人员以及计算机开发人员阅读。实例篇以信号处理为主，通过简单易懂的Python源程序，实际演示信号处理的一些基础知识和原理，因此特别适合于相关专业的学生作为扩展视野的补充阅读教材。

阅读本书的读者需要掌握Python语言的一些基础知识，下面是一个“自我检测列表”，如果读者熟悉下述内容，阅读本书的实例源代码就应该没有困难。

此外由于Python程序简单易读，即使读者没有接触过Python，也可以边阅读本书边通过其他书籍或免费教程学习Python。

基本语法：库的载入(import)、循环(for、while)、判断(if)、函数定义(def) 基本数据类型  
的用法：列表(list)、字典(dict)、元组(tuple)、字符串 面向对象的基本语法：类(class)、继承

C语言编程的基础知识 有关Python语言的基础知识，可以参考啄木鸟社区的Python图书简介。

啄木鸟社区的Python图书概览 本书所有演示程序，均在Windows XP系统下采用Python(x,y)通过测试。

如果读者觉得安装众多的Python程序库很麻烦，不妨下载安装Python(x,y)，或者直接使用本书所附光盘中的Python(x,y)安装程序。

## <<Python科学计算>>

### 内容概要

本书介绍如何用Python开发科学计算的应用程序，除了介绍数值计算之外，还着重介绍如何制作交互式的2D、3D图像，如何设计精巧的程序界面，如何与C语言编写的高速计算程序结合，如何编写声音、图像处理算法等内容。

书中涉及的Python扩展库包括NumPy、SciPy、SymPy、matplotlib、Traits、TraitsUI、Chaco、TVTK、Mayavi、VPython、OpenCV等，涉及的应用领域包括数值运算、符号运算、二维图表、三维数据可视化、三维动画演示、图像处理以及界面设计等。

书中以大量实例引导读者逐步深入学习，每个实例程序都有详尽的解释，并都能在本书推荐的运行环境中正常运行。

此外，本书附有大量的图表和插图，力求减少长篇的理论介绍和公式推导，以便读者通过实例和数据学习并掌握理论知识。

## <<Python科学计算>>

### 作者简介

张若愚，毕业于华中理工大学(现华中科技大学)通信工程专业，2004年获日本姬路工业大学(现兵库县立大学)硕士学位。

毕业后于日本神户制钢综合研究所从事研究开发工作至今，研究方向为：嵌入式DSP信号处理系统开发，嵌入式MCU控制系统开发，工业控制软件开发，信号处理、数据处理以及生产系统的计算机模拟。

在工作中他积极采用Python作为主要编程语言，在数据处理、信号分析、工业控制、算法模拟等领域取得了较好的研究成果。

已完成的研究课题有：嵌入式声音分离系统、车载音响设备、超声波探伤系统、压缩机系统的数字模拟等。

## &lt;&lt;Python科学计算&gt;&gt;

## 书籍目录

## 第1章 软件包的安装和介绍1

## 1.1 Python简介1

## 1.2 安装软件包2

## 1.2.1 Python(x,y)2

## 1.2.2 Enthought Python Distribution (EPD)3

## 1.3 方便的开发工具3

## 1.3.1 IPython4

## 1.3.2 Spyder8

## 1.3.3 Wing IDE 10112

## 1.4 函数库介绍13

## 1.4.1 数值计算库13

## 1.4.2 符号计算库14

## 1.4.3 界面设计14

## 1.4.4 绘图与可视化14

## 1.4.5 图像处理和计算机视觉15

## 第2章 NumPy——快速处理数据16

## 2.1 ndarray对象16

## 2.1.1 创建数组16

## 2.1.2 存取元素21

## 2.1.3 多维数组24

## 2.1.4 结构数组29

## 2.1.5 内存结构32

## 2.2 ufunc运算35

## 2.2.1 四则运算37

## 2.2.2 比较和布尔运算39

## 2.2.3 自定义ufunc函数40

## 2.2.4 广播42

## 2.2.5 ufunc函数的方法46

## 2.3 多维数组的下标存取48

## 2.3.1 下标对象48

## 2.3.2 整数数组作为下标49

## 2.3.3 一个复杂的例子51

## 2.3.4 布尔数组作为下标53

## 2.4 庞大的函数库54

## 2.4.1 求和、平均值、方差54

## 2.4.2 最值和排序55

## 2.4.3 多项式函数57

## 2.4.4 分段函数60

## 2.4.5 统计函数62

## 2.5 线性代数65

## 2.5.1 各种乘积运算65

## 2.5.2 解线性方程组67

## 2.6 掩码数组69

## 2.7 文件存取72

## &lt;&lt;Python科学计算&gt;&gt;

- 2.8 内存映射数组75
- 第3章 SciPy——数值计算库79
  - 3.1 常数和特殊函数79
  - 3.2 优化——optimize81
    - 3.2.1 最小二乘拟合81
    - 3.2.2 函数最小值84
    - 3.2.3 非线性方程组求解86
  - 3.3 插值——interpolate88
    - 3.3.1 B样条曲线插值88
    - 3.3.2 外推和Spline拟合90
    - 3.3.3 二维插值91
  - 3.4 数值积分——integrate93
    - 3.4.1 球的体积93
    - 3.4.2 解常微分方程组95
  - 3.5 信号处理——signal97
    - 3.5.1 中值滤波97
    - 3.5.2 滤波器设计98
  - 3.6 图像处理——ndimage100
    - 3.6.1 膨胀和腐蚀101
    - 3.6.2 Hit和Miss102
  - 3.7 统计——stats105
    - 3.7.1 连续和离散概率分布105
    - 3.7.2 二项、泊松、伽玛分布108
  - 3.8 嵌入C语言程序——weave112
- 第4章 SymPy——符号运算好帮手115
  - 4.1 从例子开始115
    - 4.1.1 封面上的经典公式115
    - 4.1.2 球体体积117
  - 4.2 数学表达式119
    - 4.2.1 符号119
    - 4.2.2 数值121
    - 4.2.3 运算符和函数122
  - 4.3 符号运算125
    - 4.3.1 表达式变换和化简125
    - 4.3.2 方程128
    - 4.3.3 微分129
    - 4.3.4 微分方程130
    - 4.3.5 积分131
  - 4.4 其他功能133
    - 4.4.1 平面几何133
    - 4.4.2 绘图135
- 第5章 matplotlib——绘制精美的图表139
  - 5.1 快速绘图139
    - 5.1.1 使用pyplot模块绘图139
    - 5.1.2 以面向对象方式绘图142
    - 5.1.3 配置属性143

## &lt;&lt;Python科学计算&gt;&gt;

- 5.1.4 绘制多个子图145
- 5.1.5 配置文件147
- 5.1.6 在图表中显示中文149
- 5.2 Artist对象152
  - 5.2.1 Artist对象的属性154
  - 5.2.2 Figure容器155
  - 5.2.3 Axes容器156
  - 5.2.4 Axis容器159
  - 5.2.5 Artist对象的关系163
- 5.3 坐标变换和注释164
  - 5.3.1 4种坐标系167
  - 5.3.2 坐标变换的步骤169
  - 5.3.3 制作阴影效果173
  - 5.3.4 添加注释174
- 5.4 绘图函数简介177
  - 5.4.1 对数坐标图177
  - 5.4.2 极坐标图178
  - 5.4.3 柱状图179
  - 5.4.4 散列图180
  - 5.4.5 图像181
  - 5.4.6 等值线图184
  - 5.4.7 三维绘图187
- 第6章 Traits——为Python添加类型定义190
  - 6.1 开发背景190
  - 6.2 Trait属性的功能192
  - 6.3 Trait类型对象196
  - 6.4 Trait的元数据198
  - 6.5 预定义的Trait类型200
  - 6.6 Property属性204
  - 6.7 Trait属性监听206
  - 6.8 Event和Button属性210
  - 6.9 Trait属性的从属关系211
  - 6.10 动态添加Trait属性213
  - 6.11 创建自己的Trait类型215
    - 6.11.1 从TraitType继承215
    - 6.11.2 使用Trait()217
    - 6.11.3 定义TraitHandler类219
- 第7章 TraitsUI——轻松制作用户界面221
  - 7.1 默认界面221
  - 7.2 用View定义界面222
    - 7.2.1 外部视图和内部视图222
    - 7.2.2 多模型视图226
    - 7.2.3 Group对象228
    - 7.2.4 配置视图231
  - 7.3 用Handler控制界面和模型232

## &lt;&lt;Python科学计算&gt;&gt;

- 7.3.1 用Handler处理事件233
- 7.3.2 Controller和UIInfo对象237
- 7.3.3 响应Trait属性的事件238
- 7.4 属性编辑器240
  - 7.4.1 编辑器演示程序241
  - 7.4.2 对象编辑器243
  - 7.4.3 字符串列表编辑器248
  - 7.4.4 对象列表编辑器250
- 7.5 菜单、工具条和状态栏252
- 7.6 设计自己的编辑器255
  - 7.6.1 Trait编辑器的工作原理255
  - 7.6.2 制作matplotlib的编辑器259
  - 7.6.3 CSV数据绘图工具262
- 第8章 Chaco——交互式图表264
  - 8.1 面向脚本绘图264
  - 8.2 面向应用绘图265
    - 8.2.1 多条曲线267
    - 8.2.2 Plot对象的结构271
    - 8.2.3 编辑绘图属性275
    - 8.2.4 容器(Container)276
  - 8.3 添加交互工具279
    - 8.3.1 平移和缩放279
    - 8.3.2 选取范围282
    - 8.3.3 选取数据点284
    - 8.3.4 套索工具287
  - 8.4 二次开发289
    - 8.4.1 用Kiva库在数组上绘图290
    - 8.4.2 Enable库的组件292
    - 8.4.3 设计圆形选择工具297
    - 8.4.4 制作动画演示301
- 第9章 TVTK——数据的三维可视化303
  - 9.1 流水线(Pipeline)304
    - 9.1.1 显示圆锥304
    - 9.1.2 用ivtk观察流水线307
  - 9.2 数据集(Dataset)313
    - 9.2.1 ImageData313
    - 9.2.2 RectilinearGrid318
    - 9.2.3 StructuredGrid319
    - 9.2.4 PolyData321
  - 9.3 可视化实例324
    - 9.3.1 切面325
    - 9.3.2 等值面330
    - 9.3.3 流线333
  - 9.4 TVTK的改进337
    - 9.4.1 TVTK的基本用法338
    - 9.4.2 Trait属性339
    - 9.4.3 序列化(Pickling)339



## &lt;&lt;Python科学计算&gt;&gt;

- 9.4.4 集合迭代340
- 9.4.5 数组操作341
- 第10章 Mayavi——更方便的可视化343
  - 10.1 用mlab快速绘图343
    - 10.1.1 点和线343
    - 10.1.2 Mayavi的流水线345
    - 10.1.3 二维图像的可视化348
    - 10.1.4 网格面352
    - 10.1.5 修改和控制流水线356
    - 10.1.6 标量场358
    - 10.1.7 矢量场361
  - 10.2 Mayavi和TVTK之间的关系363
    - 10.2.1 显示TVTK流水线363
    - 10.2.2 两条流水线之间的关系365
  - 10.3 Mayavi应用程序367
    - 10.3.1 操作流水线368
    - 10.3.2 命令行和对象浏览器371
  - 10.4 将Mayavi嵌入到界面中374
- 第11章 VPython——制作3D演示动画378
  - 11.1 场景、物体和照相机378
    - 11.1.1 控制场景窗口380
    - 11.1.2 控制照相机383
    - 11.1.3 模型的属性384
    - 11.1.4 三维模型387
  - 11.2 制作动画演示390
    - 11.2.1 简单动画390
    - 11.2.2 盒子中反弹的球391
  - 11.3 与场景交互393
    - 11.3.1 响应键盘事件394
    - 11.3.2 响应鼠标事件394
  - 11.4 用界面控制场景397
  - 11.5 创建复杂模型400
    - 11.5.1 faces()的用法400
    - 11.5.2 读入模型数据402
- 第12章 OpenCV——图像处理和计算机视觉408
  - 12.1 存储图像数据的Mat对象409
    - 12.1.1 Mat对象和NumPy数组410
    - 12.1.2 像素点类型414
    - 12.1.3 其他数据类型415
    - 12.1.4 Vector类型417
    - 12.1.5 在图像上绘图418
  - 12.2 图像处理421
    - 12.2.1 二维卷积421
    - 12.2.2 形态学运算424

## &lt;&lt;Python科学计算&gt;&gt;

- 12.2.3 填充——floodFill426
- 12.2.4 去瑕疵——inpaint427
- 12.3 图像变换428
  - 12.3.1 几何变换428
  - 12.3.2 重映射——remap430
  - 12.3.3 直方图统计433
  - 12.3.4 二维离散傅立叶变换437
- 12.4 图像识别440
  - 12.4.1 用霍夫变换检测直线和圆440
  - 12.4.2 图像分割444
  - 12.4.3 用SURF进行特征匹配450
- 第13章 数据和文件453
  - 13.1 声音的输入输出453
    - 13.1.1 读写WAV文件453
    - 13.1.2 用pyAudio播放和录音456
  - 13.2 视频的输入输出459
    - 13.2.1 读写视频文件459
    - 13.2.2 安装视频编码464
  - 13.3 读写HDF5文件465
  - 13.4 读写Excel文件469
    - 13.4.1 写Excel文件469
    - 13.4.2 读Excel文件471
- 第14章 数字信号系统473
  - 14.1 FIR和IIR滤波器473
  - 14.2 FIR滤波器设计477
    - 14.2.1 用firwin()设计滤波器479
    - 14.2.2 用remez()设计滤波器481
    - 14.2.3 滤波器的级联483
  - 14.3 IIR滤波器设计485
    - 14.3.1 巴特沃斯低通滤波器485
    - 14.3.2 双线性变换487
    - 14.3.3 滤波器的频带转换490
  - 14.4 数字滤波器的频率响应494
  - 14.5 二次均衡滤波器设计工具497
  - 14.6 零相位滤波器500
  - 14.7 重取样501
- 第15章 频域信号处理505
  - 15.1 FFT演示程序505
    - 15.1.1 FFT知识复习505
    - 15.1.2 合成时域信号509
    - 15.1.3 三角波FFT演示程序511
  - 15.2 观察信号的频谱512
    - 15.2.1 窗函数515
    - 15.2.2 频谱平均517
    - 15.2.3 谱图519
  - 15.3 卷积运算522

## &lt;&lt;Python科学计算&gt;&gt;

- 15.3.1 快速卷积522
- 15.3.2 分段运算524
- 15.4 信号处理526
  - 15.4.1 基本框架527
  - 15.4.2 频域滤波器528
  - 15.4.3 频率变调处理530
  - 15.4.4 用谱图差减法降噪531
- 15.5 Hilbert变换532
- 第16章 用C语言提高计算效率537
  - 16.1 用ctypes调用DLL库537
  - 16.2 用Weave嵌入C++程序541
    - 16.2.1 Weave的工作原理541
    - 16.2.2 处理NumPy数组543
    - 16.2.3 使用blitz()提速546
    - 16.2.4 扩展模块548
  - 16.3 用Cython将Python编译成C549
    - 16.3.1 编译Cython程序549
    - 16.3.2 提高计算效率550
    - 16.3.3 快速访问NumPy数组553
  - 16.4 用SWIG创建扩展模块555
    - 16.4.1 SWIG的调用方法和实例555
    - 16.4.2 SWIG基础558
    - 16.4.3 SWIG处理NumPy数组566
- 第17章 自适应滤波器571
  - 17.1 自适应滤波器简介571
    - 17.1.1 系统识别571
    - 17.1.2 信号预测572
    - 17.1.3 信号均衡572
  - 17.2 NLMS计算公式573
  - 17.3 用NumPy实现NLMS算法575
    - 17.3.1 系统辨识模拟577
    - 17.3.2 信号均衡模拟579
    - 17.3.3 卷积逆运算581
  - 17.4 用C语言加速NLMS运算583
    - 17.4.1 用SWIG编写扩展模块583
    - 17.4.2 用Weave嵌入C++程序586
- 第18章 单摆和双摆模拟588
  - 18.1 单摆模拟588
    - 18.1.1 小角度时的摆动周期589
    - 18.1.2 大角度时的摆动周期590
  - 18.2 双摆模拟592
    - 18.2.1 公式推导592
    - 18.2.2 微分方程的数值解595
    - 18.2.3 动画演示598
- 第19章 分形几何599

## <<Python科学计算>>

- 19.1 Mandelbrot集合599
  - 19.1.1 使用NumPy加速计算601
  - 19.1.2 使用Weave加速计算603
  - 19.1.3 连续的逃逸时间604
  - 19.1.4 Mandelbrot演示程序605
- 19.2 迭代函数系统(IFS)606
  - 19.2.1 二维仿射变换610
  - 19.2.2 迭代函数系统设计器610
- 19.3 L-System分形613
- 19.4 分形山脉616
  - 19.4.1 一维中点移位法616
  - 19.4.2 二维中点移位法618
  - 19.4.3 菱形方形算法619

## &lt;&lt;Python科学计算&gt;&gt;

## 章节摘录

版权页：插图：为了减少计算时间，我们不在迭代循环中计算选择迭代方程的随机数，而是事先通过每个迭代方程的概率，计算出选择数组select。

注意这里使用accumulate（）先将概率累加，然后产生一组0到1之间的随机数，通过判断随机数所在的区间来选择不同的方程下标。

也可以使用SciPy的stats模块中的离散随机变量来产生这个随机下标数组。

最后通过scattero将得到的坐标点绘制成散列图。

其中：s参数是每个散列点的大小，因为我们要绘制10万个点，为了提高绘图速度，我们选择点的大小为1个像素；c参数是点的颜色，这里选择绿色；marker参数是点的形状，“s”表示正方形，方形的绘制是最快的；linewidths参数是点的边框宽度，0表示没有边框。

此外，c参数还可以传入一个数组，作为每个点的颜色值。

我们将计算坐标的公式下标传入，这样可以直观地看出点是由哪个公式迭代产生的。

## <<Python科学计算>>

### 媒体关注与评论

在书中，作者为我们介绍了科学计算编程所需的各个方面。从NumPy库和SciPy算法工具库的基础开始，介绍了任何科学计算应用程序所需的基本工具。之后，作者很恰当地介绍了二维绘图以及三维可视化库——matplotlib、Chaco和Mayavi。用Traits和TraitsUI进行应用程序和界面开发，以及用Cython、Weave、ctypes和SWIG等与传统的C语言库相结合等内容在书中也有很好的介绍。除了这些核心的工具之外，本书还介绍了使用SymPy进行数学符号运算以及其他各种有用的主题。  
——Eric Jones

## &lt;&lt;Python科学计算&gt;&gt;

## 编辑推荐

《Python科学计算》主要特色：如果您有一定的Python编程经验，经常需要对数值数据做一些处理、分析、可视化的工作，那么《Python科学计算》的内容十分适合您阅读。

如果您是程序员，即使工作与科学计算无关，《Python科学计算》所介绍的各种扩展库也将能丰富您的工具箱，为您的工作提供新的解决方案。

如果您是科研人员，即使对Python或编程并不十分了解，但只要花一点时间学习Python语言以及《Python科学计算》介绍的内容，就能让您的工作效率大幅提高。

附赠光盘中包含书中用到的Python扩展程序、学习《Python科学计算》时用到的工具软件以及书中所有实例的源程序。

历时三年，精心编写。

Enthought公司CEO Eric Jones作序推荐，《Python科学计算》所提供的一站式服务，能够指导读者从最初的入门直到创建一个漂亮的、全功能的分析与模拟应用程序。

NumPy——快速处理数据，SymPy——符号运算好帮手，Traits——为Python添加类型定义，Chaco——交互式图表，Mayavi——更方便的可视化，OpenCV——图像处理和计算机视觉，数字信号、滤波器、频域处理，用C语言提高计算效率，SciPy——数值计算库，matplotlib——绘制精美的图表，TraitsUI——轻松制作用户界面，TVTK——数据的三维可视化，VPython——制作3D演示动画，声音与视频数据处理，动画模拟、分形几何。

《Python科学计算》读者对象：《Python科学计算》适合科研人员阅读，但书中介绍的NumPy数据处理、matplotlib绘图、TraitsUI界面应用程序开发、各种格式的数据文件的处理以及用C语言编写扩展等内容，并非局限于科学计算领域，也适合一般的Python开发人员阅读。

#### 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>