

# Python天文应用

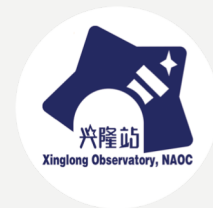
## ② 工具包

郑捷（小林）@ 中国科学院国家天文台兴隆观测基地

2018-12-18 NAOC

# Python天文应用

- 语言基础
- 天文工具包
- 天文数据处理与应用

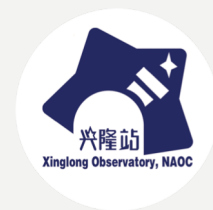




# 软件包 概述

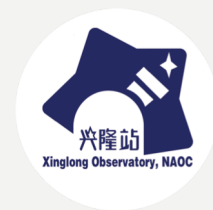
# 软件包-概述

- Python语言大量依赖软件包
- 生态环境
- 和其他语言的扩展包相似
  
- 内置软件包
- 常用软件包
- 专业领域软件包



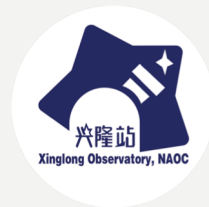
# 软件包-内置包

- 在基础语法之外的基础功能补充
- 例如：
  - sys, os, math, ...
- 内置包一般不升级
- numpy算半内置
  - 一般自动预装，可以升级



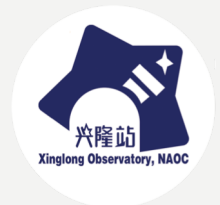
# 软件包-常用包

- 非语言基础
- 功能通用，并且常用
- numpy, scipy, matplotlib, pandas, ...
- 基本上是标配



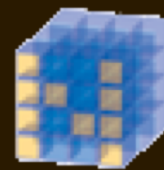
# 软件包-专业领域包

- python应用广的精髓
- 根据各个领域需要建设和使用
- **astropy**
- 其他领域
  - Web: django, flask, ...
  - AI: scikit-learn, tensorflow, ...
  - .....



# 软件包

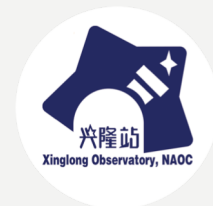
# numpy





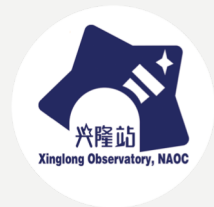
# numpy-概述

- numpy是一个数学工具包
- 与math不同：提供数组运算
- 是几乎所有科学计算、机器学习、绘图等包的基础
  
- 常用引入语句
  - `import numpy as np`



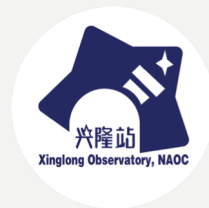
# numpy-创建数组

- 从列表创建
  - `a = np.array([1,2,3,4,5])`
- 直接创建
  - `b = np.zeros((10, 15), dtype=int)`
  - `c = np.ones(15, dtype=np.float32)`
  - `d = np.empty_like(c)`
- 创建之后无差别



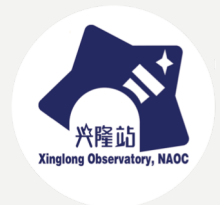
# numpy-数组特性

- 可以一维也可以多维
- 下标可以使用切片 slice
- 元素必须是相同类型
- 元素可以是记录，而不一定是单值
- 可以做为整体运算，也可以是元素运算
- 会进行必要的广播操作 broadcast
- 注意和列表的联系和区别



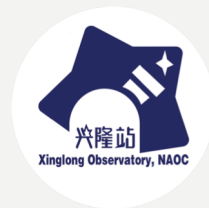
# numpy-数组信息

- 对数组a:
  - a.shape, a.ndim, a.size, a.nbytes
  - a.dtype, a.itemsize
  - len(a)



# numpy-数组运算

- 一切数学运算自动对数组有效  
(利用面向对象的重载运算符特性)
- 数组和数组运算
  - 要求两个数组维度相同
  - 逐个元素运算, 非矩阵、向量运算
- 数组和单值运算
  - 单值被广播到数组的每个元素



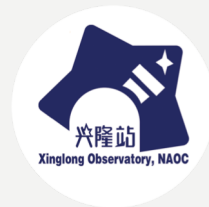
# numpy-数组数学函数

- 大部分math中的数学函数在numpy中有对应函数
- 可以针对标量或者数组
- 例如: `np.sin`
- 注意: 三角函数单位是弧度
- `np.deg2rad` `np.rad2deg`



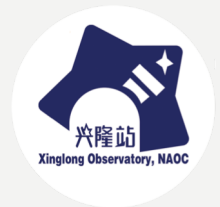
# numpy-数组统计函数

- 简单统计
  - sum, min, max, mean, median, std
  - 特别地: argmax, argmin
- 排序
  - sort argsort



# numpy-查找

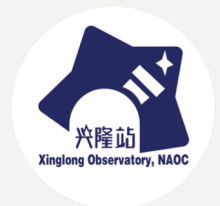
- 切片和选择性访问
- 从数组中找到符合条件的元素
  - np.where
- 布尔类型下标
- 多维数组的where





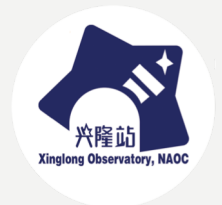
# numpy-引用和复制

- python中一切都是对象
- 除基础数据类型外，都是引用
- $a = \text{np.array}([1, 2, 3])$   
 $b = a$   
 $b[1] = 100$
- 以上a、b实际上是同一个对象
- 必要时使用copy函数



# numpy-记录数组

- 表示一个二维表的数组
- 生成
  - `dt = [...]` 列描述
  - `a = np.empty((size), dtype=dt)`
  - `b = np.genfromtxt(filename, dtype=dt)`
- 可以是自行生成，逐个填充，也可以从文本读出，或者从fits文件读出



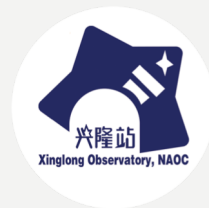
# numpy-记录数组

	记录数组	二维数组
维数	一维	二维
每个单元格数据类型	可以不一致	必须一致
按列访问	可以，是一维数组	可以，是一维数组
按行访问	可以，是一个记录	可以，是一维数组
数组运算	不可以	可以
矩阵运算	不可以	可以
转置、变形等	不可以	可以
内容	数据表等	图像等



# numpy-随机数

- np.random
- 提供随机数相关的各种功能
  - rand
  - randint
  - randn
  - shuffle
  - ...

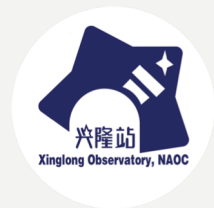


# 软件包 scipy



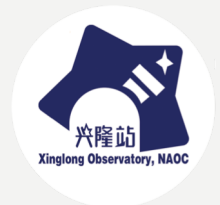
# scipy-概述

- 科学计算
- 在np基础上，进行计算
- 复制（或者说引入）了np的基本功能
- 提供积分、拟合、插值、统计等等
  
- `import scipy.xxx`
- 注意要引入子包，而不是总包



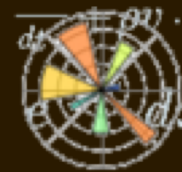
# scipy-子包

- Basic functions
- Special functions (scipy.special)
- Integration (scipy.integrate)
- Optimization (scipy.optimize)
- Interpolation (scipy.interpolate)
- Fourier Transforms (scipy.fftpack)
- Signal Processing (scipy.signal)
- Linear Algebra (scipy.linalg)
- Sparse Eigenvalue Problems with ARPACK
- Compressed Sparse Graph Routines (scipy.sparse.csgraph)
- Spatial data structures and algorithms (scipy.spatial)
- Statistics (scipy.stats)
- Multidimensional image processing (scipy.ndimage)
- File IO (scipy.io)



# 软件包

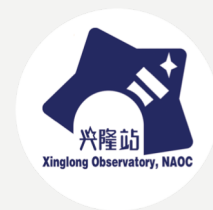
# matplotlib





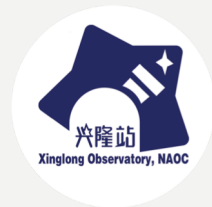
# matplotlib-概述

- 脱胎于matlib的2D绘图工具包
- 包括
  - 曲线图
  - 散点图
  - 直方图
  - 灰度图
  - ...



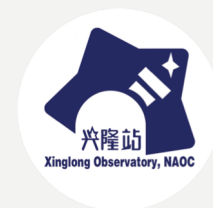
# matplotlib-图结构

- 常用的导入方式
  - `import matplotlib.pyplot as plt`
- 最简单的绘图方式
  - `plt.plot(...)`
- 使用figure和axis的绘图方式
  - `fig = plt.figure()`
  - `ax = fig.add_subplot(...)`
  - `ax.plot(...)`



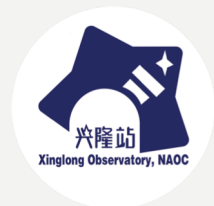
# matplotlib-基本绘图

- 曲线图
  - `ax.plot(x, y)`
- 如果只有y没有x, 那么x默认为0~n-1
- 默认为实线
- 绘制后:
  - jupyter中, 如果设置了`%matplotlib inline`, 会直接显示在界面上
  - 在shell中, 需要用`plt.show()`



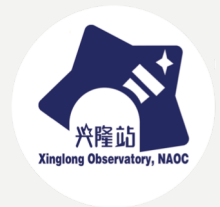
# matplotlib-基本绘图

- plot函数可以通过参数控制
- `ax.plot(x, y, 'rx')`
- 用红色绘制散点图，图例为x
- 第三个参数为简单的控制
  - 颜色: `rgbmcykw`
  - 线型: `-- - -. :` 等等
  - 散点标记: `x + , . o * s ^ < >` 等等



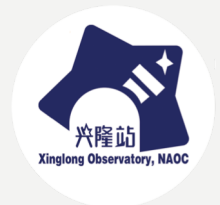
# matplotlib-图例

- plot函数的label参数
  - `ax.plot(x, y, label='xxxxx')`
  - `ax.legend()`
- 通过label给每个曲线加标签
- legend函数在图像上加图例
- 图例的位置是自动的，也可以指定



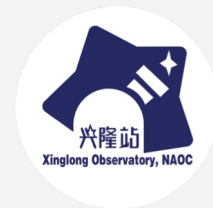
# matplotlib-图像控制

- 可以设置图框、坐标轴等
  - `ax.set_xlim` 坐标轴范围
  - `ax.set_xlabel` 坐标轴标签
  - `ax.set_xticks` 坐标轴数值
  - `ax.set_xscale` 坐标轴缩放
  - 以上x可以换成y



# matplotlib-文字

- `set_xlabel`, `set_title` 等
- `text`: 直接写文字
- `annotate`: 标注文字, 文字加线
  - `ax.annotate('$\alpha$', xy=(2, 1), xytext=(3, 1.5), arrowprops=dict(facecolor='black', shrink=0.05), )`
- 文字可以使用LaTeX符号



# matplotlib-直方图

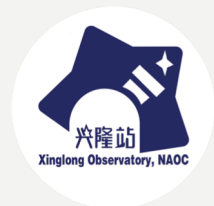
- 直方图自动进行统计并绘图
- `ax.hist(values, range=(low, high), bins=n)`
- 一般需要指定range和bins, 否则默认为数据的最小值到最大值, 分10份
- 前面的参数均可用





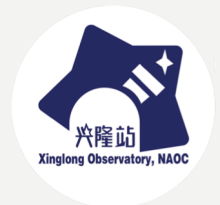
# matplotlib-显示位图

- 逐个像素表示的图叫位图
- 日常照片都是位图
- 直接记录绘制的方法的叫矢量图
- `ax.imshow`
- 严格来说是显示灰度图



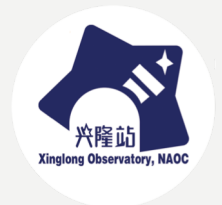
# matplotlib-多子图绘图

- `ax = fig.add_subplot`
- 可以在一个fig中加多个子图(subplot), 或者叫坐标系(axes)
- 多子图可以规则排列, 也可以不规则
  - `add_subplot(331)`
  - `add_subplot(3, 2, 2)`
  - `add_subplot(111, position=...)`



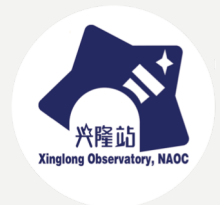
# matplotlib-图像保存

- `fig.savefig`
- 支持eps, pdf, pgf, png, ps, raw, rgba, svg, svgz等格式
- 自动根据扩展名识别
- 一般绘图结束后要关闭
  - `plt.close(fig)`



# matplotlib-网上资源

- <https://matplotlib.org/>
- <http://astroplotlib.stsci.edu/>



谢谢