



TAIJI LABORATORY
FOR GRAVITATIONAL WAVE UNIVERSE



ICTP-AP
International Centre
for Theoretical Physics Asia-Pacific
国际理论物理中心-亚太地区



中国科学院大学
University of Chinese Academy of Sciences

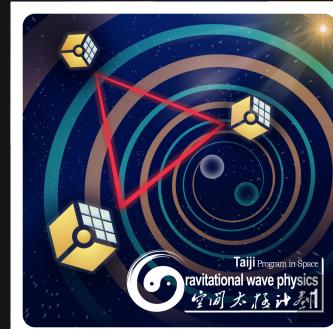
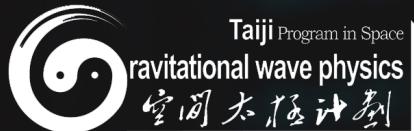
引力波数据探索：编程与分析实战训练营

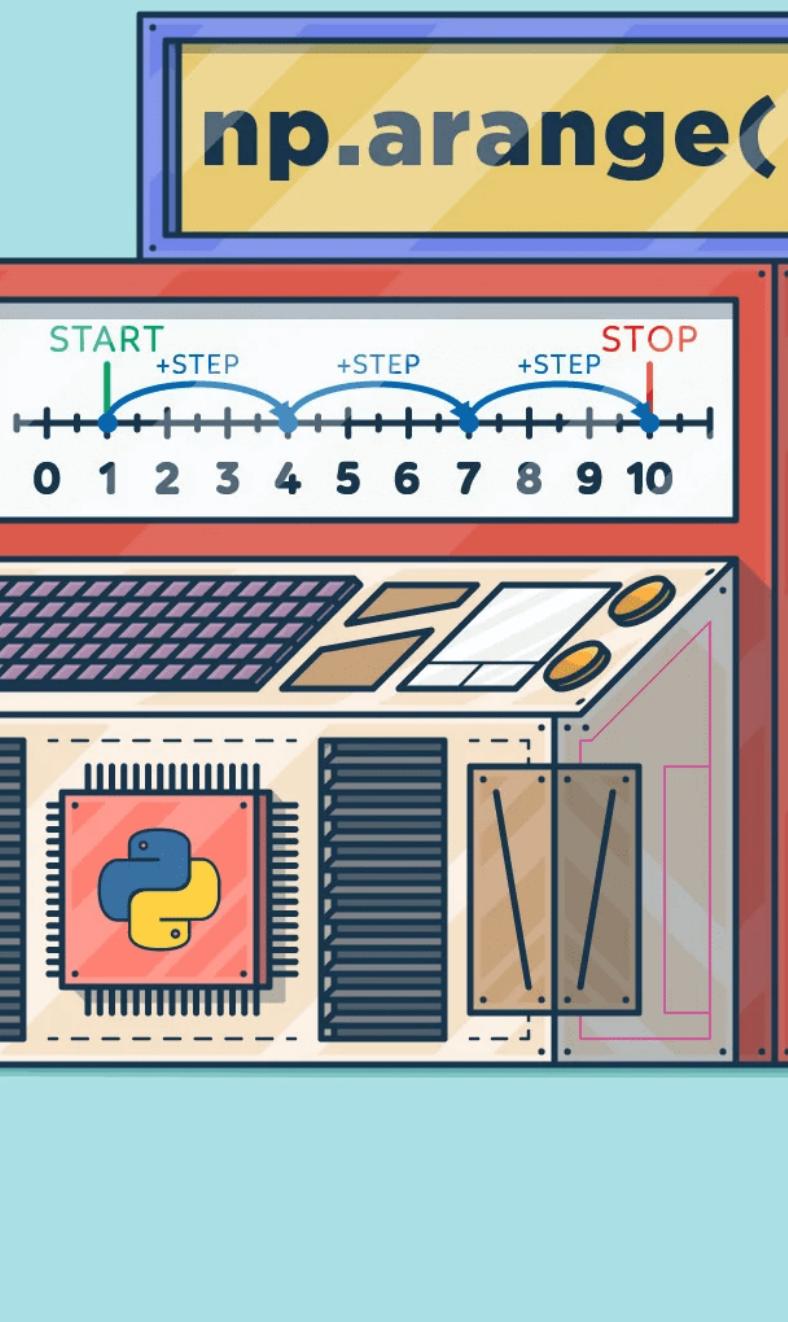
第 2 部分 基于 Python 的数据分析基础 数据分析实训之 Numpy

主讲老师：王赫

ICTP-AP, UCAS

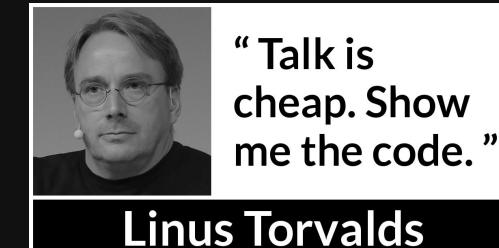
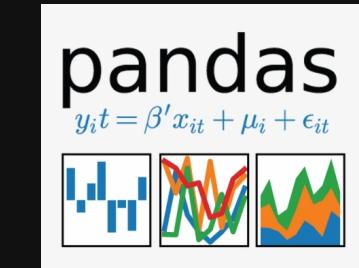
2023/12/01





数据分析实训之 Numpy

- Numpy 的理解与解读
 - Numpy 中的数据结构
 - Numpy 中的数据切片
 - Numpy 中的广播机制
 - Numpy 中的线性代数与数理统计



Linus Torvalds



Numpy 简介



- NumPy 是 Python 语言的一个扩展程序库。支持高阶大量的维度数组与矩阵运算，此外也针对数组运算提供大量的数学函数库。

- Numpy 官方网址: <https://numpy.org>
- Numpy 中文文档: <https://www.numpy.org.cn/>

- Numpy 中的数据结构: **ndarray** (即 n-dimensional array, 多维数组)

功能强大的N维数组对象



集成 C/C+和 Fortran 代码的工具

精密广播功能函数

强大的线性代数、傅立叶变换和随机数功能

- Numpy 运算速度快:

- 它将常用的数学函数都支持向量化运算，使得这些数学函数能够直接对数组进行操作，将本来需要在Python级别进行的循环，放到C语言的运算中，明显地提高了程序的运算速度。

- Numpy 的 ndarray 与 Python 的 list:

- **list** 列表本身是为了处理更广泛、更通用的目的而构建的
- **ndarray** 对于处理数组类型结构的数据会更加方便

```
In [1]: import numpy as np
import time

In [5]: def func(values):
    result = []
    for v in values:
        result.append(v * v)
    return result

data = range(10000)
%timeit func(data)
980 µs ± 30.7 µs per loop (mean ± std. dev. of 7 runs, 1000 loops each)

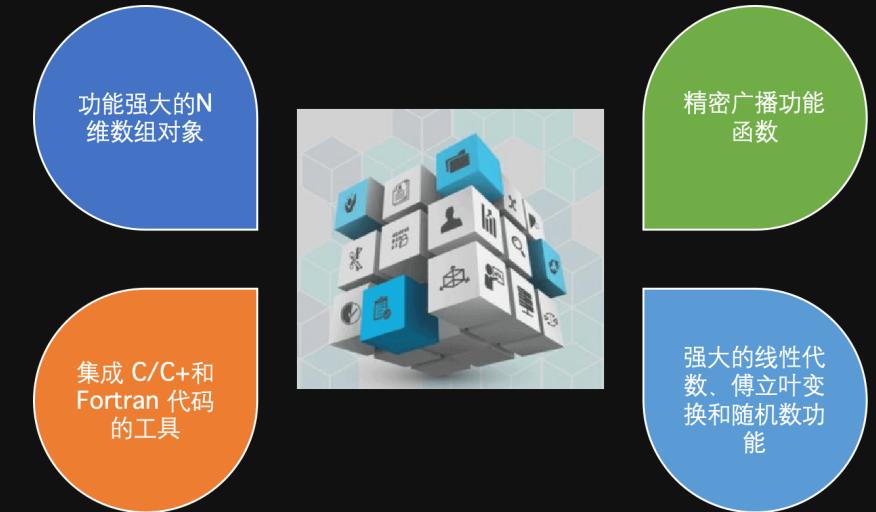
In [6]: arr = np.arange(0,10000)
%timeit arr ** arr
322 µs ± 7.82 µs per loop (mean ± std. dev. of 7 runs, 1000 loops each)
```



Numpy 简介

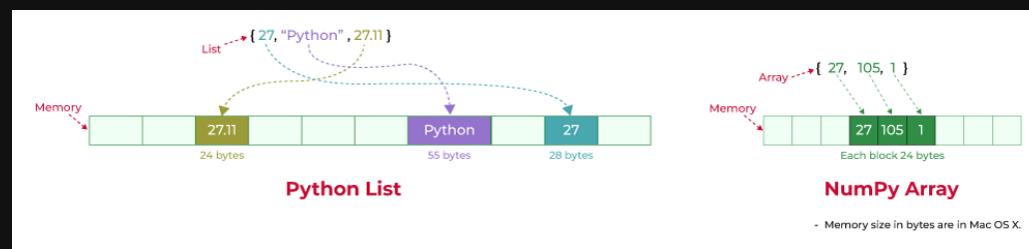


- NumPy 是 Python 语言的一个扩展程序库。支持高阶大量的维度数组与矩阵运算，此外也针对数组运算提供大量的数学函数库。
 - Numpy 官方网址: <https://numpy.org>
 - Numpy 中文文档: <https://www.numpy.org.cn/>
- Numpy 中的**数据结构**: **ndarray** (即 n-dimensional array, 多维数组)



Numpy 要点：

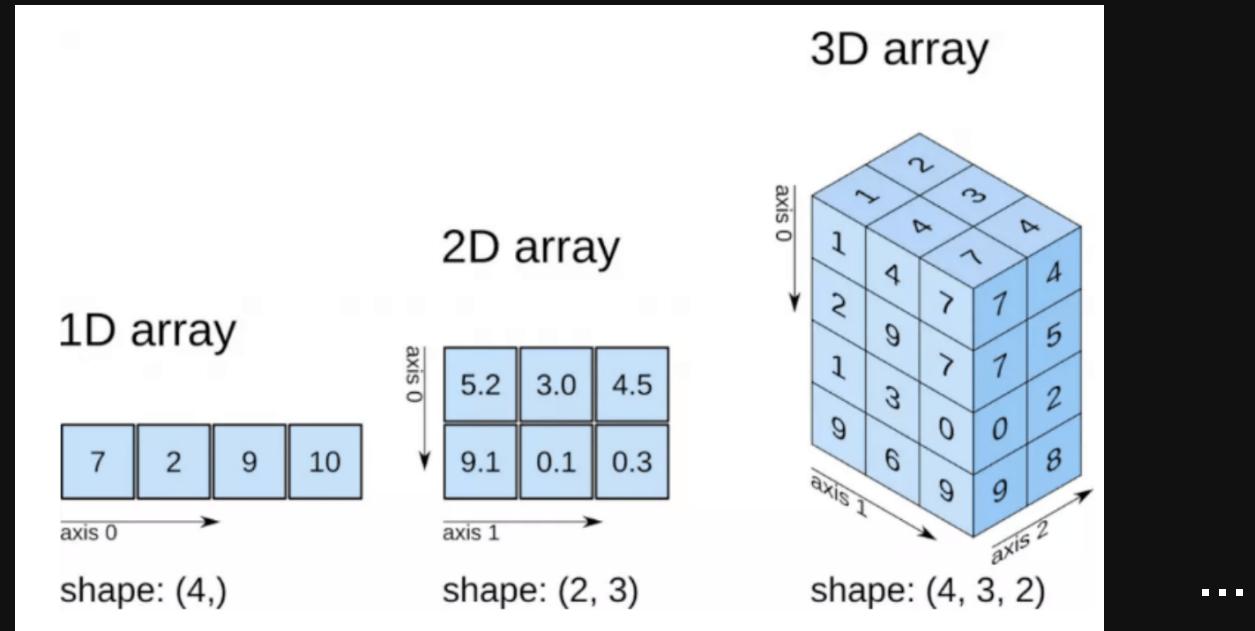
- 我们可以使用NumPy.array()在Python中创建一个N维数组。
- 数组默认是**同构的** (Homogeneous)，这意味着数组内的数据必须是相同的数据类型。（注意：你也可以创建一个结构化数组）
- 支持**逐元素** (Element-wise) 操作。
- NumPy 数组具有各种函数、方法和变量，以简化矩阵计算任务。
- 数组的元素**在内存中是连续存储的**。例如，二维数组的所有行必须具有相同数量的列。三维数组必须在每个轴上具有相同数量的行和列。





Numpy 的理解与解读

- NumPy 是 Python 语言的一个扩展程序库。支持高阶大量的维度数组与矩阵运算，此外也针对数组运算提供大量的数学函数库。
 - Numpy 官方网址: <https://numpy.org>
 - Numpy 中文文档: <https://www.numpy.org.cn/>
- Numpy 中的**数据结构**: **ndarray** (即 n-dimensional array, 多维数组)





Numpy 的理解与解读

- NumPy 是 Python 语言的一个扩展程序库。支持高阶大量的维度数组与矩阵运算，此外也针对数组运算提供大量的数学函数库。
 - Numpy 官方网址: <https://numpy.org>
 - Numpy 中文文档: <https://www.numpy.org.cn/>
- Numpy 中的**数据结构**: **ndarray** (即 n-dimensional array, 多维数组)
- **ndarray** 的关键属性有维度 (ndim)、形状 (shape) 和数值类型 (dtype) 等



<https://numpy.org>

```

1 # 引入 numpy 模块
2 import numpy as np
3 np1 = np.array([1, 2, 3])
4 np2 = np.array([3, 4, 5])
5
6 # array相加
7 print(np1 + np2) # [4 6 8]
8
9 # 显示相关信息
10 print(np1.ndim, np1.shape, np1.dtype) # 1 (3,) int64 => 一维array, 三个元素, 数值类型
11
12 np3 = np.array([1, 2, 3, 4, 5, 6])
13
14 # 改变 array 的维度
15 np3 = np3.reshape([2, 3])
16 print(np3.ndim, np3.shape, np3.dtype) # 2 (2, 3) int64

```



Numpy 的理解与解读

- NumPy 是 Python 语言的一个扩展程序库。支持高阶大量的维度数组与矩阵运算，此外也针对数组运算提供大量的数学函数库。
 - Numpy 官方网址: <https://numpy.org>
 - Numpy 中文文档: <https://www.numpy.org.cn/>
- Numpy 中的**数据结构**: **ndarray** (即 n-dimensional array, 多维数组)
- **ndarray** 的关键属性有维度 (ndim)、形状 (shape) 和数值类型 (dtype) 等



<https://numpy.org>

创建ndarray

```
np.array(list) : 从list/tuple创建
np.arange(start,stop,interval) : 创建序列
np.zero([d1,d2,...]) : 创建指定维度的全0矩阵
np.ones([d1,d2,...]) : 创建指定维度的全1矩阵
```

属性操作

```
arr.size : 数组形状, 即 (x,y,z) (返回元组)
arr.dtype : 数据类型
arr.size : 元素个数, 即 x*y*z
arr.ndim : 维度大小, 即 len(x,y,z)
arr.astype(np.int64) : 改变数据类型
arr.reshape(m,n) : 改变形状
```

基本运算

```
+ 加, - 减, * 乘, / 除, // 向下取整除, % 模, ** 乘方
标量和ndarray数组之间的运算 (所有元素进行运算)
两个ndarray数组之间的运算 (对应位置元素进行运算)
```



Numpy 的理解与解读

- NumPy 是 Python 语言的一个扩展程序库。支持高阶大量的维度数组与矩阵运算，此外也针对数组运算提供大量的数学函数库。
 - Numpy 官方网址: <https://numpy.org>
 - Numpy 中文文档: <https://www.numpy.org.cn/>
- Numpy 中的**数据结构**: **ndarray** (即 n-dimensional array, 多维数组)
- **ndarray** 的关键属性有维度 (ndim)、形状 (shape) 和数值类型 (dtype) 等
- **ndarray** 的切片



<https://numpy.org>

```
>>> a[0, 3:5]
array([3, 4])

>>> a[4:, 4:]
array([[44, 55],
       [54, 55]])

>>> a[:, 2]
a([2, 12, 22, 32, 42, 52])

>>> a[2::2, ::2]
array([[20, 22], [40, 42], [40, 42]])
```

0	1	2	3	4	5
10	11	12	13	14	15
20	21	22	23	24	25
30	31	32	33	34	35
40	41	42	43	44	45
50	51	52	53	54	55



Numpy 的理解与解读

- NumPy 是 Python 语言的一个扩展程序库。支持高阶大量的维度数组与矩阵运算，此外也针对数组运算提供大量的数学函数库。
 - Numpy 官方网址: <https://numpy.org>
 - Numpy 中文文档: <https://www.numpy.org.cn/>
- Numpy 中的**数据结构**: **ndarray** (即 n-dimensional array, 多维数组)
- **ndarray** 的关键属性有维度 (ndim)、形状 (shape) 和数值类型 (dtype) 等
- **ndarray** 的切片



<https://numpy.org>

```
>>> a[(0,1,2,3,4), (1,2,3,4,5)]
array([1, 12, 23, 34, 45])

>>> a[3:, [0,2,5]]
array([[30, 32, 35],
       [40, 42, 45],
       [50, 52, 55]])

>>> mask = np.array([1,0,1,0,0,1], dtype=bool)
>>> a[mask, 2]
array([2, 22, 52])
```

0	1	2	3	4	5
10	11	12	13	14	15
20	21	22	23	24	25
30	31	32	33	34	35
40	41	42	43	44	45
50	51	52	53	54	55



Numpy 的理解与解读

- NumPy 是 Python 语言的一个扩展程序库。支持高阶大量的维度数组与矩阵运算，此外也针对数组运算提供大量的数学函数库。
 - Numpy 官方网址: <https://numpy.org>
 - Numpy 中文文档: <https://www.numpy.org.cn/>
- Numpy 中的**数据结构**: **ndarray** (即 n-dimensional array, 多维数组)
- ndarray** 的关键属性有维度 (ndim)、形状 (shape) 和数值类型 (dtype) 等
- ndarray** 的切片
- ndarray** 里的 broadcasting (广播机制)
 - 更多官方说明: <https://numpy.org/doc/stable/user/basics.broadcasting.html>

简单的说，对两个阵进行操作时，Numpy 逐元素地比较他们的形状。

只有两种情况下Numpy会认为两个矩阵(向右对齐后)内的两个对应维度是兼容的：

- 它们相等；
- 其中一个是1维。

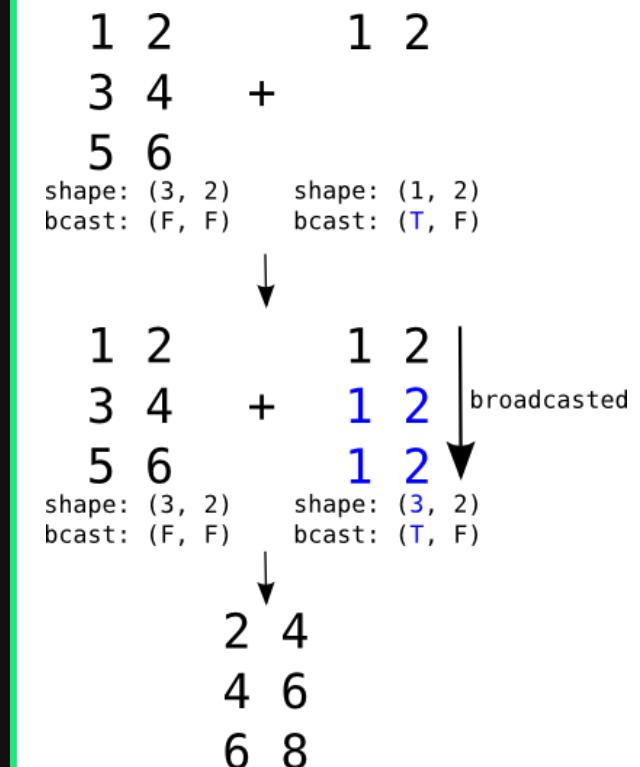
举个牛逼的例子：

```
A      (4d array):  8 x 1 x 6 x 1
B      (3d array):    7 x 1 x 5
Result (4d array):  8 x 7 x 6 x 5
```

当任何一个维度是1，那么另一个不为1的维度将被用作最终结果的维度。也就是说，尺寸为1的维度将延展或“逐个复制”到与另一个维度匹配。



<https://numpy.org>





Numpy 的理解与解读

- NumPy 是 Python 语言的一个扩展程序库。支持高阶大量的维度数组与矩阵运算，此外也针对数组运算提供大量的数学函数库。
 - Numpy 官方网址: <https://numpy.org>
 - Numpy 中文文档: <https://www.numpy.org.cn/>
- Numpy 中的**数据结构**: **ndarray** (即 n-dimensional array, 多维数组)
- **ndarray** 的关键属性有维度 (ndim)、形状 (shape) 和数值类型 (dtype) 等
- **ndarray** 的切片
- **ndarray** 里的 broadcasting (广播机制)
- **ndarray** 线性代数与数理统计

argmax, argmin, argsort, corrcoef, dot, floor, inner, inv, lexsort, max, maximum, mean, median, min, minimum, nonzero, prod, re, round, sort, std, sum, trace, transpose, ...



<https://numpy.org>

