

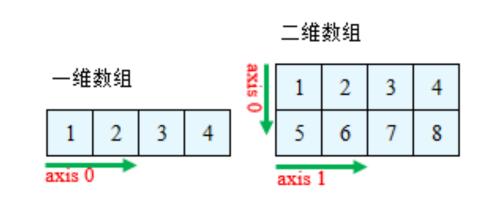
万永权

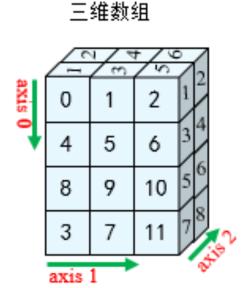
目录 CONTENTS

- 1. 数据的特性
- 2. Numpy

数据维度

- ■维度: 一组数据的组织形式
- ■分类
 - ●一维数据
 - ●二维数据
 - ●多维数据
 - ●高维数据





一维数据

- 由一组有序或无序数据构成,采用线性方式组织
- 对应列表、集合等概念

```
1, 1, 2, 3, 5, 8, 13, 21, 34 ... ...
```

90, 87, 65, 78, 98, 100, 54, 82, 76, 85

一维数据存储

- 存储方式一:空格分隔
 - 使用一个或多个空格分隔进行存储,不换行
 - 缺点:数据中不能存在空格

亚洲 欧洲 北美洲 南美洲 南极洲 非洲 大洋洲

- 存储方式二: 逗号分隔
 - 使用英文半角逗号分隔数据进行存储,不换行
 - 缺点:数据中不能有英文逗号

亚洲,欧洲,北美洲,南美洲,南极洲,非洲,大洋洲

- 存储方式三:其他方式
 - 使用其他符号或符号组合分隔,建议采用特殊符号
 - 缺点:需要根据数据特点定义,通用性较差

亚洲\$欧洲\$北美洲\$南美洲\$南极洲\$非洲\$大洋洲

二维数据

- 由多个一维数据构成,是一维数据的组合形式
- 表格是典型的二维数据
- 其中,表头是二维数据的一部分

序号	学号	姓名	高数	英语	程序设计
1	20191001	张三	98	88	94
2	20191002	李四	87	90	76
3	20191003	王五	79	84	87
4	20191004	赵六	64	68	75

二维数据格式化(1)

- ◆二维列表
- ◆CSV数据存储格式
 - CSV: Comma-Separated Values
 - ➤ 国际通用的一二维数据存储格式,一般.csv扩展名
 - ▶ 每行一个一维数据,采用逗号分隔,无空行

序号	学号	姓名	高数	英语	程序设计
1	20191001	张三	98	88	94
2	20191002	李四	87	90	76
3	20191003	王五	79	84	87
4	20191004	赵六	64	68	75

序号,学号,姓名,高数,英语,程序设计

1,20191001,张三,98,88,94

2,20191002,李四,87,90,76

3,20191003,王五,79,84,87

4,20191004,赵六,64,68,75



多维数据

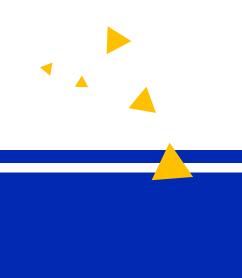
• 由一维或二维数据在新维度上扩展形成

时间维度

2019 \Longrightarrow 2020

序号	学号	姓名	高数	英语	程序设计
1	20191001	张三	98	88	94
2	20191002	李四	87	90	76
3	20191003	王五	79	84	87
4	20191004	赵六	64	68	75

序号	学号	姓名	高数	英语	程序设计
1	20191001	张三	76	90	89
2	20191002	李四	83	88	74
3	20191003	王五	87	90	92
4	20191004	赵六	78	98	87



- ◆NumPy 是 Numerical **Python** 的简称,是高性能计算和数据分析的基础包。
- ◆ NumPy是Python的一个扩充程序库。支持高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运算效率极好,是大量机器学习框架的基础库。

包括:

- 1.一个强大的N维数组对象ndarray;
- 2.比较成熟的(广播)函数库;
- 3.用于整合C/C++和Fortran代码的工具包;
- 4.实用的线性代数、傅里叶变换和随机数生成函数。

- ◆NumPy的强大功能主要基于底层的一个ndarray结构,其可以生成N 维数组对象。
- ◆ndarray数组的属性及其说明如下表。

	属性名称	属性说明
数组维度	ndim	返回int,表示数组的维数
数组形状	shape	返回tuple,表示数组形状的阵列,对于n行m列的矩阵,形状为(n,m)
数组大小	size	返回int,表示数组的元素总数,等于数组形状的乘积
数据类型	dtype	返回data-type,表示数组中元素的数据类型
	itemsize	返回int,表示数组的每个元素的大小(以字节为单位),例如,一个元素 类型为float64的数组的itemsiz属性值为8(float64占用64个bits,每个字节长 度为8,所以64/8,占用8个字节),一个元素类型为complex32的数组的 itemsiz属性值为4,即32/8



NumPy 数组属性

- 1. 常用术语
- 1) 轴(axis):每一个线性数组称为一个轴,轴即数组的维度(dimensions)。 例如将二维数组看作一维数组,此一维数组中每个元素又是一个一维数组。 则每个一维数组是 NumPy 中的一个轴(axis)。第一个轴相当于是底层数组, 第二个轴是底层数组中的数组。
- 2) 秩(rank): 秩描述NumPy数组的维数,即轴的数量。一维数组的秩为1, 二维数组的秩为 2. 以此类推。

常用函数

◆Numpy提供的函数

函数	描述
np.array([x,y,z],dtype=int)	从Python列表和元组创造数组
np.arange(x,y,i)	创建一个由x到y,以i为步长的数组
np.linspace(x,y,n)	创建一个由x到y,等分成n个元素的数组
np.indices((m,n))	创建一个m行n列的矩阵
np.random.rand(m,n)	创建一个m行n列的随机数组
np.ones((m,n),dtype)	创建一个m行n列全1的数组,dtype是数据类型
np.empty((m,n),dtype)	创建一个m行n列全0的数组,dtype是数据类型

【例1】建立一个一维ndarray数组。 import numpy as np

a = np.array([1,2,3])print(a)

举例: 创建2维数组

```
>>> import numpy as np
>>> a = np.ones((4,5))
>>> print(a)
[[1, 1, 1, 1, 1, 1,
[1. 1. 1. 1. 1.]
[1. 1. 1. 1. 1.]
[1. 1. 1. 1. 1.]]
>>> a.ndim
>>> a.shape
(4, 5)
>>> a.dtype
dtype('float64')
```

产生数列的函数

——arange函数、linspace函数、range函数

在进行科学运算时,经常用到基本的简单数列。Python中提供了range()函数。NumPy中也有类似的函数,如arange()、linspace()函数等。

1) range()函数

Python内置的range函数可以创建一维数组,指定开始值、终值和步长。注意数组不包括终值。

函数形式: range(start, stop [,step])

可以生成一个数组,从start开始,到stop-1结束,间隔为step。缺省情况下从0开始。step需要是整数,默认为1。

例如: arr1=range(0,5,1)

2) arrange()函数

NumPy的arange函数功能与range函数类似,在start开始到stop范围内,生成一个 ndarray数组。

函数形式: arange([start,] stop [, step,], dtype=None)

【例3.16】生成3到9之间、步长为0.2的数组。

import numpy as np

arr2=np.arange(3,9,0.2)

arr2

3) linspace()函数

格式: numpy.linspace(start, stop, num=50, endpoint=True, retstep=False, dtype=None)。

其中start为序列的起始值,stop为结束值,num是生成的样本数。

【例3.17】生成1到5之间的10个数。

import numpy as np arr3=np.linspace(1, 5, 10) arr3

5. 使用随机函数创建数组

除了简单的顺序数列,NumPy还在random子模块中提供了随机数函数,常见的随机函数见表。

常用的NumPy随机函数

函数	描述
rand(d0,d1,,dn)	随机产生指定维度的浮点数组
randint(low[,high,size,dtype])	随机产生[low,high]范围内的整数
random([size])	随机产生[0.0, 1.0)之间的浮点数
uniform(start,end,size)	随机产生一组[start,end)范围内的均匀
	分布的浮点数
normal (loc, scale, size)	基于给定的均值和方差,随机产生一
	组正态分布的浮点数

【例3.18】创建随机数组。

#生成2行3列的随机浮点数组 np.random.rand(2,3) #生成2行2列的随机整数数组 np.random.randint(0,10,(2,2)) #生成2行3列、正态分布的随机数组 np.random.uniform(1,2,(2,3))

6.其他数据结构转换成ndarray

NumPy中,可以通过array()函数将数值序列转换为ndarray数组,如List(列表)和Tuple(元组)等。

```
【例3.19】List类型转换成ndarray。
    import numpy as np
    # List
    data = [[2000, 'Ohino', 1.5],
         [2002, 'Ohino', 3.6],
         [2002, 'Nevada', 2.9]]
    print(type(data))
    # List to array
    ndarr = np.array(data)
    print(type(ndarr))
```



通过索引访问数组

NumPy通常以提供高效率的数组著称,这主要归功于索引的易用性。

1. 一维数组的索引

▶一维数组的方法很简单,与Python中的list的索引方法一致。

2. 多维数组的索引

- >多维数组的每一个维度都有一个索引,各个维度的索引之间用逗号隔开。
- >多维数组同样也可以使用整数序列和布尔值索引进行访问。



数组示意图

 $_{nl}$

1

2

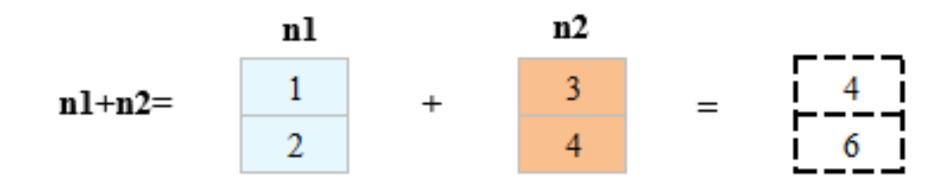
n2

3

-4



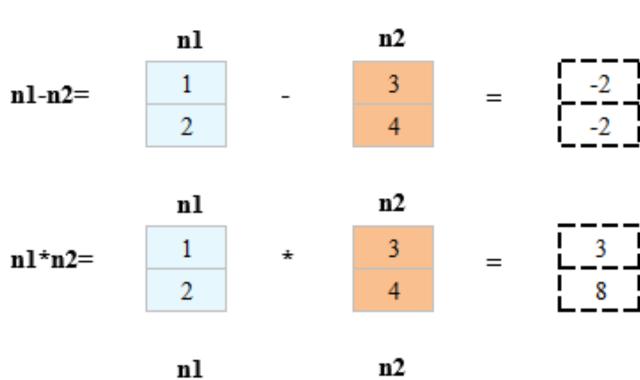
数组加法运算示意图







数组的减法、乘法和除法运算示意图



切片、迭代

切片是指对数据序列对象取一部分的操作,ndarray数组与其他数据序列类似,也可以进行索引、切片和迭代。

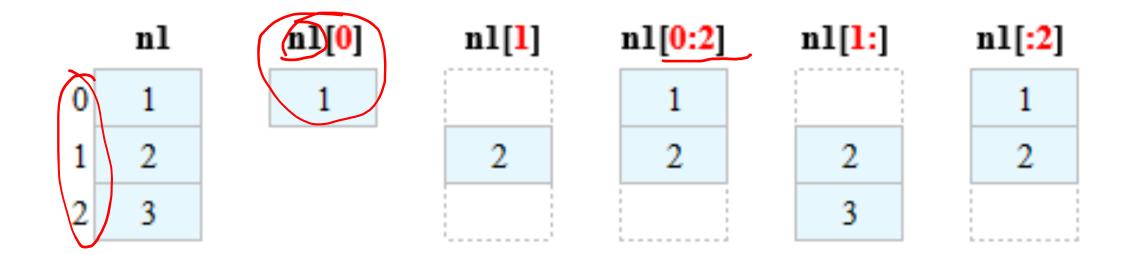
1. 切片 (slice)

对ndarray进行切片操作与一维数组相同,用索引标记切片的起始和终止位置。





切片式索引示意图



二维数组切片操作

因为 ndarray可以是多维数组,在进行切片时,通常需要设定每个维度上的切片位置。

```
【例3.20】二维ndarray的切片。
   import numpy as np
   # 创建一个4行6列的二维数组
   arr = np.arange(24).reshape(4,6)
   print('arr =\n',arr)
   # 截取第2行到最后一行,第1列到第4列构成的ndarray
   arr1 = arr[1:, :3]
   print('B = \n', arr1)
```



【例3.26】简单条件运算。 import numpy as np stus_score = np.array([[80, 88], [82, 81], [84, 75], [86, 83], [75, 81]]) result=[stus_score> 80] print(result)

【例3.27】np.where函数实现数据筛选。
import numpy as np
num = np.random.normal(0, 1, (3,4))
print(num)
num[num<0.5]=0
print(num)
print(np.where(num>0.5,1,0))

2. 统计计算

NumPy提供了丰富的统计函数,常用统计函数如下表3.7所示。

函数	描述
argmax	求最大值的索引
argmin	求最小值的索引
cumsum	从第一元素开始累加各元素
max	求最大值
mean	求算术平均值
min	求最小值
std	求数组元素沿给定轴的标准偏差
Sum	求和

ndarray形态操作方法



方法	描述
ndarray.reshape(n,m)	不改变数组ndarray,返回一个维度为(n,m)的数组
ndarray.resize(new_shape)	与reshape()作用相同,直接修改数组ndarray
ndarray.flatten()	对数组进行降维,返回一个展开后的一维数组

数组重塑示意图

ndarray.reshape(4,5)

20

