扩展数据类型

北京理工大学计算机学院　高玉金

2019年3月

dtype类型对象简介

- dtype不是Python内置的数据类型对象
- dtype在第三方库NumPy库中定义的数据的类型
- NumPy是Numerical Python的简称，是目前Python数值计算中最重要的基础包
- 学习大数据处理必须了解dtype类型对象

```
In [47]: import numpy as np

In [48]: arr = np.arange(10)

In [49]: arr.dtype
Out[49]: dtype('int32')

In [50]: arr
Out[50]: array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

NumPy为什么重要

- 与Python本身的数据结构不同，NumPy在内部将数据存储在连续的内存块上
- NumPy的算法库是C语言写的，无需类型坚持和其他管理操作
- NumPy的数组使用的内存量小于其他Python内建序列
- NumPy可以针对全量数组进行复杂计算

```
tic = time.time()
for _ in range(10):
    np_arr2= np_arr *2
toc = time.time()
print(toc-tic)
```

```
for _ in range(10):
    list_arr2 = [x*2 for x in list_arr]
twc =time.time()
print(twc-toc)
```

dtype对象介绍

- Dtype对象描述与数组对应的内存区域如何使用，包括：
  - 数据的类型（如：整数、浮点数或 Python 对象）
  - 数据的大小（如： 整数使用多少个字节存储）
  - 数据的字节顺序（如：小端法或大端法）
  - 如果是结构化类型，指定每个字段的名称、数据类型和对应的内存块
  - 如果数据类型是子数组，则指定其形状和数据类型
- 字节顺序是通过对数据类型预先设定"<"或">"来决定的
  - “<”表明使用小端法，即低位组放在最前面
  - “>”表明使用大端法，即高位组放在最前面

dtype可以使用的数据类型

- Numpy 支持的数据类型比 Python 内置类型要多很多，基本上可以和 C 语言的数据类型对应，其中部分类型对应为 Python 内置类型
- Python中的float使用8字节或64位存储，在dtype中表示为np.float64
- NumPy提供更精确的类型，如float16，float32，float64和float128，分别表示为f2，f4，f8和f16
- NumPy支持int8/uint8/16/32/64（有符号数和无符号数），分别表示为i1/2/4/8和u1/2/4/8
- 字符串支持String_，表示为大写S
- Unicode_，表示为大写U，如果长度为20，则表示为U20，S20等
- 小写o表示object
- 其他类型请参阅Numpy的数据类型手册

使用dtype()函数构造数据类型

- numpy.dtype(object, align, copy)
  - object ——要转换为的数据类型对象
  - align —— 如果为 true，填充字段使其类似 C 的结构体。
  - copy ——复制 dtype 对象 ，如果为 false，则是对内置数据类型对象的引用
- S20表示20位ASCII码，i1表示int8，f4为32位标准精度浮点

```
In [75]: type('i4')
Out[75]: str

In [76]: type(np.dtype('i4'))
Out[76]: numpy.dtype
```

```
In [58]: student = np.dtype([('name','S20'),
('age',  'i1'),  ('marks',  'f4')])

In [59]: student
Out[59]: dtype([('name', 'S20'), ('age', 'i1'),
('marks', '<f4')])
```

组合类型的使用

```
In [60]: a = np.array([('Diego',  12,  90),('Eugene',  40,  75)], dtype = student)
```

```
array([(b'Diego', 12, 90.), (b'Eugene', 40, 75.)],
      dtype=[('name', 'S20'), ('age', 'i1'), ('marks', '<f4')])
```

```
In [62]: a['name']
    ...:
Out[62]: array([b'Diego', b'Eugene'], dtype='|S20')
```

```
In [65]: a['marks']
Out[65]: array([90., 75.], dtype=float32)
```

注：把类型S20改为U20，重新赋值之后，b消失（bytes）

使用dtype的astype( )方法转换数据类型

- 把多维数组对象里面的数据类型转变成其他类型
- 如从int32到float64

```
In [48]: arr = np.arange(10)

In [49]: arr.dtype
Out[49]: dtype('int32')

In [50]: arr
Out[50]: array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])

In [51]: farr=arr.astype(np.float64)

In [52]: farr.dtype
Out[52]: dtype('float64')

In [53]: arr.dtype
Out[53]: dtype('int32')

In [54]: farr
Out[54]: array([0., 1., 2., 3., 4., 5., 6., 7.,
8., 9.])
```