

# python科学计算

- 1, Numpy提供对数组进行运算和操作的函数集。
- 2, SciPy在Numpy的基础之上添加了许多科学计算函数库。  
例如: 线性代数使用LAPACK库。  
快速傅里叶变换, FFTPACK库。  
常微分方程求解, ODEPACK库。  
非线性方程组求解及其最小值求解, MINPACK库。

SymPy; 数学符号运算扩展库。

## NumPy库

- 1, ndarray(数组)。
  - 2, ufunc对数组进行处理的函数。
- 1, 输入输出数组。
  - 2, 获取或者更改数组形状。
  - 3, 获得数组类型。
  - 4, arange函数创建数组。(具有一定特征)
  - 5, linspace(),指定初始值, 终值, 和元素个数创建数组。通过endpoint参数指定是否包含终值。
  - 6, logspace()。
  - 7,zeros(),ones(),empty()。
  - 8,frombuffer(),fromstring,fromfile()等可以从字节序列或者文件创建数组。
  - 9, fromfunction()第一个参数为计算每个数组元素的函数, 第二个为指定数组形状。
  - 10, slice () 对象

## ufunc运算

- 1, `x=np.linspace(0,2*np.pi,10)`  
`y=np.sin(x)`  
`y`  
注意:这里输出y是创建了一个新数组, 如果想改变输出元素的存储位置, 用out来指定。
- 2, math 库和NumPy各有优缺点。  
`import *`为全部导入  
`import NumPy as np`,导入, 并选择合适的函数。
- 3, 四则运算。  
NumPy 为数组定义了各种数学运算操作符, 因此计算两个数组相加可以简单地写为 `a+b`, 而 `np.add(a,b,a)`则可以用 `a+=b` 表示。表 2-1 列出了数组的运算符及其对应的 ufunc 函数, 注意除号"/"的意义根据是否激活 `__future__.division` 会有所不同。

表 2-1 数组的运算符及其对应的 ufunc 函数

运 算 符	对应的 ufunc 函数
<code>y = x1 + x2</code>	<code>add(x1, x2 [, y])</code>
<code>y = x1 - x2</code>	<code>subtract(x1, x2 [, y])</code>
<code>y = x1 * x2</code>	<code>multiply(x1, x2 [, y])</code>
<code>y = x1 / x2</code>	<code>divide(x1, x2 [, y])</code> , 如果两个数组的元素为整数, 那么用整数除法
<code>y = x1 / x2</code>	<code>true_divide(x1, x2 [, y])</code> , 总是返回精确的商
<code>y = x1 // x2</code>	<code>floor_divide(x1, x2 [, y])</code> , 总是对返回值取整
<code>y = -x</code>	<code>negative(x [, y])</code>
<code>y = x1**x2</code>	<code>power(x1, x2 [, y])</code>
<code>y = x1 % x2</code>	<code>remainder(x1, x2 [, y])</code> ,或 <code>mod(x1, x2 [, y])</code>

## 4, 布尔运算

每个比较运算符都与一个 ufunc 函数对应, 下面是比较运算符与其 ufunc 函数的对照表:

表 2-2 比较运算符及其对应的 ufunc 函数

比较运算符	ufunc 函数
<code>y = x1 == x2</code>	<code>equal(x1, x2 [, y])</code>
<code>y = x1 != x2</code>	<code>not_equal(x1, x2 [, y])</code>
<code>y = x1 &lt; x2</code>	<code>less(x1, x2 [, y])</code>
<code>y = x1 &lt;= x2</code>	<code>less_equal(x1, x2 [, y])</code>
<code>y = x1 &gt; x2</code>	<code>greater(x1, x2 [, y])</code>
<code>y = x1 &gt;= x2</code>	<code>greater_equal(x1, x2 [, y])</code>

- 5, frompyfunc()将一个计算单个元素的函数转换为ufunc()函数。
- 6, 广播。
- 7, ogrid产生二维坐标网格。

- 8, sum求和
  - 9, mean求平均数。
  - 10, std标准差。
  - 11, var方差。
  - 12, min, max最小最大值。
  - 13, ptp最大最小之间的差。
  - 14, median中值。
- 

#### 多项式函数

- 1, poly1d将系数实例化。
  - 2, deriv()微分, integ()积分。
  - 3, roots求解多元函数的根。
  - 4, poly()将根转回系数。
- 

#### 分段函数

- 1, where () 和 piecewise () 计算三角形波。
- 

#### 统计函数

- 1, unique()返回其参数数组中所有不同的值, 并且按照从小到大的顺序排列。
  - 2, histogram()返回两个一维数组, 第一个是每个区间的统计结果, 第二个数组长度为len(list)+1, 每两个相邻数值构成一个统计区间。
- 

#### 线性代数

- 1, matrix对象, 创建的是矩阵。