

NumPy 广播(Broadcast)

广播(Broadcast)是 numpy 对不同形状(shape)的数组进行数值计算的方式，对数组的算术运算通常在相应的元素上进行。

如果两个数组 a 和 b 形状相同，即满足 **a.shape == b.shape**，那么 a*b 的结果就是 a 与 b 数组对应位相乘。这要求维数相同，且各维度的长度相同。

实例

```
import numpy as np
a = np.array([1,2,3,4])
b = np.array([10,20,30,40])
c = a * b
print (c)
```

输出结果为：

```
[ 10  40  90 160]
```

当运算中的 2 个数组的形状不同时，numpy 将自动触发广播机制。如：

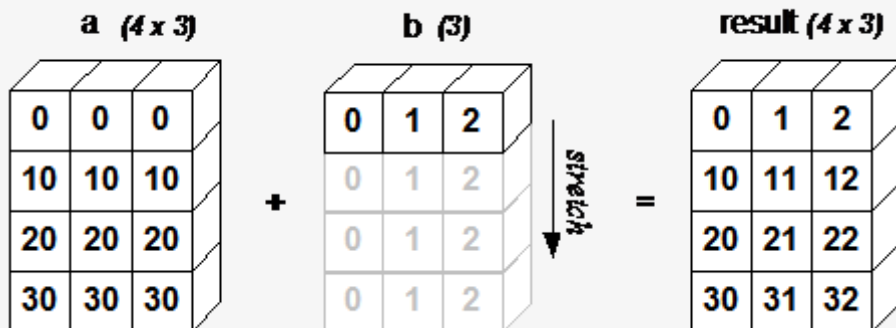
实例

```
import numpy as np
a = np.array([[ 0, 0, 0],
[10,10,10],
[20,20,20],
[30,30,30]])
b = np.array([1,2,3])
print(a + b)
```

输出结果为：

```
[[ 1  2  3]
[11 12 13]
[21 22 23]
[31 32 33]]
```

下面的图片展示了数组 b 如何通过广播来与数组 a 兼容。



4x3 的二维数组与长为 3 的一维数组相加，等效于把数组 b 在二维上重复 4 次再运算：

实例

```
import numpy as np
a = np.array([[ 0, 0, 0],
              [10,10,10],
              [20,20,20],
              [30,30,30]])
b = np.array([1,2,3])
bb = np.tile(b, (4, 1))
print(a + bb)
```

输出结果为：

```
[[ 1  2  3]
 [11 12 13]
 [21 22 23]
 [31 32 33]]
```

广播的规则：

- 让所有输入数组都向其中形状最长的数组看齐，形状中不足的部分都通过在前面加 1 补齐。
- 输出数组的形状是输入数组形状的各个维度上的最大值。
- 如果输入数组的某个维度和输出数组的对应维度的长度相同或者其长度为 1 时，这个数组能够用来计算，否则出错。
- 当输入数组的某个维度的长度为 1 时，沿着此维度运算时都用此维度上的第一组值。

简单理解：对两个数组，分别比较他们的每一个维度（若其中一个数组没有当前维度则忽略），满足：

- 数组拥有相同形状。
- 当前维度的值相等。
- 当前维度的值有一个是 1。

若条件不满足，抛出 "ValueError: frames are not aligned" 异常。

 点我分享笔记