کتابخانه NumPy

معرفی کتابخانه NumPy

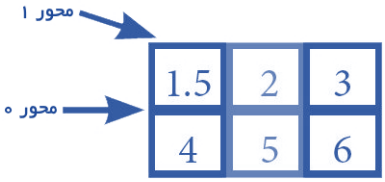
NumPy، کتابخانه پایه‌ای برای محاسبات علمی در زبان برنامه‌نویسی پایتون است. این کتابخانه آرایه‌های چندبُعدی با کارایی بالا و ابزارهای کار با این آرایه‌ها را فراهم می‌کند. از دستور زیر برای وارد کردن (Import) کتابخانه NumPy استفاده می‌شود. (توجه: اندیس آرایه در پایتون از صفر شروع می‌شود.)

```
>>> import numpy as np
```

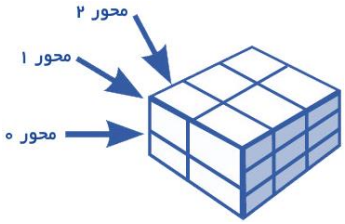
آرایه‌های NumPy

یک‌بُعدی

دوبُعدی



سه‌بُعدی



ساخت آرایه‌ها

```
>>> a = np.array([1,2,3])
>>> b = np.array([(1.5,2,3), (4,5,6)], dtype = float)
>>> c = np.array([[(1.5,2,3), (4,5,6)], [(3,2,1), (4,5,6)]], dtype = float)
```

متغیرهای فرانحوی اولیه (Initial Placeholders)

ساخت آرایه‌ای با ابعاد ۳×۴ از صفرها	`>>> np.zeros((3,4))`
ساخت آرایه‌ای با ابعاد ۲×۳×۴ از یک‌ها	`>>> np.ones((2,3,4),dtype=np.int16)`
ساخت آرایه‌ای از مقادیر با فاصله‌های یکسان (بر اساس گام)	`>>> d = np.arange(10,25,5)`
ساخت آرایه‌ای از مقادیر با فاصله‌های یکسان (بر اساس تعداد نمونه‌ها)	`>>> np.linspace(0,2,9)`
ساخت یک آرایه ثابت با ابعاد ۲×۲ و درایه‌های برابر با ۷	`>>> e = np.full((2,2),7)`
ساخت ماتریس همانی با ابعاد ۲×۲	`>>> f = np.eye(2)`
ساخت آرایه با ابعاد ۲×۲ و مقادیر تصادفی	`>>> np.random.random((2,2))`
ساخت یک آرایه خالی با ابعاد ۳×۲	`>>> np.empty((3,2))`

ورودی/خروجی

ذخیره‌سازی و بارگذاری آرایه روی حافظه	ذخیره‌سازی و بارگذاری فایل‌های متنی
>>> np.save('my_array', a)	>>> np.loadtxt("myfile.txt")
>>> np.savez('array.npz', a, b)	>>> np.genfromtxt("my_file.csv", delimiter=',')
>>> np.load('my_array.npy')	>>> np.savetxt("myarray.txt", a, delimiter=" ")

انواع داده

عدد صحیح ۶۴ بیتی علامت‌دار	>>> np.int64
ممیز شناور استاندارد با دقت دو برابر (Double-Precision)	>>> np.float32
اعداد مختلط با ۱۲۸ ممیز شناور	>>> np.complex
نوع دودویی ذخیره‌کننده مقادیر TRUE و FALSE	>>> np.bool
نوع شی پایتون	>>> np.object
رشته با طول ثابت	>>> np.string_
رشته یونیکد با طول ثابت	>>> np.unicode_

بازرسی آرایه

شکل آرایه a	>>> a.shape
طول آرایه a	>>> len(a)
ابعاد آرایه b	>>> b.ndim
تعداد عناصر آرایه e	>>> e.size
نوع داده عناصر آرایه b	>>> b.dtype
نام نوع داده آرایه b	>>> b.dtype.name
تبدیل نوع داده آرایه به یک نوع دیگر (در این مثال int)	>>> b.astype(int)

درخواست راهنما

```
>>> np.info(np.ndarray.dtype)
```

محاسبات روی آرایه‌ها

عملیات ریاضی

تفریق	>>> g = a - b array([[-0.5, 0. , 0.], [-3. , -3. , -3.]])
تفریق	>>> np.subtract(a,b)
جمع	>>> b + a array([[2.5, 4. , 6.], [5. , 7. , 9.]])
جمع	>>> np.add(b,a)

تقسیم	>>> a / b array([[0.66666667, 1. , 1.], [0.25 , 0.4 , 0.5]])
تقسیم	>>> np.divide(a,b)
ضرب	>>> a * b array([[1.5, 4. , 9.], [4. , 10. , 18.]])
ضرب	>>> np.multiply(a,b)
توان	>>> np.exp(b)
ریشه دوم	>>> np.sqrt(b)
پرینت کردن سینوس یک آرایه	>>> np.sin(a)
کسینوس مولفه‌ای	>>> np.cos(b)
لگاریتم طبیعی مولفه‌ای	>>> np.log(a)
ضرب داخلی	>>> e.dot(f) array([[7., 7.], [7., 7.]])
مقایسه	
مقایسه مولفه‌ای	>>> a == b array([[False, True, True], [False, False, False]], dtype=bool)
مقایسه مولفه‌ای	>>> a < 2 array([True, False, False], dtype=bool)
مقایسه آرایه‌ای	>>> np.array_equal(a, b)
توابع تجمیعی	
مجموع آرایه a	>>> a.sum()
مقدار کمینه آرایه‌ای	>>> a.min()
مقدار بیشینه یک سطر آرایه (در اینجا آرایه b)	>>> b.max(axis=0)
مجموع تجمعی عناصر	>>> b.cumsum(axis=1)
میانگین	>>> a.mean()
میانه	>>> b.median()
ضریب همبستگی	>>> a.corrcoef()
انحراف معیار (در اینجا آرایه b)	>>> np.std(b)
کپی کردن آرایه‌ها	
ساخت نمایشی از آرایه با داده‌های مشابه	>>> h = a.view()
ساخت یک کپی از آرایه	>>> np.copy(a)
ساخت یک کپی عمیق از آرایه	>>> h = a.copy()

مرتب‌سازی آرایه‌ها

مرتب‌سازی یک آرایه	>>> a.sort()
مرتب‌سازی عناصر یک محور آرایه	>>> c.sort(axis=0)

زیرمجموعه‌سازی، برش‌زنی و اندیس‌گذاری

زیرمجموعه‌سازی		
	>>> a[2] 3	انتخاب عنصر در اندیس دوم
	>>> b[1,2] 6.0	انتخاب عنصر در سطر ۱ و ستون ۲ (برابر با b[1][2])
برش‌زنی		
	>>> a[0:2] array([1, 2])	انتخاب درایه در اندیس ۰ و ۱
	>>> b[0:2,1] array([2., 5.])	انتخاب درایه در سطر ۰ و ۱ در ستون ۱
	>>> b[:1] array([[1.5, 2., 3.]])	انتخاب همه درایه‌های سطر ۰ (برابر با b[0:1, :]) مشابه [:,:,۱]
	>>> c[1,...] array([[[3., 2., 1.], [4., 5., 6.]]])	
	>>> a[: :-1] array([3, 2, 1])	ایجاد یک آرایه با ترتیب معکوس شده آرایه a
اندیس‌گذاری دودویی		
	>>> a[a<2] array([1])	انتخاب عناصر آرایه a با مقادیر کمتر از ۲
اندیس‌گذاری Fancy		
	>>> b[[1, 0, 1, 0],[0, 1, 2, 0]] array([4. , 2. , 6. , 1.5])	انتخاب عناصر بر اساس اندیس سطر و ستون
	>>> b[[1, 0, 1, 0]][:,[0,1,2,0]] array([[4. ,5. , 6. , 4.], [1.5 ,2. , 3. , 1.5], [4. ,5. , 6. , 4.], [1.5, 2., 3., 1.5]])	انتخاب زیر مجموعه‌ای از سطرها و ستون‌های ماتریس

تغییرات روی آرایه‌ها

ترانهاده آرایه	
>>> i = np.transpose(b)	ساخت ترانهاده یک آرایه
>>> b.T	ساخت ترانهاده یک آرایه
تغییر شکل آرایه	
>>> b.ravel()	مسطح کردن آرایه
>>> g.reshape(3,-2)	شکل دهی مجدد آرایه، بدون تغییر داده‌ها
اضافه/حذف عناصر	
>>> h.resize((2,6))	تغییر ابعاد آرایه h به (۶، ۲)
>>> np.append(h,g)	الحاق درایه به انتهای آرایه
>>> np.insert(a, 1, 5)	درج درایه در آرایه
>>> np.delete(a,[1])	حذف درایه‌های یک آرایه
ترکیب آرایه‌ها	
>>> np.concatenate((a,d),axis=0) array([1, 2, 3, 10, 15, 20])	الحاق آرایه‌ها بر اساس محور
>>> np.vstack((a,b)) array([[1. , 2. , 3.], [1.5, 2. , 3.], [4. , 5. , 6.]])	ادغام آرایه‌ها به صورت عمودی
>>> np.r_[e,f]	ادغام آرایه‌ها به صورت عمودی (بر اساس سطر)
>>> np.hstack((e,f)) array([[7., 7., 1., 0.], [7., 7., 0., 1.]])	ادغام آرایه‌ها به صورت افقی (بر اساس ستون)
>>> np.column_stack((a,d)) array([[1, 10], [2, 15], [3, 20]])	ساخت آرایه‌های ادغام شده بر اساس ستون
>>> np.c_[a,d]	ساخت آرایه‌های ادغام شده بر اساس ستون

تفکیک آرایه‌ها	
>>> np.hsplit(a,3) [array([1]),array([2]),array([3])]	تفکیک آرایه به صورت افقی از اندیس سوم
>>> np.vsplit(c,2) [array([[[1.5, 2. , 1.], [4. , 5. , 6.]]]), array([[[3., 2., 3.], [4., 5., 6.]]])]	تفکیک آرایه به صورت عمودی در اندیس دوم

مجموعه آموزش‌های داده‌کاوی فرادرس (+کلیک کنید)

برای مشاهده دیگر «تقلب‌نامه‌های» مجله فرادرس، به این لینک مراجعه فرمایید.

جهت آگاهی از آخرین تقلب‌نامه‌های منتشر شده، در کانال تلگرام مجله فرادرس عضو شوید.

تهیه و تنظیم: مجله فرادرس

منبع