arange در نامپای رو توضیح میدی

[In reply to KHz]

بله! در کتابخانه NumPy، تابع arange برای ایجاد آرایه‌های عددی با فواصل مشخص استفاده می‌شود. این تابع مشابه range در پایتون است، اما به جای لیست، یک آرایه‌ی NumPy تولید می‌کند.

سینتکس:

numpy.arange([start, ]stop, [step, ]dtype=None)

پارامترها:

start (اختیاری): مقدار شروع (پیش‌فرض 0 است).

stop: مقدار پایانی (شامل نمی‌شود).

step (اختیاری): فاصله بین اعداد (پیش‌فرض 1 است).

dtype (اختیاری): نوع داده‌ای آرایه.

مثال‌ها:

۱. ساده‌ترین حالت:

import numpy as np

arr = np.arange(5)

print(arr)

خروجی:

[0 1 2 3 4]

(اعداد از 0 تا 4 ایجاد شده‌اند.)

۲. مشخص کردن مقدار شروع و پایان:

arr = np.arange(2, 10)

print(arr)

خروجی:

[2 3 4 5 6 7 8 9]

(از 2 شروع شده و تا 10 (شامل نمی‌شود) ادامه دارد.)

۳. تعیین مقدار گام (step):

arr = np.arange(1, 10, 2)

print(arr)

خروجی:

[1 3 5 7 9]

(اعداد با گام 2 افزایش می‌یابند.)

۴. استفاده از اعداد اعشاری:

arr = np.arange(0, 1, 0.2)

print(arr)

خروجی:

[0. 0.2 0.4 0.6 0.8]

(اعداد اعشاری با گام 0.2 ایجاد شده‌اند.)

نکات مهم:

اگر step عدد اعشاری باشد، بهتر است از np.linspace برای دقت بالاتر استفاده کنید.

مقدار stop در خروجی قرار نمی‌گیرد.

اگر سوالی داری بگو!

random.randint رو توضیح میدی

[In reply to KHz]

در NumPy، برای تولید اعداد تصادفی از تابع numpy.random.randint استفاده می‌شود. این تابع اعداد صحیح تصادفی در یک بازه‌ی مشخص ایجاد می‌کند.

سینتکس:

numpy.random.randint(low, high=None, size=None, dtype=int)

پارامترها:

low: مقدار حداقل (شامل می‌شود).

high (اختیاری): مقدار حداکثر (شامل نمی‌شود). اگر مقدار high مشخص نشود، low به عنوان حداکثر در نظر گرفته شده و مقدار حداقل 0 خواهد بود.

size (اختیاری): تعداد اعداد تصادفی که می‌خواهیم تولید کنیم. می‌تواند یک مقدار عددی یا یک tuple برای تولید آرایه‌های چندبعدی باشد.

dtype (اختیاری): نوع داده‌ای خروجی (معمولاً int است).

مثال‌ها:

۱. تولید یک عدد صحیح تصادفی بین دو مقدار مشخص

import numpy as np rand\_num = np.random.randint(1, 10) # تولید عددی بین 1 و 9 print(rand\_num)

خروجی (مثال):

5

۲. تولید یک آرایه‌ی یک‌بعدی از اعداد تصادفی

arr = np.random.randint(1, 10, size=5) # آرایه‌ای با 5 عدد تصادفی print(arr)

خروجی (مثال):

[3 7 2 5 9]

۳. تولید یک آرایه‌ی دوبعدی تصادفی

arr\_2d = np.random.randint(1, 100, size=(3, 4)) # آرایه ۳×۴ از اعداد تصادفی بین 1 تا 99 print(arr\_2d)

خروجی (مثال):

[[23 56 12 78] [45 34 67 89] [90 4 32 76]]

۴. استفاده از dtype برای تعیین نوع داده

arr = np.random.randint(0, 256, size=(2, 3), dtype=np.uint8) # مقادیر بین 0 تا 255 با نوع داده‌ای uint8 print(arr)

خروجی (مثال):

[[123 45 67] [210 89 34]]

(این روش برای تولید تصاویر سیاه و سفید در OpenCV کاربرد دارد.)

نکات مهم:

مقدار high در خروجی شامل نمی‌شود.

اگر فقط یک مقدار (low) بدهید، مقدار high برابر low در نظر گرفته می‌شود و حداقل مقدار 0 خواهد بود.

می‌توان آرایه‌های چندبعدی با استفاده از size=(rows, cols, ...) تولید کرد.

اگر سوالی داشتی، بپرس!

ndim در نامپای رو توضیح میدی

در NumPy، ویژگی ndim برای مشخص کردن تعداد بعدهای (Dimensions) یک آرایه استفاده می‌شود. این ویژگی تعداد محورهای (Axes) آرایه را برمی‌گرداند.

---

سینتکس:

array.ndim

> مقدار ndim نشان می‌دهد که آرایه چندبعدی است.

---

مثال‌ها:

۱. آرایه یک‌بعدی (بردار)

import numpy as np

arr1 = np.array([1, 2, 3, 4, 5])

print(arr1.ndim) # خروجی: 1

✅ چون آرایه فقط یک لیست ساده از اعداد است، ndim مقدار 1 را برمی‌گرداند.

---

۲. آرایه دوبعدی (ماتریس)

arr2 = np.array([[1, 2, 3], [4, 5, 6]])

print(arr2.ndim) # خروجی: 2

✅ این آرایه دارای ۲ محور (سطر و ستون) است، بنابراین مقدار ndim برابر 2 است.

---

۳. آرایه سه‌بعدی (تنسور)

arr3 = np.array([[[1, 2], [3, 4]], [[5, 6], [7, 8]]])

print(arr3.ndim) # خروجی: 3

✅ این آرایه دارای ۳ محور (ارتفاع، سطر، ستون) است.

---

بررسی چند نمونه دیگر

arr4 = np.array(42) # یک مقدار عددی (اسکالر)

print(arr4.ndim) # خروجی: 0

arr5 = np.array([[[[1, 2], [3, 4]]]]) # آرایه چهار بعدی

print(arr5.ndim) # خروجی: 4

✅ مقدار ndim برای arr4 برابر 0 است، چون فقط یک مقدار عددی دارد (اسکالر).

✅ arr5 یک آرایه چهاربعدی است.

---

جمع‌بندی:

ndim = 0 → مقدار اسکالر (یک عدد)

ndim = 1 → آرایه یک‌بعدی (بردار)

ndim = 2 → آرایه دوبعدی (ماتریس)

ndim = 3 → آرایه سه‌بعدی (تنسور)

و غیره...

اگر سوالی داشتی، بپرس!