

- [首页](#)

- [技术问答](#) 当前位置: [首页](#) > [代码示例](#) > [用法及示例精选](#) > 正文

- [技术教程](#)

- [系统&架构](#)

- [算法&结构](#)

- [编程示例](#)

- [实用工具](#)

- [全部](#)

Python Pytorch arange()用法及代码示例

PyTorch是由Facebook开发的开源机器学习库。它用于深度神经网络和自然语言处理。

函数`torch.arange()`返回大小的一维张量 $\left[\frac{\text{end} - \text{start}}{\text{step}} \right]$

从间隔的值 $(start, end)$ 从开始就采取共同的差异步骤。

$$out_{i+1} = out_i + step$$

用法: `torch.arange(start=0, end, step=1, out=None)`

参数:

start: 点集的起始值。默认值: 0。

end: 点集的最终值

step: 每对相邻点之间的间隙。默认值: 1。

out(Tensor, optional): 输出张量

返回类型: 张量

代码1:

```
# Importing the PyTorch library
import torch

# Applying the arange function and
# storing the resulting tensor in 't'
a = torch.arange(3)
print("a = ", a)

b = torch.arange(1, 6)
print("b = ", b)

c = torch.arange(1, 5, 0.5)
print("c = ", c)
```

输出:

```
a = tensor([0, 1, 2])
b = tensor([1, 2, 3, 4, 5])
c = tensor([1.0000, 1.5000, 2.0000, 2.5000, 3.0000, 3.5000, 4.0000, 4.5000])
```

请注意, 与末尾比较时, 非整数步骤易受浮点舍入误差的影响; 为避免不一致, 在这种情况下, 建议在末端增加一个小的epsilon。

相关用法