**Tensor.clone()**

返回tensor的拷贝，返回的新张量和原张量具有同样的大小和数据类型，但存储在不同位置，修改一个不对对另一个产生影响。 若原tensor的requires\_grad=True，则clone()返回的tensor的梯度会流向原tensor，即返回的tensor的梯度会叠加在原tensor上

a = torch.tensor([1.0, 2.0, 3.0],requires\_grad=True)

print(a.grad)

out1 = a.sum()

out1.backward()

print(a.grad)

out2 = out1.clone()*#添加clone()，c的requires\_grad为True*

out2.backward()

print(a.grad)

>>>

None

tensor([1., 1., 1.])

tensor([2., 2., 2.])