

矩阵求导

计算机视觉第四次作业 | 2101212840 游盈萱

矩阵求导

问题描述

推导过程

1. 推导 $\frac{\partial f}{\partial W}$, f 对 W 求偏导
2. 推导 $\frac{\partial f}{\partial X}$, f 对 X 求偏导
3. 推导 $\frac{\partial f}{\partial Y}$, f 对 Y 求偏导

程序验证

1. 验证 $\frac{\partial f}{\partial W}$
2. 验证 $\frac{\partial f}{\partial X}$
3. 验证 $\frac{\partial f}{\partial Y}$

问题描述

目标函数: $f = ||\max(XW, 0) - Y||_F^2$

手动写出以下表达式, 并用PyTorch进行验证:

$$\frac{\partial f}{\partial W} \quad \frac{\partial f}{\partial X} \quad \frac{\partial f}{\partial Y}$$

推导过程

$$\begin{aligned} f &= ||XW - Y||_F^2 \\ &= \text{tr}[(XW - Y)^T(XW - Y)] \end{aligned}$$

对方程两边求微分,

$$\begin{aligned} df &= d\{\text{tr}[(XW - Y)^T(XW - Y)]\} \\ &= \text{tr}\{d[(XW - Y)^T(XW - Y)]\} \end{aligned}$$

1. 推导 $\frac{\partial f}{\partial W}$, f 对 W 求偏导

$$df = \text{tr}\{d[(XW - Y)^T(XW - Y)]\}$$

f 对 W 求偏导,

$$\begin{aligned} df &= \text{tr}[(XdW)^T(XW - Y) + (XW - Y)^T(XdW)] \\ &= \text{tr}[2(XW - Y)^T XdW] \end{aligned}$$

得到 $\frac{\partial f}{\partial W}$,

$$\Rightarrow \frac{\partial f}{\partial W} = 2X^T(XW - Y)$$

2. 推导 $\frac{\partial f}{\partial X}$, f 对 X 求偏导

$$df = \text{tr}\{d[(XW - Y)^T(XW - Y)]\}$$

f 对 X 求偏导,

$$\begin{aligned} df &= \text{tr}[d(XW)^T(XW - Y) + (XW - Y)^T d(XW)] \\ &= \text{tr}[2(XW - Y)^T d(XW)] \\ &= \text{tr}[2W(XW - Y)^T dW] \end{aligned}$$

得到 $\frac{\partial f}{\partial X}$,

$$\Rightarrow \frac{\partial f}{\partial X} = 2(XW - Y)W^T$$

3. 推导 $\frac{\partial f}{\partial Y}$, f 对 Y 求偏导

$$df = \text{tr}\{d[(XW - Y)^T(XW - Y)]\}$$

f 对 Y 求偏导,

$$\begin{aligned} df &= \text{tr}[-d(Y)^T(XW - Y) - (XW - Y)^T dY] \\ &= \text{tr}[2(Y - XW)^T dY] \end{aligned}$$

得到 $\frac{\partial f}{\partial Y}$,

$$\Rightarrow \frac{\partial f}{\partial Y} = 2(Y - XW)$$

程序验证

```
1 import torch
2 torch.manual_seed(0)
3
4 x = torch.randn(10, 4, requires_grad = True)
5 w = torch.randn(4, 4, requires_grad = True)
6 y = torch.randn(10, 4, requires_grad = True)
7
8 x, w, y
```

```
(tensor([[ -1.1258, -1.1524, -0.2506, -0.4339],
         [ 0.8487, 0.6920, -0.3160, -2.1152],
         [ 0.3223, -1.2633, 0.3500, 0.3081],
         [ 0.1198, 1.2377, 1.1168, -0.2473],
         [-1.3527, -1.6959, 0.5667, 0.7935],
         [ 0.5988, -1.5551, -0.3414, 1.8530],
         [-0.2159, -0.7425, 0.5627, 0.2596],
         [-0.1740, -0.6787, 0.9383, 0.4889],
         [ 1.2032, 0.0845, -1.2001, -0.0048],
         [-0.5181, -0.3067, -1.5810, 1.7066]], requires_grad = True),
 tensor([[ 0.2055, -0.4503, -0.5731, -0.5554],
         [ 0.5943, 1.5419, 0.5073, -0.5910],
         [-1.3253, 0.1886, -0.0691, -0.4949],
         [-1.4959, -0.1938, 0.4455, 1.3253]], requires_grad = True),
 tensor([[ 1.5091, 2.0820, 1.7067, 2.3804],
         [-1.1256, -0.3170, -1.0925, -0.0852],
         [ 0.3276, -0.7607, -1.5991, 0.0185],
         [-0.7504, 0.1854, 0.6211, 0.6382],
         [-0.0033, -0.5344, 1.1687, 0.3945],
         [ 1.9415, 0.7915, -0.0203, -0.4372],
         [-0.2188, -2.4351, -0.0729, -0.0340],
         [ 0.9625, 0.3492, -0.9215, -0.0562],
         [-0.6227, -0.4637, 1.9218, -0.4025],
         [ 0.1239, 1.1648, 0.9234, 1.3873]], requires_grad = True))
```

```

1 M0 = torch.zeros(10, 4)
2 w.grad.zero_()
3 x.grad.zero_()
4 y.grad.zero_()
5 f = torch.norm(torch.mm(x, w) - y) ** 2
6 f.backward()
7 w.grad, x.grad, y.grad

```

```

(tensor([[22.4769, 10.3214, 0.5826, -5.9407],
         [50.8333, 35.6112, 3.9501, -30.7748],
         [-14.4824, 11.2645, 7.2040, -7.0126],
         [-53.4309, -28.8103, 0.7109, 42.7320]]),
 tensor([[6.1743, -11.9859, 4.3394, -0.0597],
         [4.7394, 14.5691, -10.1215, -25.4544],
         [-1.4891, -6.4197, 3.7353, 9.1782],
         [1.2874, 8.9058, 1.9930, -8.3078],
         [-1.2455, -11.9822, 5.9854, 15.0334],
         [-2.3351, -22.5698, 8.3787, 26.0864],
         [-2.4884, 1.9020, 3.6204, 5.2850],
         [-2.0456, -7.8526, 7.6394, 13.2748],
         [3.7401, -0.3867, -6.6651, -8.9618],
         [-0.9880, -9.7293, -0.5763, 9.1050]]),
 tensor([[2.8885, 6.6300, 3.6443, 3.0501],
         [-10.5886, -2.7045, -0.0733, 6.8840],
         [3.8740, 2.6523, -1.7731, -1.5687],
         [-0.8009, -3.8551, 0.4984, 4.6333],
         [6.4413, 3.0368, 1.8791, -4.2604],
         [10.1243, 7.7652, 0.5255, -7.2968],
         [2.8020, -2.8862, 0.2066, -1.3166],
         [6.7530, 2.4705, -1.6596, -1.4750],
         [-5.0359, 0.3463, 4.9753, -0.5440],
         [1.7406, 4.0666, -0.1749, -4.2519]]))

```

1. 验证 $\frac{\partial f}{\partial W}$

```

1 w.grad == 2 * torch.mm(x.t(), (torch.mm(x, w) - y))

```

```

(tensor([[True, True, True, True],
         [True, True, True, True],
         [True, True, True, True],
         [True, True, True, True]]))

```

说明 $\frac{\partial f}{\partial W} = 2X^T(XW - Y)$ 推导正确

2. 验证 $\frac{\partial f}{\partial X}$

```
1 | x.grad == 2 * torch.mm((torch.mm(X, W) - Y), W.t())
```

```
tensor([[True, True, True, True],
        [True, True, True, True],
        [True, True, True, True],
        [True, True, True, True],
        [True, True, True, True],
        [True, True, True, True],
        [True, True, True, True],
        [True, True, True, True],
        [True, True, True, True]])
```

说明 $\frac{\partial f}{\partial X} = 2(XW - Y)W^T$ 推导正确

3. 验证 $\frac{\partial f}{\partial Y}$

```
1 | Y.grad == 2 * (Y - torch.mm(X, W))
```

```
tensor([[True, True, True, True],
        [True, True, True, True],
        [True, True, True, True],
        [True, True, True, True],
        [True, True, True, True],
        [True, True, True, True],
        [True, True, True, True],
        [True, True, True, True],
        [True, True, True, True],
        [True, True, True, True]])
```

说明 $\frac{\partial f}{\partial Y} = 2(Y - XW)$ 推导正确