



TensorFlow

数学运算

主讲人：龙良曲

Outline

- $+ - * /$
 - $**$, pow, square
 - sqrt
 - $//$, %
 - exp, log
 - @, matmul
 - linear layer
-

Operation type

- element-wise

- $+ - * /$

对应元素的加减乘除

- matrix-wise

- $@$, matmul

$[b, 3, 5] @ [b, 5, 4] \rightarrow [b, 3, 4]$

- dim-wise

- $\text{reduce_mean}/\text{max}/\text{min}/\text{sum}$

+ - * / % //

```
In [134]: b=tf.fill([2,2],2.)
```

```
In [135]: a=tf.ones([2,2])
```

```
In [136]: a+b,a-b,a*b,a/b
```

```
(<tf.Tensor: id=462, shape=(2, 2), dtype=float32, numpy=
  array([[3., 3.],
         [3., 3.]], dtype=float32)>,
 <tf.Tensor: id=463, shape=(2, 2), dtype=float32, numpy=
  array([[ -1.,  -1.],
         [ -1.,  -1.]], dtype=float32)>,
 <tf.Tensor: id=464, shape=(2, 2), dtype=float32, numpy=
  array([[2., 2.],
         [2., 2.]], dtype=float32)>,
 <tf.Tensor: id=465, shape=(2, 2), dtype=float32, numpy=
  array([[0.5, 0.5],
         [0.5, 0.5]], dtype=float32)>)
```

```
In [137]: b//a,b%a
```

```
(<tf.Tensor: id=470, shape=(2, 2), dtype=float32, numpy=
  array([[2., 2.],
         [2., 2.]], dtype=float32)>,
 <tf.Tensor: id=471, shape=(2, 2), dtype=float32, numpy=
  array([[0., 0.],
         [0., 0.]], dtype=float32)>)
```

tf.math.log tf.exp

(loge)

```
● ● ●  
In [138]: a  
<tf.Tensor: id=461, shape=(2, 2), dtype=float32, numpy=  
array([[1., 1.],  
       [1., 1.]], dtype=float32)>  
  
In [140]: tf.math.log(a)  
<tf.Tensor: id=475, shape=(2, 2), dtype=float32, numpy=  
array([[0., 0.],  
       [0., 0.]], dtype=float32)>  
  
In [141]: tf.exp(a)  
<tf.Tensor: id=477, shape=(2, 2), dtype=float32, numpy=  
array([[2.7182817, 2.7182817],  
       [2.7182817, 2.7182817]], dtype=float32)>
```

log2, log10?

换底公式实现



```
In [22]: tf.math.log(8.)/tf.math.log(2.)  
Out[22]: <tf.Tensor: id=54, shape=(), dtype=float32,  
numpy=3.0>  
In [23]: tf.math.log(100.)/tf.math.log(10.)  
Out[23]: <tf.Tensor: id=60, shape=(), dtype=float32,  
numpy=2.0>
```

pow, sqrt

```
In [142]: b
<tf.Tensor: id=458, shape=(2, 2), dtype=float32, numpy=
array([[2., 2.],
       [2., 2.]], dtype=float32)>

In [143]: tf.pow(b, 3)
<tf.Tensor: id=481, shape=(2, 2), dtype=float32, numpy=
array([[8., 8.],
       [8., 8.]], dtype=float32)>

In [144]: b**3
<tf.Tensor: id=484, shape=(2, 2), dtype=float32, numpy=
array([[8., 8.],
       [8., 8.]], dtype=float32)>

In [145]: tf.sqrt(b)
<tf.Tensor: id=486, shape=(2, 2), dtype=float32, numpy=
array([[1.4142135, 1.4142135],
       [1.4142135, 1.4142135]], dtype=float32)>
```

@ matmul



```
In [146]: a,b  
(<tf.Tensor: id=461, shape=(2, 2), dtype=float32, numpy=  
  array([[1., 1.],  
         [1., 1.]], dtype=float32)>,  
 <tf.Tensor: id=458, shape=(2, 2), dtype=float32, numpy=  
  array([[2., 2.],  
         [2., 2.]], dtype=float32)>)
```

```
In [147]: a@b  
<tf.Tensor: id=490, shape=(2, 2), dtype=float32, numpy=  
  array([[4., 4.],  
         [4., 4.]], dtype=float32)>
```

```
In [148]: tf.matmul(a,b)  
<tf.Tensor: id=492, shape=(2, 2), dtype=float32, numpy=  
  array([[4., 4.],  
         [4., 4.]], dtype=float32)>
```


@ matmul



```
In [150]: a=tf.ones([4,2,3])
In [151]: b=tf.fill([4,3,5], 2.)

In [152]: a@b
<tf.Tensor: id=503, shape=(4, 2, 5), dtype=float32, numpy=
array([[[6., 6., 6., 6., 6.],
...
[6., 6., 6., 6., 6.]])], dtype=float32)> 4个矩阵相乘并行计算

In [153]: tf.matmul(a,b)
<tf.Tensor: id=505, shape=(4, 2, 5), dtype=float32, numpy=
array([[[6., 6., 6., 6., 6.],
...
[6., 6., 6., 6., 6.]])], dtype=float32)>
```

With broadcasting



```
In [164]: a.shape # TensorShape([4, 2, 3])
```

```
In [165]: b.shape # TensorShape([3, 5])
```

```
In [166]: bb=tf.broadcast_to(b, [4,3,5])
```

```
In [167]: a@bb
```

```
<tf.Tensor: id=516, shape=(4, 2, 5), dtype=float32, numpy=
array([[[6., 6., 6., 6., 6.],
        [6., 6., 6., 6., 6.]], ...

       [[6., 6., 6., 6., 6.],
        [6., 6., 6., 6., 6.]]], dtype=float32)>
```

Recap

- $y = w * x + b$
 - $Y = X@W + b$
 - $\begin{bmatrix} x_0^0 & x_0^1 \\ x_1^0 & x_1^1 \end{bmatrix} \begin{bmatrix} w_{00} & w_{01} & w_{02} \\ w_{10} & w_{11} & w_{12} \end{bmatrix} + [b_0, b_1, b_2] \rightarrow \begin{bmatrix} y_0^0 & y_0^1 & y_0^2 \\ y_1^0 & y_1^1 & y_1^2 \end{bmatrix}$
 - $[b, 2] \rightarrow [b, 3]$
-

$$Y = X@W + b$$



```
In [168]: x=tf.ones([4,2])
```

```
In [169]: W=tf.ones([2,1])
```

```
In [170]: b=tf.constant(0.1)
```

```
In [171]: x@W+b
```

```
<tf.Tensor: id=526, shape=(4, 1), dtype=float32, numpy=
array([[2.1],
       [2.1],
       [2.1],
       [2.1]], dtype=float32)>
```

$$\textit{out} = \textit{relu}(X@W + b)$$

```

In [171]: x@W+b
<tf.Tensor: id=526, shape=(4, 1), dtype=float32, numpy=
array([[2.1],
       [2.1],
       [2.1],
       [2.1]], dtype=float32)>

In [172]: out=x@W+b
In [173]: out=tf.nn.relu(out)
<tf.Tensor: id=530, shape=(4, 1), dtype=float32, numpy=
array([[2.1],
       [2.1],
       [2.1],
       [2.1]], dtype=float32)>

```

下一课时

前向传播（张量）
实战

Thank You.
