



# TensorFlow

## 数据统计

---

主讲：龙良曲

# Outline

- `tf.norm`
  - `tf.reduce_min/max`
  - `tf.argmax/argmin`
  - `tf.equal`
  - `tf.unique`
-

# Vector Norm

**Eukl. Norm**  $\|x\|_2 = [\sum_k x_k^2]^{1/2}$

**Max.norm**  $\|x\|_\infty = \max_k |x_k|$

**$L_1$ -Norm**  $\|x\|_1 = \sum_k |x_k|$

---

# tf.norm

- Here talks about Vector Norm



```
In [50]: a=tf.ones([2,2])
```

```
In [52]: tf.norm(a)
```

```
Out[52]: <tf.Tensor: id=192, shape=(), dtype=float32, numpy=2.0>
```

```
In [54]: tf.sqrt(tf.reduce_sum(tf.square(a)))
```

```
Out[54]: <tf.Tensor: id=197, shape=(), dtype=float32, numpy=2.0>
```

```
In [55]: a=tf.ones([4,28,28,3])
```

```
In [56]: tf.norm(a)
```

```
Out[56]: <tf.Tensor: id=206, shape=(), dtype=float32, numpy=96.99484>
```

```
In [57]: tf.sqrt(tf.reduce_sum(tf.square(a)))
```

```
Out[57]: <tf.Tensor: id=211, shape=(), dtype=float32, numpy=96.99484>
```

# L1 Norm

```
In [68]: b = tf.ones([2,2])
```

```
In [69]: tf.norm(b) 默认求L2范数
```

```
Out[69]: <tf.Tensor: id=250, shape=(), dtype=float32, numpy=2.0>
```

```
In [73]: tf.norm(b,ord=2,axis=1)
```

```
Out[73]: <tf.Tensor: id=271, shape=(2,), dtype=float32, numpy=array([1.4142135, 1.4142135], dtype=float32)>
```

```
In [70]: tf.norm(b,ord=1)
```

```
Out[70]: <tf.Tensor: id=255, shape=(), dtype=float32, numpy=4.0>
```

```
In [71]: tf.norm(b,ord=1,axis=0)
```

```
Out[71]: <tf.Tensor: id=260, shape=(2,), dtype=float32, numpy=array([2., 2.], dtype=float32)>
```

```
In [72]: tf.norm(b,ord=1,axis=1)
```

```
Out[72]: <tf.Tensor: id=265, shape=(2,), dtype=float32, numpy=array([2., 2.], dtype=float32)>
```

# reduce\_min/max/mean



```
In [76]: a=tf.random.normal([4,10])
```

```
In [78]: tf.reduce_min(a),tf.reduce_max(a),tf.reduce_mean(a)
```

```
Out[78]:
```

```
(<tf.Tensor: id=283, shape=(), dtype=float32, numpy=-1.1872448>,  
 <tf.Tensor: id=285, shape=(), dtype=float32, numpy=2.1353827>,  
 <tf.Tensor: id=287, shape=(), dtype=float32, numpy=0.3523524>)
```

```
In [79]: tf.reduce_min(a,axis=1),tf.reduce_max(a,axis=1),tf.reduce_mean(a,axis=1)
```

```
Out[79]:
```

```
(<tf.Tensor: id=292, shape=(4,), dtype=float32, numpy=array([-0.3937837,  
 -1.1872448, -1.0798895, -1.1366792], dtype=float32)>,  
 <tf.Tensor: id=294, shape=(4,), dtype=float32, numpy=array([1.9718986, 1.1612172,  
 2.1353827, 2.0984378], dtype=float32)>,  
 <tf.Tensor: id=296, shape=(4,), dtype=float32, numpy=array([ 0.61504304,  
 -0.01389184, 0.606747 , 0.20151143], dtype=float32)>)
```

# argmax/argmin



```
In [80]: a.shape
```

```
Out[80]: TensorShape([4, 10])
```

```
In [81]: tf.argmax(a).shape
```

默认axis=0

```
Out[81]: TensorShape([10])
```

```
In [83]: tf.argmax(a)
```

```
Out[83]: <tf.Tensor: id=305, shape=(10,), dtype=int64, numpy=array([0, 0, 2, 3, 1, 3, 0, 1, 2, 0])>
```

```
In [82]: tf.argmin(a).shape
```

```
Out[82]: TensorShape([10])
```

# tf.equal



```
In [44]: a=tf.constant([1,2,3,2,5])
```

```
In [45]: b=tf.range(5)
```

```
In [46]: tf.equal(a,b)
```

```
Out[46]: <tf.Tensor: id=170, shape=(5,), dtype=bool, numpy=array([False, False, False, False, False])>
```

```
In [47]: res=tf.equal(a,b)
```

```
In [48]: tf.reduce_sum(tf.cast(res, dtype=tf.int32))
```

```
Out[48]: <tf.Tensor: id=175, shape=(), dtype=int32, numpy=0>
```



# Accuracy

```
● ● ●  
In [99]: a  
<tf.Tensor: id=308, shape=(2, 3), dtype=float32, numpy=  
array([[0.1 , 0.2 , 0.7 ],  
       [0.9 , 0.05, 0.05]], dtype=float32)>  
  
In [100]: pred=tf.cast(tf.argmax(a,axis=1), dtype=tf.int32)  
Out[101]: <tf.Tensor: id=324, shape=(2,), dtype=int32, numpy=array([2, 0])>  
  
In [110]: y  
Out[110]: <tf.Tensor: id=328, shape=(2,), dtype=int32, numpy=array([2, 1],  
dtype=int32)>  
  
In [112]: tf.equal(y,pred)  
Out[112]: <tf.Tensor: id=335, shape=(2,), dtype=bool, numpy=array([ True, False])>  
  
In [113]: correct=tf.reduce_sum(tf.cast(tf.equal(y,pred),dtype=tf.int32))  
  
In [114]: correct  
Out[114]: <tf.Tensor: id=340, shape=(), dtype=int32, numpy=1>  
  
In [115]: correct/2  
Out[115]: <tf.Tensor: id=345, shape=(), dtype=float64, numpy=0.5>
```

# tf.unique

```
In [116]: a=tf.range(5)
```

```
In [117]: tf.unique(a)
```

```
Out[117]: Unique(y=<tf.Tensor: id=351, shape=(5,), dtype=int32, numpy=array([0, 1, 2, 3, 4], dtype=int32)>, idx=<tf.Tensor: id=352, shape=(5,), dtype=int32, numpy=array([0, 1, 2, 3, 4], dtype=int32)>)
```

```
In [118]: a=tf.constant([4,2,2,4,3])
```

```
In [119]: tf.unique(a)
```

```
Out[119]: Unique(y=<tf.Tensor: id=356, shape=(3,), dtype=int32, numpy=array([4, 2, 3], dtype=int32)>, idx=<tf.Tensor: id=357, shape=(5,), dtype=int32, numpy=array([0, 1, 1, 0, 2], dtype=int32)>)
```

利用 `tf.gather(Unique, idx)` 可还原

下一课时

---

数据排序

**Thank You.**

---