



TensorFlow

数据类型

主讲人：龙良曲

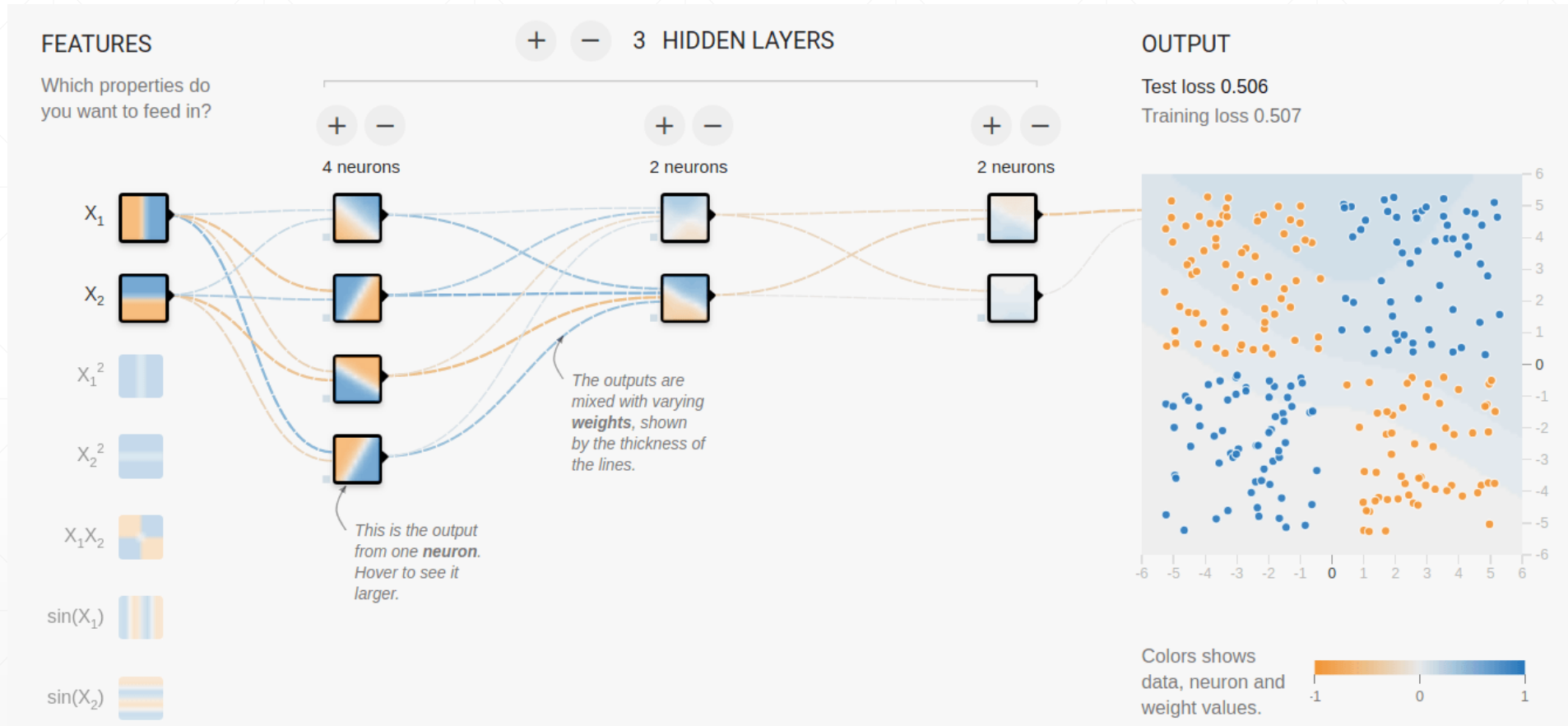
Data Container

- list
 - np.array
 - tf.Tensor
-

What's Tensor

- scalar: 1.1
 - vector: [1.1], [1.1, 2.2, ...]
 - matrix: [[1.1, 2.2], [3.3, 4.4], [5.5, 6.6]]
 - tensor: $rank > 2$
-

Tensor Flow in Graph



TF is a computing lib

- int, float, double
 - bool
 - string
-

Create

```
In [3]: tf.constant(1)
```

```
Out[3]: <tf.Tensor: id=2, shape=(), dtype=int32, numpy=1>
```

```
In [4]: tf.constant(1.)
```

```
Out[4]: <tf.Tensor: id=4, shape=(), dtype=float32, numpy=1.0>
```

```
In [5]: tf.constant(2.2, dtype=tf.int32)
```

```
# TypeError: Cannot convert provided value to EagerTensor.
```

```
# Provided value: 2.2 Requested dtype: int32
```

```
In [6]: tf.constant(2., dtype=tf.double)
```

```
Out[6]: <tf.Tensor: id=7, shape=(), dtype=float64, numpy=2.0>
```

```
In [7]: tf.constant([True, False])
```

```
Out[7]: <tf.Tensor: id=9, shape=(2,), dtype=bool, numpy=array([ True, False])>
```

```
In [11]: tf.constant('hello,world.')
```

```
Out[11]: <tf.Tensor: id=14, shape=(), dtype=string, numpy=b'hello,world.'>
```

Tensor Property

```
1 In [35]: with tf.device("cpu"):
2         ...:     a=tf.constant([1])
3 In [36]: with tf.device('gpu'):
4         ...:     b=tf.range(4)
5
6 In [37]: a.device #'/job:localhost/replica:0/task:0/device:CPU:0'
7 In [38]: b.device #'/job:localhost/replica:0/task:0/device:GPU:0'
8
9 In [39]: aa=a.gpu()
10 In [40]: aa.device #'/job:localhost/replica:0/task:0/device:GPU:0'
11
12 In [41]: bb=b.cpu()
13 In [43]: bb.device #'/job:localhost/replica:0/task:0/device:CPU:0'
14
15 In [44]: b.numpy()
16 Out[44]: array([0, 1, 2, 3], dtype=int32)
17
18 In [46]: b.ndim
19 Out[46]: 1
20 In [47]: tf.rank(b)
21 Out[47]: <tf.Tensor: id=20, shape=(), dtype=int32, numpy=1>
22
23 In [48]: tf.rank(tf.ones([3,4,2]))
24 Out[48]: <tf.Tensor: id=25, shape=(), dtype=int32, numpy=3>
25
26 In [49]: b.name
27 AttributeError: Tensor.name is meaningless when eager execution is enabled.
```

Check Tensor Type

```
1 In [15]: a=tf.constant([1.])
2 In [16]: b=tf.constant([True, False])
3 In [17]: c=tf.constant('hello,world.')
4 In [22]: d=np.arange(4)
5
6 In [19]: isinstance(a,tf.Tensor)
7 Out[19]: True
8 In [20]: tf.is_tensor(b)
9 Out[20]: True
10
11 In [23]: tf.is_tensor(d)
12 Out[23]: False
13
14 In [24]: a.dtype,b.dtype,c.dtype
15 Out[24]: (tf.float32, tf.bool, tf.string)
16
17 In [25]: a.dtype=tf.float32
18 Out[25]: True
19
20 In [26]: c.dtype=tf.string
21 Out[26]: True
```


Convert



```
In [15]: a=np.arange(5)
# array([0, 1, 2, 3, 4])
```

```
In [17]: a.dtype
Out[17]: dtype('int64')
```

```
In [19]: aa=tf.convert_to_tensor(a)
# <tf.Tensor: id=19, shape=(5,), dtype=int64, numpy=array([0, 1, 2, 3, 4])>
```

```
In [21]: aa=tf.convert_to_tensor(a,dtype=tf.int32)
# <tf.Tensor: id=21, shape=(5,), dtype=int32, numpy=array([0, 1, 2, 3, 4], dtype=int32)>
```

```
In [23]: tf.cast(aa, dtype=tf.float32)
Out[23]: <tf.Tensor: id=23, shape=(5,), dtype=float32, numpy=array([0., 1., 2., 3., 4.],
dtype=float32)>
```

```
In [26]: aaa=tf.cast(aa, dtype=tf.double)
# <tf.Tensor: id=27, shape=(5,), dtype=float64, numpy=array([0., 1., 2., 3., 4.])>
```

```
In [27]: tf.cast(aaa,dtype=tf.int32)
Out[27]: <tf.Tensor: id=28, shape=(5,), dtype=int32, numpy=array([0, 1, 2, 3, 4], dtype=int32)>
```

bool \diamond int



```
In [28]: b=tf.constant([0,1])
```

```
In [29]: tf.cast(b,dtype=tf.bool)
```

```
Out[29]: <tf.Tensor: id=31, shape=(2,), dtype=bool, numpy=array([False,  True])>
```

```
In [30]: bb=tf.cast(b,dtype=tf.bool)
```

```
In [31]: tf.cast(bb,tf.int32)
```

```
Out[31]: <tf.Tensor: id=34, shape=(2,), dtype=int32, numpy=array([0, 1], dtype=int32)>
```

tf.Variable

```

In [34]: a=tf.range(5)
# <tf.Tensor: id=41, shape=(5,), dtype=int32, numpy=array([0, 1, 2, 3, 4], dtype=int32)>

In [36]: b=tf.Variable(a)
In [37]: b.dtype # tf.int32
In [38]: b.name # 'Variable:0'

In [39]: b=tf.Variable(a, name='input_data')
In [40]: b.name # 'input_data:0'
In [41]: b.trainable # True

In [42]: isinstance(b, tf.Tensor) # False
In [43]: isinstance(b, tf.Variable) # True
In [44]: tf.is_tensor(b) # True

In [45]: b.numpy()
Out[45]: array([0, 1, 2, 3, 4], dtype=int32)
```

To numpy



```
In [92]: a.numpy()  
array([[ 0.03739073, -1.0016401 ],  
       [ 0.26954213, -0.21734552]], dtype=float32)
```

```
In [93]: b.numpy()  
array([[ 0.03739073, -1.0016401 ],  
       [ 0.26954213, -0.21734552]], dtype=float32)
```

```
In [94]: a=tf.ones([])
```

```
In [95]: a.numpy()
```

```
Out[95]: 1.0
```

```
In [96]: int(a)
```

```
Out[96]: 1
```

```
In [97]: float(a)
```

```
Out[97]: 1.0
```

下一课时

创建Tensor

Thank You.
