

# 输出方式

---

主讲：龙良曲

# Outline

- $y \in R^d$
  - $y_i \in [0, 1], i = 0, 1, \dots, y_d - 1$
  - $y_i \in [0, 1], \sum_{i=0}^{y_d} y_i = 1, i = 0, 1, \dots, y_d - 1$
  - $y_i \in [-1, 1], i = 0, 1, \dots, y_d - 1$
-

1.  $y \in R^d$

- linear regression
  - naïve classification with MSE
  - other general prediction
  - $out = relu(X@W + b)$ 
    - logits
-

## 2. $y_i \in [0, 1]$

- binary classification

- $y > 0.5, \rightarrow 1$
- $y < 0.5, \rightarrow 0$

- Image Generation

- rgb

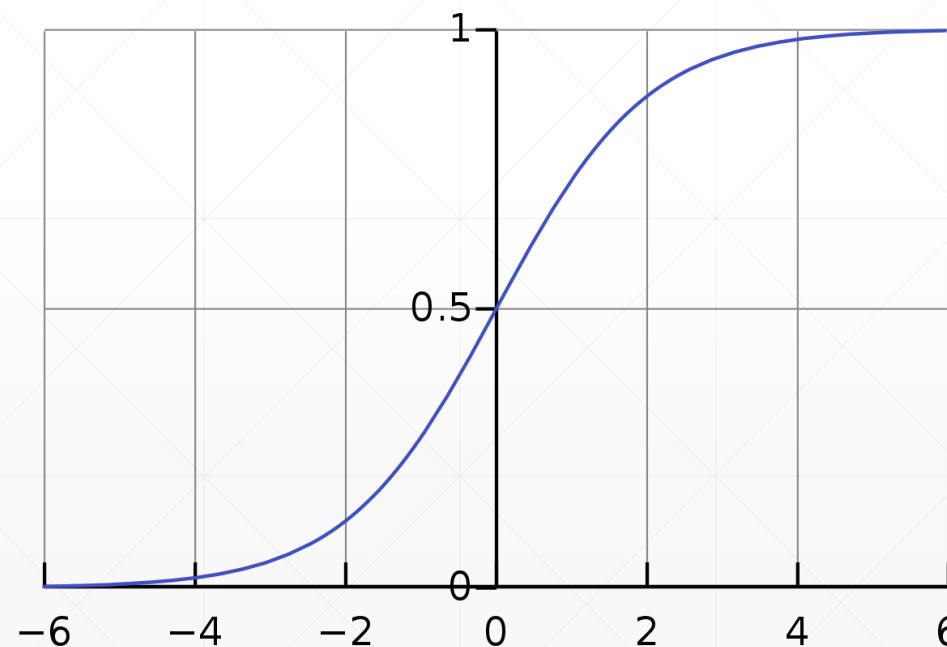


2.  $y_i \in [0, 1]$

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

- sigmoid

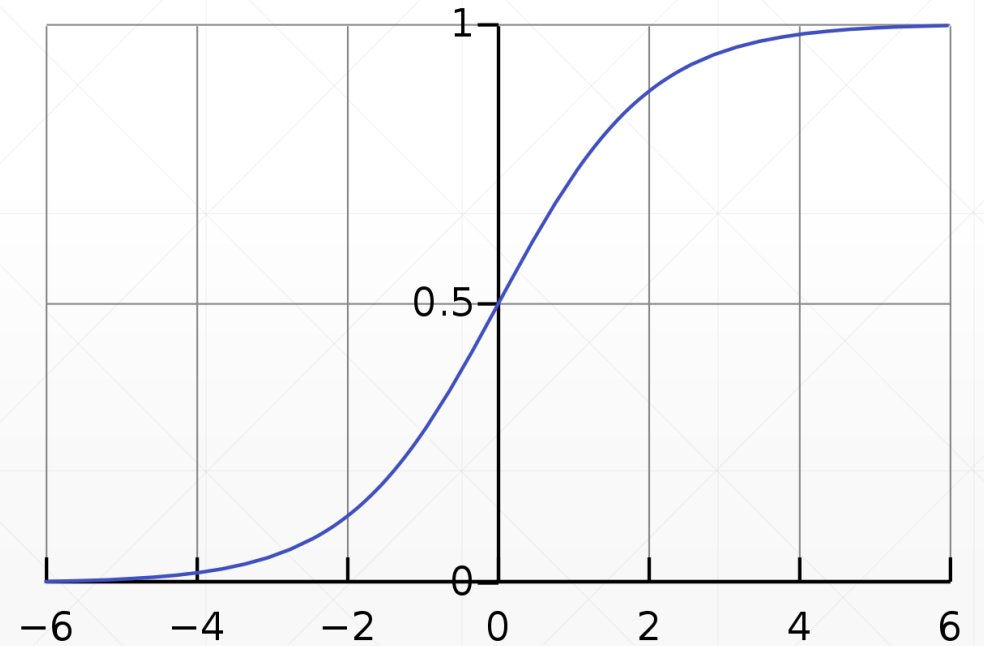
- $out' = \text{sigmoid}(out)$



# sigmoid

- `tf.sigmoid`

$$f(x) = \frac{1}{1 + e^{-x}}$$





```
In [10]: a=tf.linspace(-6.,6,10)
<tf.Tensor: id=19, shape=(10,), dtype=float32, numpy=
array([-6.          , -4.6666665, -3.3333333, -2.          , -0.6666665,
        0.6666667 ,  2.          ,  3.3333334 ,  4.6666667 ,  6.          ]
```

```
In [12]: tf.sigmoid(a)
<tf.Tensor: id=21, shape=(10,), dtype=float32, numpy=
array([0.00247264, 0.00931597, 0.03444517, 0.11920291, 0.33924365,
        0.6607564 , 0.8807971 , 0.96555483, 0.99068403, 0.9975274 ]
```

```
In [13]: x=tf.random.normal([1,28,28])*5
In [15]: tf.reduce_min(x), tf.reduce_max(x)
(<tf.Tensor: id=32, shape=(), dtype=float32, numpy=-18.78872>,
 <tf.Tensor: id=34, shape=(), dtype=float32, numpy=15.466431>)
```

```
In [16]: x=tf.sigmoid(x)
In [17]: tf.reduce_min(x), tf.reduce_max(x)
(<tf.Tensor: id=39, shape=(), dtype=float32, numpy=0.0>,
 <tf.Tensor: id=41, shape=(), dtype=float32, numpy=0.99999976>)
```

3.  $y_i \in [0, 1]$ ,  $\sum y_i = 1$

### ■ sigmoid

```
● ● ●  
In [21]: a=tf.linspace(-2.,2,5)  
In [22]: tf.sigmoid(a)  
<tf.Tensor: id=54, shape=(5,), dtype=float32, numpy=  
array([0.11920291, 0.26894143, 0.5          , 0.7310586 , 0.880797  ],  
      dtype=float32)>
```

### ■ softmax

```
● ● ●  
In [23]: tf.nn.softmax(a)  
<tf.Tensor: id=56, shape=(5,), dtype=float32, numpy=  
array([0.01165623, 0.03168492, 0.08612854, 0.23412167, 0.6364086 ],  
      dtype=float32)>
```

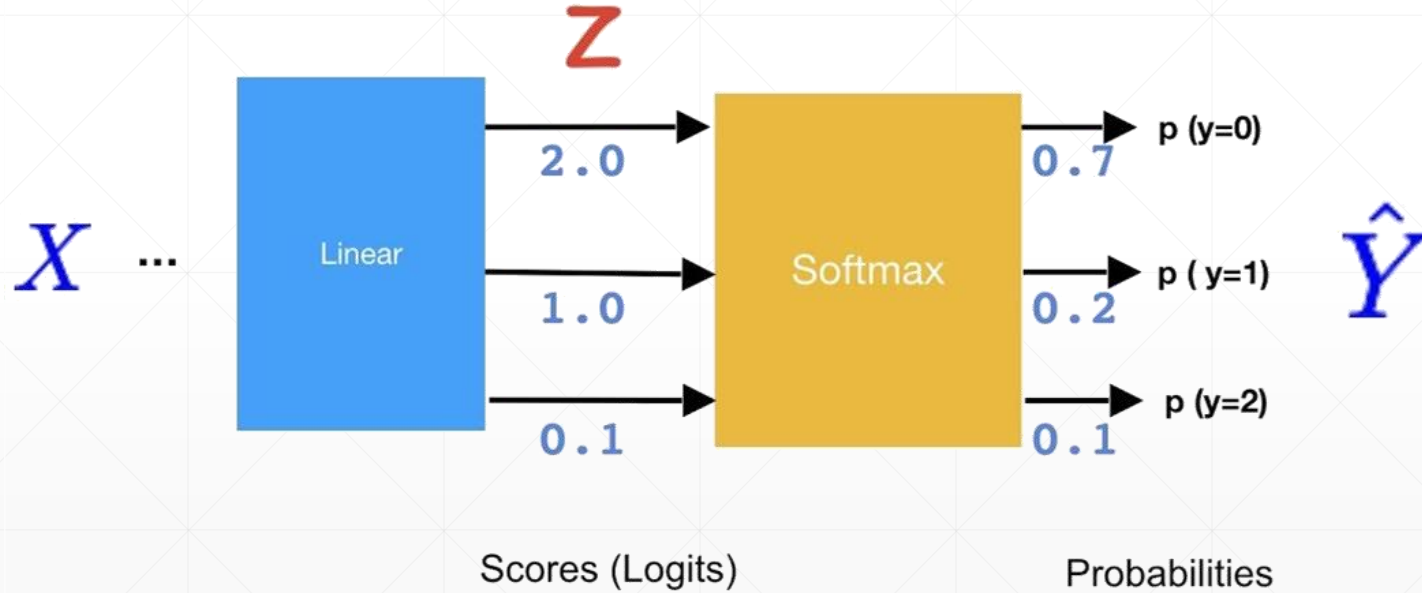
---



# softmax

## Meet Softmax

$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad \text{for } j = 1, \dots, K.$$



# Classification



```
In [24]: logits=tf.random.uniform([1,10],minval=-2,maxval=2)
<tf.Tensor: id=64, shape=(1, 10), dtype=float32, numpy=
array([[ -1.5842109 ,  1.6475668 , -0.71567106,  1.5819931 ,  0.35972595,
         0.12536812,  1.0662012 , -0.70131207, -1.5194197 ,  1.5553613 ]],
      dtype=float32)>

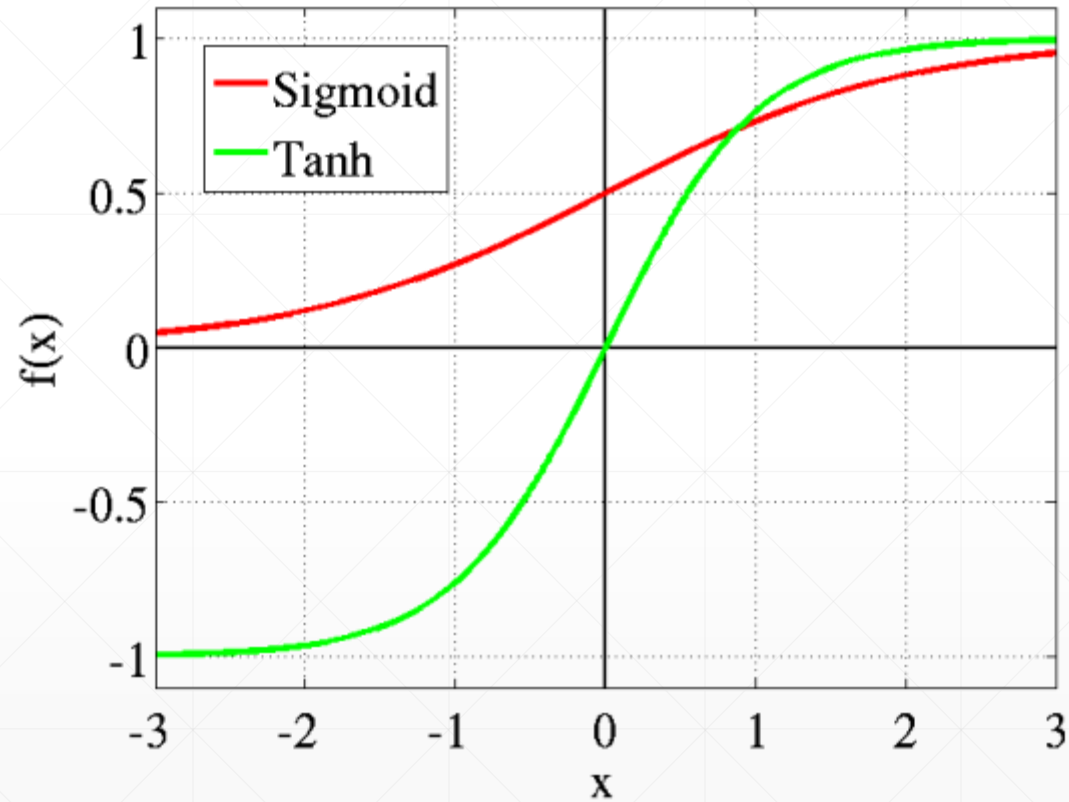
In [27]: prob=tf.nn.softmax(logits,axis=1)
<tf.Tensor: id=67, shape=(1, 10), dtype=float32, numpy=
array([[0.00946281, 0.23964217, 0.02255392, 0.22443208, 0.06610908,
        0.05229748, 0.13399215, 0.02288011, 0.01009621, 0.21853393]],
      dtype=float32)>

In [29]: tf.reduce_sum(prob,axis=1)
Out[29]: <tf.Tensor: id=70, shape=(1,), dtype=float32, numpy=array([0.99999994],
      dtype=float32)>
```

---

#### 4. $y_i \in [-1, 1]$

- tanh



$$\tanh(x) = \sinh(x)/\cosh(x) = (e^x - e^{-x}) / (e^x + e^{-x})$$

---



```
In [30]: a
```

```
Out[30]: <tf.Tensor: id=53, shape=(5,), dtype=float32, numpy=array([-2., -1.,  0.,  
  1.,  2.], dtype=float32)>
```

```
In [33]: tf.tanh(a)
```

```
<tf.Tensor: id=73, shape=(5,), dtype=float32, numpy=  
array([-0.9640276, -0.7615942,  0.          ,  0.7615942,  0.9640276],  
      dtype=float32)>
```

---

下一课时

---

误差计算

**Thank You.**

---