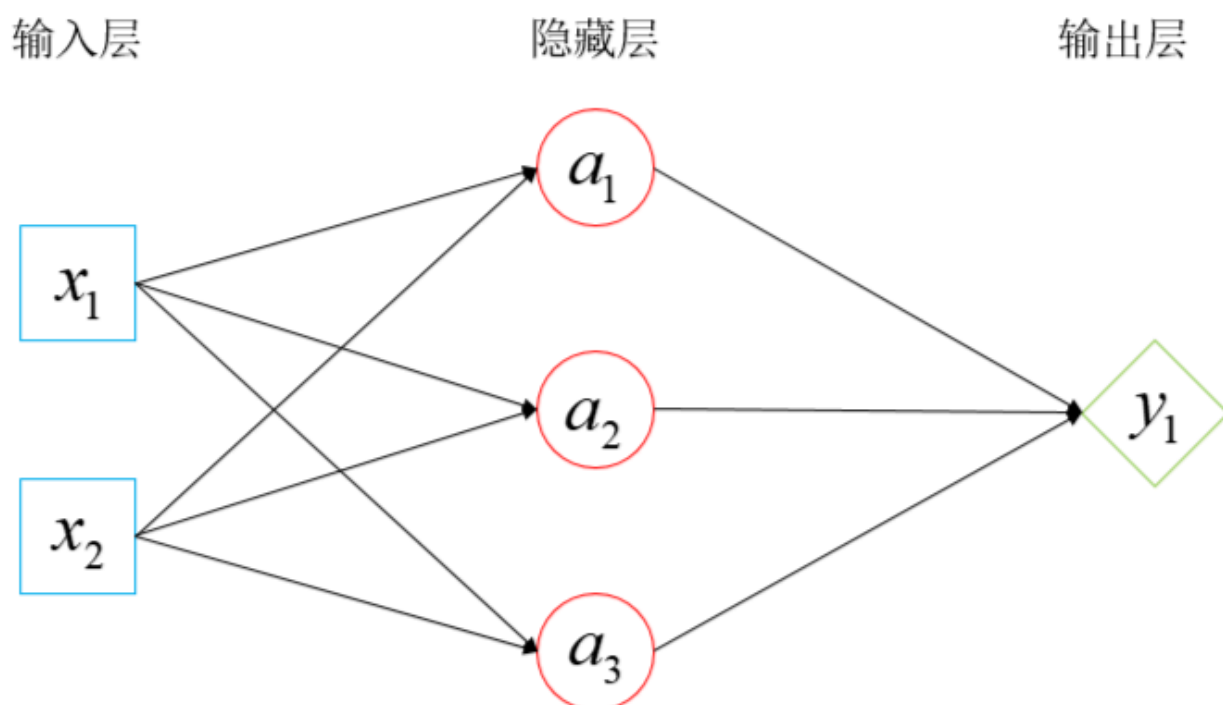


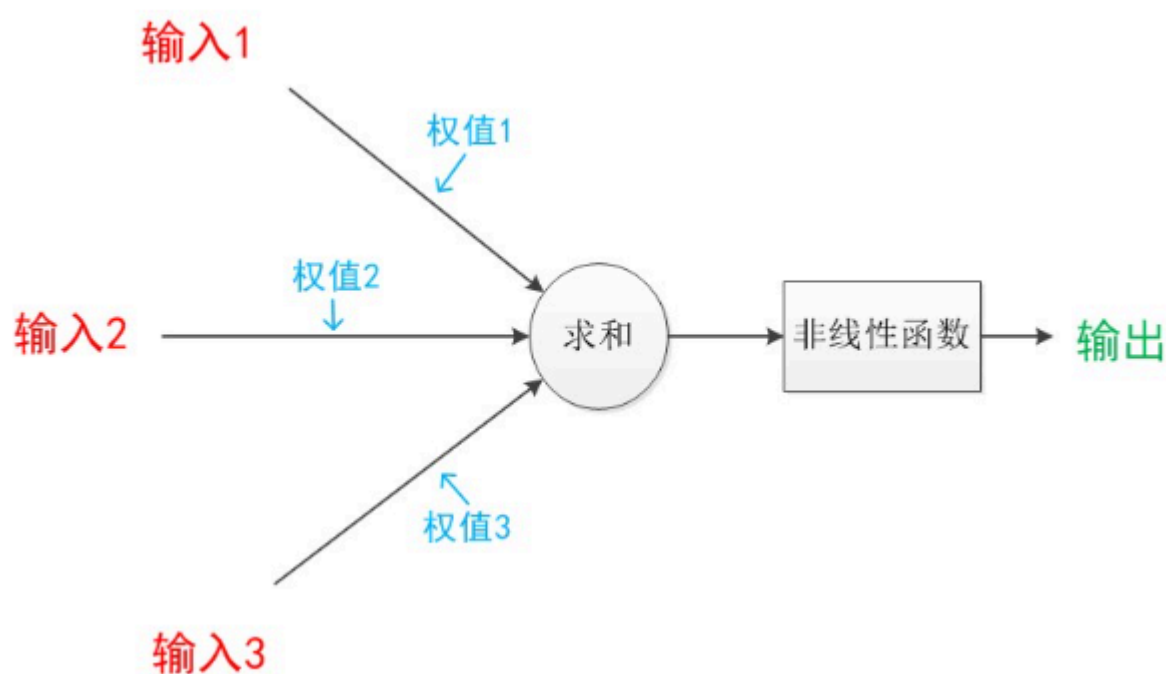
全连接神经网络

一、全连接神经网络结构

全连接神经网络的整体结构如下所示：其包括输入层、隐藏层和输出层，其特点是**每一层的每一个节点都与上下层节点全部连接**，其中隐藏层可以有很多层。



对于其中的每一个节点（即单独的一个神经元结构）如下图所示：



这个结构实际上就是前面章节的逻辑回归模型结构，求和后需要经过一个激活函数，如果网络中没有使用激活函数，每一层的节点的输入都是上层输出的线性函数，无论神经网络中的隐含层有多少，最后的输出结果都是网络输入的线性拟合，即隐含层没有发挥其作用。

二、常见的激活函数特点

1.sigmoid函数（S型函数）

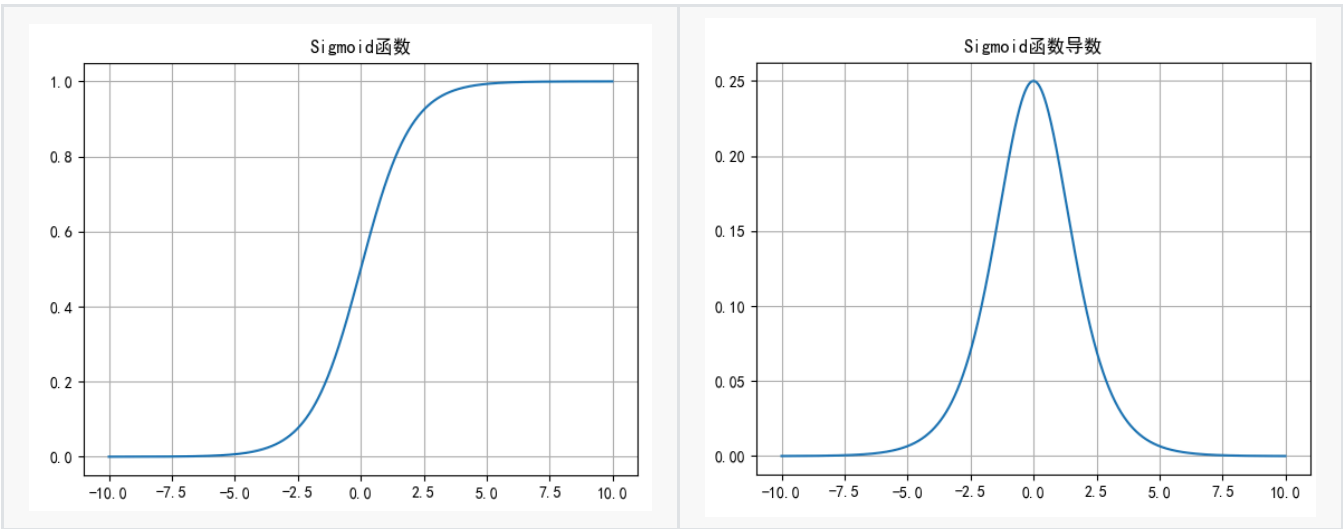
sigmoid函数的定义如下：

$$g(Z) = \frac{1}{1 + e^{-z}} \tag{1}$$

由于在使用梯度下降法求解最优参数时会对损失函数求导，而损失函数中会包含激活函数，因此需要研究激活函数的导数形式：

$$g' = g(1 - g) \tag{2}$$

其函数图像和导数图像如下所示：



观察图像可以看出，其输出在[0-1]之间，非常适合**二分类任务**；但是其导数值域在[0-0.25],因此会出现**梯度消失的问题**（隐藏层数过多时，参数更新过程中导数叠乘导致值趋于0，参数不再更新），同时由于其输出关于0不对称，会导致参数更新速度较慢。

2.Tanh函数（双曲正切函数）

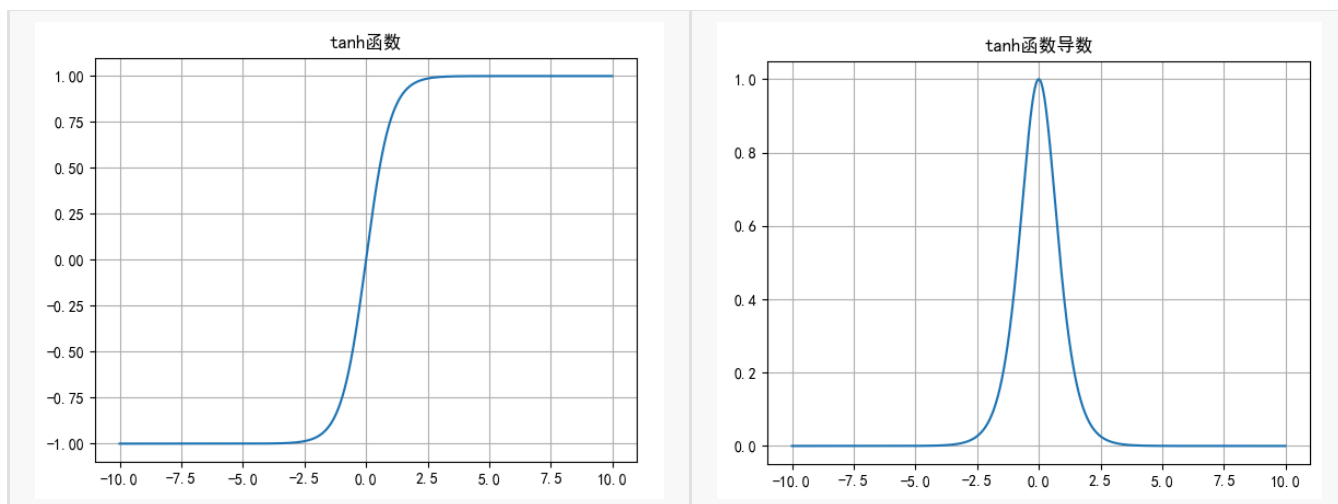
Tanh函数的定义如下：

$$g(Z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \tag{3}$$

导数公式如下：

$$g' = 1 - g^2 \tag{4}$$

其函数图像和导数图像如下所示：



其与sigmoid函数非常相似，输出范围在 $[-1,1]$ 关于0对称，训练时收敛速度更快，其同样存在梯度消失的问题。

3.ReLU函数

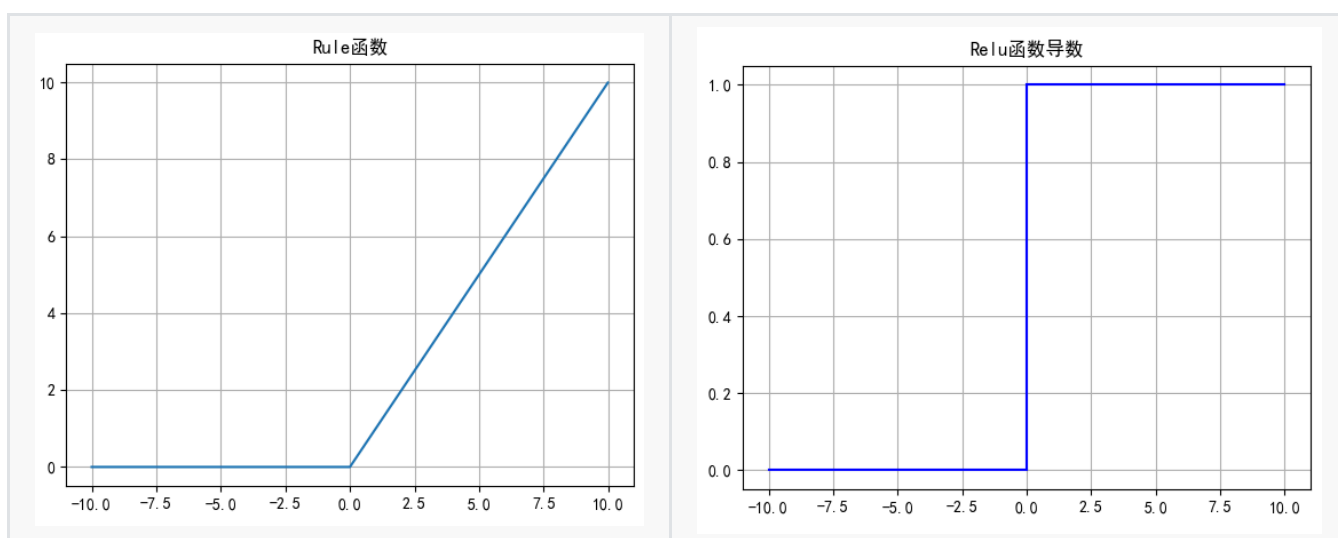
Tanh函数的定义如下：

$$g(Z) = \begin{cases} Z & \text{if } Z > 0 \\ 0 & \text{if } Z \leq 0 \end{cases} \quad (5)$$

导数公式如下：

$$g' = \begin{cases} 1 & \text{if } Z > 0 \\ 0 & \text{if } Z \leq 0 \end{cases} \quad (6)$$

其函数图像和导数图像如下所示：



其解决了梯度消失的问题，但是当输入为负时，梯度为0。这个神经元及之后的神经元梯度永远为0，不再对任何数据有所响应，导致相应参数永远不会被更新，即可能出现神经元死亡。

4.Softmax函数

Softmax函数的定义如下：

$$g_i = \frac{e^{z_i}}{\sum_{j=1}^N e^{z_j}} \quad (7)$$

导数公式如下：

$$g'_i = g_i(1 - g_i) \quad (8)$$

Softmax激活函数主要针对**多分类问题**，其中N为分类的个数， g_i 为属于对应类别的概率输出，Softmax函数满足每个输出值在[0-1],所有输出值和为1。

三、前向传播和反向传播

1.前向传播

神经网络中的前向传播实际上就是**通过输入的特征X和设定的参数w计算出预测输出Y值的过程**。

2.反向传播

神经网络中的反向传播实际上就是**利用前向传播计算出的值与真实值之间误差进行梯度更新参数的过程**。

进行具体的数学计算时只需要按照求导的链式法则，依次对每一层进行损失函数的求导即可。链式法则如下所示：

$$\frac{dy}{dx} = \frac{dy}{du} \frac{du}{dx} \quad (9)$$

三、代码实践问题

1.Numpy和TensorFlow版本冲突：

TensorFlow ≤2.10 时配合 **NumPy ≥1.24** 会出充图问题，具体报错如下所示：

```
E:\Software\anaconda\envs\keras\lib\site-packages\tensorflow\python\framework\dtypes.py:513:
FutureWarning: In the future `np.object` will be defined as the corresponding NumPy scalar.
  np.object,
```

解决方法：使用视频中指定的版本

```
python==3.8
tensorflow==2.4.0
keras==2.4.3
numpy==1.19.5
pandas==1.3.5
matplotlib==3.4.2
sklearn==0.0
```

卸载安装对应版本numpy

```
pip uninstall numpy -y  
pip install numpy==1.19.5
```