博客网址：<https://www.cnblogs.com/missidiot/p/9378079.html>

什么是激活函数？

-激活函数是神经网络中极其重要的概念，决定了某个神经元是否被激活，这个神经元接受到的消息是否是有用的，是否该留下或者是该抛弃。

-激活函数式我们对输入做的一种非线性的转换。转换的结果输出，bong当做下一个隐藏层的输入。

-引入激活函数是为了增加神经网络模型的非线性。

-没有激活函数的每层都相当于矩阵相乘。就算你叠加了若干层之后，无非还是个矩阵相乘罢了。

为什么要用激活函数？

-如果不用激活函数，每一层输出都是上层输入的线性函数，无论神经网络有多少层，输出都是输入的线性组合，这种情况就是最原始的感知机（Perceptron）。

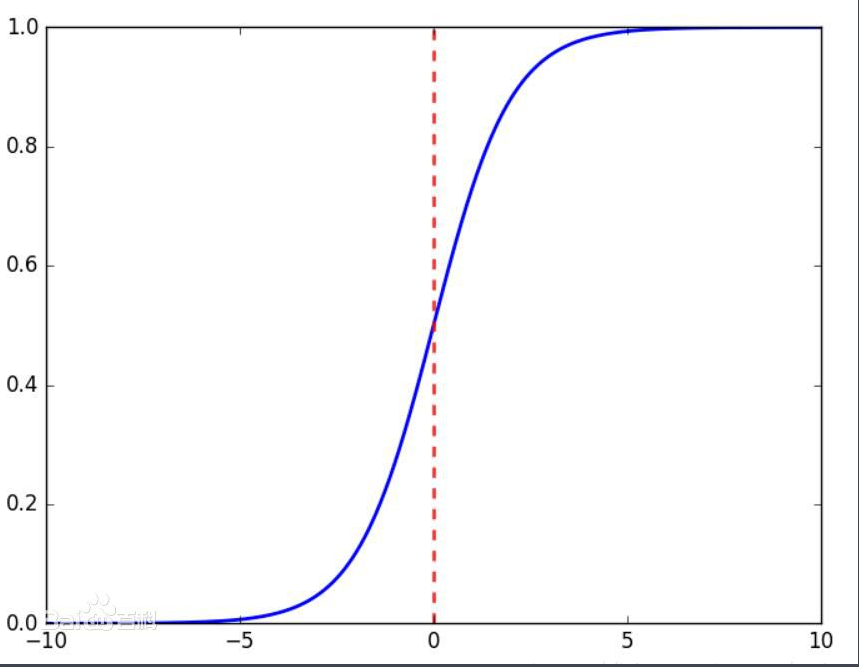
-如果使用的话，激活函数给神经元引入了非线性因素，使得神经网络可以任意逼近任何非线性函数，这样神经网络就可以应用到众多的非线性模型中。

常用的激活函数：

-1：Sigmoid函数：

-Sigmoid函数是一个在生物学中常见的S型函数，也称为S型生长曲线。在信息科学中，由于其单增以及反函数单增等性质，Sigmoid函数常被用作神经网络的阈值函数，将变量映射到0,1之间。sigmoid 是使用范围最广的一类激活函数，具有指数函数形状，它在物理意义上最为接近生物神经元。，sigmoid函数连续，光滑，严格单调，以(0,0.5)中心对称，是一个非常良好的阈值函数。

-公式：

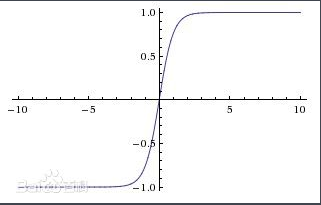
-图像：

-然而，sigmoid也有其自身的缺陷，最明显的就是饱和性：其两侧导数逐渐趋近于0，具有这种性质的称为软饱和激活函数，由于在后向传递过程中，sigmoid向下传导的梯度包含了一个 f′(x)f′(x) 因子（sigmoid关于输入的导数），因此一旦输入落入饱和区，f′(x)f′(x) 就会变得接近于0，导致了向底层传递的梯度也变得非常小。此时，网络参数很难得到有效训练。这种现象被称为梯度消失。一般来说， sigmoid 网络在 5 层之内就会产生梯度消失现象，此外，sigmoid函数的输出均大于0，使得输出不是0均值，这称为偏移现象，这会导致后一层的神经元将得到上一层输出的非0均值的信号作为输入。

-2：Tanh函数：

-Tanh是双曲函数中的一个，Tanh()为双曲正切。在数学中，双曲正切“Tanh”是由基本双曲函数双曲正弦和双曲余弦推导而来。

-公式：

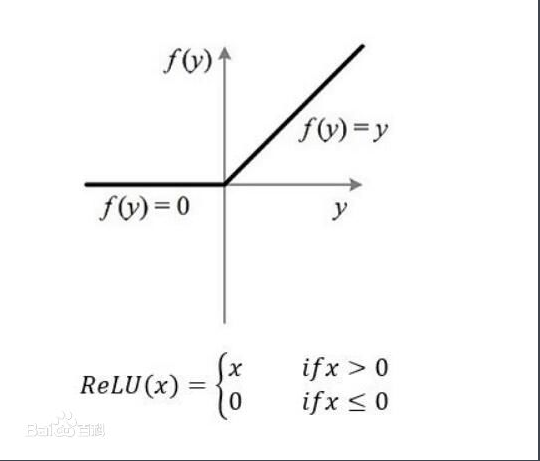
-图像：

-与sigmoid相比，它的输出均值是0，使得其收敛速度要比sigmoid快，减少迭代次数。然而，从图中可以看出，tanh一样具有软饱和性，从而造成梯度消失

-3：ReLU函数：

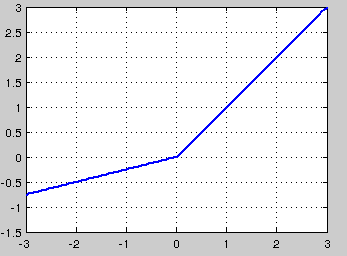
-Relu激活函数（The Rectified Linear Unit），用于隐层神经元输出。

-公式：

-图像：

-从图中可以看出当x<0时，ReLU硬饱和，而当x>0时，则不存在饱和问题。所以，ReLU 能够在x>0时保持梯度不衰减，从而缓解梯度消失问题。这让我们能够直接以监督的方式训练深度神经网络，而无需依赖无监督的逐层预训练。然而，随着训练的推进，部分输入会落入硬饱和区，导致对应权重无法更新。这种现象被称为“神经元死亡”。与sigmoid类似，ReLU的输出均值也大于0，偏移现象和 神经元死亡会共同影响网络的收敛性。

针对在x<0的硬饱和问题，我们对ReLU做出相应的改进，使得



这就是Leaky-ReLU,