

原

# 几种激活函数的比较

2017年04月14日 17:21:30 [guorongronghe](#) 阅读数：5772

版权声明：本文为博主原创文章，未经博主允许不得转载。

<https://blog.csdn.net/guorongronghe/article/details/70174476>

激活函数：用来加入非线性因素的，因为线性模型的表达能力不够

比如下图的数据比较简单是线性可分的，一条直线就可以对样本进行分类了：

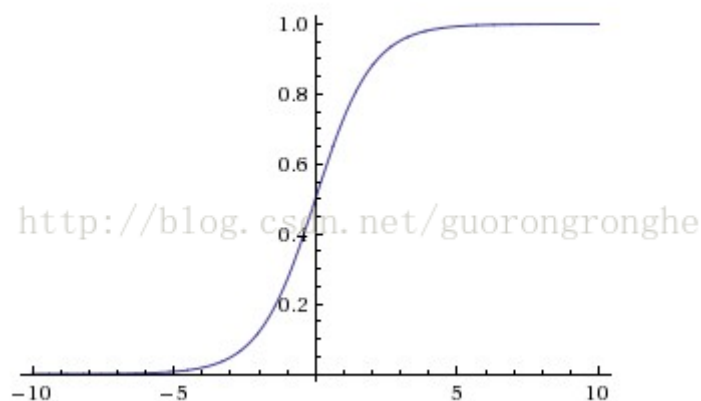
但是如果碰到下面这个情况，数据变成线性不可分的时候，一条直线根本无法很好的对数据进行分类，这样就引入非线性因素，而激活函数恰好能够做到这点：

下面介绍几个常用的激活函数：

1.sigmoid函数：用于隐层神经元输出

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

函数图像为：



优缺点：输出映射在 (0, 1) 之间，用于输出层，求导容易

缺点：容易产生梯度消失，导致训练出现问题，输出不是以0为中心的

2.tanh函数：

$$\tanh(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}}$$

区间位于 $[-1,1]$ 之间，函数图像为：

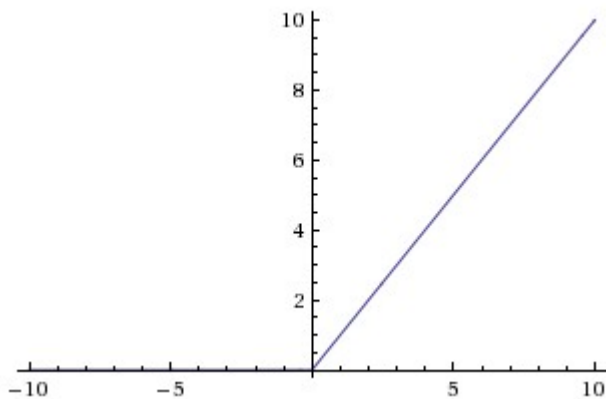
函数的优缺点：

这个函数和sigmoid相比收敛速度快，但是和sigmoid一样有个缺点是：容易产生梯度消失

3.ReLU：用于隐层神经元输出

$f(x) = \max(0, x)$  即 if  $x \geq 0$  时  $f(x) = x$ , 否则  $f(x) = 0$

对应的图像为：



或者可表示为：

优缺点：

优点：这个是线性,非饱和的形式和前面两个相比它能够快速收敛，另外ReLU有效的缓解了梯度消失的问题，而且它提供了神经网络的稀疏表达能力

缺点：随着训练的进行，可能会出现神经元死亡，权重无法更新的情况。如果发生这种情况，那么流经神经元的梯度从这一点开始将永远是0。也就是说，ReLU神经元在训练中不可逆地死亡了。

4.softmax函数：用于多分类神经网络输出

$$f(x) = \log(1 + \exp(x))$$

以上就是这几个激活函数的比较，最后将这几个函数显示在同一个图像上，方便比较

