上篇笔记讲到，要优化Deep Learning的神经网络首先要搞清楚问题出在哪里，总体上分为两部分：针对Training data上性能的优化和针对Testing data的性能优化。本文将介绍针对Training data上性能优化的一种方法：替换激活函数（New activation function）。

**1 问题背景**

早期的神经网络中激活函数一般都使用的，当Deep Learning的Hidden Layer变多后，就会出现图1所示的情况。



图1 手写分类准确率

图1可以看出，当神经网络深度超过7层后，性能显著下降。**有一个认识误区是：因为网络变深之后，参数数量增加，导致了Over-fitting。**实际上，并不是Over-fitting，因为我们是在Training data上训练的性能变差的，并不是Over-fitting，换句话说，10层网络如果训练得当，至少性能也应该和5层网络一样，因为只需要保持前面5层网络相同，剩下的5层网络构成一个Identity network就可以了；但是层数增加反而性能下降，就是Training没有做好。

**2 问题原因**

主要是“梯度消失”（Vanishing Gradient Problem）现象导致的。如图2所示。这个现象产生的原因在于利用BP法后向求解梯度时有一个对激活函数求导数的操作，图2（a）中的，网络深度越深，在Input的地方中对操作累乘的次数就越多，导致值就很小。



图2（a） BP算法中的后向求导操作



图2（b） Vanish Gradient的影响

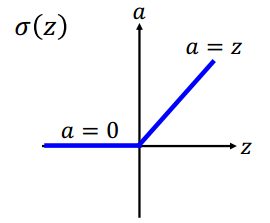
由于离Output近的layer求梯度时累乘的少，所以梯度（）较大，这些layer的参数就变化的快，而靠近Input的layer，中对操作累乘的次数很多，导致值就很小，参数学习就慢；这样导致的一个问题就是：当前面几层的参数还处于Random状态的时候，靠近Output的layer参数很快就收敛了，导致Loss function很快就收敛到一个local minimum，这样得到的性能就会很差。

**3 解决方案**

最早的方法就是分开Training各个层的参数，例如：先Training好第一层，再Training第二层，依次类推。但这种方法现在也基本不用了，因为比较麻烦，性能也并不一定提升很多。

目前比较主流的方法就是修改下激活函数。

1. **Rectified Linear Unit（修正线性单元）**

 **Reason**：

1）计算很快；（梯度为常数）

2）生物上理由；（这个可参考paper）

3）等同于无穷多叠加的结果；

4）可以hold住Vanishing Gradient Problem。

从这个函数形式就可以看出，当时，激活函数输出为0；当时，激活函数输出为；这样对整个网络就有两个影响：1）对于输出为0的节点，对网络是没有贡献的，这点可以认为是将整个网络稀疏化了，等效于减少了网络的参数；2）对于输出不为0的节点，输出就等于激活函数的输入，即输出是输入的线性函数，所以整个网络变成了一个又瘦又长的linear的网络，这样linear的网络就不会有“梯度递减”的问题了。那么有几个问题：

1. 激活函数替换为ReLU后，网络变成了linear的了么？

实际上，整体上的网络还是非线性的，因为不同的输入会导致不同的节点输入不一样，所以整体上还是non-linear的。

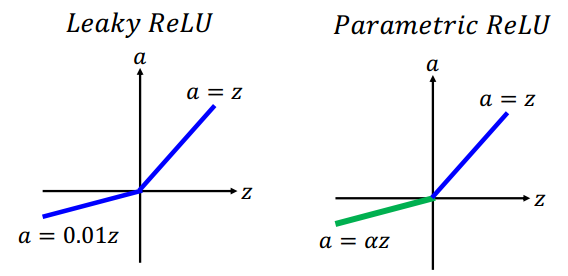
2）ReLU在0处不可微分，求导怎么求？

在实际的数据计算中，都是分段处理的，而且的情况极少遇见，即使遇见也就分段处理了。

3）会不会有参数没有Training到？

实际上也不会有参数不被Training到的情况，因为随着不同的训练数据输入后，每个节点的输入值都会不一样，总会有的情况出现，这个时候就Training到这个节点的参数了。

1. **ReLU - variant**



基本上就是对激活函数的输入的情况做一些改进，让其以一个固定的梯度（0.01）或者一个自学习得到的梯度进行参数学习。

1. **Max-out**

基本思想就是让这个activation function根据Training data自己学习出来，如图3所示：

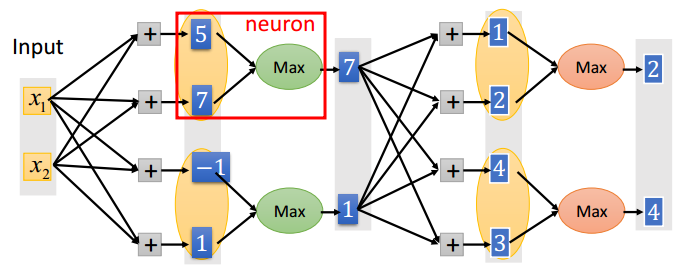


图3 Max-out实现

如上图，事先决定哪几个元素（不同的，例如5和7），分为一组，再将一组中的最大的值作为输出。实际上ReLU是Maxout中的一种特例，如图4所示：



图4（a）ReLU示意图 图4（b）Max-out示意图

将图4（b）中的输入数1的权重赋值为0即等价于图4（a），所以ReLU是Maxout中的一种特例。同时，后续将要介绍的CNN中的Pooling操作本质上就是一种Max-out，所以对CNN中如何利用梯度下降法做Pooling的问题在此迎刃而解。



2017.08.27.