Batch normalization分析

注：本文内容翻译自An Intuitive Explanation of Why Batch Normalization Really Works (Normalization in Deep Learning Part 1)

为简洁起见，下文中将batch normalization简写为BN

优点&缺点

优点：

1. BN使得训练过程中可以用更大的学习率，加速学习过程
2. 网络引入BN之后，可以利用sigmoid函数，使得容易出现的梯度淹没问题得到了解决。
3. 在此基础上出现了layer normalization和weighted normalization

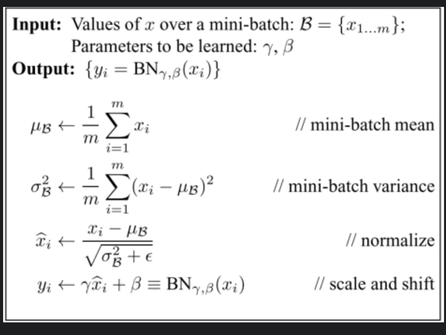
缺点：

虽然技术成熟，但在实践过程中也很容易出现问题。BN过程虽然简单，但其真正有效的原因比看起来复杂（并不只是协变量位移那么简单）。

本文旨在描述BN的工作原理，进而解答其之所以可以广泛应用的内在原因。

1. Covariate Shift（协变量位移）

Covariate shift一词出现在BN原文标题中，它是指学习算法中输入值分布的变化。这不是深度学习所独有的问题。例如当训练集和测试集数据源完全不同（比方说训练集来自网络爬图，测试集都是iphone拍摄的图像），那么二者分布就会不同。covariance shift之所以是一个问题，在于机器学习算法的行为会根据输入数据的分布而发生改变。
在机器学习上下文中，我们尤为关心网络内部节点输入的分布变化。神经网络在整个训练过程中会不断变化每层的权重。这就意味着每层激活函数输出也会变化。前一层的激活值即为后一层的输入。也就是说，在训练过程中，神经网络的每一层都面临着每步迭代其输入分布都会发生变化的局面。这样是又问题的，因为这就强迫每个中间层都需要不断地适应其输入变化的分布。
batch normalization最基本的思想就是用归一化每一层的activations来限制covariate shift（将每层输入变为均值为0，方差为1的数据分布）。这样，理论上使得每层的输入具有更稳定的分布，由此可以加速神经网络的训练过程。
实际应用过程中，如果将每层的activations都严格变为均值为0，方差为1的数据分布会限制网络的表达能力。由此，在实践中，BN允许网络学习合适其自身的网络参数β和γ，



原始的BN论文

deeplearning.ai讲座

2. 高阶影响

（待续）

BN的局限性

最关键的局限在于其取决于mini-batch。BN过程是按照每个mini-batch计算均值和方差，并将特征数据按照mini-batch数据归一化。这就意味着每个mini-batch的均值和方差不同。这种依赖会带来两个问题：

batch size不能太小

很明显，batch size为1时，无法用BN。但当batch size稍大一点儿的时候也会出问题。理论上我们想用全局的均值和方差归一化输入数据，但是每次迭代都针对整个数据库计算均值和方差，计算量太大，因此改为mini-batch形式估计均值和方法。这种简单的均值和方差估计意味着结果中有一定量的误差，并且在每个mini-batch是不同的。更小的batch size会加大这些估计的方差，也就意味着在利用BN+SGD时要注意batch size。

难以在RNN网络的recurrent connection中应用