如何理解BN？|Batch Normalization原理剖析

1. 思路

Batch Normalization，简称为BN，即批量归一化。其思路来源于**白化**。白化指的是对输入数据分布变换到零均值，单位方差的正态分布——如此，借用这个思想，神经网络便会快速收敛。

2. 原理

在机器学习中， 所有的数据都假设遵循独立且分布(i.i.d)，即假设训练数据和测试数据都满足相同的分布，这是通过训练数据获得的模型能够在测试集上获得良好性能的一个基本保障。而由于DNN在做非线性变换前的激活输入值（x=WU+B）随着网络深度加深或者在训练过程中，其分布逐渐发生偏移(shift)，之所以训练收敛慢，一般是整体分布逐渐往非线性函数的取值区间的上下限两端靠近（梯度相乘累计），所以这就导致反向传播时底层神经网络的梯度消失，这便是训练DNN收敛越来越慢的本质原因。BN的作用就是在DNN训练过程中，通过一定的归一化手段，把每层神经网络任意神经元这个输入值的分布强行拉回到均值为0方差为1的标准正态分布，使得**每一层神经网络的输入均能都保持相同的分布，同时也使得激活输入值落在非线性函数对输入比较敏感的区域，从而让梯度变大，避免梯度消失的问题。而梯度变身就意味着学习收敛速度更快，因此能同时大大的加快训练的速度。**

3. 训练

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: γ, β
Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma\widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

求一个批次所有激活值的**均值**
求一个批次所有激活值的**方差**
对该批次的激活值进行**归一化**

最后再进行**平移**和**缩放**

经过归一化后某个神经元的激活x形成了均值为0，方差为1的正态分布，目的是把值向线性区拉动，增大导数值，增强反向传播信息的流动性，加快训练收敛速度。但是，如此一来会导致网络的表达能力下降，为了防止这一点，每个神经元增加两个可调节参数（scale和shift），这两个参数是通过训练来学习得到的，用来对变换后的激活x做一个反变换，使得网络的表达能力增强，相当于做一个线性和非线性的trade-off。

4. 推理

BN，顾名思义是根据Batch来进行激活值的调整。但是在推理阶段，我们仅输入一个实例，此时无法算实例集合求出均值和方差。一个方法就是利用从所有的训练实例中所获得的的全局统计量，即均值和方差。训练完成后，模型会存储每个Mini-Batch的均值和方差统计量，我们在推理阶段对这些统计量求相应的数学期望即可获得全局统计量：

$$E[x] \leftarrow E_{\mathrm{B}}[\mu_{\mathrm{B}}]$$

$$Var[x] \leftarrow \frac{m}{m-1} E_{\mathrm{B}}[\sigma_{\mathrm{B}}^2]$$

算好了全局统计量，每个隐层神经元也已经有相应训练好的Scale和Shift参数，这样我们便可以得到输出y：

$$y = \frac{\gamma}{\sqrt{Var[x]+\varepsilon}} \cdot x + (\beta - \frac{\gamma \cdot E[x]}{\sqrt{Var[x]+\varepsilon}})$$

5. 总结

线性区可以加速模型的收敛速度，非线性区可以增强模型的表达能力，BN在线性区和非线性区做了一个很好的trade-off。
BN可以增大梯度，避免梯度消失的问题，防止模型出现过拟合。