BN(Batch Normalization)是深度学习中在训练神经网络时用到的一种技巧。为的是解决内部协方差（Internal Convariate shift）偏移的问题。当神经网络输入值的分布差异较大时（输入特征的scale差异较大），与Weights进行矩阵相乘之后，不同特征的scale差异较大，而深度学习网络需要通过训练不断更新完善，那么差异值产生的些许变化都会深深影响后层，偏离越大表现越为明显；因此，对于反向传播来说，这些现象都会导致梯度发散，从而需要更多的训练步骤来抵消scale不同带来的影响，也就是说，这种分布不一致将减缓训练速度。

* **BN的基本原理**

BN的思路是调整各层的激活值分布，使其具有适当的广度，使每一层神经网络的输入在深度神经网络训练中保持相近分布，其优点为  
1.可以使学习快速进行（可以增大学习率）  
2.不那么依赖初始值（对初始值没那么敏感）  
3.抑制过拟合（降低Dropout等的必要性）

图示, 工程绘图, 示意图

描述已自动生成

文本

低可信度描述已自动生成

这里， γ和β是参数。一开始γ = 1， β = 0，然后再通过学习调整到合适的值。

* BN放在什么位置

具体BN放的位置没有太多标准，可以是放在激活函数之前，也可以放在激活函数之后，总之没有太多限制。

* BN在训练和测试阶段有什么不同

对于BN,在训练时，是对每一批的训练数据进行归一化，即用每一批数据的均值和方差，而在测试阶段，如进行一个样本的观测，并没有batch的概念，因此这个时候用的均值和方差是全部训练数据的均值和方差，可以通过移动平均法求得。

对于BN,当一个模型训练完成后，它所有的参数都确定了。

BN为什么能防止过拟合

同样一个样本的输出不再取决于样本本身，也取决于跟这个样本属于同一个mini-batch的其他样本，同一个样本跟不同的样本组成一个mini-batch，他们的输出是不同的（仅限训练阶段），可以理解成数据增强，贯穿数据流过整个神经网络，相当于神经网络的每一层的输入都被数据增强处理了。

简单理解来说，BN就是对每一批数据归一化到相同的分布，而每一批数据的均值和方差都有一定的差别，而不是固定的值，这个差别增加了模型的鲁棒性，一定程度上减少过拟合。