背景

神经网络学习过程本质就是为了学习数据分布，一旦训练数据与测试数据的分布不同，那么网络的泛化能力也大大降低；另外一方面，一旦每批训练数据的分布各不相同(batch 梯度下降)，那么网络就要在每次迭代都去学习适应不同的分布，权重也不断更新，导致激活层输出的分布一直变化（即 Internal Covariate Shift），这样将会大大降低网络的训练速度以及增加网络过拟合的风险。

BatchNormalization 解决的正是每一层数据部分的问题，将每一层网络在输入到下一层之前进行归一化，使得数据被强制在同一分布下。这样可以保证在训练的过程中每一层的输入数据分布稳定，反传时也可以避免落入饱和区加速了网络收敛。

思路总结

①对比 mini-batch SGD 和 SGD

SGD 是每次只随机使用一个样本迭代，而单个样本不能代表全体样本的数据分布，而 min-batchSGD 每次在一个 batch 上进行优化迭代，可以实现并行化提高效率

②数据的白化操作可以解决 Internal Covariate Shift 的问题

白化操作不是处处可微且需要对整个训练集的数据进行分析因此代价大，计算复杂。因此作者提出了一个可微的且在每次参数更新后不需要对整个数据集进行参数计算的方法。

③将数据分布强行在同一分布下破坏了学习到的特征图的原始分布

为了解决这个问题，又加入了两个可学习的参数 γ 和 β 来恢复原始的某一层所学到的特征。

个人感觉是结合了 mini-bacth 处理的思想和白化的思想，同时增加了两个调节参数保证网络的表达能力提出了 BatchNormalization

方法

训练过程：

假设 mini-batch 里有 m 个样本 $\mathrm{B}=\{x_1,\cdots,x_m\}$，且每个样本有 d 维，$x=(x^{(1)},\cdots,x^{(d)})$，对这 m 个样本在维度方向计算均值 μ_{B} 和标准差 σ_{B}

$$\mu_{\mathrm{B}}=\frac{1}{m}\sum_{i=1}^{m}x_i \tag{1}$$

$$\sigma_{\mathrm{B}}=\frac{1}{m}\sum_{i=1}^{m}\left(x_i-\mu_{\mathrm{B}}\right)^2 \tag{2}$$

将原始输入 x 进行标准化处理：

$$\hat{x}_i = \frac{x_i - \mu_{\mathrm{B}}}{\sqrt{\sigma_{\mathrm{B}}^2 + \varepsilon}} \tag{3}$$

然后对 $\hat{x}_i$ 进行缩放平移操作到新的分布 y 中：

$$y_i = \gamma \hat{x}_i + \beta \tag{4}$$

网络只需要学习参数 γ 和 β 来控制数据的分布

预测过程：

均值和方差是把每个 mini-batch 的均值和方差统计量记住，然后对这些均值和方差求其对应的数学期望，每层神经元的参数 γ 和 β 通过训练得以固定，这样均值、方差、参数 γ 、β 四个值都是固定的就可以直接推理了。