人工神经网络 HW3 实验报告

计 52 路橙 2015010137

Part 1：综述

本次实验中，我用 TensorFlow 实现了 BN（Batch Normalization,）层与 CNN、MLP 的结合，并比较了 CNN 与 MLP 的区别，初步探讨了 BN 层对训练的影响。

关于 BN 层的实现，调用了 TensorFlow 的部分函数，并自己实现了均值、方差的动态更新和 one by one 策略。

Part 2：带 BN 层的 CNN 与 MLP 的比较

① **参数数量基本相同时：**

选择 CNN 网络为 kernel_size=5，channel 为(1,30)，(30,30)的两个卷积层，而 MLP 网络选择 784*50 的隐含层。

CNN 的参数个数主要由 kernel_size 和 channel_size 决定，对于 kernel_size=5，channel_size=30 的情况下，CNN 的参数个数为：

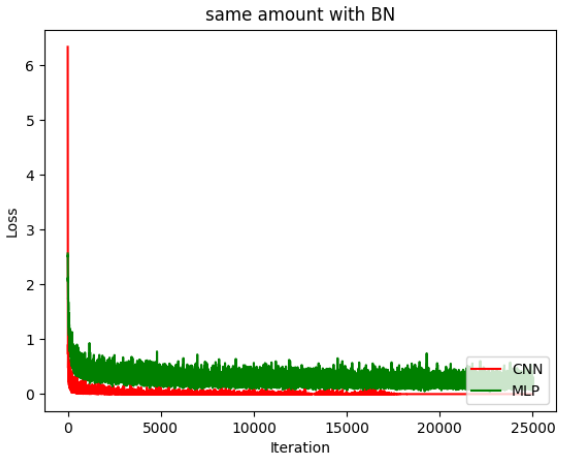
$$(1 \times 30 \times 5^2 + 30) + (30 \times 30 \times 5^2 + 30) + (7*7*30*10 + 10) = 38020$$

而 MLP 参数个数为：

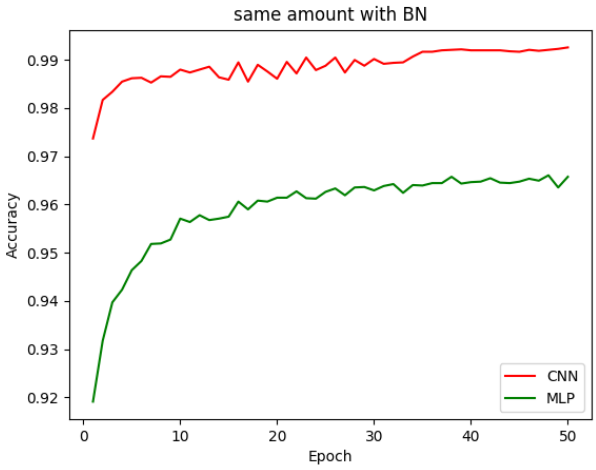
$$(784 \times 50 + 50) + (50 * 10 + 10) = 39760$$

此时，二者参数数量基本相等。当带有 BN 层进行训练时，二者结果如下：

每个 Iteration 的 Loss：



每个 Epoch 在 valid 集上的准确率：



参数选择：

- **Learning Rate:0.001**
- **Learning Rate Decay Factor:0.9995**
- **Weight variable 中 Stddev:0.1**

可以发现，在参数数量基本相等时，带有 BN 层后，CNN 与 MLP 有如下区别：

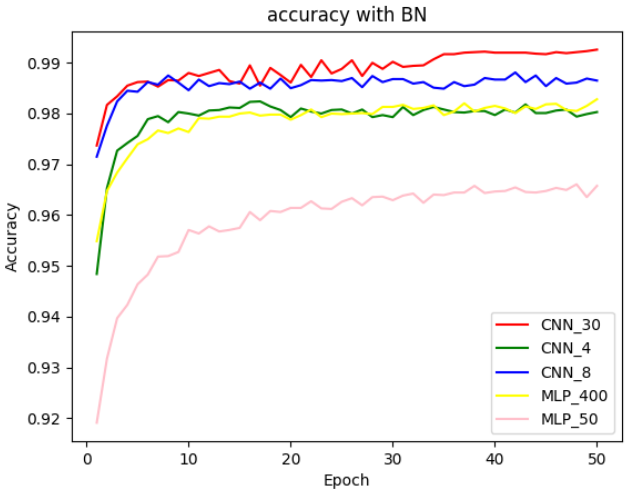
- CNN 的 loss 下降更平缓，而 MLP 的 loss 抖动很大。
- CNN 的准确率比 MLP 可以高出 3%左右，且收敛速度明显快于 MLP。这一方面可以说明，同等参数下，CNN 更易于收敛；另一方面也说明，CNN 中的参数比 MLP 中参数的耦合性小，因而同等参数下能表达的特征更多。

更进一步地，为了刻画 CNN 与 MLP 的区别，我做了不同参数的带有 BN 层的 CNN 与 MLP 的实验，对比如下：

② **不同参数下 CNN 与 BN 的比较：**

如下图所示，对于 CNN 的两个卷积层，设他们的 shape 为(1, c),(c, c)，则对于 c 取值为 4,8,30。取 kernel_size 都为 5，我分别做了 3 组对比实验。 而对于 MLP 的隐含层大小为(784, h)，我选取 h 为 50,400，做了 2 组对比实验。

以上五组对比实验的参数选择与①中相同，且都带有 BN 层。



网络参数数量表

网络类型	CNN_4	CNN_8	CNN_30	MLP_50	MLP_400
参数数量	2478	5746	38020	39760	318010

分析如下：

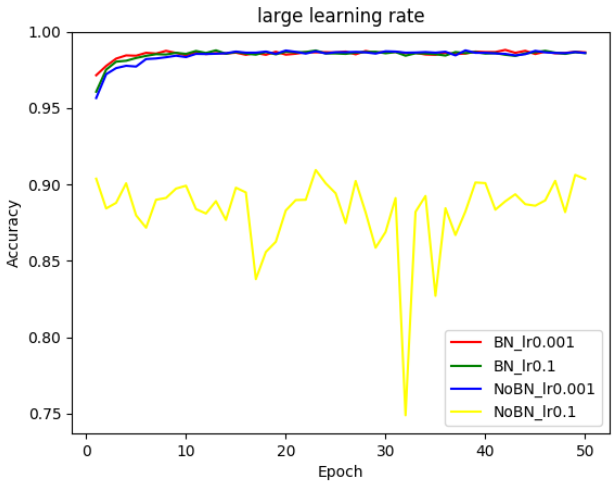
- 由图可知，以上五组实验中，CNN 的准确率都比 MLP 的准确率高，且收敛速度都快于 MLP，更重要的是，参数数量都少于 MLP。
- 观察绿线和黄线，分别对应 CNN_4 和 MLP_400，前者参数数量只有 2478，但后者参数数量高达 318010，二者比例约为 2478/318010 = **0.78%**，即 MLP 只有当参数数量达到超过 CNN 很多个数量级后才可以达到和 CNN 一样的训练准确率，而尽管如此，CNN 的收敛速度明显快于 MLP，因此 CNN 即使在很小的参数数量下也可发挥出比 MLP 更优的性能。
- **这说明，MLP 中很多参数实际上都是有很大相关性的，即 MLP 的参数用高维数组描述了一个低维特征，其中有很多维度都互相有关联，并不是本质的刻画。而 CNN 用远少于 MLP 参数个数的参数刻画了同样的特征，甚至准确率高于 MLP，这一方面可以说明原理层面上的改变比参数的调整更重要，另一方面也可以说明 CNN 在图像处理领域的强大。**

Part 3：BN 层的影响

① **CNN 中 BN 层的影响：**

1) 当初始 learning rate 较大时（初始梯度爆炸）：

选择网络为 c=8 时的 CNN（详见 Part2.②，其余参数都与该实验相同）。分别选取 learning rate 为 0.001 和 0.1，带 BN 层与不带 BN 层，共 4 个实验。

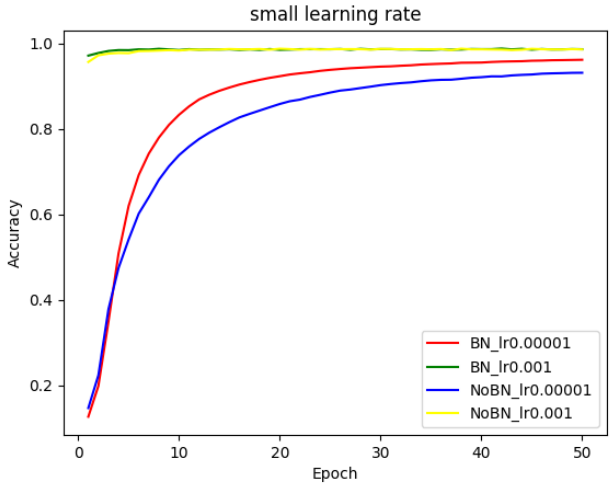


由上图可知，在 learning rate 适中（0.001）时，带或不带 BN 层的网络都可以很快收敛，且准确率可以达到很高。然而，当 learning rate 初始较大（0.1）时，不带 BN 层的网络由于梯度爆炸而无法收敛，甚至波动幅度很大，但带 BN 层的网络却可以平稳快速地收敛到较高的准确率上。

因此，BN 层在解决梯度爆炸问题上有明显的作用——通过 BN 层，将爆炸的梯度缩小至可以使网络稳健收敛的梯度。

2) 当初始 learning rate 较小时（初始梯度弥散）：

选择网络为 c=8 时的 CNN（详见 Part2.②，其余参数都与该实验相同）。分别选取 learning rate 为 0.001 和 0.00001，带 BN 层与不带 BN 层，共 4 个实验。

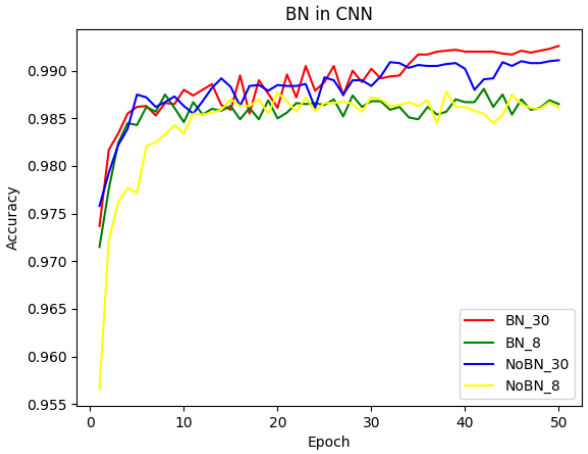


由上图可知，在 learning rate 适中（0.001）时，带或不带 BN 层的网络都可以很快收敛，且准确率可以达到很高。然而，当 learning rate 初始较小（0.00001）时，不带 BN 层的收敛速度会较慢，而带 BN 层的收敛速度快于不带 BN 层的收敛速度。尽管效果仍然不如 learning rate 为 0.001 时的情形，但加 BN 层的效果依然强于不带 BN 层的效果，收敛速度和最终准确率都较高。

因此，BN 层在解决梯度弥散问题上也有明显的作用——通过 BN 层，将很小的梯度集中到某一个方向上，从而使网络稳健地收敛。

3) 不同 Channel size 时：

选择网络为 c=8、c=30 时的 CNN（详见 Part2.②，其余参数都与该实验相同）。分别选取带 BN 层与不带 BN 层，共 4 个实验。结果如下图：

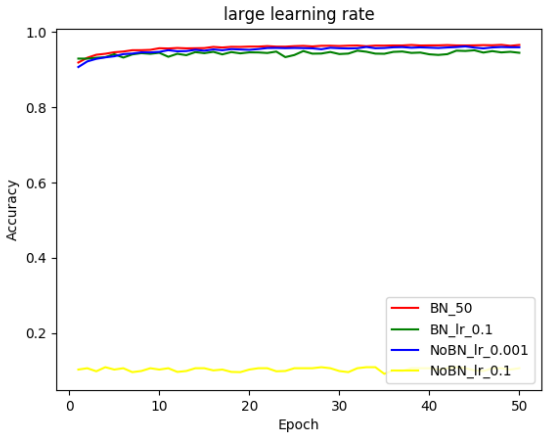


由上图可知， 在其他参数一定时，BN 层对 CNN 的最终收敛准确率并没有太大影响。但当 kernel size 较小时，BN 层对 CNN 的影响较明显：带有 BN 层的 CNN 比不带 BN 层的 CNN 有更快的收敛速度。

② **MLP 中 BN 层的影响：**

1) 当初始 learning rate 较大时（初始梯度爆炸）：

选择网络为隐含层 h=50 时的 MLP（详见 Part2.②，其余参数都与该实验相同）。分别选取 learning rate 为 0.001 和 0.1，带 BN 层与不带 BN 层，共 4 个实验。

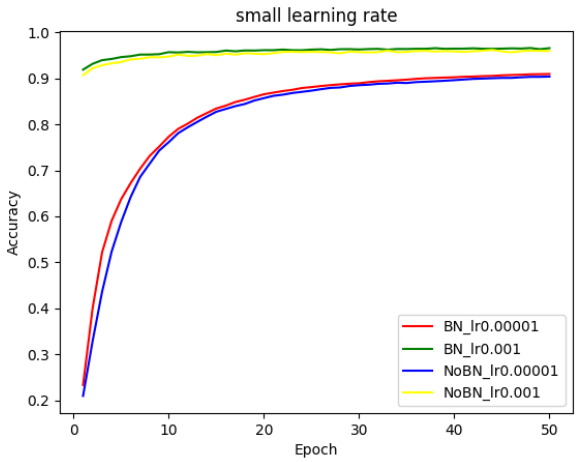


由上图可知，在 learning rate 适中（0.001）时，带或不带 BN 层的网络都可以很快收敛，且准确率可以达到很高。然而，当 learning rate 初始较大（0.1）时，不带 BN 层的网络由于梯度爆炸而完全无法收敛，但带 BN 层的网络却可以平稳快速地收敛到较高的准确率上。

因此，与 CNN 相同，BN 层在 MLP 上，在解决梯度爆炸问题上也有明显的作用——通过 BN 层，将爆炸的梯度缩小至可以使网络稳健收敛的梯度。

2) 当初始 learning rate 较小时（初始梯度弥散）：

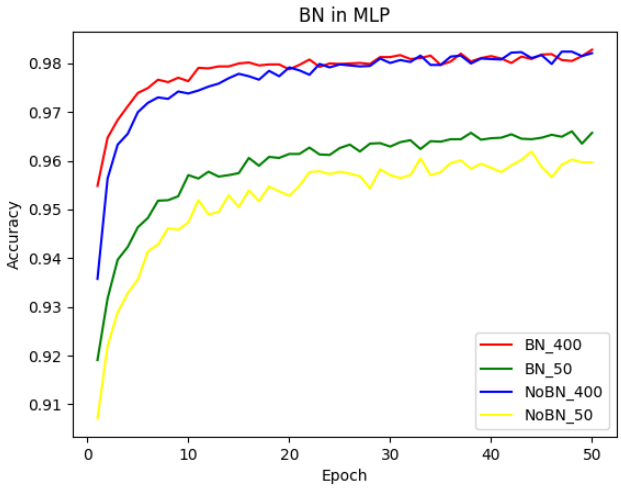
选择网络为隐含层 h=50 时的 MLP（详见 Part2.②，其余参数都与该实验相同）。分别选取 learning rate 为 0.001 和 0.00001，带 BN 层与不带 BN 层，共 4 个实验。



由上图可知，在 learning rate 适中（0.001）时，带或不带 BN 层的网络都可以很快收敛，且准确率可以达到很高。但当 learning rate 较小时，BN 层对于 MLP 的加速收敛作用不明显。猜测是因为单隐含层的 MLP 中线性度依然很强，BN 的作用更利于影响非线性分布，对线性的影响变化不大。

3) 不同隐含层大小时：

选择网络为隐含层 h=50,400 时的 MLP（详见 Part2.②，其余参数都与该实验相同）。分别选取带 BN 层与不带 BN 层，共 4 个实验。



由上图可知，在 MLP 中，当网络规模较大时，BN 层只会加速收敛；而当网络规模较小时，加了 BN 层后既可以加速收敛，也可以使最终准确率提升。

Part 4：One By One 测试

	逐 batch 测试准确率	One By One 测试准确率
CNN, channel = 30	0.9926	0.9925
CNN, channel = 8	0.9870	0.9654
MLP，隐含层 784*400	0.9807	0.9724
MLP，隐含层 784*50	0.9658	0.9014

由上表可以看出，在 one by one 测试中，准确率相对于逐 batch 测试都会有所降低。这是因为，One By One 测试中使用的均值和方差为训练数据的估计值，而非 test 集的真正的均值和方差，从而有一些偏移，会造成一定的误差。而当逐个 batch 测试时，所用的均值和方差便为一个 batch 内数据的均值和误差，误差影响会远小于 one by one 测试。