

Artificial Neural Network HW2

刘泓尊 2018011446 计 84

Department of Computer Science, Tsinghua University

2020 年 10 月 18 日

目录

1 Network Structure and Hyperparameters	1
1.1 MLP	1
1.2 CNN	2
2 Experiment Result	2
2.1 Overview	2
3 Questions	2
3.1 Explain how ‘self.training’ work, Why should training and testing be different? . .	2
3.2 Explain why training loss and validation loss are different. How does the difference help you tuning hyper-parameters?	3
3.3 Compare difference between MLP and CNN	3
3.4 Effect of Batch Normalization	3
3.5 Effect of Dropout	4
4 Summary	4

1 Network Structure and Hyperparameters

1.1 MLP

网络结构: $(32 \times 32 \times 3) \text{Linear} 784 \rightarrow \text{BN} \rightarrow \text{ReLU} \rightarrow \text{Dropout}(0.5) \rightarrow 784 \text{Linear} 10 \rightarrow \text{CELoss}$
权重初始化均为默认

超参: $lr = 1e^{-3}, weight_decay = 0.0$

您可以在项目根目录下运行如下命令复现我的结果:

```
1 python mlp/main.py --num_epochs 50 --drop_rate 0.5 --data_dir ./cifar-10
   _data
```

1.2 CNN

网络结构: Conv(300x5x5) → BN → ReLU → Dropout(0.5) → MaxPool(2) → Conv(300x3x3) → BN → ReLU → Dropout(0.5) → MaxPool(2) → (300x6x6)Linear10 → CELoss

权重初始化均为默认

超参: $lr = 1e^{-3}$, $weight_decay = 0.0$

您可以在项目根目录下运行如下命令复现我的结果:

```
1 python cnn/main.py --num_epochs 50 --drop_rate 0.5 --data_dir ./cifar-10_data
```

2 Experiment Result

实验环境: NVIDIA TITAN Xp, CUDA Version: 10.1

2.1 Overview

下面是不同模型和参数配置的结果。更多的对比和曲线图请见[3.3](#)等小节。

Model/Metrics	Valid Loss	Best Valid Acc	Final Test Loss	Final Test Acc	Time(s)/epoch
MLP+Dp(0.5)+BN	1.985	56.21(@epoch=116)	1.928	55.60	3.82
MLP+BN	1.363	54.43(@epoch=63)	1.334	54.70	3.79
MLP+Dp(0.5)	1.540	54.31(@epoch=42)	1.511	54.58	3.63
CNN+Dp(0.5)+BN	0.720	78.51(@epoch=142)	0.730	77.78	34.78
CNN+BN	1.252	78.63 (@epoch=41)	1.255	78.08	28.44
CNN+Dp(0.5)	0.883	74.96(@epoch=41)	0.893	73.21	29.87

3 Questions

3.1 Explain how 'self.training' work, Why should training and testing be different?

当调用 model.train() 的时候,PyTorch 会把模型的 self.training 参数设为 True;当调用 model.eval() 的时候, self.training = False.

self.training 可以指示处于训练阶段还是测试阶段, 主要是对 BN 层和 Dropout 层有影响。

在 train 阶段, BN 层会计算当前 batch 的 mean 和 var, 并对训练样本的 mean 和 var 进行移动平均; Dropout 层则会使比率为 p 的神经元随机失活, 并将权重放大到 $1 / (1-p)$;

在 eval 阶段, BN 层会固定 mean 和 var, 即根据训练样本得到的 mean 和 var 作为测试样本的 mean 和 var. Dropout 层则等价于恒等变换 (在我的实现中), 使用到了所有的权重。

训练阶段进行数据归一化的同时需要统计数据的 mean 和 var, 用于测试阶段, Dropout 的引入减少了过拟合的现象; 但测试阶段需要固定 BN 层和 Dropout 层, 这保证了测试结果的确定性, 让测试数据分布和训练数据分布更接近, 也不会泄露任何测试数据的信息, 保证了模型的泛化能力。

3.2 Explain why training loss and validation loss are different. How does the difference help you tuning hyper-parameters?

因为模型会一直针对 train loss 进行权重更新，优化目标是最小化 train loss，所以在训练过程中 train loss 会一直下降。

但是 train set 和 valid set 的数据分布不可能完全一致，模型的 train loss 下降过程中，泛化能力会变弱，导致 valid loss 先下降后上升，也就是所谓“过拟合”。Dropout 的作用便是将网络解耦，提高模型的泛化能力；BN 层通过归一化，也达到了让测试集分布和训练集分布更接近的效果，当然同时还会加速收敛。

通过观察 valid loss 的变化趋势，可以尽早发现过拟合，确定最佳的训练停止步数，即采用 early stop 策略，以使得 test set 的性能更优。

此外，可以通过观察不同 droprate 等超参对 valid loss 的影响，确定最佳的超参组合，比如在本实验中我通过对比 droprate=0.0, 0.1, 0.5, 0.9 的 valid loss 曲线，最终确定 droprate=0.5 是一个较优的选择；

此外，观察 train loss 也可以判断模型的收敛速度和结构的正确性。

3.3 Compare difference between MLP and CNN

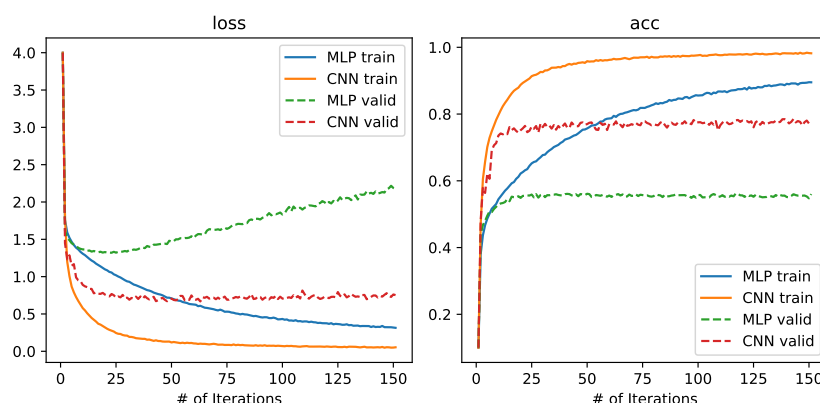


图 1: MLP 和 CNN 的对比, Dropout=0.5, with BN

从上图中可以看到，测试集准确率上 CNN 的效果大大好于 MLP，有 22% 的提升。CNN 收敛速度也比 MLP 更快，过拟合程度更低。当然，CNN 每个 epoch 的时间比 MLP 要长。

我认为 CNN 的效果好主要是因为 CNN 通过卷积操作捕捉了图像的 2 维信息，而 MLP 只能对 1 维数据进行特征提取。而且 CNN 更擅长捕捉图片的局部信息，而分类任务往往就是某个局部决定了图像类别；

同时 CNN 的平移不变性保证了：在图片的任意位置出现物体，都可以被正确识别，这就让模型泛化能力更强；而 MLP 需要在每个位置上学习权重，难以兼顾。权值共享也使得 CNN 在较少的参数量下就可以获得较高的性能和更快的收敛速度。

3.4 Effect of Batch Normalization

图2表示了有无 BN 对 loss 和 acc 的影响：从图中可以看到，BN 的存在会加快模型在训练集上的收敛速度，提升了训练的稳定性，在 CNN 上效果更为显著；BN 通过归一化操作让每层的输入分布拉回到原点附近，缓解了梯度消失的问题，从而加快收敛速度。

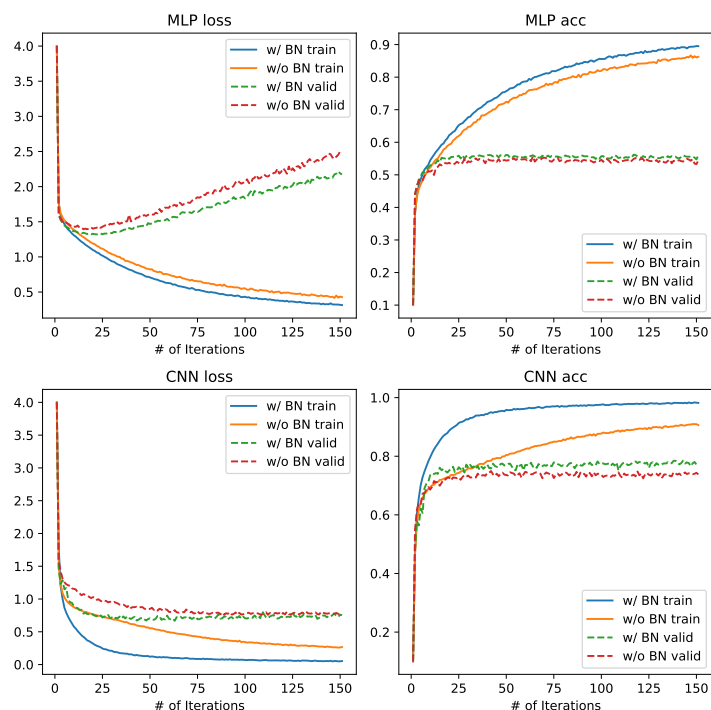


图 2: 有无 Batch Normalization 对 MLP 和 CNN 的影响

BN 的存在还会让模型更加鲁棒, 因为 BN 让每一批数据都归一化到 0 均值和单位方差, 一定程度上拉近了训练数据和测试数据的距离, 提高了泛化能力。

3.5 Effect of Dropout

图3和图4展示了不同 Dropout rate 对 loss 和 acc 的影响: 从 train loss 曲线可以看到, dropout rate 越大, 模型的收敛速度越慢, 因为模型耦合度减小, 需要学习更多的部分。如果 dropout rate 过大, 模型可能难以收敛到很优的 train loss。

dropout 是一种正则化手段, 通过忽略一部分输出来减少过拟合现象, 从而提高模型的泛化能力。从图中可以看到, 不使用 dropout 时模型在 10 个 epoch 左右便出现了过拟合, 之后 valid loss 便迅速上升, 而且训练很不稳定。而 droprate=0.5 是一个合适的参数, 它既保证了模型的收敛速度, 又可以在测试集上达到较高的准确率。

4 Summary

本次实验我深入理解了 BN 和 Dropout 的实现原理, 对 PyTorch 深度学习框架有了更熟练的掌握。掌握了减少模型过拟合的基本方法, 锻炼了通过观察曲线调参的能力。

感谢老师和助教的悉心指导!

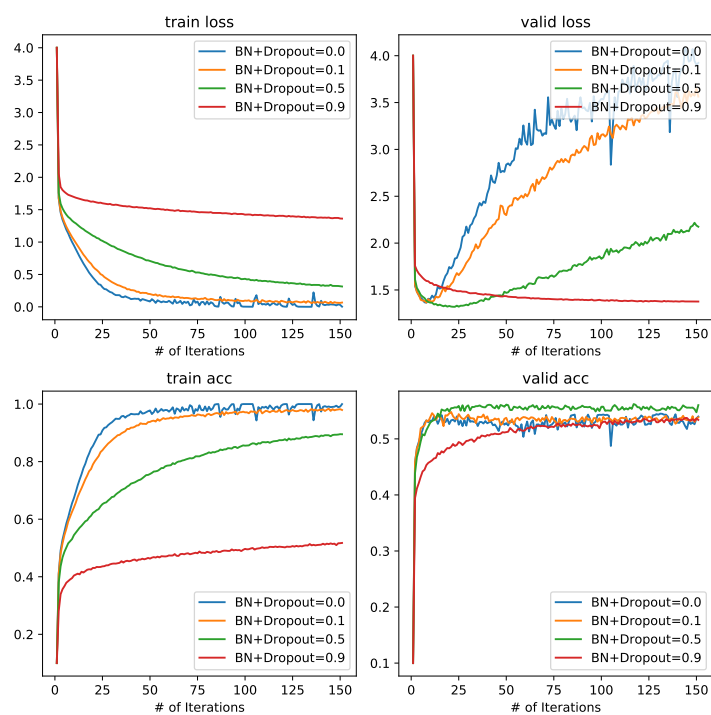


图 3: 不同 Dropout rate 对 MLP 的影响

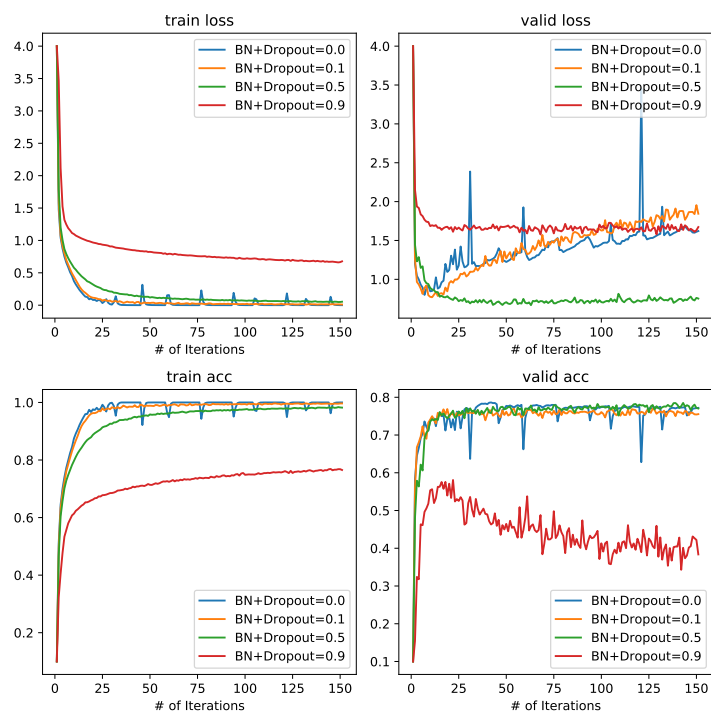


图 4: 不同 Dropout rate 对 CNN 的影响