Stochastic Weight Averaging

作者：纽约的自行车

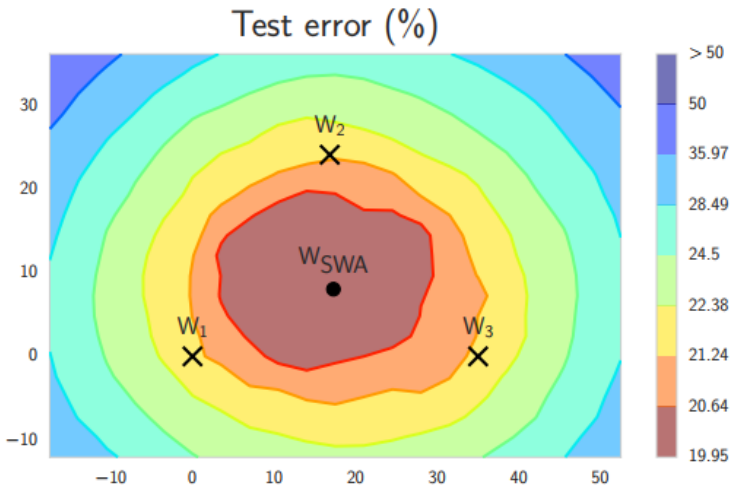
日期：2022 年 10 月 12 号

摘　　要

本文介绍一种模型参数平均方法，Stochastic Weight Averaging（SWA，随机权重平均），首先介绍该方法的灵感来源，然后介绍原理，最后给出 pytorch 中的使用方法。

1　灵感来源

常规方法训练模型时存在一个问题：局部最优解通常在最优损失面边缘。这个问题是 Pavel Izmailov [1] 在实验中发现的，并没有具体的理论依据。下面图中，显示低损失的红色区域上的点 W_1、W_2 和 W_3 是局部最优解，都处在最佳损失面边缘，中心点 W_{swa} 表示最优解。根据 W_1、W_2 和 W_3 获得 W_{swa} 便是我们希望的。



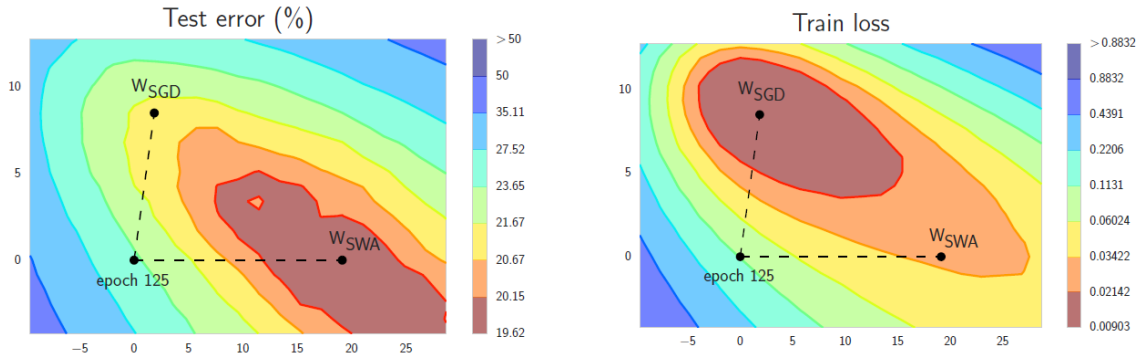
2　SWA 介绍

2.1　解决方法

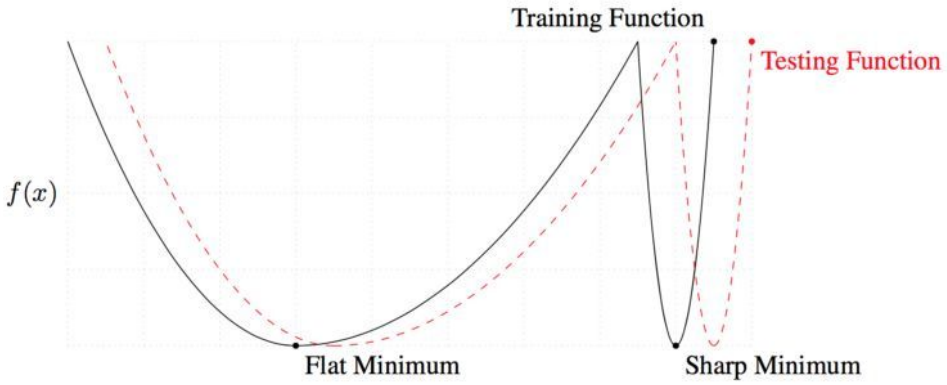
SWA [1] 是一个很朴素的想法，通过对上图中几个这样的点取平均，很有可能得到一个甚至更低损失的、全局化的通用解，即图中的 W_{swa}。这就是 SWA。。。如何对这些点取平均将在 2.2 节介绍。

实验中发现一个有意思的地方就是 SWA 在训练集上的损失比常规方法大，但是在测试集上错误率更低。如下图，点 $epoch125$ 表示用常规方法训练 125 个 epoch 后到达的收敛点，W_{SGD}

表示从 epoch 125 开始继续用常规方法训练直至收敛，W_{SWA} 表示从 epoch 125 开始继续用 SWA 方法训练直至收敛。右图中显示 SWA 方法在训练集上损失更大，左图中显示 SWA 在测试集上错误率更小。



对上述现象解释是训练集和测试集分布不一致。由于采样或者随着时间迁移在线数据分布发生变化，导致训练集和测试集（在线数据）分布存在一定的偏差，在小数据集中尤为明显。分布不一致带来的影响是训练数据和测试数据会产生类似的但并不完全一样的损失面，如下图所示。



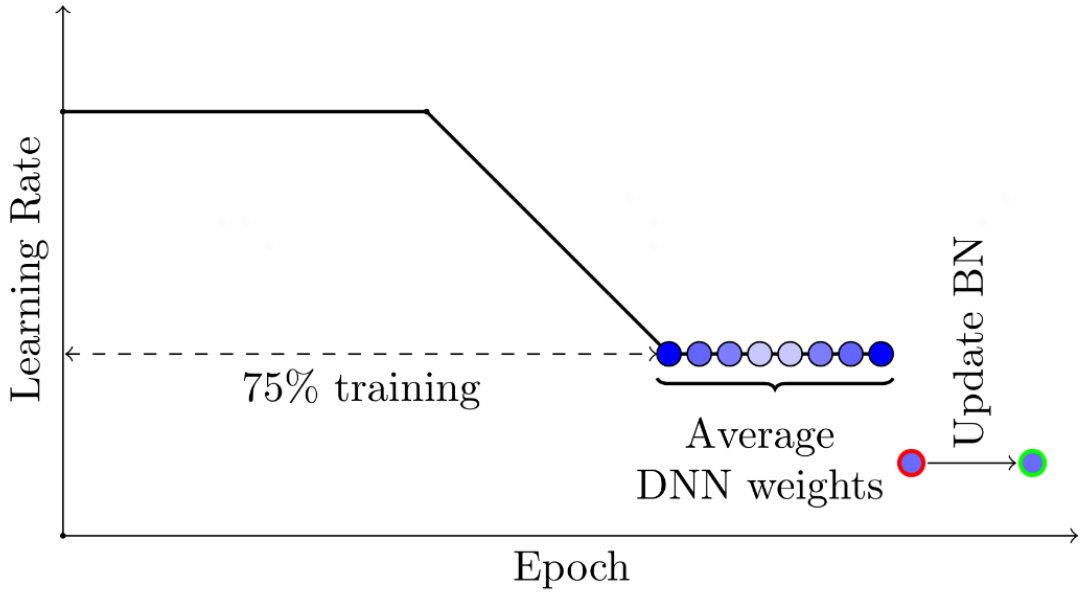
由于损失面的差异，对于一个训练集的局部最优解，在测试集损失面上会给出一个高损失值。上图中黑色点 sharp minimum 其在纵坐标上对应的红色虚线几乎在最高点，这表示测试集损失很大。

总结一下就是 SWA 在测试集上鲁棒性更好。

2.2 SWA 原理

如何将各个局部最优解取平均？一种方法是训练多次，每次得到一个局部最优解，并保存模型，预测时对各个模型输出结果取平均。这种方法缺点是需要保存多个模型，预测时推理计算量也大。SWA 用了一种更简单的方法，达到了和上述方法类似的效果。SWA 没有保存每个局部最优解的模型，而是将各个局部最优解模型的参数取平均，最终只得到一个模型。

如下图所示，在前 75% 的训练时间中和常规方法相同，后 25% 时间中保持学习率不变，模型参数照常更新，但是每隔 N 个 step 截取模型参数，并且和之前截取的模型参数取平均。



模型参数平均方式如下式所示：

$$W_{swa} := \frac{W_{swa} \cdot n_{models} + w}{n_{models} + 1}$$

训练过程红 SWA 不需要集成很多模型，只需要两个模型。第一个模型存储模型权重的平均值（公式中的 w_{swa}），这就是训练结束后的最终模型，用于预测。第二个模型（公式中的 w）变换权重空间，利用循环学习率策略找到最优权重空间。

2.3 Pytorch 实现

在常规方法上只需添加 3 行代码即可 [2]。

```
from torchcontrib.optim import SWA  # 导入包

...
...

# training loop
# 初始化优化器，只要是torch.optim中的就行
base_opt = torch.optim.SGD(model.parameters(), lr=0.1)
# 对优化器包装
opt = torchcontrib.optim.SWA(base_opt, swa_start=10, swa_freq=5, swa_lr=0.05)
for _ in range(100):
     opt.zero_grad()
     loss_fn(model(input), target).backward()
     opt.step()
opt.swap_swa_sgd()  # 计算参数平均
```

参考文献

[1] “Averaging Weights Leads to Wider Optima and Better Generalization”. In: ().

[2] “https://pytorch.org/blog/stochastic-weight-averaging-in-pytorch/”. In: ().