2022 Deep Learning Final Project

310706034 吳啓玄

1. 動機與背景描述

深度學習模型的泛化性 (Generalization)一直是這個領域追求的目標，模型泛化性的好壞，會影響到模型在測試集上的表現，那麼該如何提升深度學習模型的泛化性呢? 這可以從 Optimizer 的角度出發，如果 Optimizer 能讓參數在訓練時收斂在 Flat Minimum 的位置，則相對收斂在 Sharp Minimum，前者會有較小的 Testing Loss (如下圖 1)。

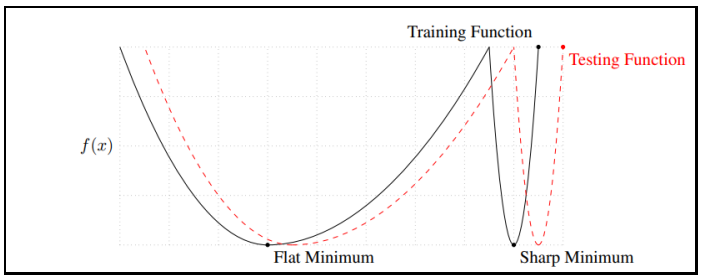

圖 1: A Conceptual Sketch of Flat and Sharp Minima. The Y-axis indicates value of the loss function and the X-axis the variables (parameters) (Nitish Shirish Keskar et al., 2017)

常見的 optimizer 包括 SGD、RMSProp、Adam，這些 optimizer 有各自的優點，舉 Adam 為例， Adam 是廣為人知的 optimizer，在 RMSProp 的基礎上結合了 momentum 的概念，並且透過額外的超參數 (beta1、beta2、epsilon) 使其穩定，在初始化或是連續遇到小的 gradient 時，可以讓參數更新更穩定，收斂速度較快，然而，Adam 在測試集資料的表現上並不是特別優秀 (Liangchen Luo et al.,2019)，如圖 2 與圖 3 (Liangchen Luo et al., 2019)，但論文內並未指出 Adam 總是收斂在 sharp minimum，只是在 test accuracy 上表現不佳。

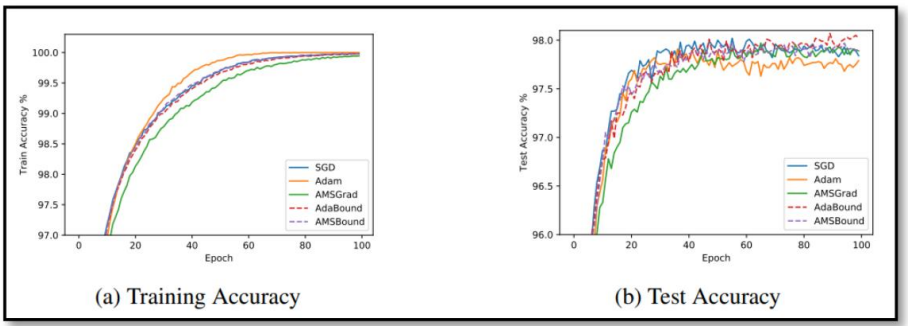

圖 2: Training and test accuracy for feedforward neural network on MNIST.

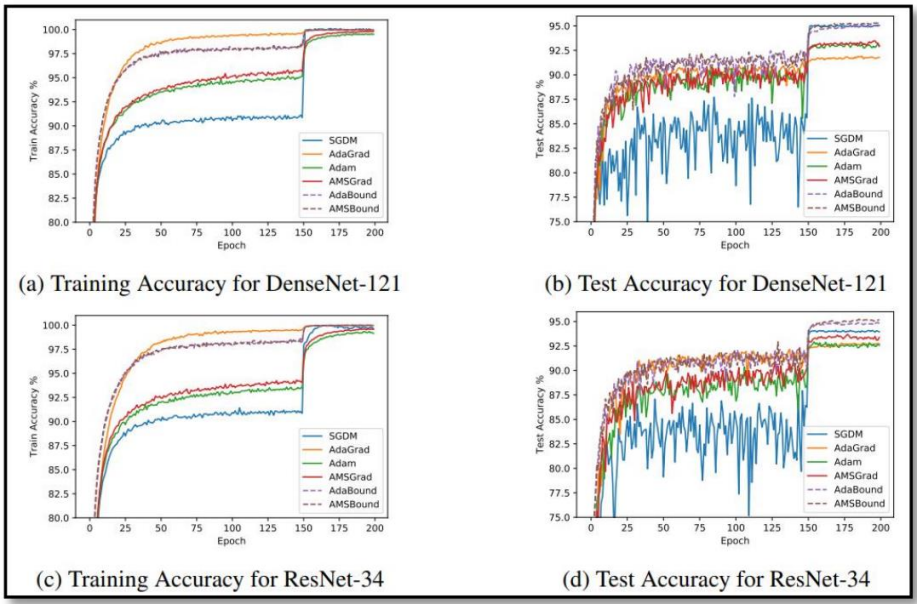

圖 3: Training and test accuracy for DenseNet-121 and ResNet-34 on CIFAR-10.

因此如果有一個新的 optimizer，它可以使模型盡量往 Sharpness 較小的地方更新，並收斂在 Flat Minimum，則這個新的 optimizer 能幫助模型提升泛化能力。

2. 研究方法

本專題的主要方法是使用 Google 團隊在 2021 ICLR 所發表的 spotlight 論文 SAM (Sharpness-Aware Minimization) (Pierre Foret et al, 2020)，SAM 的優點是在最小化 Loss value 的同時也最小化 Loss Sharpness，它是訓練時增加擾動 (perturbations) 的一種方法，以下為 SAM 做一次參數更新的演算法:

Step 1. 使用參數 w 對 batch data S 計算 gradient G

Step 2. G 除以 dual norm (所有梯度的平方和開根號) 得到 dual vector，沿著 dual vector 方向乘上 hyperparameter ρ (論文提到 $\rho = 0.05$ ，效果都不錯) 更新參數到達 w_{adv}

Step 3. 使用參數 w_{adv} 對 S 計算 gradient G'

Step 4. 用 $-G'$ 更新原本的參數 w

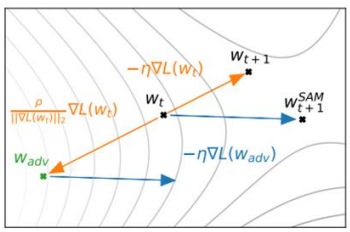

圖 4: Schematic of the SAM parameter update. (Pierre Foret et al, 2020)

以上的圖 4 是一次參數更新的示意圖，詳細的數學證明包括收斂性與 Generalization Bound 在論文內有詳述，主要概念是從論文中的第一個定理出發 (圖 5)，希望 loss 在分布 D 的期望值，必定小於目前參數周圍的最大值加上某個範圍內。

Theorem (stated informally) 1. *For any $\rho > 0$, with high probability over training set $\mathcal{S}$ generated from distribution $\mathscr{D}$,*

$$L_{\mathscr{D}}(\boldsymbol{w}) \leq \max_{\|\boldsymbol{\epsilon}\|_2 \leq \rho} L_{\mathcal{S}}(\boldsymbol{w}+\boldsymbol{\epsilon}) + h(\|\boldsymbol{w}\|_2^2/\rho^2),$$

where $h : \mathbb{R}_+ \to \mathbb{R}_+$ is a strictly increasing function (under some technical conditions on $L_{\mathscr{D}}(\boldsymbol{w})$).

圖 5: SAM Theorem 1.

首先，該如何找到 ε 會使 w + ε 得到最大 loss 值呢 ？ 我們可以先把下列第一個式子用一階泰勒展開，得到下列第二個式子，論文提到 p = 2 會得到較好的範化結果。那麼接下來就會得到一個範數問題 (norm problem)，這個範數問題可以解得 ε 為第三個式子，p 和 q 皆為 2，所以估計的 ε 就是 ρ 乘上 gradient 除以所有參數 gradient 的 L2 norm。

$$L_{\mathcal{S}}^{SAM}(\boldsymbol{w}) \triangleq \max_{||\boldsymbol{\epsilon}||_p \leq \rho} L_S(\boldsymbol{w}+\boldsymbol{\epsilon})$$

First-order Taylor expansion, p = 2 is typically optimal

$$\boldsymbol{\epsilon}^*(\boldsymbol{w}) \triangleq \arg\max_{\|\boldsymbol{\epsilon}\|_p \leq \rho} L_{\mathcal{S}}(\boldsymbol{w}+\boldsymbol{\epsilon}) \approx \arg\max_{\|\boldsymbol{\epsilon}\|_p \leq \rho} L_{\mathcal{S}}(\boldsymbol{w}) + \boldsymbol{\epsilon}^T \nabla_{\boldsymbol{w}} L_{\mathcal{S}}(\boldsymbol{w}) = \arg\max_{\|\boldsymbol{\epsilon}\|_p \leq \rho} \boldsymbol{\epsilon}^T \nabla_{\boldsymbol{w}} L_{\mathcal{S}}(\boldsymbol{w}).$$

Classical dual norm problem, p=2, q=2

$$\hat{\boldsymbol{\epsilon}}(\boldsymbol{w}) = \rho \, \text{sign}\left(\nabla_{\boldsymbol{w}} L_{\mathcal{S}}(\boldsymbol{w})\right) |\nabla_{\boldsymbol{w}} L_{\mathcal{S}}(\boldsymbol{w})|^{q-1} / \left(\|\nabla_{\boldsymbol{w}} L_{\mathcal{S}}(\boldsymbol{w})\|_q^q\right)^{1/p} \quad \text{where } 1/p + 1/q = 1$$

有了估計的 ε，接下來就可以求 SAM loss 的 gradient 了，gradient 的結果會有兩項相加，為了加速運算，刪除二階項的計算，得到最後的 SAM 的 loss gradient 近似，是 w + 估計 ε 上的 gradient。

$$\nabla_{\boldsymbol{w}} L_{\mathcal{S}}^{SAM}(\boldsymbol{w}) \approx \nabla_{\boldsymbol{w}} L_{\mathcal{S}}(\boldsymbol{w} + \hat{\boldsymbol{\epsilon}}(\boldsymbol{w})) = \frac{d(\boldsymbol{w} + \hat{\boldsymbol{\epsilon}}(\boldsymbol{w}))}{d\boldsymbol{w}} \nabla_{\boldsymbol{w}} L_{\mathcal{S}}(\boldsymbol{w})|_{\boldsymbol{w}+\hat{\boldsymbol{\epsilon}}(w)}$$

$$= \nabla_w L_{\mathcal{S}}(\boldsymbol{w})|_{\boldsymbol{w}+\hat{\boldsymbol{\epsilon}}(\boldsymbol{w})} + \frac{d\hat{\boldsymbol{\epsilon}}(\boldsymbol{w})}{d\boldsymbol{w}} \nabla_{\boldsymbol{w}} L_{\mathcal{S}}(\boldsymbol{w})|_{\boldsymbol{w}+\hat{\boldsymbol{\epsilon}}(\boldsymbol{w})}.$$

Accelerate the computation, drop the second-order terms

Final gradient approximation:

$$\nabla_{\boldsymbol{w}} L_{\mathcal{S}}^{SAM}(\boldsymbol{w}) \approx \nabla_{\boldsymbol{w}} L_{\mathcal{S}}(w)|_{\boldsymbol{w}+\hat{\boldsymbol{\epsilon}}(\boldsymbol{w})}.$$

3. 研究資料集與模型

因為 SAM 論文內已實作過許多經典的影像辨識資料集，例如 Cifar-10、Cifar-100，並得到數據給出結論，因此本次期末專題資料集欲使用 2022 深度學習第一次作業的 Stanford Dogs 和 Fashion Mnist Dataset，並使用 ResNet 架構 (Kaiming He et al., 2015)，包括 Resnet-18 與 Resnet-34，實驗觀察模型訓練完成後，使用 SAM 是否會比 SGD + momentum optimizer 有更好的泛化能力，也就是有更佳的測試集準確率，藉此得到結論。會使用 Stanford Dogs 和 Fashion Mnist Dataset 這兩個 Dataset 最主要原因是想觀察如果 SAM 用在非經典資料集的狀況下，是否會和 SAM 原論文有相同好的泛化效果，想藉此應用 SAM 在往後深度學習訓練時使用之優化器。Stanford Dogs 有 8 個類別的狗種類，前處理將每一張圖片像素轉為 256 x 256，訓練集資料有 1,328 張影像，測試集資料有 288 張影像。Fashion Mnist 是 28 x 28 的灰階圖像，有 5 個類別，訓練集資料有 30,000 張影像，測試集資料有 5,000 張影像。圖 6 為 Stanford Dogs Datasets，圖 7 為 Fashion Mnist Datasets。

圖 6: Stanford Dogs Datasets

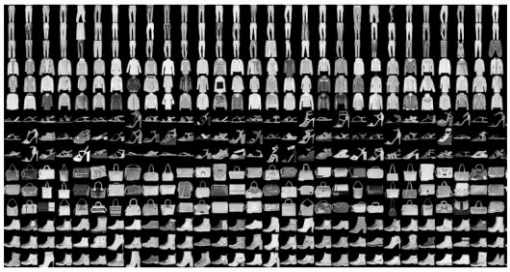

圖 7: Fashion Mnist Datasets

訓練模型和超參數設定方面，模型訓練使用的 batch size 為 128，Fashion Mnist train 20 個 epochs，Stanford Dogs train 30 個 epochs，Loss 計算皆使用 cross entropy。learning rate scheduler 採用 reduce learning rate on plateau，其方法是如果在設定的 epoch 內，train loss 沒有達到目前最小 loss，則調降 learning rate。本次作業在訓練上採取 3 個 epochs 內，test loss 如果皆未達到目前最小 loss，則將 learning rate 設為當前的 0.5 倍，並且設置最低 learning rate 為 10^{-6}。另外，優化器的超參數部分，SGD 和 SAM 的初始 learning rate 設為 0.001、momentum 使用 0.9，SAM 的 $\rho = 0.05$，SAM 使用 open source code (https://github.com/davda54/sam)。

本次實驗將比較 Resnet-18、Resnet-34 和 SGD、SAM 組合在 Stanford Dogs、Fashion Mnist 兩個 Dataset 的表現，一個組合重複訓練 5 次，並利用 two sample T-test 檢定 SAM 有無顯著在測試集表現優於 SGD。藉此探討 SAM 的泛化能力，是否優於傳統的優化器。

4. 研究結果

下圖為本次實驗的結果，首先在訓練時間方面，使用 SAM 的訓練時間比 SGD 多了一倍，這也符合預期，因為 SAM 需要兩次的梯度運算。再來是在 Stanford Dogs Datasets 的表現，把 Resnet-18 和 Resnet-34 都訓練 30 個 epochs 且接近收斂時，Resnet-18 和 Resnet-34 在使用 SAM 作為優化器時都顯著比使用 SGD 作為優化器時的 testing accuracy 高，大約是高 1.1%左右。在 Fashion Mnist 方面，把 Resnet-18 和 Resnet-34 都訓練 20 個 epochs 且接近收斂時，Resnet-18 在使用 SAM 作為優化器時並沒有顯著比使用 SGD 作為優化器時的 testing accuracy 高，但在 Resnet-34 在使用 SAM 作為優化器時有顯著比使用 SGD 作為優化器時的 testing accuracy 高，可以推測使用較複雜模型時，SAM 效果也會較好。

		SGD + Momentum			SAM			
		train	test	Cost Time (s)	train	test	Cost Time (s)	p-value
Stanford Dogs (epoch = 30)	Resnet-18	99.992 ± 0.053	77.082 ± 0.425	245	99.802 ± 0.217	78.124 ± 0.887	406	0.028
	Resnet-34	100.000 ± 0.000	79.792 ± 1.215	365	99.963 ± 0.075	80.993 ± 0.519	601	0.031
Fashion Mnist (epoch = 20)	Resnet-18	99.996 ± 0.005	99.384 ± 0.118	669	99.910 ± 0.019	99.440 ± 0.068	1197	0.196
	Resnet-34	99.842 ± 0.353	99.188 ± 0.237	1134	99.908 ± 0.026	99.472 ± 0.056	1999	0.030

圖 8: 實驗結果

圖 9 是 Fashion Mnist 的 Learning Curve，Resnet-18 和 Resnet-34 在用 SAM 訓練時的收斂速度並沒有 SGD 來得快，但在泛化效果方面，無論是在 Resnet-18 或 Resnet-34，SAM 都比 SGD 來得好。

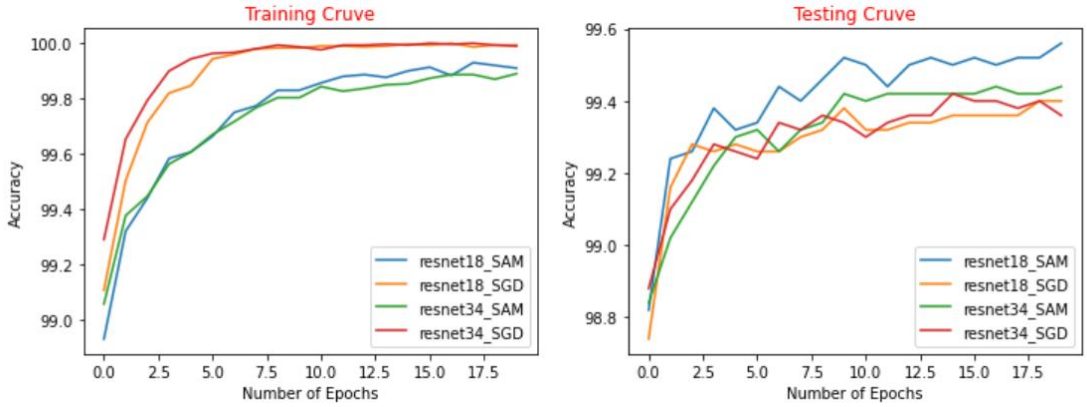

圖 9: Fashion Mnist Learning Curve

圖 10 是 Stanford Dogs 的 Learning Curve，Resnet-34 的收斂速度沒有 Resnet-18 來得快，但在泛化效果方面，Resnet-34 是優於 Resnet-18，並且在 Resnet-34 中，使用 SAM 的測試集曲線在訓練後期有優於 SGD 的趨勢。

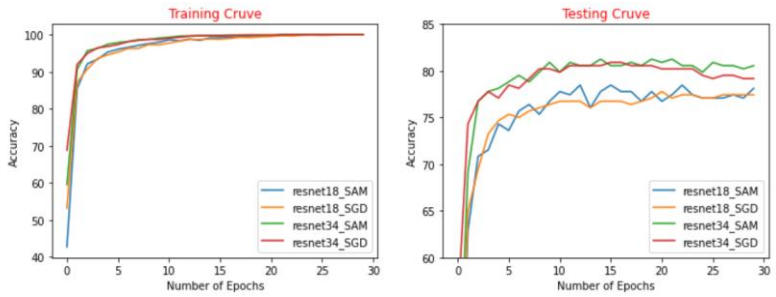

圖 10: Stanford Dogs Learning Curve

5. 結論與建議

在使用 SAM 時，存在使用者可以做取捨的部分，因為 SAM 雖然可以得到較好的模型泛化性，但也需要花比較多時間來訓練模型，讓模型收斂，因為 SAM 本身的特性是必須收斂在平坦點，因此在追求訓練時間快速，並且捨棄一點準確率的狀況下，SAM 並不會是最好的優化器選擇，舉本次實驗為例，與 SGD 優化器相比，花費將近一倍的訓練時間，可以換來顯著多大約 1%的準確率。

6. 參考資料

Liangchen Luo, Yuanhao Xiong, Yan Liu, Xu Sun. Adaptive Gradient Methods With Dynamic Bound Of Learning Rate, 2019
https://openreview.net/pdf?id=Bkg3g2R9FX

Pierre Foret, Ariel Kleiner, Hossein Mobahi, Behnam Neyshabur. Sharpness-Aware Minimization for Efficiently Improving Generalization, 2020
https://arxiv.org/abs/2010.01412

Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. Deep Residual Learning for Image Recognition, 2015
https://arxiv.org/abs/1512.03385

Liangchen Luo, Yuanhao Xiong, Yan Liu, Xu Sun. Adaptive Gradient Methods With Dynamic Bound Of Learning Rate, 2019
https://openreview.net/pdf?id=Bkg3g2R9FX