

1. (2%) 請以中文說明一下 **lifelong learning** 的中心概念是什麼？

Lifelong learning主要的目的簡而言之就是希望可以讓機器在學習不同**task**的時候使用相同的**model**來解決(**task**之間可以有關係也可以沒關係)，以改善每次學習不同的**task**都要換一個不同的**model**成本太高的問題。其中 **lifelong learning**有三個希望達到的目標。

第一個就是希望機器在學習**task A**之後，學習**task B**的時候不會忘記**task A**的東西(**Knowledge Retention**)，但是同時也希望他可以將**task B**學好(**But Not Intransigence**)。

第二個是**Knowledge Transfer**，簡單來說就是希望機器在學習的過程中可以觸類旁通，因為有過去的經驗，所以在訓練新的**task**的時候可以表現得更好，而因為有了新學習的知識，所以過去的**task**不僅不會忘記還會表現得更好。

第三個則是**Model Expansion**，希望機器在有需要的時候可以適度的增加**model**的大小，但是在這次的作業中我們並沒有做這個部分。

2. (2%) 列出 **EWC, MAS** 的作法是什麼？根據你的理解，說明一下大概的流程該怎麼做 (不要貼 **code**)。

EWC 的做法雖然只有做一次微分，但是卻可以帶來二次微分的效果。當 **train** 完 **task A** 要 **train B** 時將 **A** 的資料放到 **model** 中，得到他的 **output** 之後和 **label** 做 **log softmax** 得到 **loss**。之後 **backward** 將 **gradient** 取平方除以 **task A** 的資料數量，然後一直加到一個矩陣之中，最後就是我們想要求得的矩陣了！

MAS 的做法帶來的效益就是一次微分。作法也是在 **Task A** 和 **Task B** 的交界處將 **A** 的資料放到 **model** 裡面，算他的 **output** 取 **L2 norm** 平方得到 **loss**，**backward** 後將 **gradient** 取平方除以 **task A** 的資料數量，然後一直加到一個矩陣之中，最後就是我們想要求得的矩陣。

其實這兩個方法都是想要求得一個矩陣當作參數的守衛，如果矩陣的 **entry** 很大表示該值非常重要，不能做太大的更動，如果很小的話表示該值沒有很重要可以改比較多。

其實要得到這個矩陣的方式有兩種，可以做一次微分也可以做兩次微分，一次微分可以得到斜率趨近於零的點，但是如過最後停在鞍部的地方，就算他的一次微分等於零，他動一點對於整體的改變還是會很大的，所以這時候取二次微分就可以避免這個問題。在**model train**足夠好的狀況下，二次微分應

該會比一次微分來得好些。

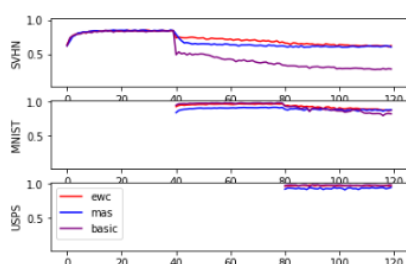
3. (1%) EWC 和 MAS 所需要的資料最大的差異是什麼？

他們最大的差別就是需不需要用到 label 資料，因為 ewc 會需要將 output 跟 label 做 log softmax 得到 loss，所以需要用到 label 資料。而 mas 則不需要。

4. (5%) 秀出 part1 及 part2 最後結果比較圖，並分析一下結果，以及你跑的實驗中有什麼發現。

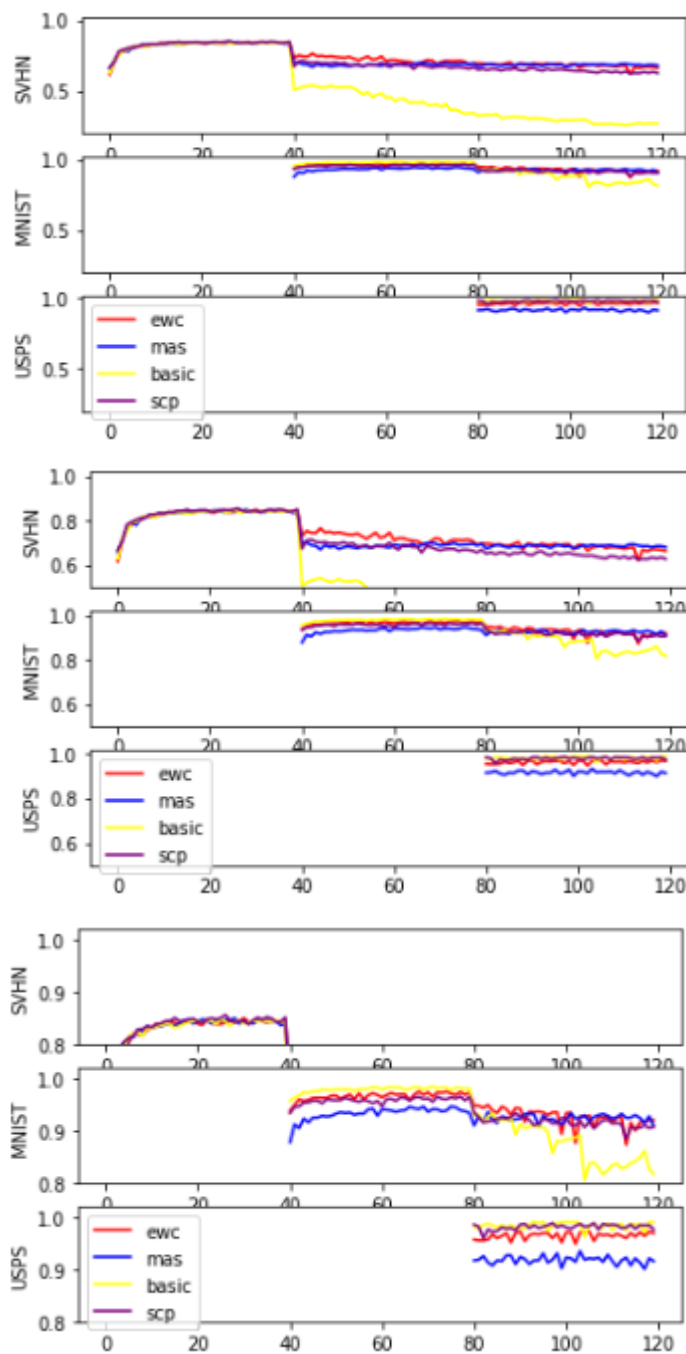
(EWC, MAS, baseline 比較圖 (2%) 與 EWC、MAS、SCP (或是你自己實做的演算法)、baseline 比較圖 (3%))

Part1 :



經過一連串實驗之後，我的 `coeff_list` 決定設為 `[1, 400, 0, 15]` (mas, ewc, baseline, scp) 他們各自的效果比較好。從結果可以看到不管是 ewc 或是 mas 他們的表現都比 Basic 來得好上許多，仔細看 ewc 的表現還略勝 mas 一點，可能是跟二次微分有關係。從圖中可以看到 mas 學習第二個 task 和第三個 task 的效果也沒有其他來的那麼好(有些固執)。

Part2 :



(下面兩張圖只是最上面那張圖的放大圖而已)

從時做結果可以看到，不管是 ewc、mas、scp 都完全是輾壓 basic 的，scp 在 SVHN 後段的時候忘得比較快，但是其實也沒有差到很多，他的效果差不多就介在 ewc 或 mac 中間，最好的時候也差不多跟 ewc 一樣，可見 ewc 或許真的是個不錯的方法做 lifelong learning，mas 雖然效果不錯(不太會忘記)，但是還是有固執的問題，正確率有點上不去，這真的是 mas 的硬傷。