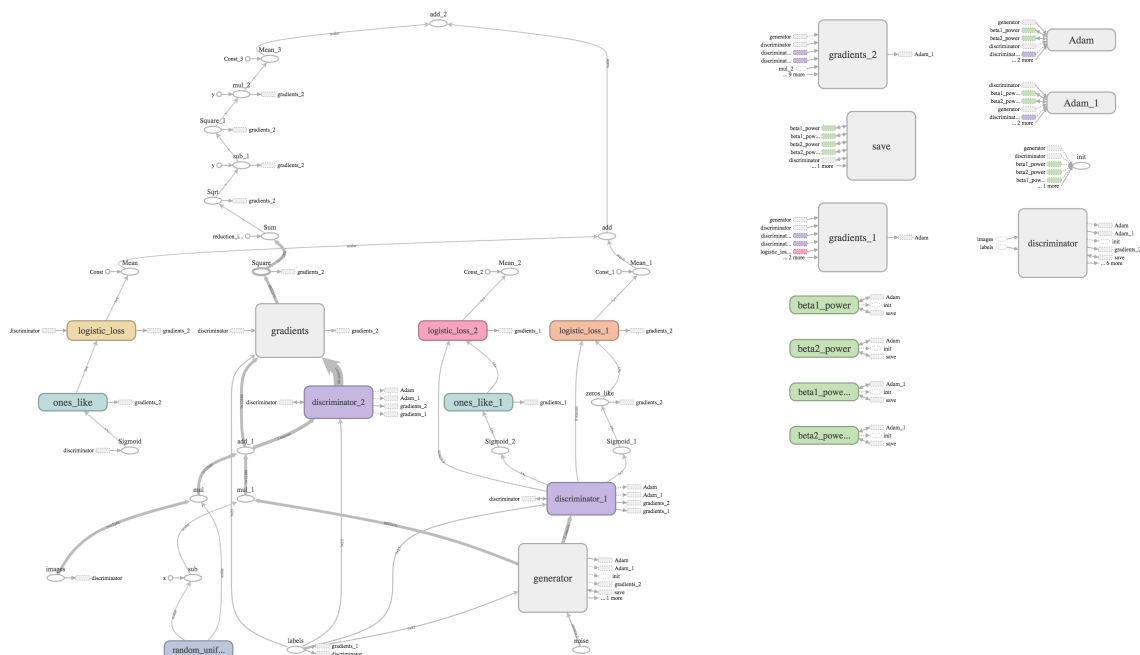


## 作業四 Generative Adversarial Networks

劉彥廷 B03902036

## 1. 模型敘述

以下為本次作業採用的模型預覽



其本質上為 conditional WGAN-GP (使用了 gradient penalty 的 WGAN)。文字的編碼直接採用助教會評量的 23 種 tags 構成的 one-hot vector。吐出的影像為 64x64 的結果。

根據報告規範，要寫出 loss functions

$$L_D^{WGAN-GP} = E[D(x)] - E[D(G(z))] - \lambda E[(\|\nabla D(\epsilon x + (1 - \epsilon)G(z))\| - 1)^2]$$

其中  $\lambda$  在程式裡頭為 scale 此一變數。

$$L_G^{WGAN-GP} = E[D(G(z))]$$

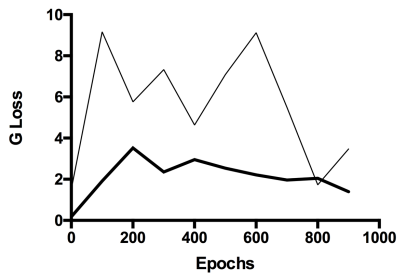
## 2. 效能改善

最開始的時候參考了 [2] 與 [3] 實作了 Conditional WGAN。但在調整 generator 與 discriminator 的更新比例（以下簡稱 D/G 比）時，出現了 mode collapse 且收斂時間很久，所以轉而更新模型為 Conditional WGAN-GP（WGAN 換為投影片當中的 Improved WGAN）。

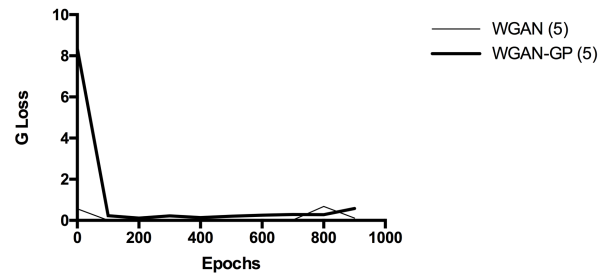


Figure 1: WGAN 的 D/G 比 0.2 發生的 mode collapse

以下為 WGAN 與 WGAN-GP 兩者之間的 G loss 與 D loss 比較。



(a) G loss



(b) D loss

### 3. 實驗設置與觀察

最終繳交的模型每更新 5 次 discriminator 就會更新 1 次 generator，該模型總共經歷過了 10000 次這個循環。

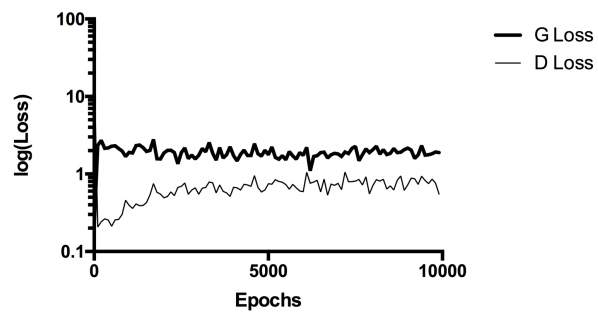
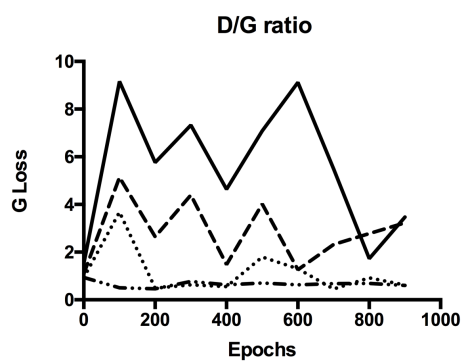


Figure 3: WGAN-GP 的 log loss

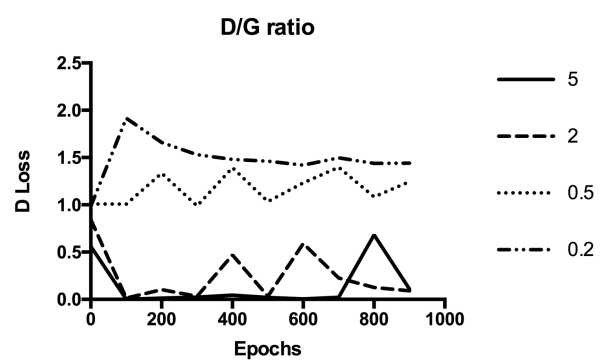
以下為最後的結果



在本次作業當中針對 WGAN 實驗了不同的更新比例並觀察其 loss。D/G 比變高代表了 discriminator 有比較多的時間學習，loss 預期會下降，反之則為 generator 的 loss 下降。

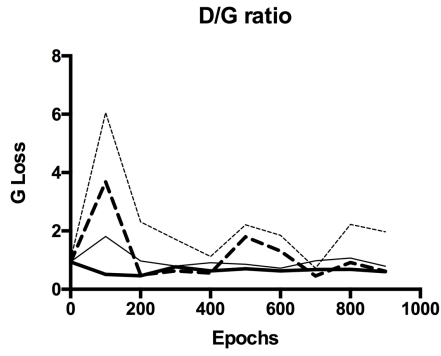


(a) G loss

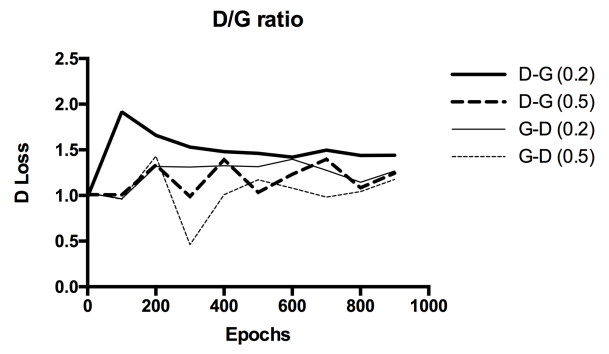


(b) D loss

但有趣的是，先更新 generator 似乎 loss 都會相較於先更新 discriminator 來得高。推測是基於 discriminator 後更新的話，在剛開始的 epochs 當中，generator 的表現會變差因為錯誤的判決，而錯誤不斷累積最終造成了整體的 GAN 表現不佳。



(a) G loss



(b) D loss

#### 4. 参考文献

- [1] Kristiadi, A. (2017). Wasserstein gan implementation in tensorflow and pytorch. <https://wiseodd.github.io/techblog/2017/02/04/wasserstein-gan/>. (Accessed on 01/06/2018).
- [2] Song, J. Tensorflow implementation of wasserstein gan (and improved version in wgan\_v2). <https://github.com/jiamings/wgan>. (Accessed on 01/06/2018).
- [3] Zhang, J. The test code for conditional convolutional adversarial networks. <https://github.com/zhangqianhui/Conditional-Gans>. (Accessed on 01/06/2018).