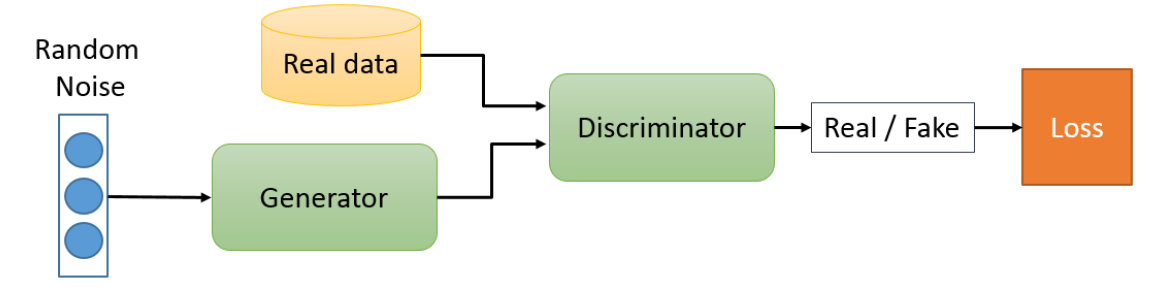
黃維新

A061738

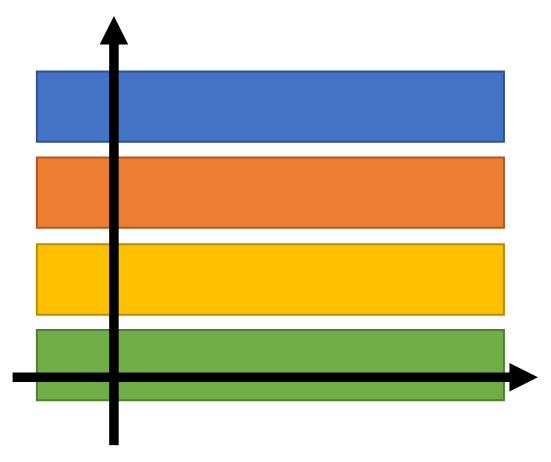
InfoGAN

1. **Introduction**

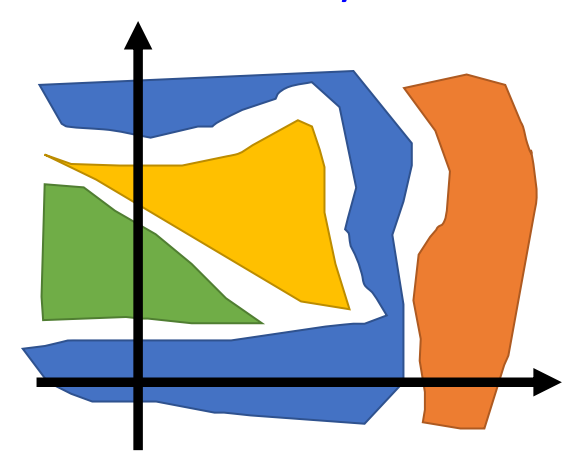
GAN為生成對抗網路，其觀念可以比喻成畫家和鑑賞家之間的關係，畫家想要模仿名畫家的畫，騙取大家的錢，而鑑賞家要判斷哪張畫是真的，哪張是假的，一開始畫家的技術很差，很容易被鑑賞家發現是假畫，但是由於鑑賞家功力也不強，因此還是有一些些畫被當成真畫，因此畫家仔細觀察蒙騙過去的畫有甚麼特別的，發現了一些重要的關鍵後，畫家的技術提升了，而鑑賞家此時功力還停留在原地，到了下次，畫家成功騙過鑑賞家，鑑賞家損失慘重，於是開始觀察真正的畫跟假畫的差異，而功力大幅提升，周而復始……畫家跟鑑賞家互相對抗，功力大幅提升，而畫家跟鑑賞家在本次的實作中，為兩個網路：generator以及discriminator，使這兩個網路互相對抗，也因此有了生成對抗網路這個名稱。基本架構圖如下↓



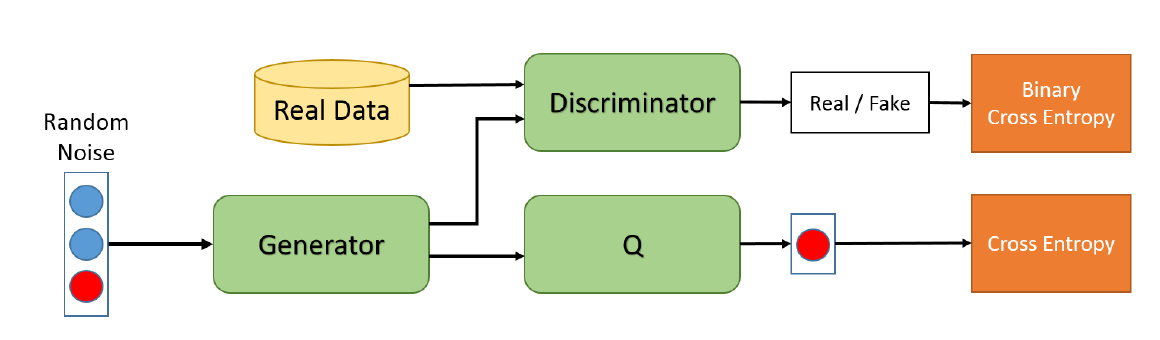
首先會輸入一連串隨機的數字，經過generator後生成fake圖片，然後discriminator接收real 圖片以及fake圖片，給出評價。訓練好後，generator就可以生成照片，其中Random Noise的每個維度都代表不一樣的feature，比如說生成的是人臉，那麼可能第一維代表的是頭髮的長度，第二個維度是嘴巴的大小，因此我們想像維度對應到feature的關係如下圖↓



但是實際上卻不是這樣，實際上會長得像下圖↓



因此我們不能很值觀的判斷哪個維度代表甚麼意思，於是有了infoGAN。

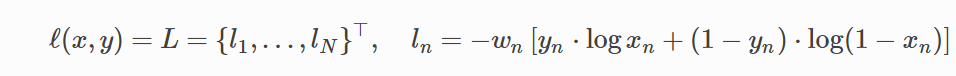
InfoGAN的架構圖如下↓

紅色的點為我們加入的非random的值，一起送入generator後，生成的data給Q網路去做分類，並且訓練Q網路要能分辨出我們加入的值會是多少，因此最後訓練出的網路，我們能直接調整非random值，讓data的變化在可控制範圍內。

**2. Experiment setups**

A. How you implement InfoGAN

1. Adversarial loss

Loss function of generator and discriminator was expressed using BCELoss，the function is 

Loss function of Q was expressed using BCELoss，the function is CrossEntropyLoss。

2. Maximizing mutual information

在訓練的時候要注意的就是不能同時train generator和discriminator，以上面的infoGAN的架構圖來說，如果同時train generator 和 Q網路，那麼generator只要原封不動把值傳給Q網路，那麼Q網路就能輕易判別人為加入的data是多少，為了要避免這種情況，因此train generator時discriminator和Q網路不能更新，反之亦然。

3. How you generate fixed noise and images

Noise我使用noise = torch.FloatTensor(opt.batchSize, 54).cuda()

10-D discrete one hot vector 我使用

dis\_c = torch.FloatTensor(opt.batchSize, 10).cuda()

B. Which loss function of generator you used

I use -> nn.BCELoss().

**3. Results**

A. Results of your samples