

# 程式語言-期中專題進度報告（第 12 組）

## 1. 成員：

組長：資工 2B 108502571 楊佳峻

組員：資工 2B 108502570 吳榕憲

組員：資工 2B 108502572 吳秉鴻

## 2. 題目：GAN 手寫數字生成

## 3. 實作方法：

使用 Python3.8、TensorFlow 套件

網站：<https://bit.ly/3lAsvUe>

參考論文方法說明：

GAN 可以看成是一個框架，它包含一個 generator 和 discriminator。Generator 負責透過模型輸入，生成能夠媲美真實數據的結果；discriminator 則負責防偽，也就是能辨識 generator 的輸出與真實數據。在兩種模型的對抗下，generator 的輸出可以越來越逼近真實數據，而 discriminator 則越來越精明，能辨識細微的真偽。

## 4. GAN 生成模型與辨識模型

Generator:總共有兩層，分別為輸入與輸出層，輸入的資料為  $1 \times 100$  的 random noise data，輸出為  $28 \times 28$  pixel 的圖片。分別以 relu 與 sigmoid 做為激勵函數。輸入層節點個數為 128，輸出層節點數為  $28 \times 28$ 。

```

### Generator
generator = Sequential([
    Dense(128, activation='relu', input_shape=[100]),
    Dense(28 * 28, activation='sigmoid')
])

```

Discriminator: 總共有兩層，分別為輸入與輸出層，輸入為一個 28\*28pixel 的圖片，輸出是否為 generator 生成的圖檔，一樣分別以 relu 與 sigmoid 做為激勵函數，輸入層節點個數為 128，輸出層節點數為 1 個，代表是或否。

```

### discriminator
discriminator = Sequential([
    Dense(128, activation='relu', input_shape=[28 * 28]),
    Dense(1, activation='sigmoid')
])

```

Loss function:

為了訓練兩種模型，會需要用到 loss function

$$L_D(x, z) = -\log(D(x)) - \log(1 - D(G(z)))$$

$$L_G(z) = -\log(D(G(z)))$$

其中， $L_D$  代表 discriminator 的 loss function， $x$  為真實數據， $D(x)$  代表 discriminator 判斷  $x$  為 true 的機率， $z$  為隨機輸入， $G(z)$  代表 generator 輸出的資料， $L_G$  代表 generator 的 loss function。我們的目標是最小化 loss function(損失函數)。

```

### d_loss
d_loss = (
    - log_clip(discriminator(x_input))
    - log_clip(1.0 - g)
)

```

```

### g_loss
g_loss = (
    - log_clip(g)
)

```

## 5. 目前進度與未來期望:

- 目前進度
  - 完成 GAN 基礎環境架設

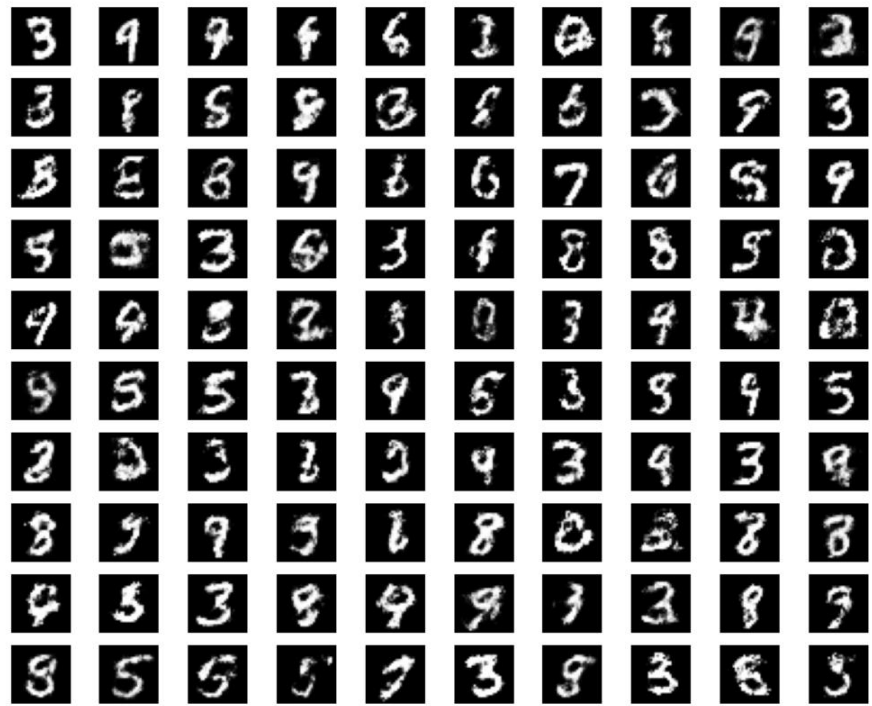
## ◆ 建構及訓練模型、未優化成果

### ● 未來期望

- 新增影像生成
- 提高生成數字精確度
- 使用 CGAN 新增條件輸入，控制生成的數字，而不是隨機生成

## 6. 目前生成手寫數字

Figure 1



## 7. 甘特圖：

