

# DSP HW1 Report

B04901056 張承洋

工作環境: ubuntu 16.04 terminal

執行方式:

1. 輸入指令 `make train` (生成 `train` 執行檔)
2. 輸入指令 `make test` (生成 `test` 執行檔)
3. 輸入指令 `make execute (training)`  
(生成 `model_0x.txt`)

```
execute : train
./train 1200 model_init.txt seq_model_01.txt model_01.txt;\
./train 1200 model_init.txt seq_model_02.txt model_02.txt;\
./train 1200 model_init.txt seq_model_03.txt model_03.txt;\
./train 1200 model_init.txt seq_model_04.txt model_04.txt;\
./train 1200 model_init.txt seq_model_05.txt model_05.txt
```

4. 輸入指令 `make predict (get result)`

```
predict : test
./test modellist.txt testing_data1.txt testing_answer.txt result1.txt acc.txt
./test modellist.txt testing_data2.txt testing_answer.txt result2.txt acc2.txt
rm acc2.txt
```

(生成 `result1.txt`, `result2.txt`, `acc.txt`)

分析結果:

**accuracy= 0.7504**

我發現 **training** 的次數跟準確度沒有絕對的關係，只能看出一個趨勢，在 **iteration** 大約為 **100** 時有些微下降，但再繼續提高 **iteration** 卻又得到更好的結果，推測是像 **linear regression** 的概念，從 **local minimum** 到另一個更低的 **minimum** 的過程中，準確度下降。

另外，我在這次作業的實作方式是動態陣列，為了方便

`function` 的 `input, output`，但實際上好像造成執行速度上表現不佳，推測是 `dynamic programming` 所拿到的記憶體位置較分散，呼叫並更新值的時候比較費時。