

機器學習期末專題報告

題目簡介

- DengAI: Predicting Disease Spread
- Hosted By DrivenData
- 預測兩個城市的登革熱案件數量

隊伍資訊

- 隊伍名稱 - NTU_b03902089_hanhan (NTU_b03902089_hanhanA)
- 隊員與分工
 - B03902089 - 林良翰 - 模型設計 (Neural Network & Adaptive Boosting Regressor)
 - B03902015 - 簡瑋德 - 前處理與特徵選擇 (Preprocessing & Feature Selection)
 - B03902007 - 鄭德馨 - 實驗設計與觀察 (Experiment Design & Observation)
 - B03902032 - 周家宇 - 實驗設計與觀察 (Experiment Design & Observation)

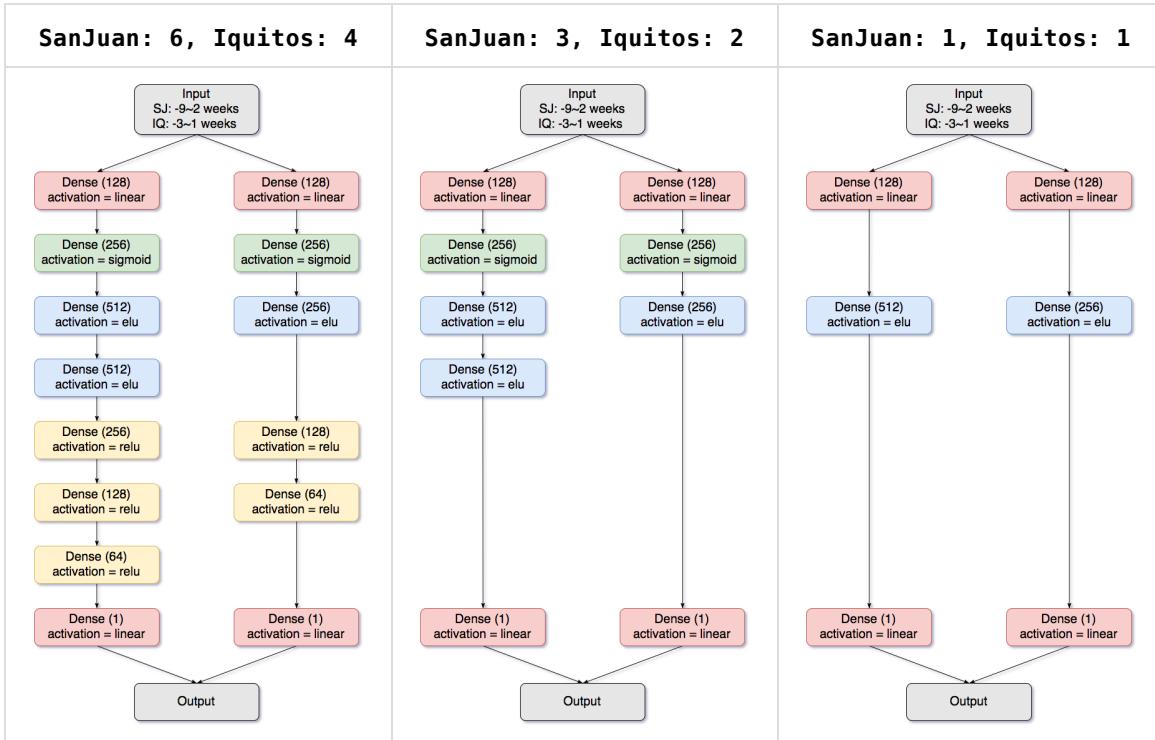
前處理與特徵選擇

- 原始資料大約有20項數值，經過篩選後，我們最後決定只取其中4個比較重要的指標，分別是
 - reanalysis_dew_point_temp_k - 平均露點溫度
 - reanalysis_specific_humidity_g_per_kg - 比濕（即水氣的值量與該團空氣總值量的比值）
 - station_avg_temp_c - 平均攝氏溫度
 - station_min_temp_c - 最低攝氏溫度
- 另外，因為訓練集與測試集的資料在時間上都是連續的（一週接著一週的資料），因此在選擇特徵時，除了該星期的資料，我們也會加入「之前幾週」與「之後幾週」的資料
 - 城市 SanJuan - 前9週至後2週，共12週48項數值
 - 城市 Iquitos - 前3週至後1週，共5週20項數值
- 考慮到資料中有一些欠缺的欄位（沒有紀錄的數值），我們透過線性內差把數值補回去。舉例來說，前一週的溫度是20度，後一週的溫度是26度，本週沒有相關紀錄的話，我們可以透過線性內差，替本週的溫度補上23這個數值
- 最後，考慮各項特徵的數值分布範圍差異甚大，例如溫度可能大部分都是兩位數，但比值卻常常是0到1之間的數值，故針對各項特徵進行正規化（標準化）。標準化的流程即是很常見的「減平均在除以標準差」，套用之後，各個特徵的數值會有「平均為零、標準差為一」的性值

模型架構

模型一 「Deep Nerual Network」

- 使用 keras.models.Sequential
- 把前處理好的資料shuffle (seed=3318)
- 「validation」的比例取10%
- 兩個城市各自訓練一個模型
- 實驗用的模型: (SanJuan: <SanJuan hidden layers>, Iquitos: <Iquitos hidden layers>)



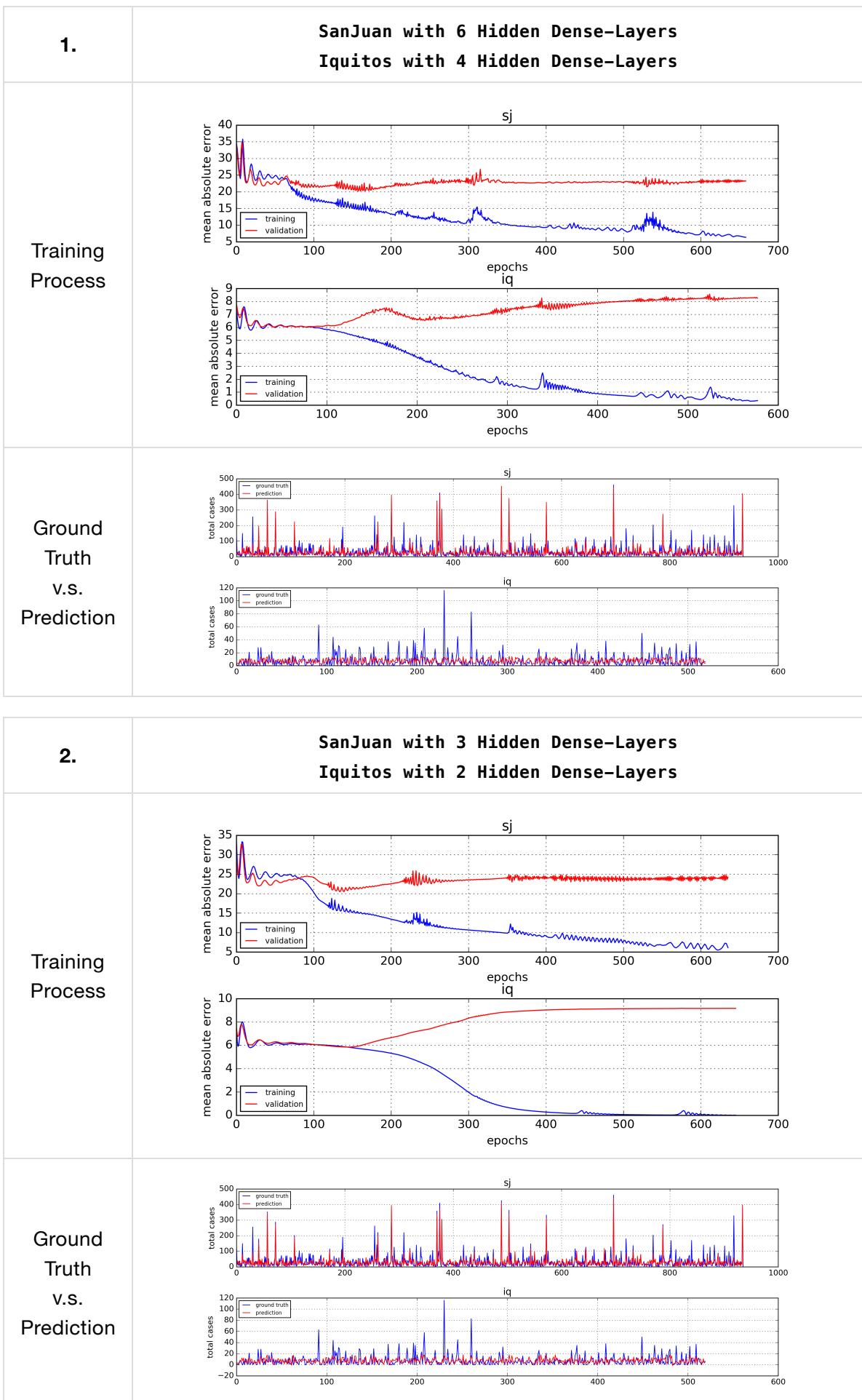
模型二 「Adaptive Boosting Regressor」

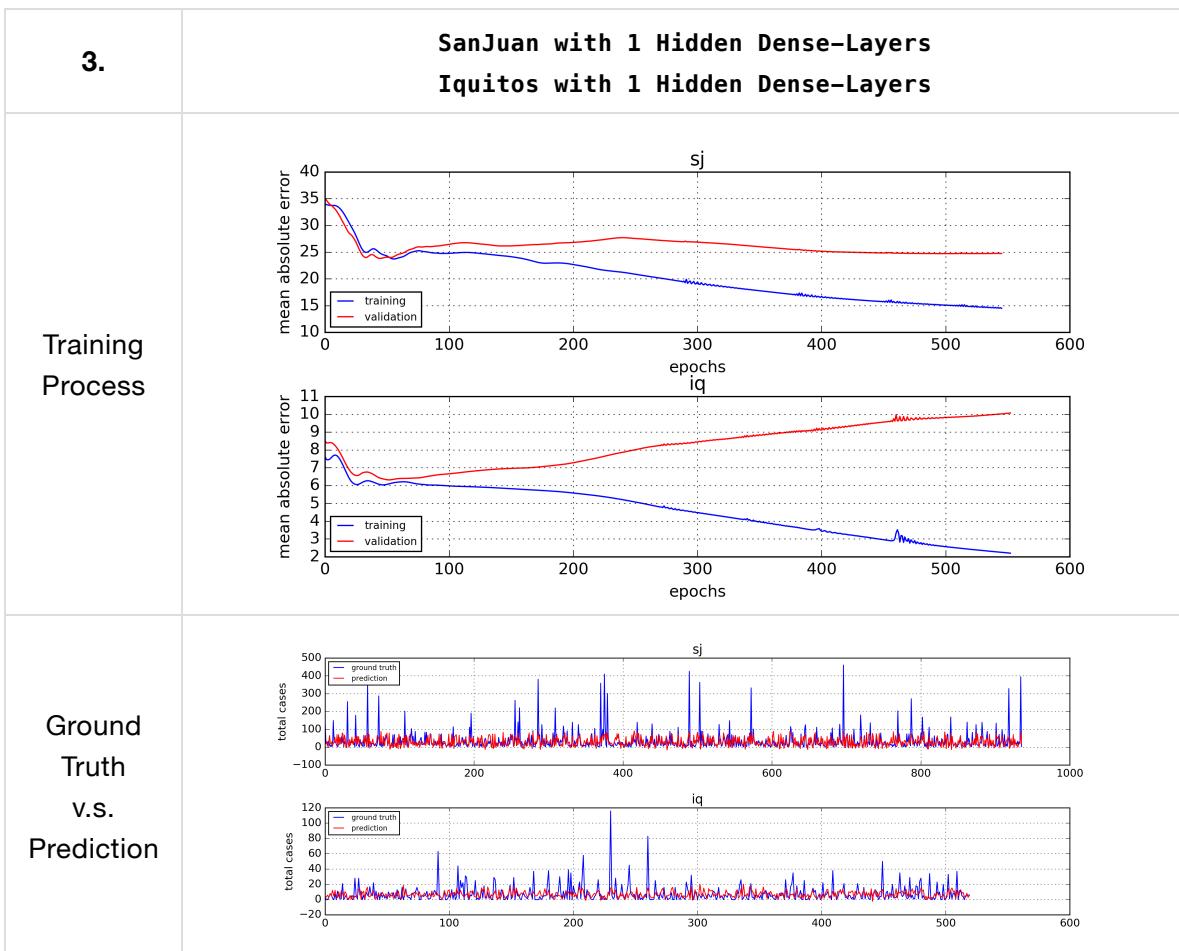
- 使用 `sklearn.ensemble.AdaBoostRegressor`
- 「validation」的比例取10%
- 兩個城市各自訓練一個模型
- `base_estimator` 使用 `DecisionTreeRegressor`
- SanJuan 的參數 - `n_estimators=50, max_depth=5`
- Iquitos 的參數 - `n_estimators=150, max_depth=6`
- 「Public Leader Board」的表現 - MAE=23.4856

實驗設計與觀察

「DNN」的模型，我們嘗試調整模型的深度

- 實驗數據與圖表



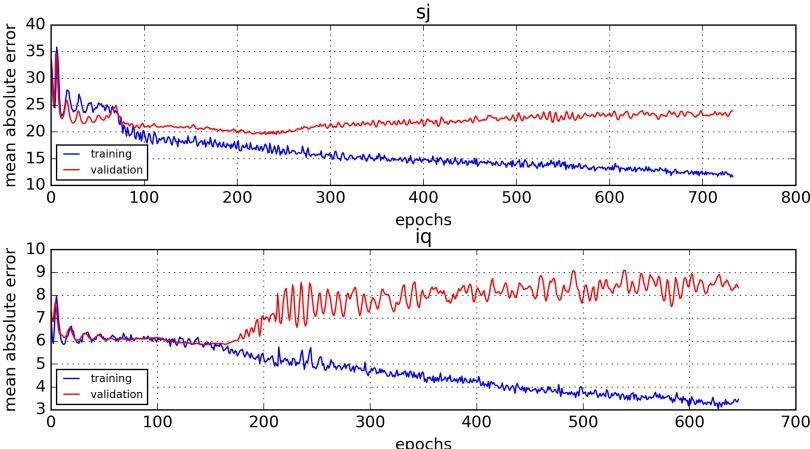
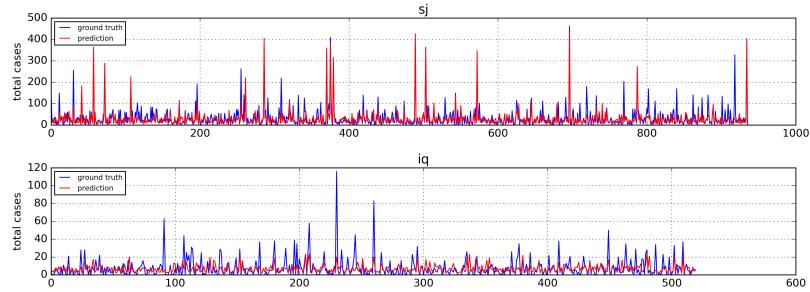
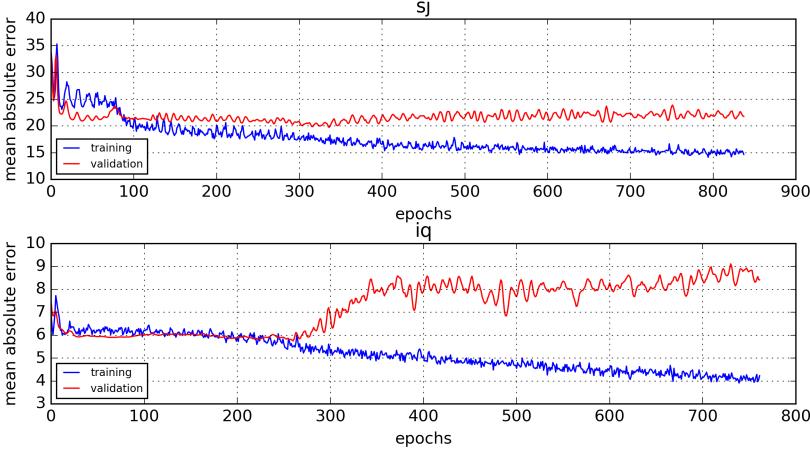
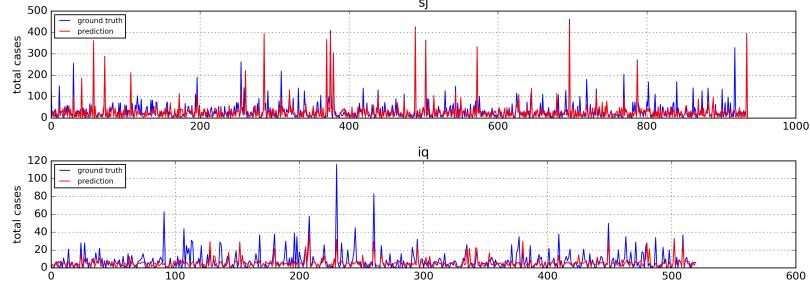


- 觀察與分析

1. 疊得越深的模型，訓練集的表現越好，測試集則沒有明顯的進步。同時，因為參數數量增加，對一些「Noise」特別敏感，從第一張圖就能發現，很容易出現小幅度的震盪
2. 模型的深淺，除了可能影響「overfit」的程度，也會影響「overfit」發生的時間。六層隱藏層的模型，城市「SanJuan」在僅僅50個「epoch」時就有過擬合的傾向，反觀只有一層隱藏層的模型，到了近200個「epoch」才出現「overfit」的問題

既然「DNN」的模型過擬合相當嚴重，我們嘗試加入「Dropout」

- 實驗數據與圖表

	1.	除了 Input Layer 之外，都加上 Dropout(0.2)
Training Process		
		
Ground Truth V.S. Prediction	2.	除了 Input Layer 之外，都加上 Dropout(0.4)
		
Ground Truth V.S. Prediction		

- 觀察與分析

1. 「Dropout」確實如我們預期的，縮小了訓練集和測試集的表現差距，但仔細觀察後，測試集的表現並沒有顯著的進步，只不過是訓練集的表現受到影響而變差。我們認為，面對「過

度擬合」，比起「Dropout」，「Early-Stopping」可能是一個更有效的解決方法 – 藉由觀察「validation」的表現，決定該把模型訓練、收斂到甚麼樣的程度

2. 除了發現「Dropout」的幫助不如預期，從圖中還可以發現，加了「Dropout」之後，震盪有變頻繁的趨勢。這個結果也許和「Dropout」的隨機性有關係，畢竟捨棄的「unit」是隨機選擇的，在訓練的過程中，多少添加了一些「隨機變因」，以致「錯誤曲線」不如原本那麼平滑

「**Adaptive Boosting Regressor**」的模型，我們調整了「弱回歸器」的數量以及深度

- 實驗數據與圖表

1. 50顆決策樹

最大深度	2	3	4	5	6
SanJua MAE	34.547	34.755	28.762	22.906	23.390
Iquitos MAE	10.312	8.894	7.129	6.665	6.446

2. 100顆決策樹

最大深度	2	3	4	5	6
SanJua MAE	35.213	36.390	29.431	24.903	23.250
Iquitos MAE	10.134	9.578	7.345	6.660	6.767

3. 150顆決策樹

最大深度	2	3	4	5	6
SanJua MAE	42.265	34.747	28.925	25.476	22.987
Iquitos MAE	11.144	8.803	7.453	6.839	6.329

4. 200顆決策樹

最大深度	2	3	4	5	6
SanJua MAE	42.696	37.087	30.601	24.790	24.287
Iquitos MAE	11.159	9.683	7.596	6.889	6.446

5. 300顆決策樹

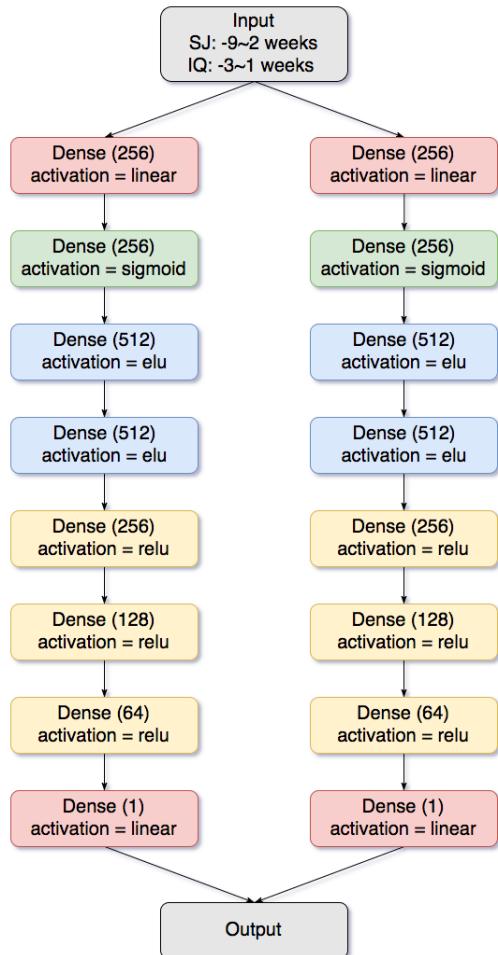
最大深度	2	3	4	5	6
SanJua MAE	34.633	35.988	30.799	24.509	23.381
Iquitos MAE	11.096	9.413	7.698	6.857	6.412

- 觀察與分析

1. 決策樹的數量對表現的影響有限，反而是樹的深度似乎和表現呈正相關。我們猜測，樹的數量只要達到一定的數量，就已經能夠展現「ensemble」的效果，影響結果好壞的，反而是弱回歸器各自的表現
2. 不同的城市，適合的參數也不一樣。如表中所示， SanJua 在50顆最大深度為5的決策樹時有最佳的「validation performance」；而城市 Iquitos 最好的情況則落在150顆最大深度為6的決策樹

最佳結果

- 使用多個更為複雜的DNN模型，並用不同的shuffle seed去訓練，最後ensemble出結果



- 使用的models

- sj_12.9573.h5 + iq_5.5605.h5：上傳後的MAE=22.2740
- sj_16.1113.h5 + iq_5.5494：上傳後的MAE=22.4832

- Ensemble結果

- 上傳後的MAE=22.0841