1. How do you select features for your model input, what preprocessing did you do ?

a. preprocessing：觀察 train 和 test_X 後即發現有多筆資料，因此需要採取一些規則來補 missing data：
   i. 24hr 都缺失 data：直接 copy 前一天的 data，作為今日的資料
   ii. 單個或連續數個缺失的 data：取下一筆 data 當作該時段的資料，若遇到連續幾個時段都缺失，將會一直往下找，直到找到 data。

b. 初次進行訓練時只取了 pm2.5 當作 feature，但結果發現 RMSE 居然高達 16.81741，因此推估其他 features 都有可能影響 pm2.5，至於誰影響多誰影響小，可以藉由計算 weight 來推估(在第二題將繼續討論)。

c. 數據的正規化與標準化：
   i. Max-min standardization：將原始資料的最大、最小值 mapping 至區間[0,1]，目的在於讓不同單位的 features 可以進行比較，但最終結果效果並不優秀(RMSE 為 3.57772 左右)。
   ii. Z-score standardization：利用原始資料的平均值(mean)和標準差(standard deviation)進行資料的標準化，適用於資料的最大值和最小值未知的情況，或是有 outlier 的情況。此作法最終都比 Max-min 有更佳的表現，因此在後續的測試都採用此方法來做預處理(RMSE 最佳表現為 3.233)

2. Compare the impact of different training data amounts on the PM2.5 prediction accuracy, visualize the result and explain it.

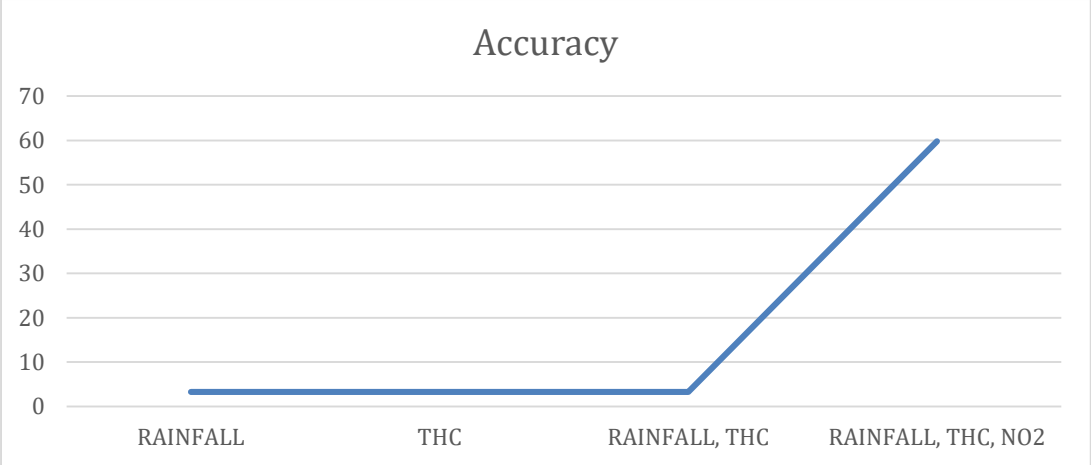
a. 以 test_X 計算出的 weight 比較各 features 的影響權重，剔除影響度最小的前幾個 feature，只找重要的 feature 來訓練試試看，以下是獲得的結果：

Feature	1	2	3	4	5
AMB_TWEP	13.97	0.66	0.43	0.8	0.54
CH4	0.86	0.05	0.21	0.31	0.32
CO	0.2	0.74	0.21	0.25	0.56
NMHC	1.07	0.16	0.19	0.19	0.42
NO	0.61	0.17	0.11	0.05	0.32
NO2	0.54	0.51	0.19	0.18	0
NOx	0.01	0.4	0.17	0.12	0.08
O3	0.1	0.57	0.4	0.02	0.13
PM10	0.05	0.35	0.22	0.72	0.9
PM2.5	3.08	0.56	0.64	0.44	1.06
RAINFALL	5	0.02	0	0.03	0.05
RH	0.12	0.46	0.63	0.34	0.04
SO2	0.51	0.24	0.16	0.16	0.05
THC	0.04	0.04	0.05	0.11	0.04
WD_HR	0.11	0.09	0.1	0.06	0.2
WIND_DIREC	0.3	0.11	0.26	0.07	0.12
WIND_SPEED	0.11	0.02	0.45	0.06	0.09
WS_HR	0.18	0.1	0.31	0.16	0.07

經過各小時分開排序後可知 RAINFALL、THC、NO2、NOx、O3 等 features 的權重比較低，其中 RAINFALL、THC 特別低，因此首要嘗試分別刪除這兩個因子後會得到什麼結果：

刪除的 feature	**Accuracy**
RAINFALL	3.30216
THC	3.30396
RAINFALL, THC	3.29527
RAINFALL, THC, NO2	59.80006

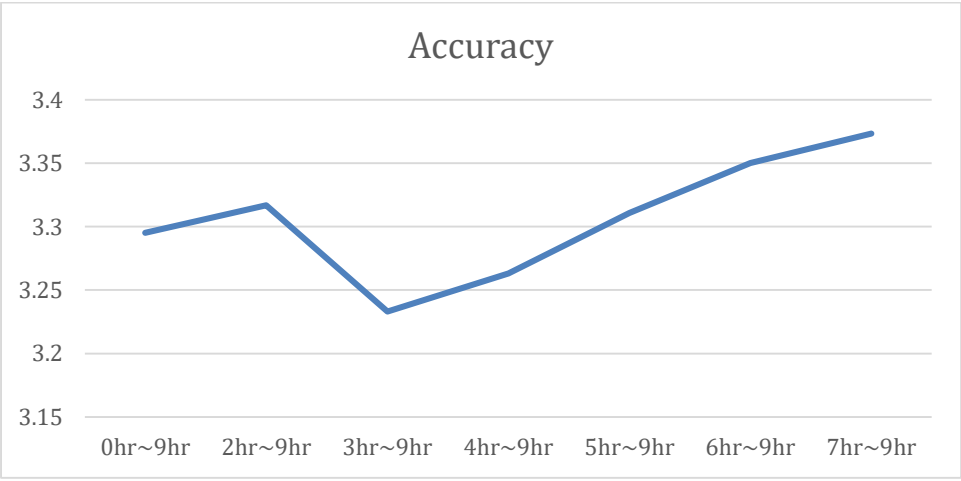
可以由測試結果推論刪除 RAINFALL 和 THC 可以有效降低 RMSE，再刪除 NO2 就會偏離預測方向了，因此後面就不在做刪除 features 的測試。



b. 比較不同的預測時間長度：（均以刪除 RAINFALL、THC 這兩個 features 為前提）

預測的時間長度(以預測第 10hr 為例）	**Accuracy**
0hr~9hr	3.29527
2hr~9hr	3.31688
3hr~9hr	3.233
4hr~9hr	3.26311
5hr~9hr	3.31074
6hr~9hr	3.35018
7hr~9hr	3.37343

原先的想法是假設離預測的第 10hr 越近，可能會讓預測值更準確，但是要取多長的預測時長需要一一測試，最終發現不是以前 9 個小時預測第 10 個小時的資料會最準確，而是取前 7 個小時的資料來預測下一個小時的資料會最準確，正確率的曲線如下：



推測 0hr~2hr 的資料可能會導致 overfitting 的情況。

3. Discuss the impact of regularization on PM2.5 prediction accuracy.

Regularization 是一種防止 overfitting 的方式，當模型過度學習訓練數據並無法適應新數據時，就會發生 overfitting。

Regularization 有多種技術可以應用於 PM2.5 預測模型，包括 L1 regularization、L2 regularization 和 Dropout regularization。L1 regularization 對大權重進行懲罰，可以幫助消除不必要的特徵並降低模型的複雜度。L2 regularization 會讓 weight 每次都變小一點，簡化模型，但最終不會只留下某個權重，而是讓所有權重都處於有效的狀態。Dropout regularization 在訓練期間隨機放棄一些神經元，可以通過鼓勵模型學習更健壯的特徵，從而防止過度擬合。

但是，如果應用過多的 regularization，模型可能會對數據進行欠擬合，無法捕獲重要特徵，從而降低精度。因此適量的使用 regularization 是一門藝術，需要通過多次實驗和調整才能找到最佳平衡。