Homework 1 Report – PM2.5 Prediction

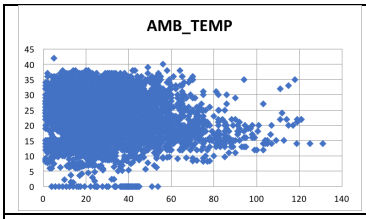
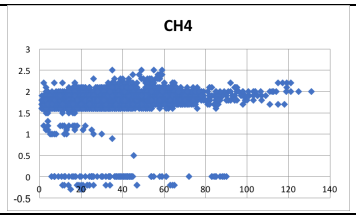
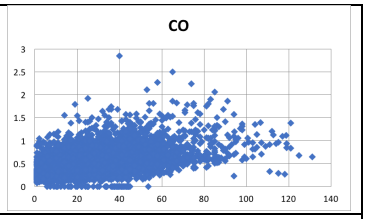
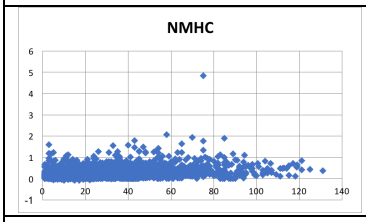
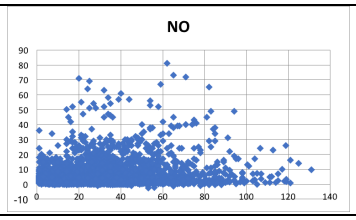
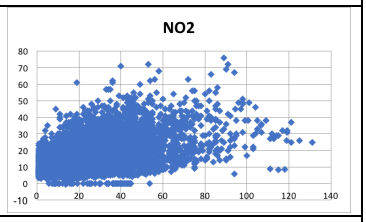
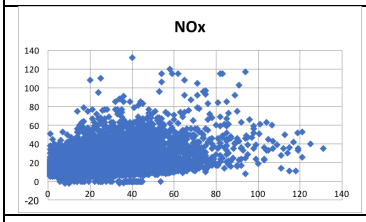
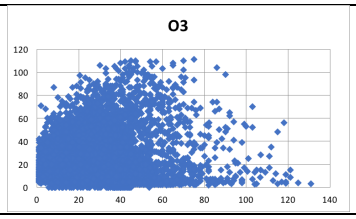
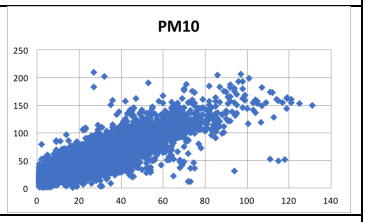
學號：b04502031　系級：電機二　姓名：施力維

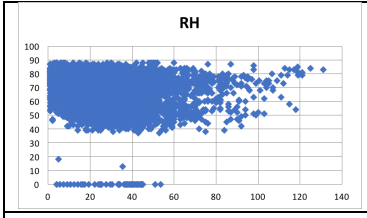
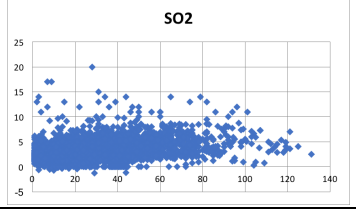
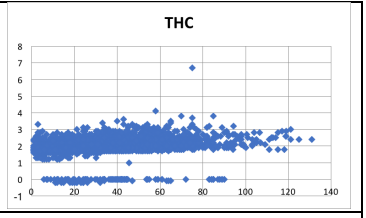
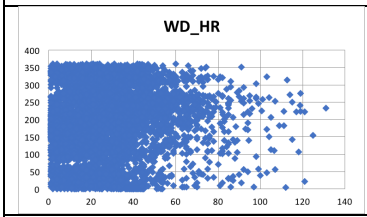
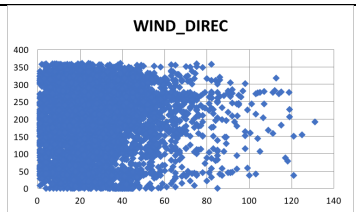
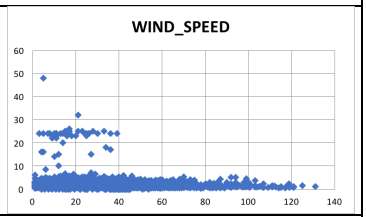
1. (1%) 請分別使用每筆 data9 小時內所有 feature 的一次項（含 bias 項）以及每筆 data9 小時內 PM2.5 的一次項（含 bias 項）進行 training，比較並討論這兩種模型的 root mean-square error（根據 kaggle 上的 public/private score）。

這邊是直接使用 raw data 來進行測試，並沒有使用任何的 preprocessing 或是 feature scaling。

僅使用前 9 筆資料 training 在 kaggle 上的分數是(9.56/9.69)；使用全部 feature 的前 9 小時來做 training 在 kaggle 上的分數是(13.02/10.33)，可以看出來使用全部的 feature 來 train 並不會得到比較好的結果，反而有一些落差。

仔細分析會發現，其實原因出在大部份的 feature 跟 PM2.5 的相關性很低，關於這點可以取 PM2.5 跟任意一筆資料來做圖以及計算相關係數來得知：

		
r = -0.13902959	r = 0.140474655	r = 0.484659035
		
r = 0.354880528	r = 0.168428377	r = 0.469270392
		
r = 0.410215527	r = 0.118966582	r = 0.829270028

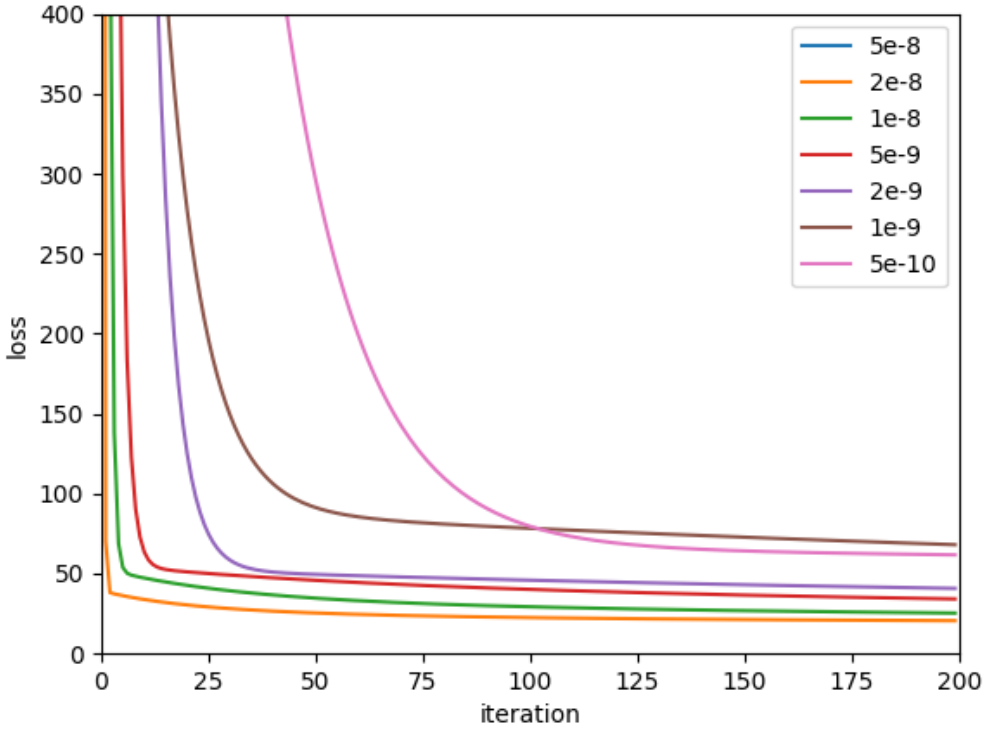
		
r = -0.109630976	r = 0.407790309	r = 0.279799374
		
r = 0.080282721	r = 0.068869233	r = -0.118722067

從上表可以觀察到，PM2.5 跟大部分 feature 的相關係數都很低，只有跟 PM10 是強相關(r = 0.83)，跟 CO、NO2、SO2 勉強算是弱相關，除此之外的資料在圖上大都呈現水平分布的狀態，也就是接近毫無關係，因此我們在使用所有 feature 來 training 時，train 出來的效果就會不理想，因為使用到了沒有意義的數據來做 training。

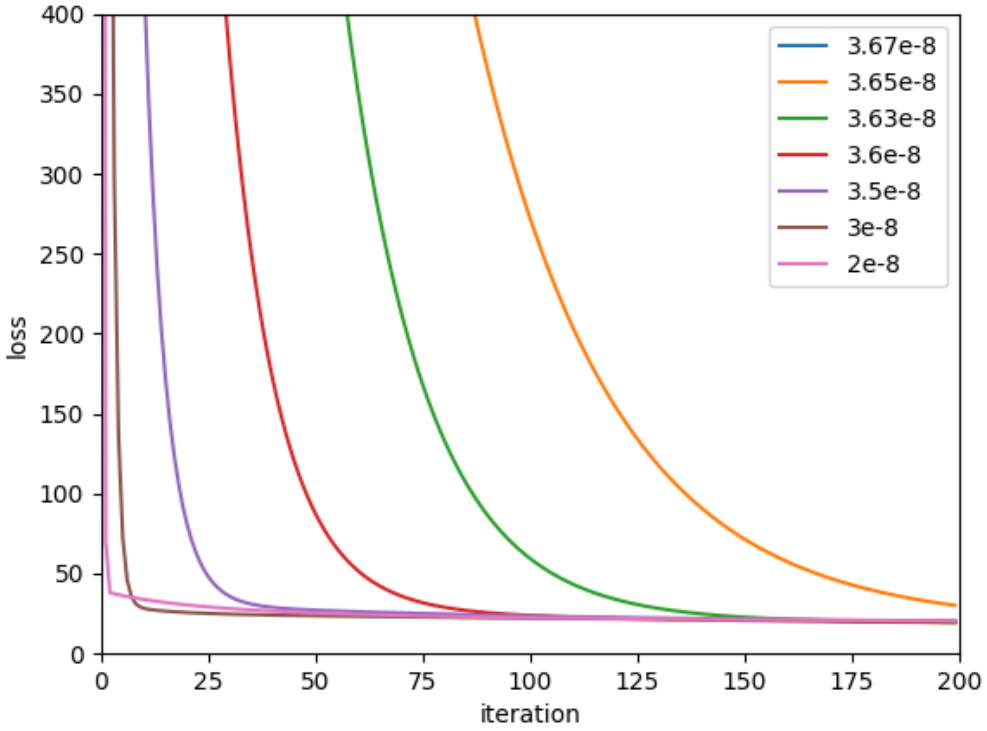
2. (2%) 請分別使用至少四種不同數值的 learning rate 進行 training（其他參數需一致），作圖並且討論其收斂過程。

做測試時使用的模型為 learning model，有對 data 做 precessing，feature 的選用為僅用前 9 筆 PM2.5 的形式，loss 為 RMSE。

使用的 learning rate 分別為5×10^{-10}~5×10^{-8}之間總共 8 筆資料，所做出的圖形如下表所示：



大致可以看出來，隨著 learning rate 逐漸增加，gradient decent 收斂的會越快，在圖中當 learning rate 為 2e−8 時，收斂非常快，在 10 代以內就收斂了；然而超過之後的 5e−8 則會直接發散，因此無法於圖上見到。當 learning rate 太小時(5e−10)，loss 收斂的並不夠快，在 100 個 iteration 才收斂，兩者相差了 10 倍。除此之外，最後收斂到的值也有明顯差異，2e−8 可以收斂到比較低的值。



為了瞭解在 2e−8 與 5e−8 中間的情況，繼續將中間的 learning rate 做出來，入上圖所示。當 learning rate 為 3.67e−8 時，loss 已經發散超過圖上了，而介於 2e−8 與 3.67e−8 是呈現逐漸增加的形式，從上面兩個測試可以知道：loss 對於 learning rate 會有一個局部最小值，在這個模型當中大約是 2e−8 左右，太慢會收斂的很慢，太快的話則很容易跨出去而無法收斂。

3. (1%) 請分別使用至少四種不同數值的 regulization parameter λ 進行 training（其他參數需一致），討論其 root mean−square error（根據 kaggle 上的 public/private score）。

這邊的模型採用經過 preprocessing 的 training data，feature 的選用為僅用前 9 筆 PM2.5 的形式，gradient decent 則是使用 adagrad 的方式來調整 learning rate，eta 設為 100，training 的 iteration 都設為 5000。

測試的 regulization parameter λ 分別為 0, 0.1, 1, 10, 100, 1000, 10000，在 kaggle 上的分數分別如下表所示：

	Test1	Test2	Test3	Test4	Test5	Test6	Test7
λ	0	0.1	1	10	100	1000	10000

Public	7.35	7.35	7.35	7.37	7.57	9.47	12.99
Private	8.02	8.02	8.02	8.05	8.42	10.91	14.73

以 Public/Private 分數作為縱軸做成圖如下，橫軸為取對數後的刻度：

觀察可以發現，當λ變大時，RMSE 也會變大，這是因為 regulization parameter 增加會使得 weight 會傾向於變小，這會使得 train 出來的 model 會傾向於平坦，train 出來的 RMSE 也會變大。反之當λ變小時 weight 可能會增大，有可能會因為 overfitting 而使得 RMSE 也會增加。然而在這次的測試中並沒有出現這種情形，推測可能是因為二次的函數還不足以發生 overfitting。

4. (1%) 請這次作業你的 best_hw1.sh 是如何實作的？ (e.g. 有無對 Data 做任何 Preprocessing？Features 的選用有無任何考量？訓練相關參數的選用有無任何依據？)

a. Preprocessing

觀察 Training data 可以發現在觀測資料當中有很多壞掉的資料，這可能是感應器失靈或是數據沒有存入資料庫當中等因素所致。壞掉的資料對於 training 的負面影響也很大，數據突然不連續的斷掉會使的 model 在嘗試 fitting 時產生過多變形，而且這些變形還是對於預判結果沒有什麼幫助的。Training data 在 12/13 時有一大片的無效資料，如下圖所示：

	A	B	C	L	M	N	O	P	Q	R	S	T	U	V	
1	日期	測站	測項	8	9	10	11	12	13	14	15	16	17	18	
4176	2014/12/12	大里	WIND_SPE	3.5	3.9	3.5	4.5	4.7	3.2	3.3	4.1	2.8	3.9	3.4	
4177	2014/12/12	大里	WS_HR	0.9	1.2	1.2	2.2	1	1.8	1.3	0.7	0.5	0.9	1.1	
4178	2014/12/13	大里	AMB_TEM	12	13	0	0	0	0	0	0	0	11	11	
4179	2014/12/13	大里	CH4	1.7	1.7	0	0	0	0	0	0	0	-0.2	-0.2	
4180	2014/12/13	大里	CO	0.51	0.51	0	0	0	0	0	0	0	0.61	0.7	
4181	2014/12/13	大里	NMHC	0.13	0.14	0	0	0	0	0	0	0	0.03	0.03	
4182	2014/12/13	大里	NO	3.2	4.6	0	0	0	0	0	0	0	0.7	1.4	
4183	2014/12/13	大里	NO2	15	15	0	0	0	0	0	0	0	6.9	32	
4184	2014/12/13	大里	NOx	18	20	0	0	0	0	0	0	0	7.6	33	
4185	2014/12/13	大里	O3	24	20	0	0	0	0	0	0	0	1	2.8	
4186	2014/12/13	大里	PM10	47	44	0	0	0	0	0	0	0	0	56	
4187	2014/12/13	大里	PM2.5	34	-3	0	0	0	0	0	0	0	34	47	
4188	2014/12/13	大里	RAINFALL		NR	0	0	0	0	0	0	0	0	NR	NR
4189	2014/12/13	大里	RH	56	40	0	0	0	0	0	0	0	57	59	
4190	2014/12/13	大里	SO2	3.2	4	0	0	0	0	0	0	0	1.5	6.3	
4191	2014/12/13	大里	THC	1.9	1.9	0	0	0	0	0	0	0	-0.1	-0.1	
4192	2014/12/13	大里	WD_HR	10	15	0	0	0	0	0	0	0	327	337	
4193	2014/12/13	大里	WIND_DIR	327	0.5	0	0	0	0	0	0	0	338	326	
4194	2014/12/13	大里	WIND_SPE	2.4	0.1	0	0	0	0	0	0	0	2.2	2.2	
4195	2014/12/13	大里	WS_HR	1.2	0.8	0	0	0	0	0	0	0	0.8	0.7	
4196	2014/12/14	大里	AMB_TEM	12	16	19	22	23	24	23	22	20	18	18	

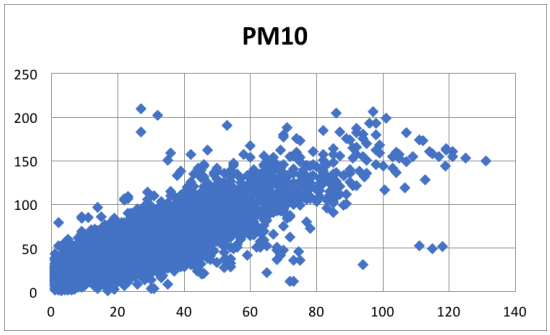
為了避免 model 被壞掉的資料所影響，我們可以對資料進行修補的動作，修補方式的核心主要是使用內差法，當資料出現不尋常的狀況(突然變成 0、出現負數、突然爆衝到不合理的數字時)，就將該筆資料判定為無效資料，並使用最接近的兩筆資料來進行內差補齊。

與此同時，testing data 也會有資料爛掉的情況存在，因此在用 model 預測 testing data 前也要先對 testing data 做 preprocessing。

Preprocessing 的效果十分顯著，單一以 PM2.5 作為 feature，採用 linear model 做測試：沒有 preprocessing 時 kaggle 上的分數為 (public/private) = (9.56/9.70)，做完 preprocessing 之後則是 (6.77/7.14)，在 RMSE 上有超過 2.5 的進步，由此可知壞掉的資料對於 model 影響性之大，以及 preprocessing 的重要性。

b. Feature 的選擇

選擇有用的 feature 對於 training 來說是不可或缺的事情，沒有用的 data 只會 train 出沒有用的 model，因此這邊參考於第一題中的比較，選擇與 PM2.5 關聯性最高的 PM10 來作為 feature，其相關係數為 0.83。



只使用 PM2.5 前 9 筆來做也可以得到不錯的結果(public/private) =

(6.77/7.14)，但是把 PM10 搭配 PM2.5 一起使用，所得到的結果更為優秀(6.46/6.27)。

比較合理的解釋可能是因為 PM10 指的是小於 10 微米的可吸入空氣懸浮顆粒物；PM2.5 則是小於 2.5 微米的可吸入空氣懸浮顆粒物，因此 PM10 的數值中有包含 PM2.5，所以將 PM10 的數值列入考量能夠使 PM2.5 的判斷更為準確。

c. Training 的模型

Training 採用 Linear model 來實作，gradient decent 則是使用 adagrad 的方式來調整 learning rate，eta 的初始值數為 100，初始權重由在[0, 1]區間中亂數產生，不過由於是 linear regression，並不會有局部極大極小值的問題，因此 initial point 的重要性並不高。

最後實作的結果(public/private) = (6.46/6.27)，kaggle 的名次分別為 21/7。