學號：b05901070　系級：電機四　　姓名：蔡昌廷

備註：

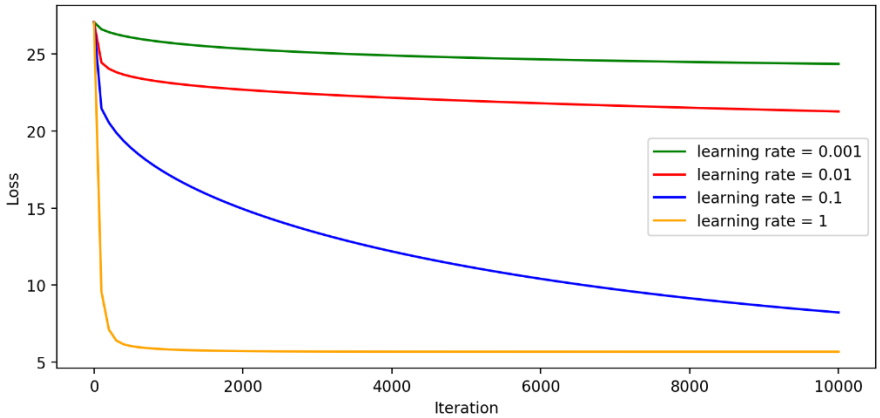
a. 1~3 題的回答中，NR 請皆設為 0，其他的數值不要做任何更動。

b. 可以使用所有 advanced 的 gradient descent 技術（如 Adam、Adagrad）。

c. 1~3 題請用 linear regression 的方法進行討論作答。

1. (2%) 使用四種不同的 learning rate 進行 training (其他參數需一致)，作圖並討論其收斂過程（橫軸為 iteration 次數，縱軸為 loss 的大小，四種 learning rate 的收斂線請以不同顏色呈現在一張圖裡做比較）。

在第一題的比較，我先設定 iteration 次數為 10000 次來進行，learning rate 分別使用 [0.001, 0.01, 0.1, 1]四個不同數值來分析其對 loss 大小的影響，作圖如下。



由上圖可發現當 learning rate 越大時，其 loss 降低的速度越快，根據老師影片提及的 learning rate 越大，在參數更新的步伐越大，其 loss 也會相對應較低的觀念一致，至於紅色與綠色的線需要更多次的 iteration 才可以使其 loss 降低並收斂。

2. (1%) 比較取前 5 hrs 和前 9 hrs 的資料（5*18 + 1 v.s 9*18 + 1）在 validation set 上預測的結果，並說明造成的可能原因（1. 因為 testing set 預測結果要上傳 Kaggle 後才能得知，所以在報告中並不要求同學們呈現 testing set 的結果，至於什麼是 validation set 請參考：https://youtu.be/D_S6y0Jm6dQ?t=1949 2. 9hr:取前 9 小時預測第 10 小時的 PM2.5；5hr:在前面的那些 features 中，以 5~9hr 預測第 10 小時的 PM2.5。這樣兩者在相同的 validation set 比例下，會有一樣筆數的資料）。

一開始我先將作業提供的 training set 的前半 80%當作此題的 traing data，後半的 20%則是當作 validation set 作為驗證使用，其他參數分別是 iteration 30000 次，learning rate 設為 0.2，並且有使用 Adagrad 的 gradient descent 的技術，最後結果如下表。

	取前 9 小時	取前 5 小時
RMSE	5.66768	5.67721

由上表會發現取前 9 小時的 features 比取前 5 小時的 features 結果好一點，可能的原因來自 9 小時的 features 擁有比前 5 小時的 features 較多的資訊來預測 PM2.5 的值，不過實際上其差異並不會到太大。

3. (1%) 比較只取前 9 hrs 的 PM2.5 和取所有前 9 hrs 的 features（9*1 + 1 vs. 9*18 + 1）在 validation set 上預測的結果，並說明造成的可能原因。

在此題的參數設計跟第二題一樣(iteration、learning rate、adagrad)，最後結果如下表。

	取所有 features	只取 PM2.5
RMSE	5.66768	5.87603

根據上表的結果，這次只取 PM2.5 的 RMSE 跟上一題比起來反而表現的更差了，最主要的原因可能來自只取 PM2.5 的數據會忽略掉太多其他也可以幫助預測 PM2.5 的其他 features (ex: PM10 等等)，導致其最後訓練出來的表現並沒有比只取前 5 小時所有 features 來的好，可見在做 feature selection 時必須注意不要太過簡化，要留下一些可以幫助預測的 features。

Collaborator：b05901071 孫鍾恩

4. (2%) 請說明你超越 baseline 的 model(最後選擇在 Kaggle 上提交的) 是如何實作的（例如：怎麼進行 feature selection, 有沒有做 pre-processing、learning rate 的調整、advanced gradient descent 技術、不同的 model 等等）。

為了降低最後預測出來 PM2.5 的 RMSE，我在一開始 18*9 的 features 上有選取跟 PM2.5 數值相關的 features 來進行 training，至於如何選擇較相關的 features，我先用 minepy 提供的 MIC(Maximal Information Coefficient) 最大互信息係數，其 feature 係數越大代表該 feature 跟 PM2.5 的相關度越高(不僅限於線性關係)，以下是我實際對 18 個 features 做出來對 PM2.5 的 MIC score(介於 0 至 1 之間)

feature	MIC	feature	MIC
AMB_TEMP	0.04644	PM2.5	0.99973
CH4	0.08545	RAINFALL	0.02518

CO	0.16497	RH	0.06666
NMHC	0.09107	SO2	0.09943
NO	0.01945	THC	0.11313
NO2	0.12357	WD_HR	0.09968
NOX	0.10314	WIND_DIREC	0.08100
O3	0.12556	WIND_SPEED	0.02202
PM10	0.35255	WS_HR	0.01941

根據上表，我先選擇 MIC 大於 0.1 的 features 來做 training，分別有 CO、NO2、NOX、O3、PM10、PM2.5、THC，而 regression 的部分則是針對 7 個 features 皆做三次回歸式 $b + w1*x + w2* x^2 + w3*x^3$，因此 w 的 dimension 為 3*7*9+1(其中+1 為 bias)，在 training 的過程使用 Adam 的方式來幫助收斂速度，最後上傳 kaggle 的成績為 5.34417，雖然已經過 strong baseline，但是總覺得應該還可以更好，因此跟同學討論過後發現，刪除掉 CO，並新增 SO2、WD_HR，最後多項式使用四次多項式去回歸，上傳 kaggle 之 public set 成績為 5.18620。