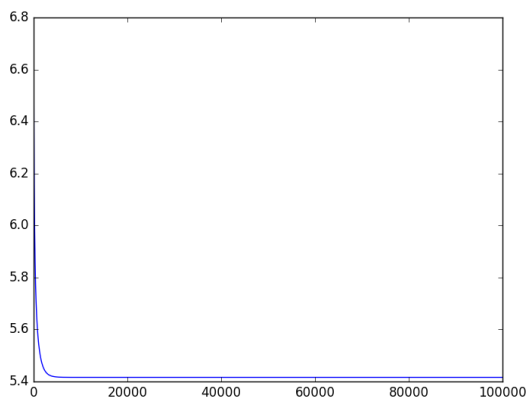


Homework 1 Report - PM2.5 Prediction

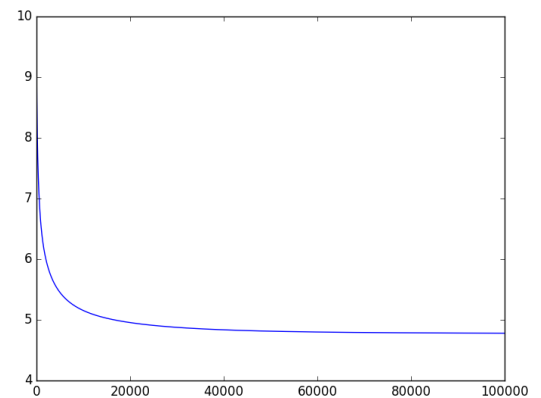
學號：b04901074 系級：電機三 姓名：吳倉永

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）。



PM2.5 當 feature

Kaggle: 8.88(pri+pub)

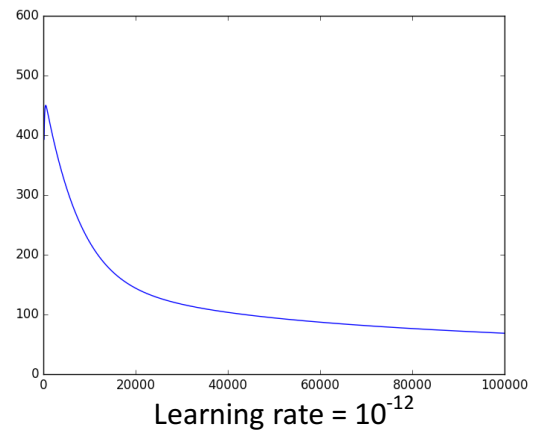
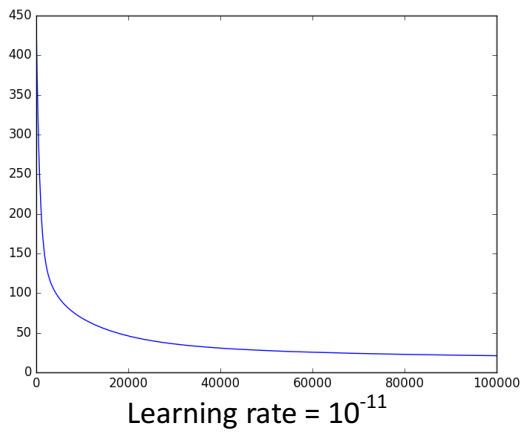
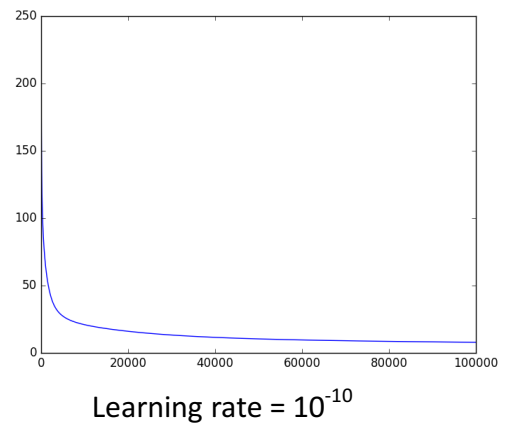
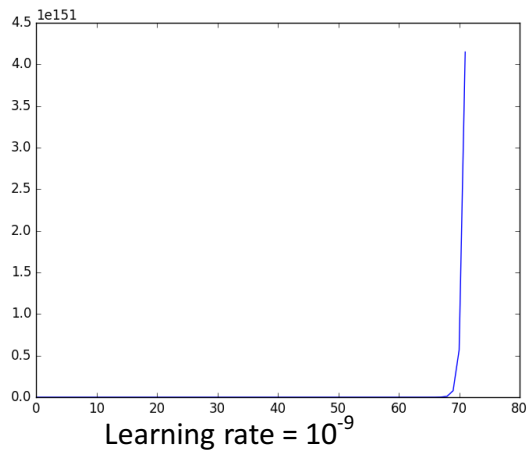


所有 18 個當 feature

Kaggle: 8.31

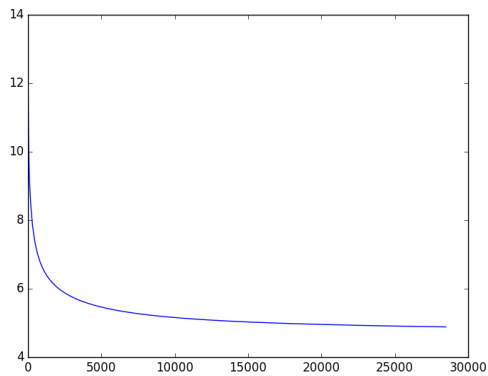
以 Kaggle 上的結果來說(test error)，用 18 個參數當 feature 的 performance 較好，而已 train error 來說，PM2.5 當參數的 performance 較佳，但在 training error 則相反。我認為原因在於，當 feature 數量少的時候，model 就比較容易去 fit training data，我們可以發現到左邊的圖很快就收斂，且 error 小，而當 feature 數量大的時候，error cost 的收斂速度就會下降。而且發現 regularization 對參數少時的影響沒那麼大

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。

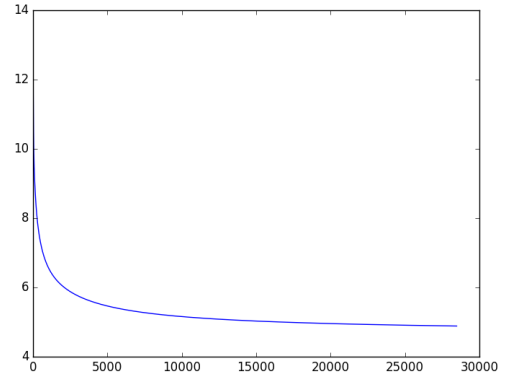


(x 軸訓練次數, y 軸 error cost)四張圖 learning rate 大小分別由左至右，由上而下增加 10 倍。可以發現到 learning rate 最大的 error cost 爆掉，這是因步伐太大，收斂不回去的原因。而在 learning rate 遞減的時候，就慢慢修正到一個比較剛好的步伐，然後 learning rate 越小，收斂速度越慢。

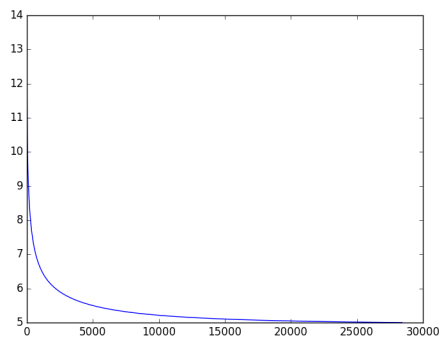
3. (1%) 請分別使用至少四種不同數值的 regularization parameter λ 進行 training (其他參數需一至), 討論其 root mean-square error (根據 kaggle 上的 public/private score)。



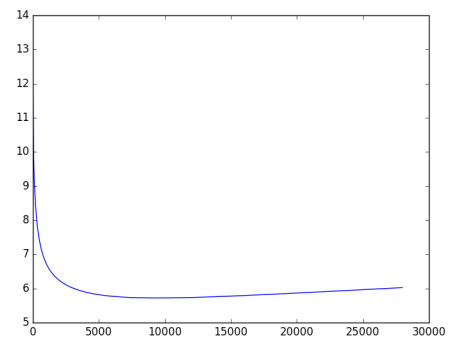
regular lambda = 0 Kaggle: 8.31



regular lambda = 1 Kaggle: 8.3



regular lambda = 100 Kaggle: 8.25

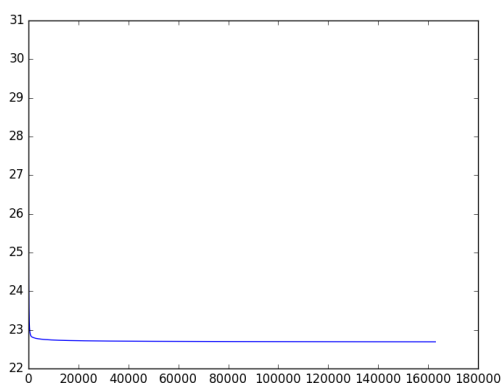


regular lambda = 10000 Kaggle: 7.94

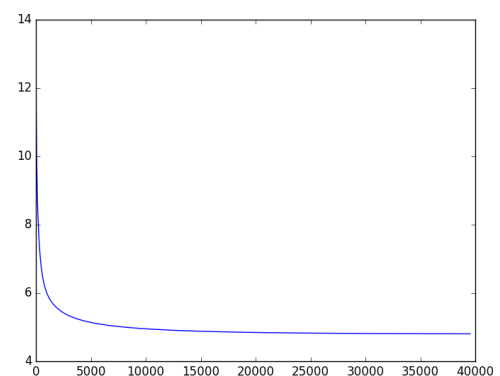
我特別用一個 regularization lambda 為 0 的做比較 error chart 做比較。可以發現, 有使用 regularization 的 Kaggle 成績都比沒用的好, 因此確實會去除雜訊的情形, 增加準確性, 而 regularization 愈大會讓 error 變更低(可以當做一個指標校正 lambda 的指標), 在 Kaggle 上的分數也有比較多的進步 (8.31→7.94)。可以知道 regularization 對 error 並不是數字上的影響(應跟 data 無關), 主要是在影響不要讓 model overfitting。

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？）

我在 data processing 及 model 的選取都有做優化。首先，由於 data 有相當程度的亂，在某幾個天中有很多連續的小時資料都是 0，因此我在取 data 時只要發現某個小時出現了 0，就 drop 跟這筆有關的數據，數據點從 5640→4836，rule 掉快 800 筆資料，而我也在 PM2.5 的值超過正常範圍時也 drop 掉此筆數據。以上將數據處理成最佳化，比較：



未處理過



處理過

可發現 data 在未處理過的 train error 就比處理過的高很多(22:6)，因 data 太髒。

而我的 model 將溫度、甲烷、一氧化碳、跟風有關的參數刪掉，然後風向取 sin 函數並與 wind speed 做相乘(有點像取 flux)，並都加入 2 次項，共 15 個 feature，再做 regularization(factor 10000)進行 predict。得到 priv+pub = 7.89 error(只取 18 個 feature 並做一樣的 regular: 7.94)