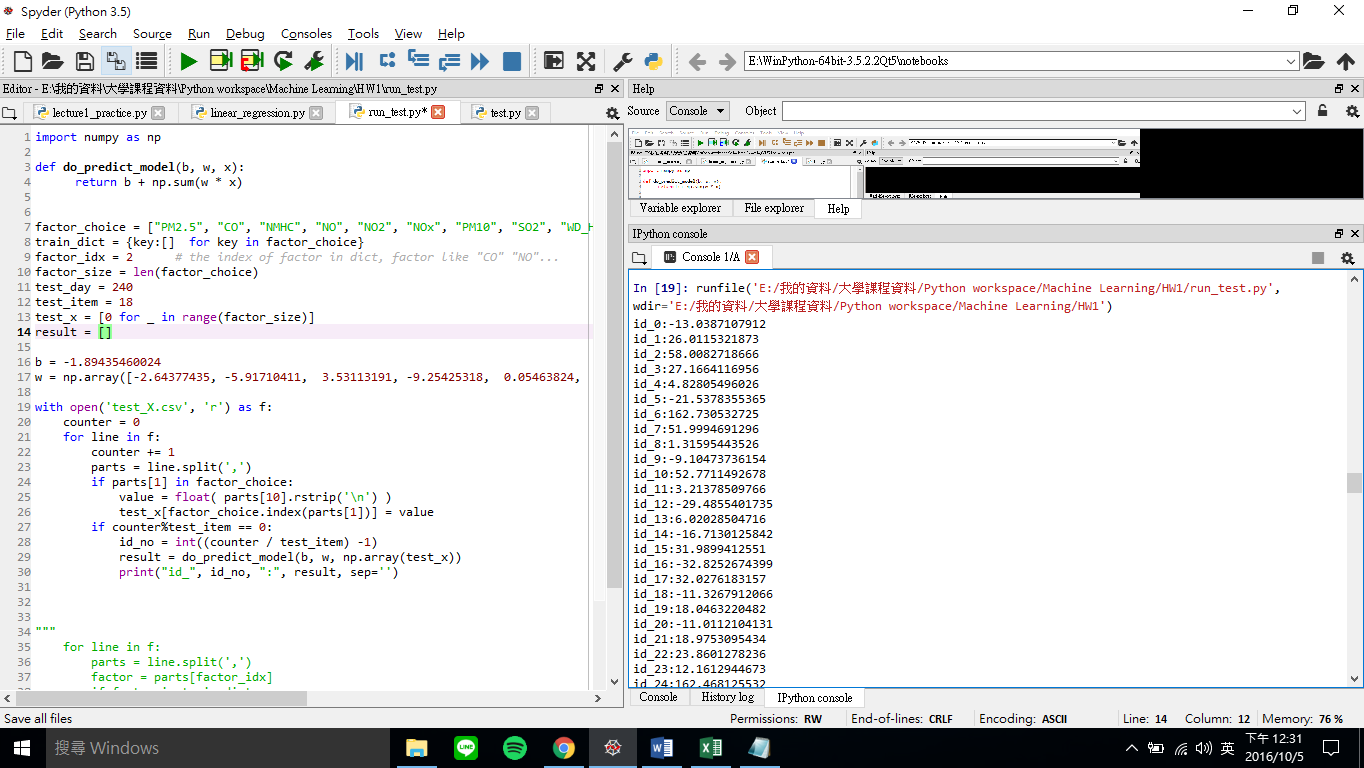
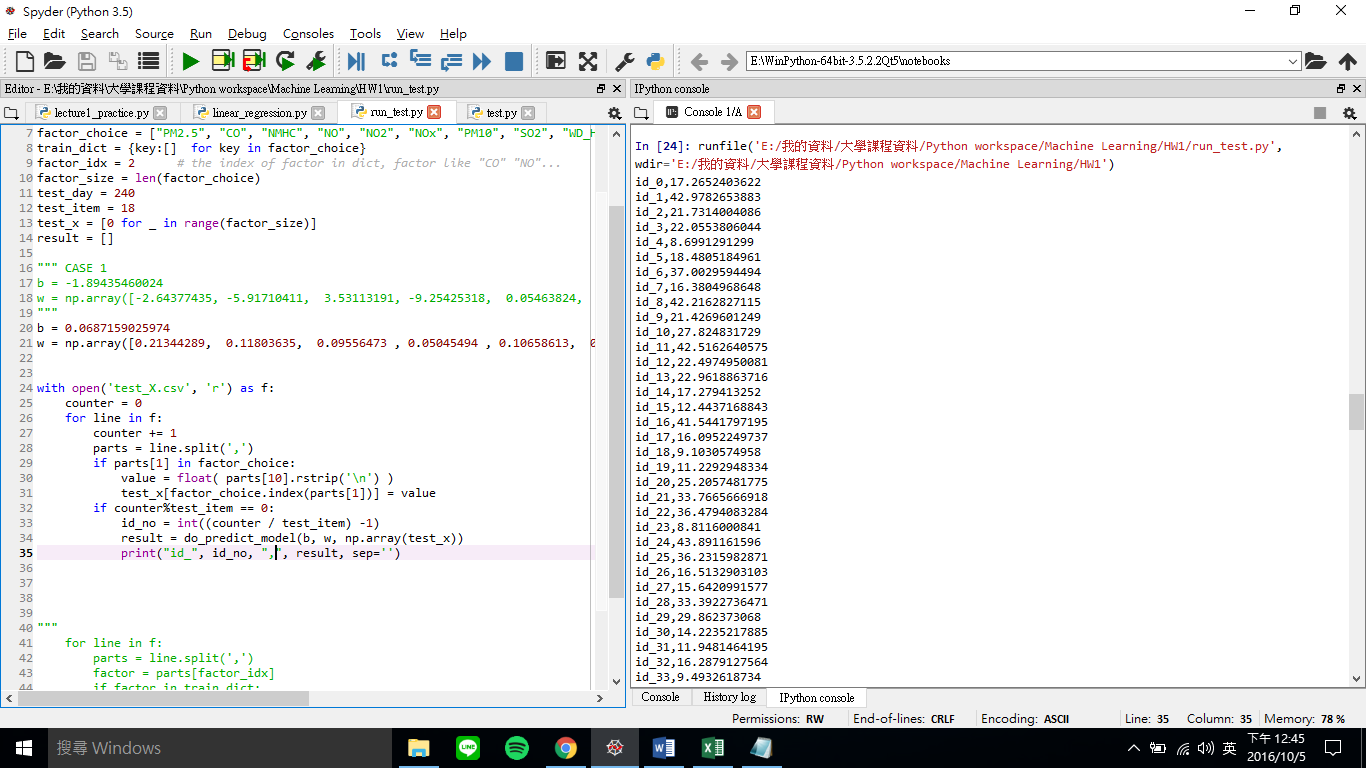
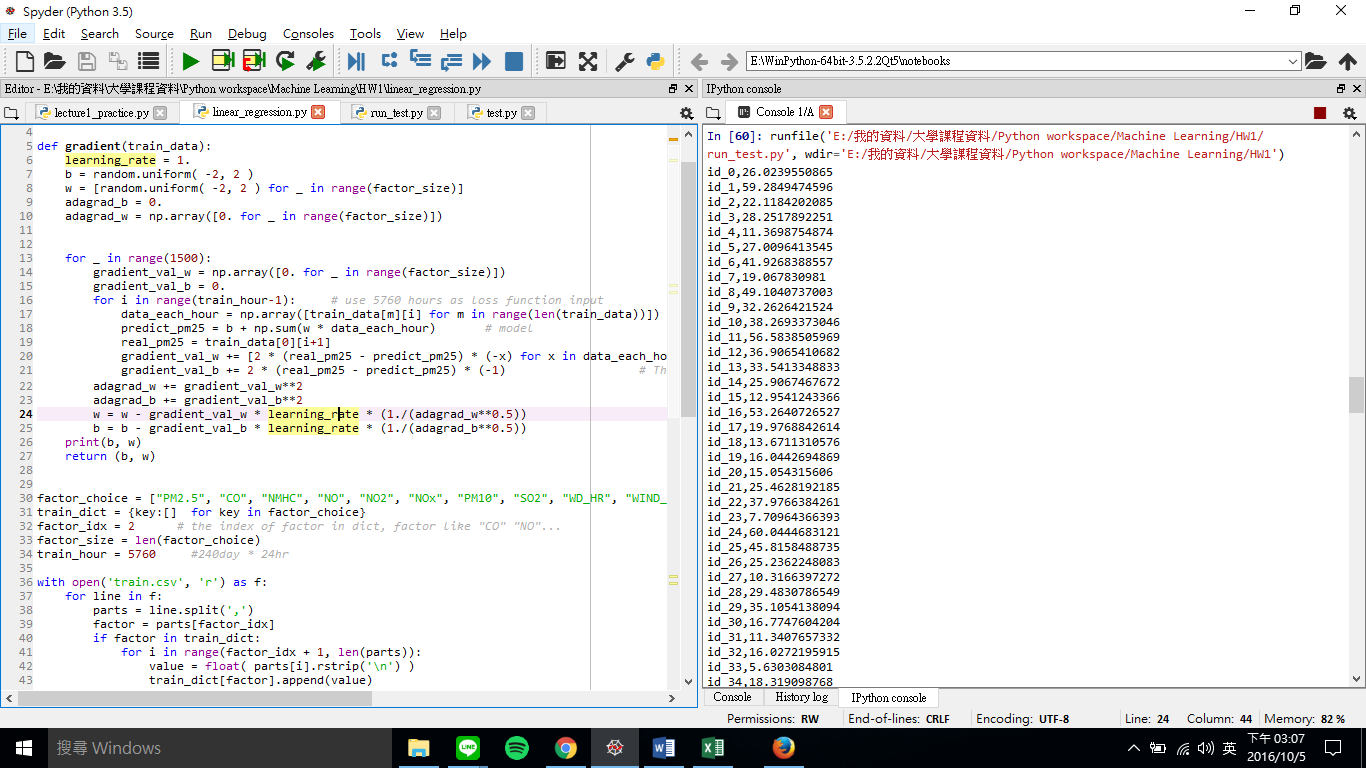
* **CASE 1 = 10/5 12:30** 
  + Model: y = b + wx
  + Feature: "PM2.5", "CO", "NMHC", "NO", "NO2", "NOx", "PM10", "SO2", "WD\_HR", "WIND\_DIREC", "WIND\_SPEED”
  + W = [-2.64377435 -5.91710411 3.53113191 -9.25425318 0.05463824 7.16766414  
    0.55116676 7.77147417 0.00990028 -0.21596661 2.77345293]
  + B = -1.89435460024
  + 資料差距太大 有負值

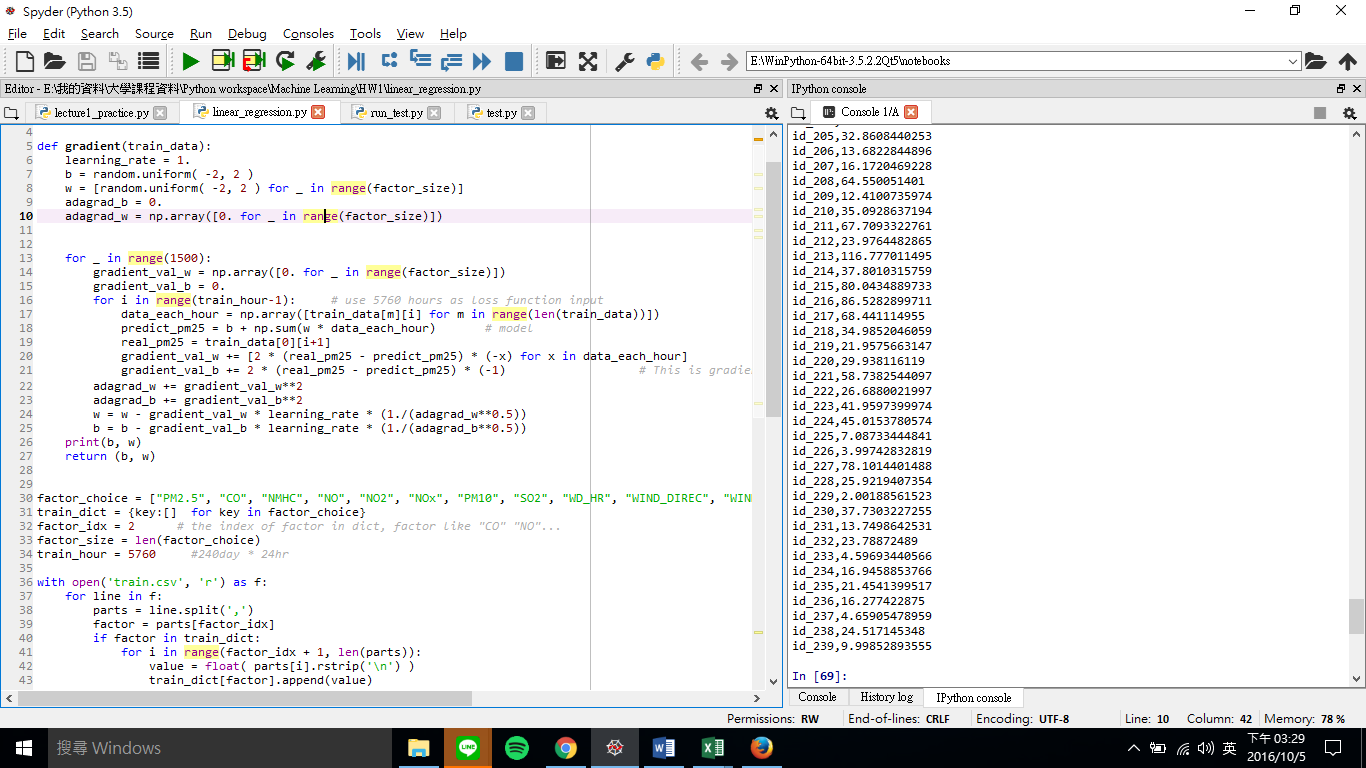


* **CASE 2 : 2016.10.5 12:45** 
  + Model: y = b + wx
  + Feature: "PM2.5", "CO", "NMHC", "NO", "NO2", "NOx", "PM10", "SO2", "WD\_HR", "WIND\_DIREC", "WIND\_SPEED”
  + W = [-2.64377435 -5.91710411 3.53113191 -9.25425318 0.05463824 7.16766414  
    0.55116676 7.77147417 0.00990028 -0.21596661 2.77345293]
  + B = -1.89435460024
  + SCORE: **12.12416，**資料看起來正常多，
* 
* CASE 3 : 2016.10.5 15:00
  + Model: y = b + wx learning\_rate=1. 1000times
  + Feature: "PM2.5", "CO", "NMHC", "NO", "NO2", "NOx", "PM10", "SO2", "WD\_HR", "WIND\_DIREC", "WIND\_SPEED”
  + b = 1.27894336385
  + w = np.array([8.08100095e-01 , 1.84496182e+00, -1.42520850e+00 , -8.93725771e-01, -9.64702230e-01 , 1.02041287e+00 , 5.46304771e-02, 1.50457841e-02, 1.06722697e-03 , 1.72807478e-03 , -4.87326585e-01])

|  |  |
| --- | --- |
| * + SCORE: | 6.77470 run 4 minutes |

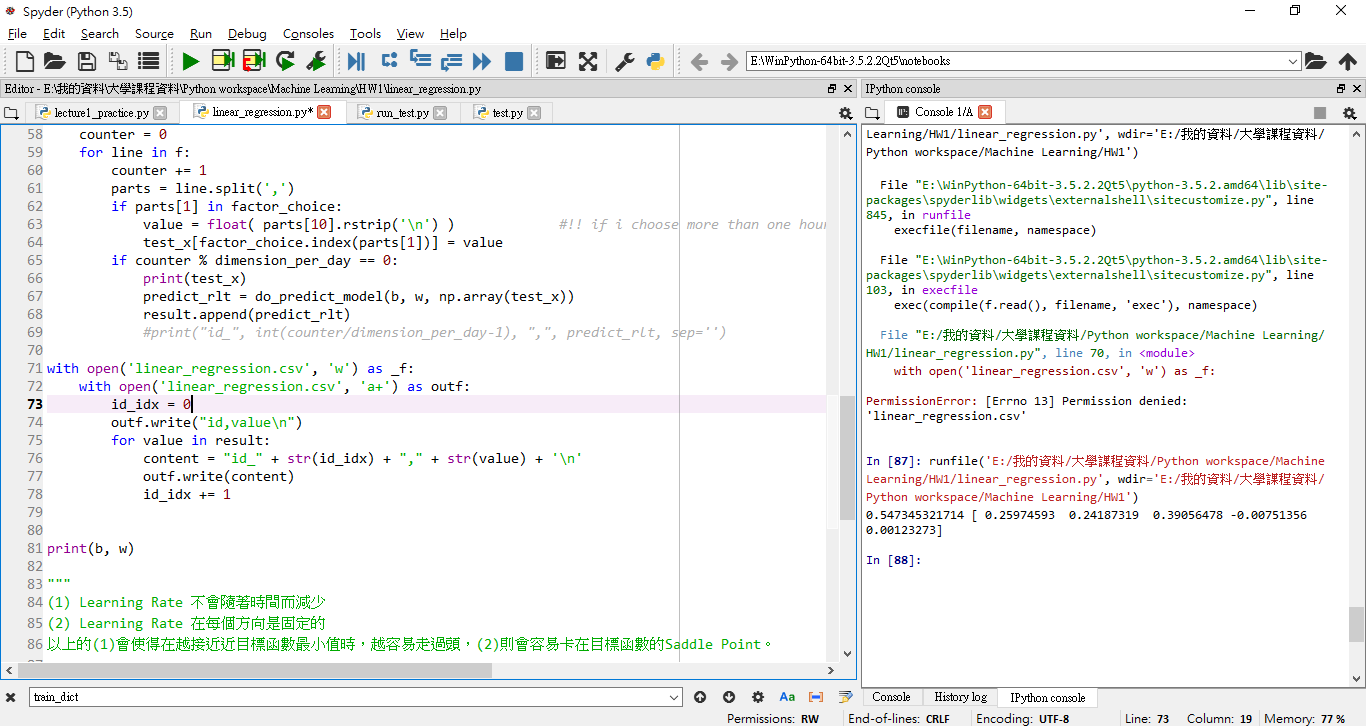
* + 
* CASE 4 : 2016.10.5 15:28 CSIEMAN
  + Model: y = b + wx learning\_rate=1. 1500times
  + Feature: "PM2.5", "CO", "NMHC", "NO", "NO2", "NOx", "PM10", "SO2", "WD\_HR", "WIND\_DIREC", "WIND\_SPEED”
  + b = -2.16391000572
  + w = np.array([8.07986354e-01 , 1.83228683e+00, 4.38743231e-01, -1.60063694e-01, -1.17341965e-01, 2.42075798e-01, 5.66894905e-02 , 1.60645585e-01, 9.79858420e-04, 2.76500861e-03, 2.40851184e-01])

|  |  |
| --- | --- |
| * + SCORE: | 6.64570 run 5 minutes 目前最佳 |

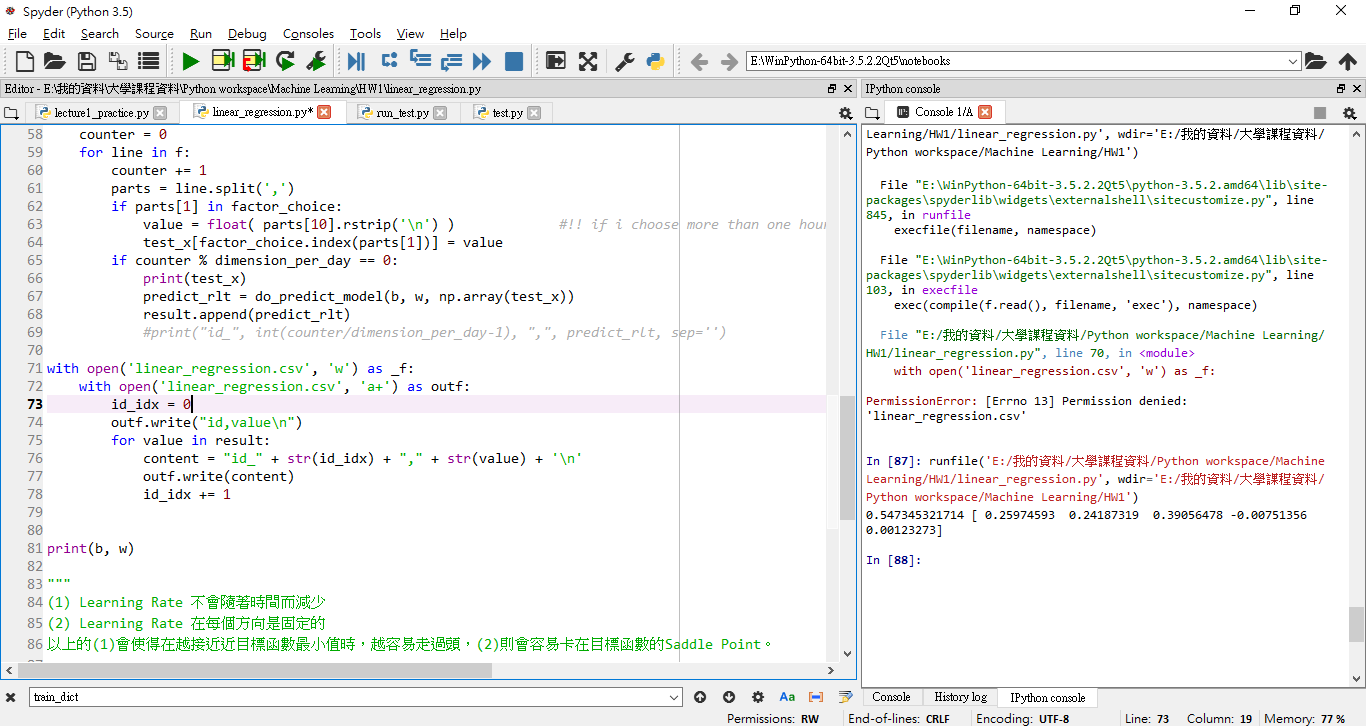
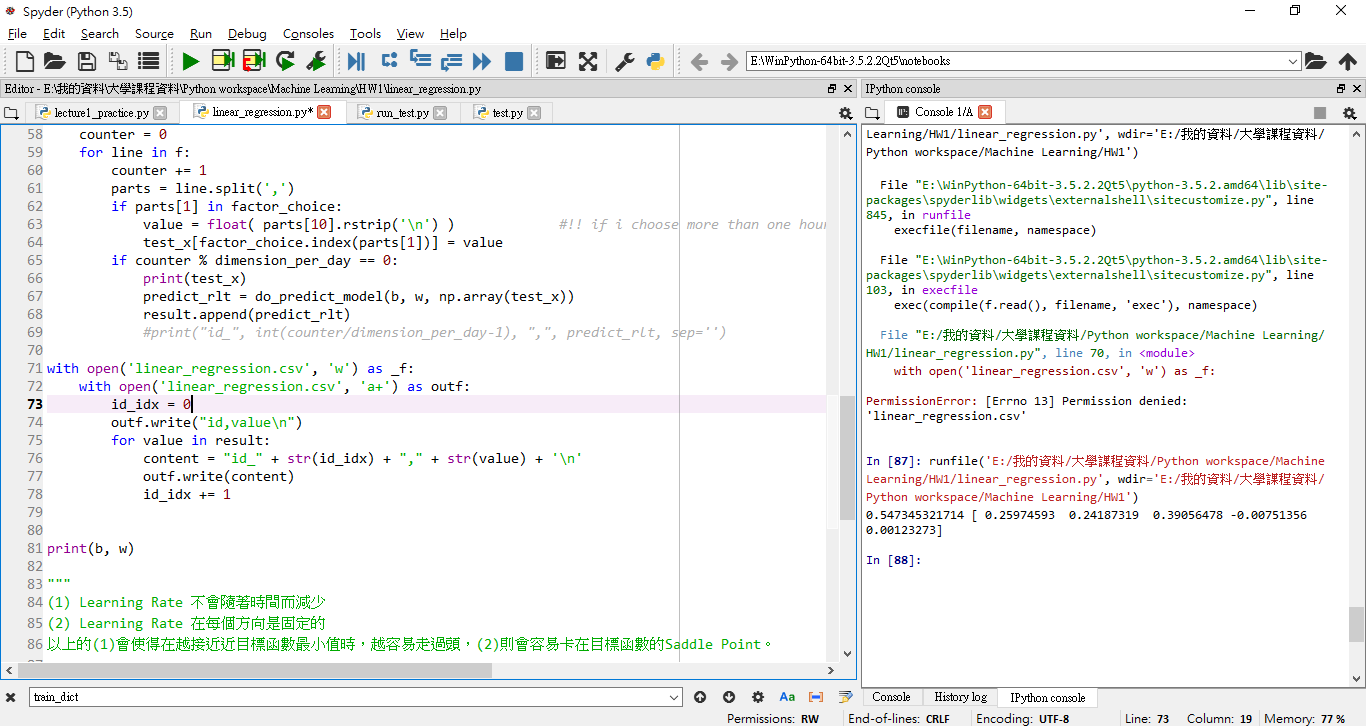


* CASE 5 : 2016.10.5 16:32 CSIEMAN
  + Model: y = b + wx learning\_rate=1. 100 times
  + Feature: "PM2.5", "CO","PM10","WD\_HR", "WIND\_DIREC"
  + b = 0.547345321714
  + w = 0.25974593 0.24187319 0.39056478 -0.00751356 0.00123273

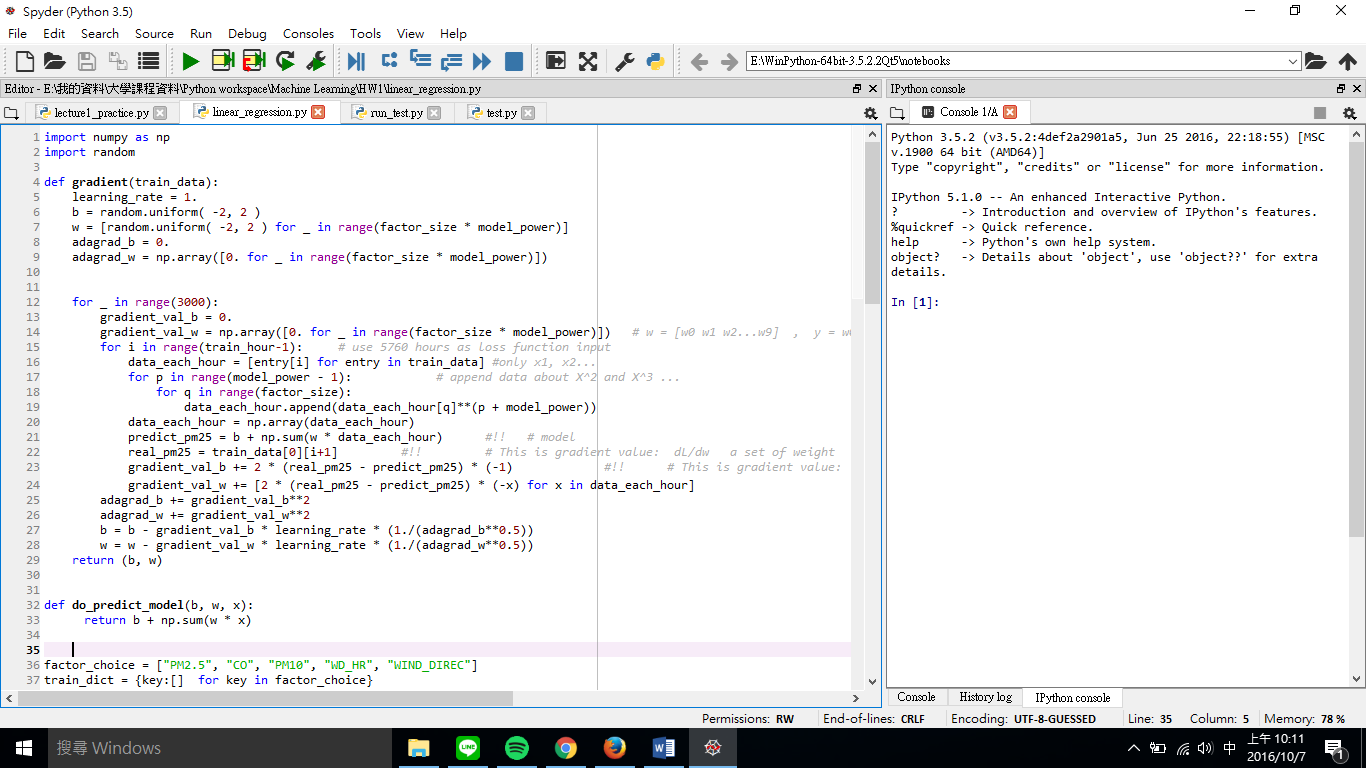
|  |  |
| --- | --- |
| * + SCORE: | **10.00700** |

*  減少feature但是run不夠多次
* CASE 6 : 2016.10.5 16:45 CSIEMAN
  + Model: y = b + wx learning\_rate=1. 1000 times
  + Feature: "PM2.5", "CO","PM10","WD\_HR", "WIND\_DIREC"
  + b = -0.567120006787
  + w = 8.15310516e-01 2.55793851e+00 6.30571705e-02 2.37966947e-03 3.06744569e-03

|  |  |
| --- | --- |
| * + SCORE: | **6.68695** run 5 minutes |

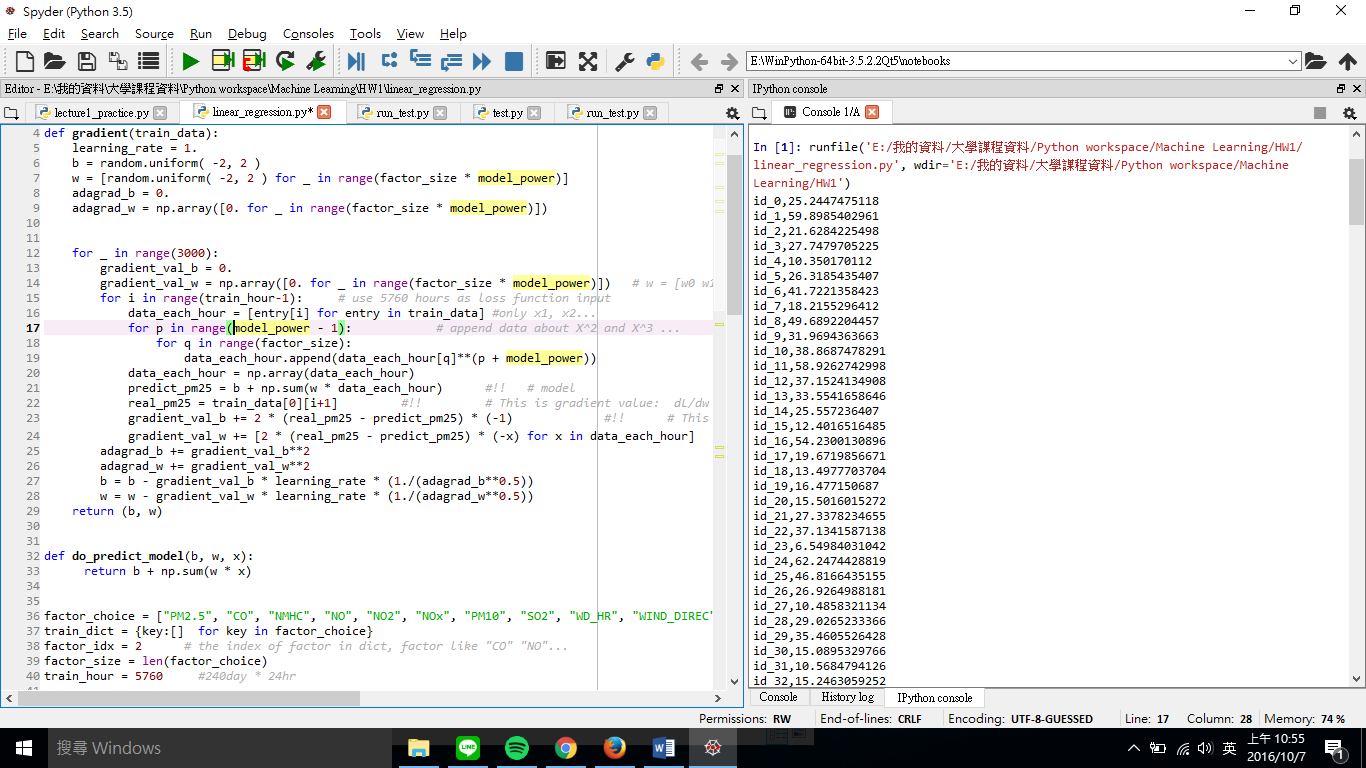
* 
* CASE 7 : 2016.10.5 17:05 CSIEMAN
  + Model: y = b + wx learning\_rate=1. 2000 times
  + Feature: "PM2.5", "CO","PM10","WD\_HR", "WIND\_DIREC"
  + b = 0.460183891334
  + w = 8.17087904e-01, 2.07985670e+00 , 5.58957478e-02 ,1.29480164e-03 , 1.51157750e-03
  + SCORE: 6.69732 run 6 min run比較多次沒比較好
* CASE 8 : 2016.10.5 23:00 CSIEMAN
  + Model: y = b + wx + wx^2 learning\_rate=1. 3000 times
  + Feature: "PM2.5", "CO","PM10","WD\_HR", "WIND\_DIREC"
  + b = ???
  + w = ???

SCORE: 33多 run 6 min 結果超差，剛換二次式懷疑寫錯

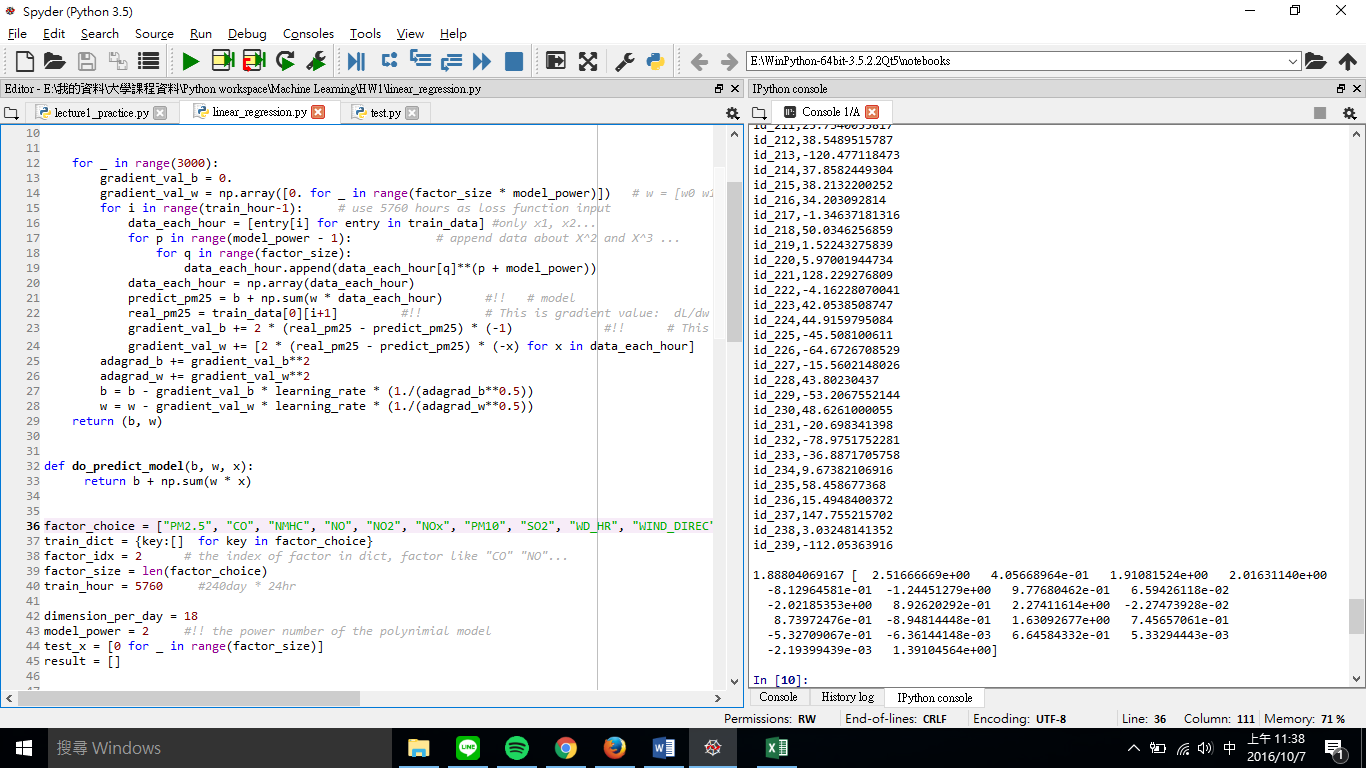


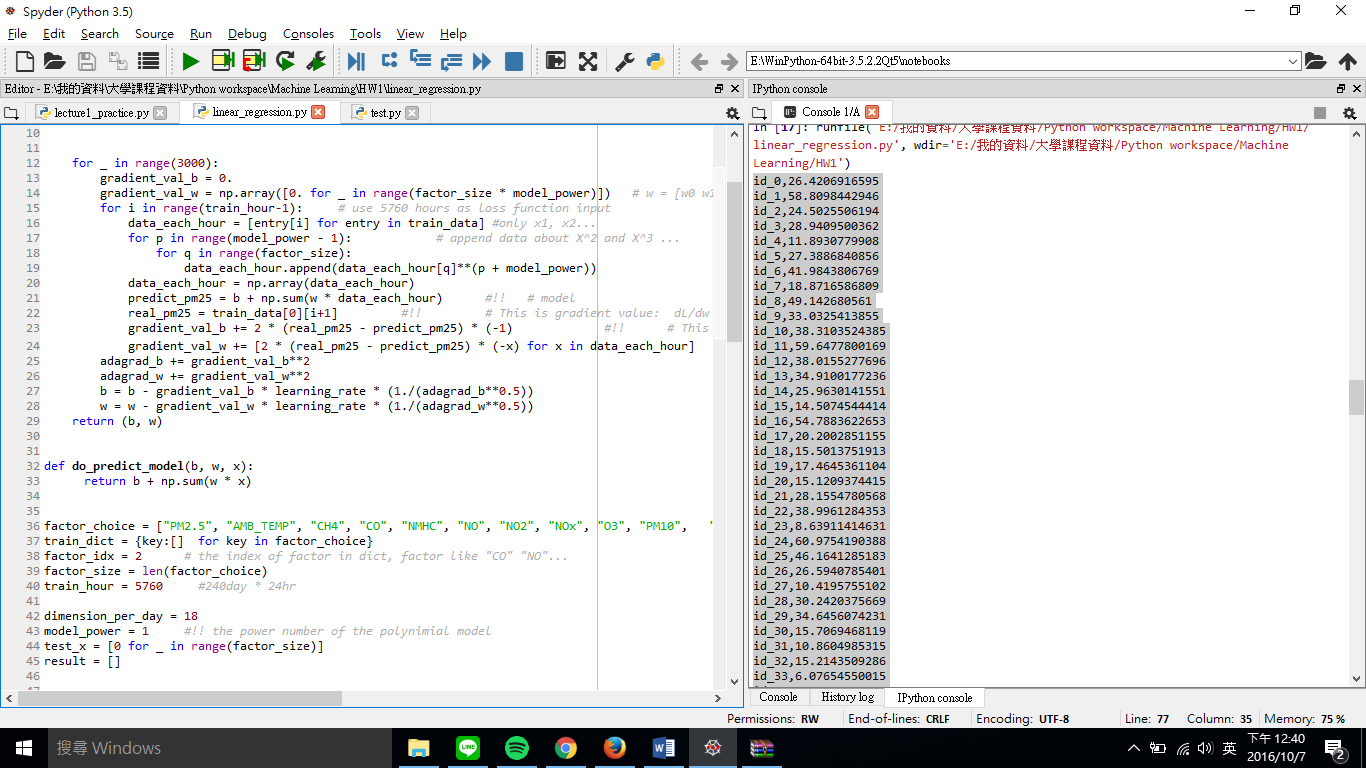
* CASE 9 : 2016.10.7 11:00 henry
  + Model: y = b + wx learning\_rate=1. 3000 times
  + Feature: "PM2.5", "CO", "NMHC", "NO", "NO2", "NOx", "PM10", "SO2", "WD\_HR", "WIND\_DIREC", "WIND\_SPEED”
  + b = -1.26561293275
  + w = 8.07875042e-01 1.26548337e+00 1.07993596e+00 1.01602640e+00 1.03365789e+00 -9.36067767e-01 5.44021460e-02 2.61461645e-01 3.01196053e-04 2.36444612e-03 4.62582875e-02

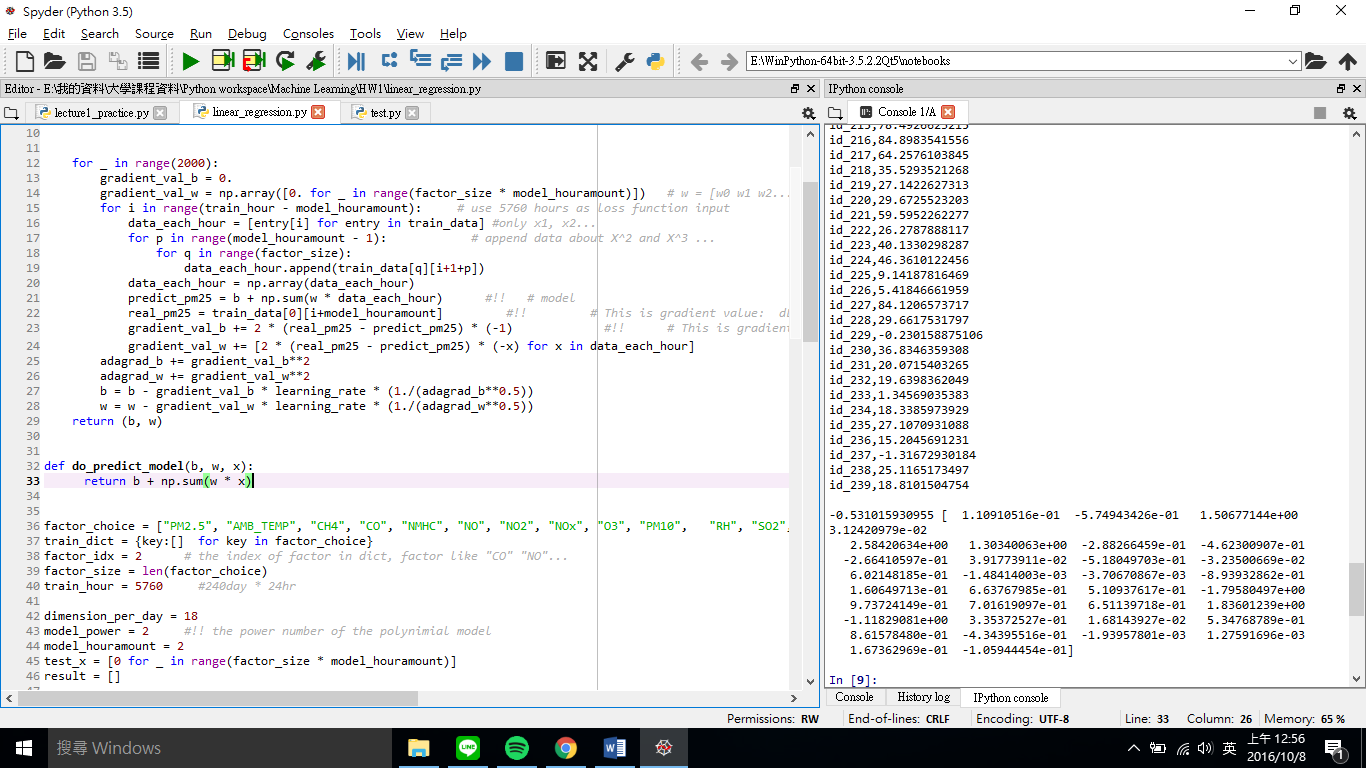
SCORE: 6.65843 這是用之前case4最好成績再跑多run次，結果差不多 這是一次式的最佳極限了



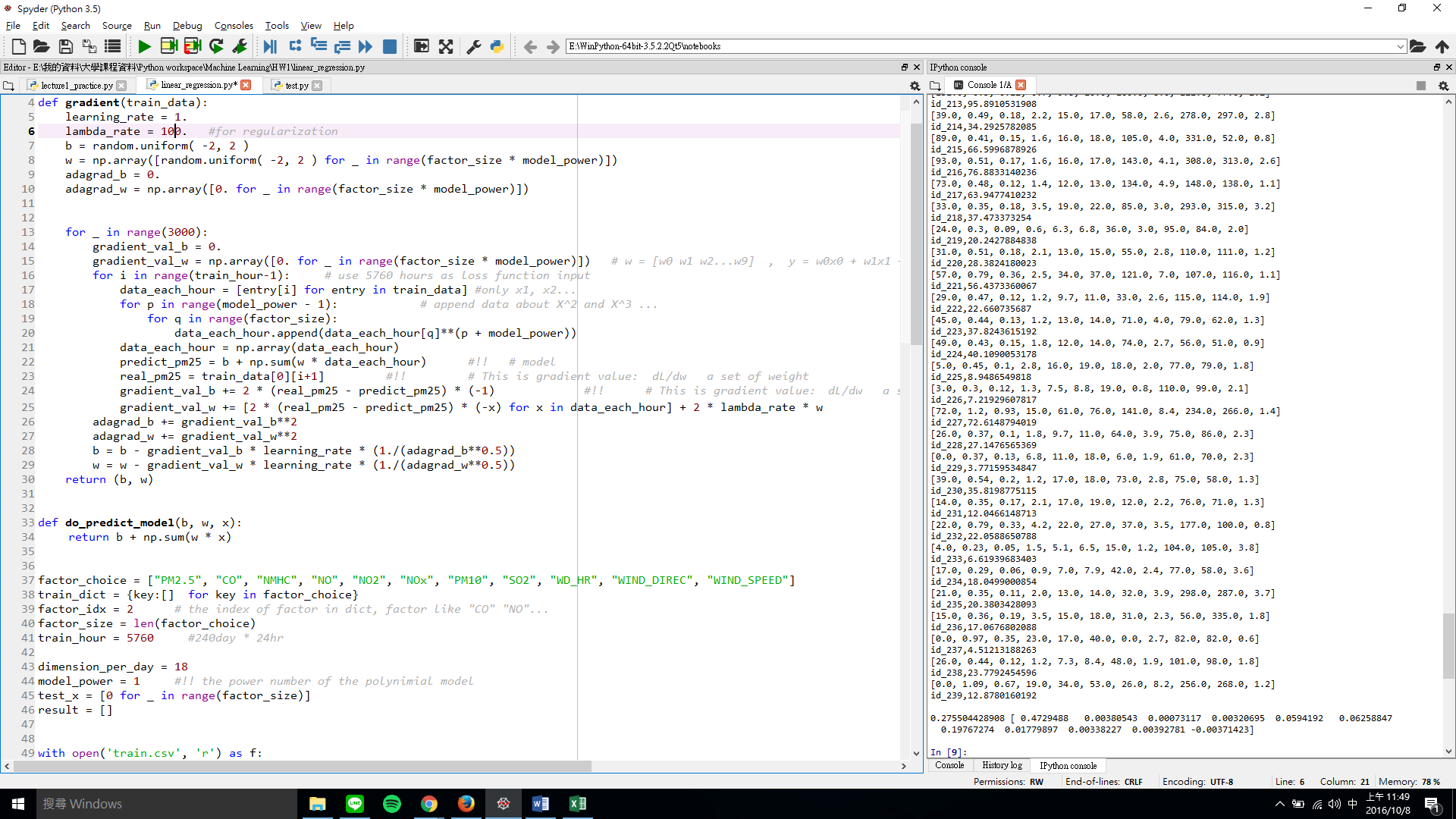
* CASE 10 : 2016.10.7 11:38 henry
  + Model: y = b + wx + wx^2 learning\_rate=1. 3000 times
  + 11 Feature: "PM2.5", "CO", "NMHC", "NO", "NO2", "NOx", "PM10", "SO2", "WD\_HR", "WIND\_DIREC", "WIND\_SPEED”
  + b = 1.88804069167
  + w = 2.51666669e+00 4.05668964e-01 1.91081524e+00 2.01631140e+00 -8.12964581e-01 -1.24451279e+00 9.77680462e-01 6.59426118e-02 -2.02185353e+00 8.92620292e-01 2.27411614e+00 -2.27473928e-02 8.73972476e-01 -8.94814448e-01 1.63092677e+00 7.45657061e-01 -5.32709067e-01 -6.36144148e-03 6.64584332e-01 5.33294443e-03 -2.19399439e-03 1.39104564e+00
  + 結果極差 沒有丟上去KAGGLE 二次式真的滿G的 可能要試試一次式然後取多個小時前



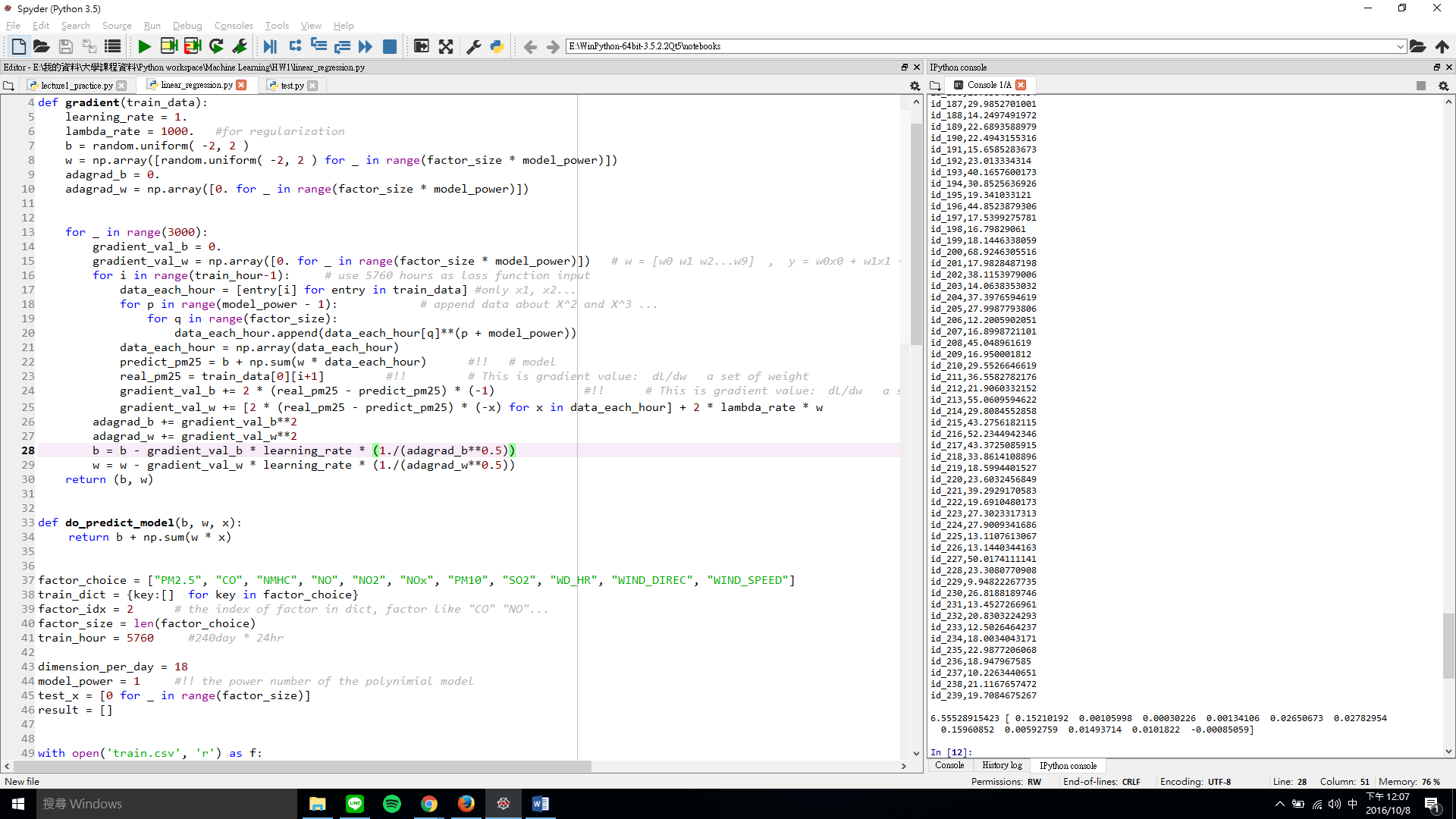
* CASE 11 : 2016.10.7 12:40 henry
  + Model: y = b + wx learning\_rate=1. 3000 times
  + 18 Feature
  + b = ?
  + w = ?
  + SCORE: 6.68573 用了全部的feature 一次式的分數還是上不去 一次式的極限就是這樣了 除非再用regularization 或是多小時前feature
* 
* CASE 12 2016.10.8 凌晨12:54 henry
  + Model: y = b + wx learning\_rate=1. 2000 times 取2小時
  + 18 Feature
  + b = -0.531015930955
  + w = 1.10910516e-01 -5.74943426e-01 1.50677144e+00 3.12420979e-02 2.58420634e+00 1.30340063e+00 -2.88266459e-01 -4.62300907e-01 -2.66410597e-01 3.91773911e-02 -5.18049703e-01 -3.23500669e-02 6.02148185e-01 -1.48414003e-03 -3.70670867e-03 -8.93932862e-01 1.60649713e-01 6.63767985e-01 5.10937617e-01 -1.79580497e+00 9.73724149e-01 7.01619097e-01 6.51139718e-01 1.83601239e+00 -1.11829081e+00 3.35372527e-01 1.68143927e-02 5.34768789e-01 8.61578480e-01 -4.34395516e-01 -1.93957801e-03 1.27591696e-03 1.67362969e-01 -1.05944454e-01
* SCORE: 6.89817 用了全部的feature 一次式 而且**取前兩個小時** 分數還是上不去



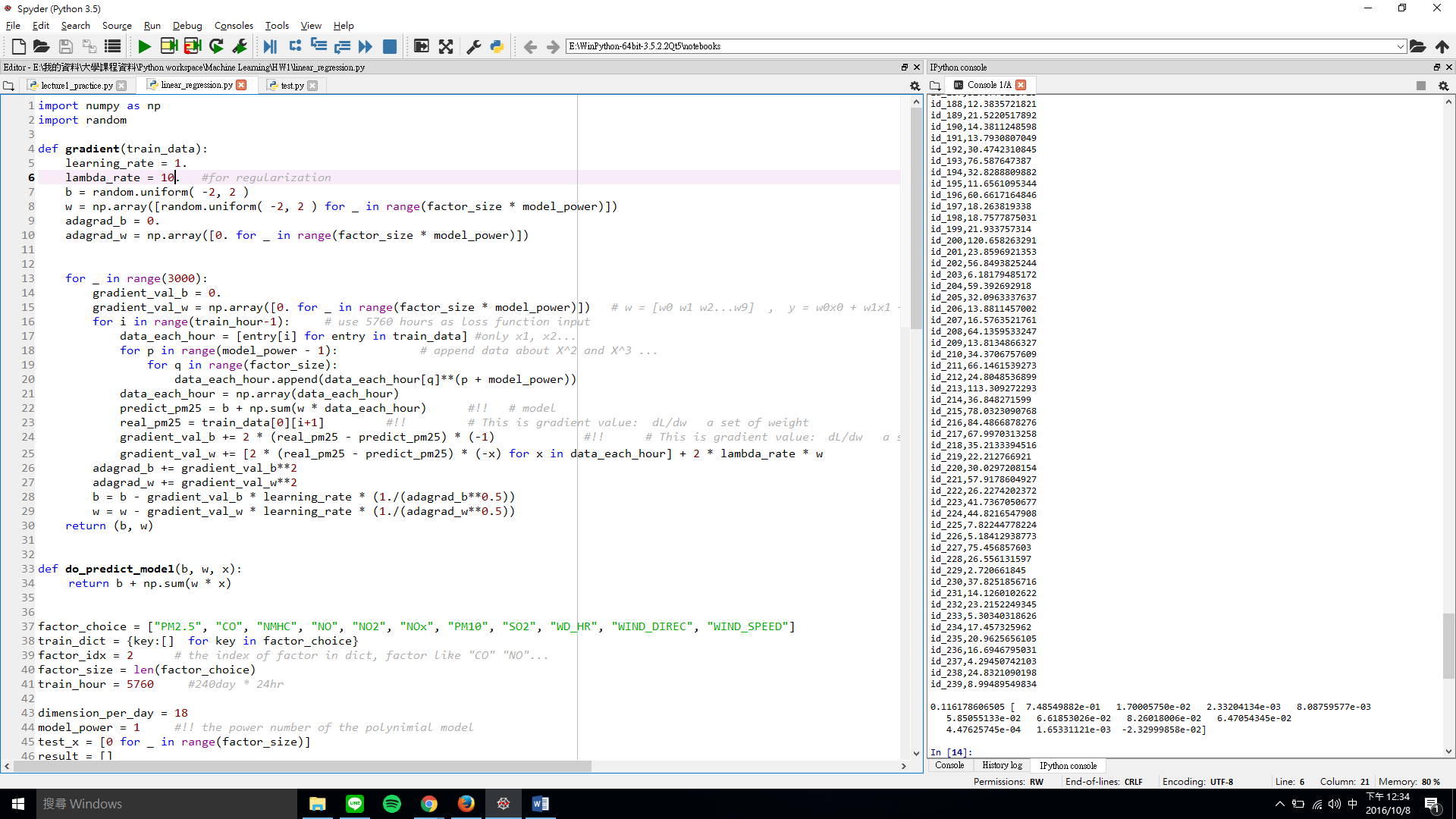
* CASE 13 2016.10.8 11:50 henry
  + Model: y = b + wx learning\_rate=1. Lambda=100 3000 times
  + 11 Feature 取1小時
  + b = 0.275504428908
  + w = [ 0.4729488 0.00380543 0.00073117 0.00320695 0.0594192 0.06258847 0.19767274 0.01779897 0.00338227 0.00392781 -0.00371423]
  + SCORE: 8.25949 改成regularization法 分數不好 要調lambda



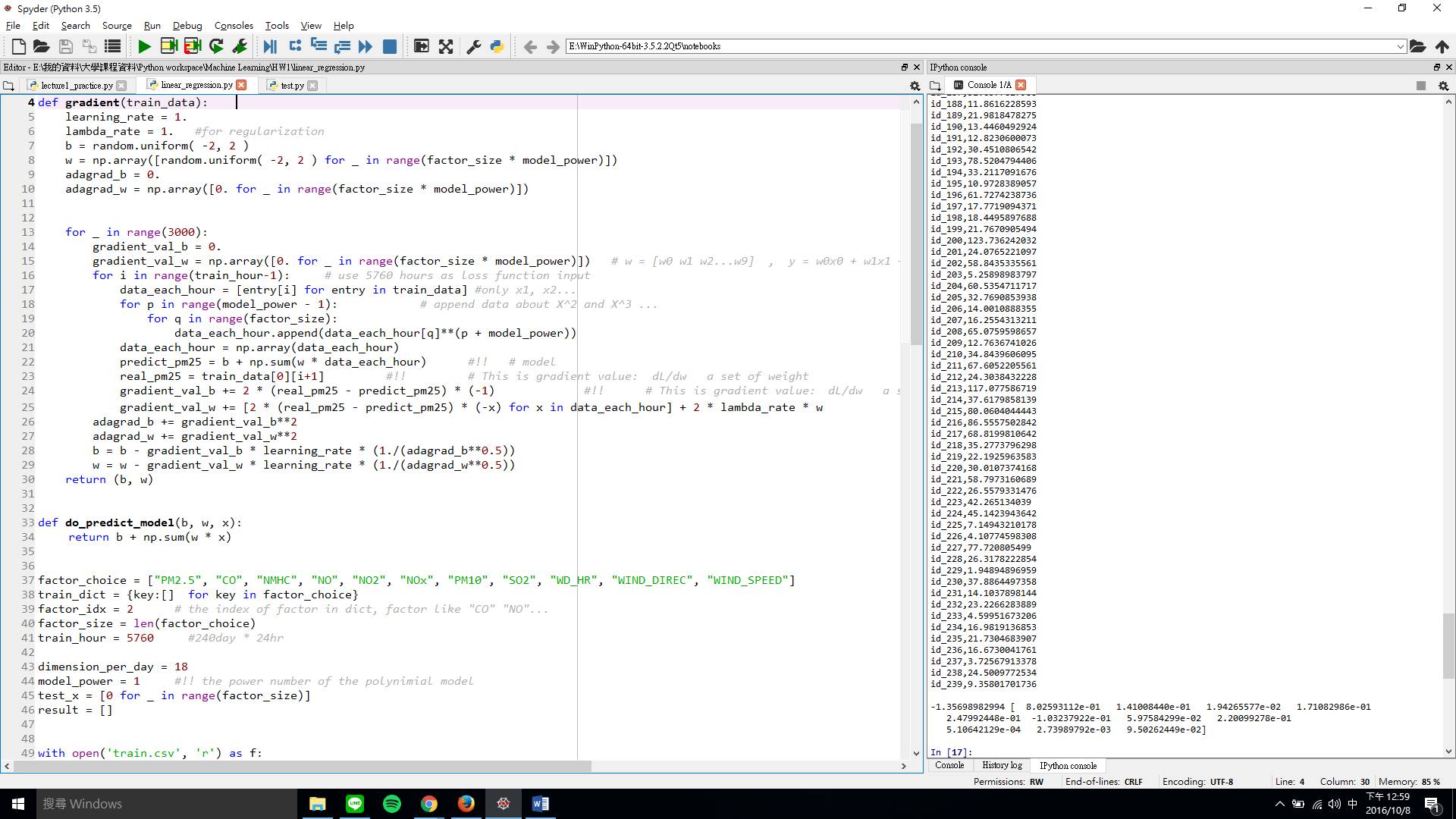
* CASE 14 2016.10.8 12:08 CSIEMAN
  + Model: y = b + wx learning\_rate=1. Lambda=1000 3000 times
  + 11 Feature 取1小時
  + b = 6.55528915423
  + w = [0.15210192 0.00105998 0.00030226 0.00134106 0.02650673 0.02782954 0.15960852 0.00592759 0.01493714 0.0101822 -0.00085059]
  + SCORE: 13.47063 改成regularization法 lambda改1000 分數更差

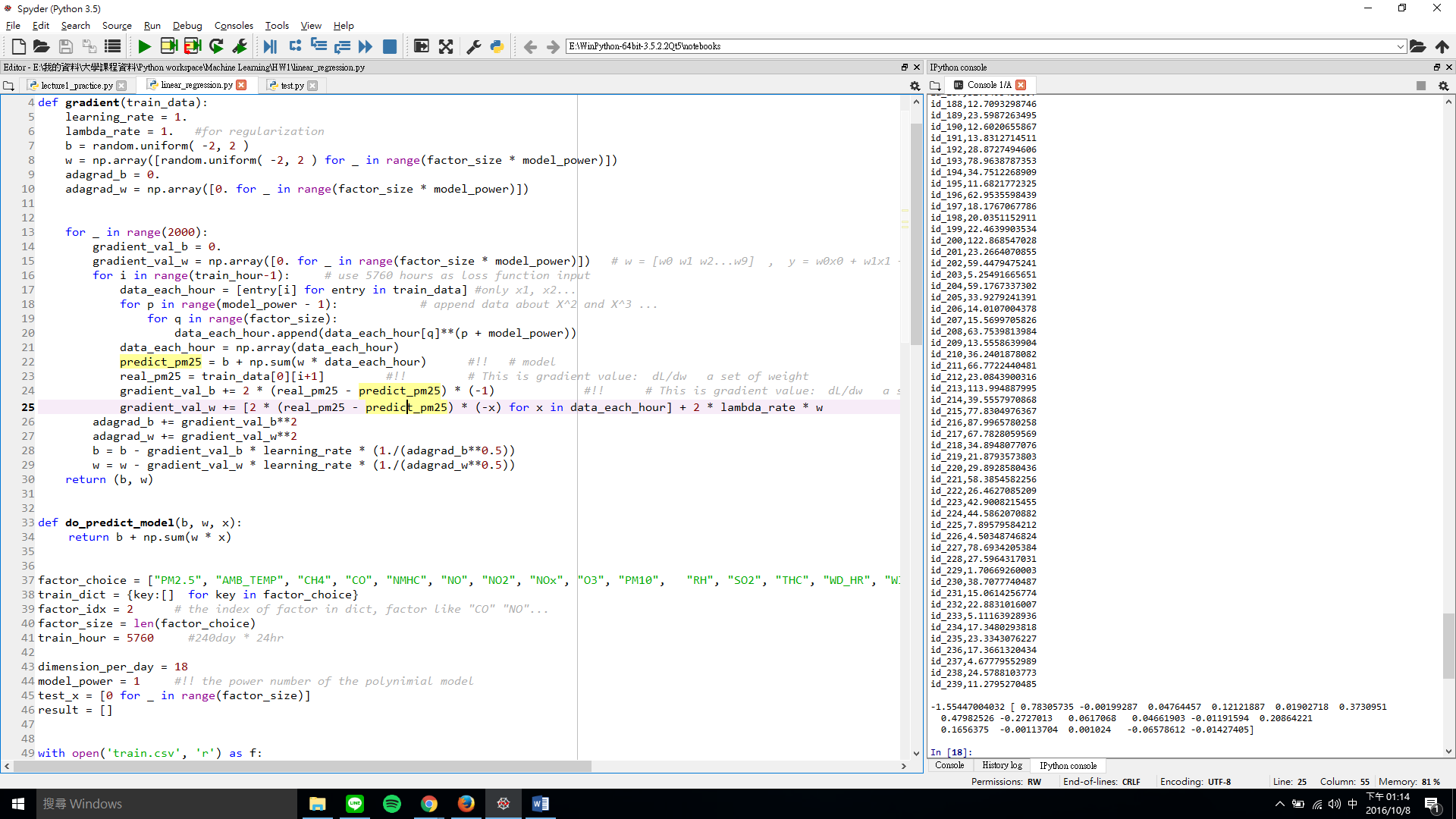


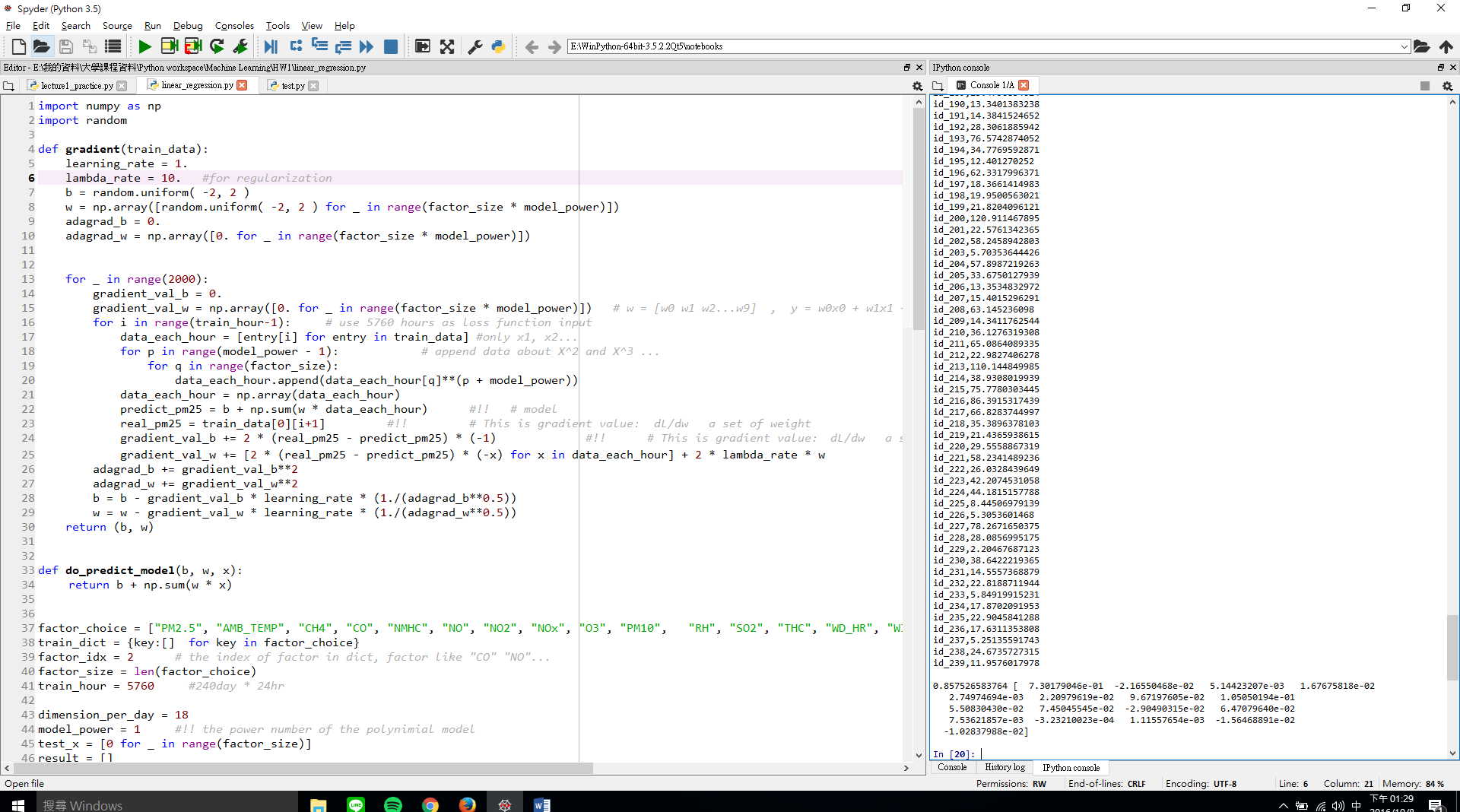
* Case 15 : 2016.10.8 12:30 CSIEMAN
  + Model: y = b + wx learning\_rate=1. Lambda=10 3000 times
  + 11 Feature 取1小時
  + B = 0.116178606505
  + W = [ 7.48549882e-01 1.70005750e-02 2.33204134e-03 8.08759577e-03 5.85055133e-02 6.61853026e-02 8.26018006e-02 6.47054345e-02 4.47625745e-04 1.65331121e-03 -2.32999858e-02]
  + SCORE: 6.84774 regularization法 lambda改10 分數好一點還是不夠



* Case 16 : 2016.10.8 12:50 CSIEMAN
  + Model: y = b + wx learning\_rate=1. Lambda=1 3000 times
  + 11 Feature 取1小時
  + B = -1.35698982994
  + W = [ 8.02593112e-01 1.41008440e-01 1.94265577e-02 1.71082986e-01 2.47992448e-01 -1.03237922e-01 5.97584299e-02 2.20099278e-01 5.10642129e-04 2.73989792e-03 9.50262449e-02]
  + SCORE: 6.68584 regularization法 lambda改1 分數好一點還是不夠



* Case 17 : 2016.10.8 1:15 CSIEMAN
  + Model: y = b + wx learning\_rate=1. Lambda=1 2000 times
  + 18 Feature 取1小時
  + B = -1.55447004032
  + W = [ 0.78305735 -0.00199287 0.04764457 0.12121887 0.01902718 0.3730951 0.47982526 -0.2727013 0.0617068 0.04661903 -0.01191594 0.20864221 0.1656375 -0.00113704 0.001024 -0.06578612 -0.01427405]
  + SCORE: 6.63914(**最佳**) regularization法 lambda改1且18feature 目前最好
  + 
* Case 18 : 2016.10.8 1:30 CSIEMAN
  + Model: y = b + wx learning\_rate=1. Lambda=10 2000 times
  + 18 Feature 取1小時
  + B = 0.857526583764
  + W = [7.30179046e-01 -2.16550468e-02 5.14423207e-03 1.67675818e-02 2.74974694e-03 2.20979619e-02 9.67197605e-02 1.05050194e-01 5.50830430e-02 7.45045545e-02 -2.90490315e-02 6.47079640e-02 7.53621857e-03 -3.23210023e-04 1.11557654e-03 -1.56468891e-02 -1.02837988e-02]
  + SCORE: 6.75505 lambda改10 分數又壞了



* Case 19 2016.10.8 15:30 CSIEMAN
  + Model: y = b + wx learning\_rate=1. Lambda=1 2000 times
  + 真正完整feature 取2小時
  + B = 1.53012311962
  + W = [ 1.93766035e-01 9.25642748e-01 -4.18943913e-02 5.49310360e-02 3.42337444e-03 1.22397272e+00 7.83032148e-01 -1.23622953e+00 -5.35977974e-01 -9.21152993e-02 7.28780302e-03 -1.03728845e+00 -1.66624374e-01 1.75179640e-01 -4.35619069e-03 -5.64282730e-03 -4.69191519e-01 1.39443472e-01   
     5.92631467e-01 -9.55163593e-01 2.95133319e-01 1.24547052e-01 1.20432459e-02 -1.18748340e-01 3.37540108e-02 5.06856205e-01 5.74910919e-01 1.37934074e-01 -3.05303846e-01 1.01275395e+00 4.28847618e-01 7.48434768e-02 5.14422754e-04 5.15137536e-03 2.34814292e-01 -2.89948315e-02
  + SCORE: 7.61004 多了RAINFALL 分數還是不夠
* Case 20 2016.10.8 16:58 yahoo
  + Model: y = b + wx learning\_rate=1. Lambda=1 2000 times
  + Feature: "PM2.5", "CO", "NO", "NO2", "NOx", "RAINFALL", "SO2", "THC" 取2小時
  + B = 0.0508857753694
  + W = [ -0.08027245 0.07324646 0.16622619 -0.067389 -0.02531188 -0.05376222 0.07439196 0.04906694 0.94323841 0.13350882 -0.07545685 0.18279842 0.05029177 -0.085203 0.26024616 0.07660513
  + SCORE: 6.43572(**目前最佳**) 改選其他feature 分數瞬間上衝
* Case 21 2016.10.8 17:29 yahoo
  + Model: y = b + wx learning\_rate=1. Lambda=1 2000 times
  + Feature: " "PM2.5", "CO","NMHC", "NO", "NO2", "NOx", "RAINFALL", "SO2", "THC"" 取3小時
  + B = 0.0795018163077
  + W = [-0.13795388 0.01620063 0.00576697 0.03600788 -0.05252885 0.00147303 -0.00815637 -0.05141201 0.08823023 0.0617765 0.06885113 0.01270805 0.02863849 -0.13109014 0.08116978 -0.04641846 0.08827193 0.08915929 0.92927864 0.12695297 0.018396 -0.15287764 0.11700625 0.11970432 -0.08848659 0.25965938 0.11725467
  + SCORE: 6.22982(**目前最佳**) 改選其他feature 分數瞬間上衝
* Case 22 2016.10.8 19:29 pig
  + Model: y = b + wx learning\_rate=1. Lambda=1 2000 times
  + Feature: PM2.5", "CH4", "CO","NMHC", "NO", " "NO2", "NOx", "RAINFALL", "SO2", "THC"] 取3小時
  + B = 0.0187582993244
  + W = [[-0.14002614 0.05180971 0.01735818 0.00633271 0.23007078 0.16788217 -0.21628144 -0.00762487 -0.05445932 0.06012844 0.06061044 0.05083399 0.06866817 0.01356127 0.22263674 0.10985388 -0.14687985 -0.04600442 0.08672146 0.06071731 0.93017613 0.07149706 0.12522793 0.01922474 -0.06394038 0.19089983 0.03838468 -0.08821245 0.26505225 0.08880045]
  + SCORE: 6.22495(**目前最佳**) 加了CH4 沒甚麼用
* 加PM10變6.29489 變差
* 去掉 THC 變6.21905 變好
* 去掉 , "CH4", "CO" 變6.43760 變差
* 加, "WD\_HR" 變好
* 加” WIND\_DIREC” 沒有變太多
* 加WIND\_SPEED 變差
* 自己切validation set，取3小時的比取2小時差
* 1 hour error: 206 246 203 229 199
* 2 hour error: 156 206 161 194 169 262
* 3 hour error: 244 206 160 296 207
* 去CH4 : 1109 1071
* 去掉CO : 1095 1089
* 去NMHC : 1087 1060
* 去掉NO : 1077 1106
* 去NO2 : 1065 1144
* 去NOX: 1081 1113
* 去RAIN:
* 去SO2:
* 去WD\_HR:
* 去RAIN 2hour 1500 : 147 159 171 205
* 去RAIN lambda=10: 1082 1073
* Powe2 hour2 pm2.5 500 1978
* Poer3 以上 爆掉
* Powe2 hour2 pm2.5+NMHC 500 1672
* Powe2 hour2 pm2.5+NMHC+NOX 500 1439
* Powe2 hour2 1000time
  + "PM2.5", "NMHC", "NO", "NOx","RAINFALL", "WIND\_SPEED" : 1163
* Powe2 hour2 2000time
  + "PM2.5", "NMHC", "NO", "NOx","RAINFALL", "WIND\_SPEED" : 1106
* 如果要用2次式 lanbda要用100000
* 其他：這幾個feature值感覺滿相關PM2.5 CO PM10 WD\_HR WIND\_DIREC(稍微) 用learn=1 time=1500因為跑了幾次，發現這幾個feature的weight都滿接近，而其他weight都很懸殊，所以猜測挺重要
* 一次式的model好像最佳就6.68左右，要試試看二次