109062318簡弘哲

1. Basic part regression equation:

(x = temperature)

City a = -0.48105859\*x\*\*2 + 24.01307842\*x - 260.79126575

City b = -0.57694117\*x + 36.11129474

City c = -3.71290268\*x + 138.97913453

1. Variables in the advanced part

y = cases, x1 = temperature, x2 = precipitation, x3 = # of Houses per km^2

我定義的變數遵循以下規則:[城市名]\_[用途]\_[y,x1,x2,x3其中之一]，型別皆為list

ex.

(1) a\_train\_x1 = a list of “temperature” of city “a” which is for model “training”

(2) b\_test\_x2 = a list of “precipitation” of city “b” which is for model “testing”

(3) c\_predict\_x3 = a list of “number of Houses per km^2” of city “c” which is for “prediction”

(4) a\_train\_y = a list of “cases” of city “a” which is for model “training”……依此類推。

basic part只用溫度當input variable，利用numpy polyfit做regression。

advanced part除溫度外多用了兩個變數precipitation, # of houses per km^2，利用sklearn linear\_model做regression。

1. 遇到的困難與解法

(1) 一開始不會實作gradient descent的演算法，直到助教說可以用numpy polyfit，才解決這個問題

(2) 在最後壓MAPE的時候數字一直壓不下去，直到我改變training data的選取範圍與數量後(選取與test跟prediction較相近的資料進行訓練)，MAPE就好看多了

4. 反思

(1) 感覺這次作業的資料雖然大部分非常集中，但好像看不出一條非常明顯的回歸線，讓我一直懷疑它適不適合用regression。

(2) 了解到該如何分配training, testing data數量的比例才會讓MAPE最小