**Regression equation in basic:**

y\_hat[i] = w[0]\*x[i] + w[1]\*y[i-1] + w[2]\*y[i-2] + w[3]\*y[i-3] + w[4]\*y[i-4] + w[5]\*y[i-5] + b

（預測y = w[0]\*溫度 + w[1]\*前一天病例 + w[2]前兩天病例 + w[3]前三天病例 + w[4]\*前四天病例 + w[5]\*前五天病例 + bias)

**Advance Variables:**

在advance part，我多加入了一個feature即precipitation降雨量，將model的equation修改成（預測y = w[0]\*溫度 + w[1]\*前一天病例 + w[2]前兩天病例 + w[3]前三天病例 + w[4]\*前四天病例 + w[5]\*前五天病例 +w[6]\*降雨量 + b），隨後training model使它可以學到降雨量與病例之間的關係，從而降低預測的失誤率。

**Difficulty：**

* Python語言及numpy函式庫的不熟悉：
* 對Machine Learning的觀念不清晰
* Hyperparameters的選擇及取值
* MAPE率難以壓下

**Summarize：**

一開始在實作的時候，對眾多事物的不熟悉讓整個實作過程進行的非常緩慢。在不斷上網查閱資料及在codelab上做實驗反復來回中，一點一滴的增進自己對python及numpy的熟悉度，並且也補足了很多linear regression的知識，而後初步的model training也就這麼笨拙的完成了，相當然而，training出來的結果MAPE很高。

隨後我便對一些hyperparameters的初始值去做一些改變，例如learning rate、iteration的次數等等，因為是初次接觸machine learning，這些hyperparameters的值不太知道該如何變化會比較符合model training，導致在很多次嘗試裡面，gradient反而往上跑導致loss的值直接溢位，然而在一次又一次的嘗試以及配合網路資料的建議及經驗下，model training出來的結果是有使MAPE下降到40~50%，但這仍不能滿足作業的條件。

於是我對data做了一些處理，因為僅靠溫度去預測登革熱病例，這兩者的關聯性並不大，所以我加入了前五天的病例當成input feature給model去學習兩者的關係。這一改變確實使MAPE率下降不少，使得train data的bias有降至15~25%，但valid data的variance卻仍居高不下。

最後我加入了Regularization，使得variance可以下降至bias的基準，甚至表現更佳，而後的training過程也是一直在調整hyperparameters的值使model可以更加精準的預測，但這一過程確實也是最花費時間的。

經由此次作業，不僅讓我更加熟悉python和numpy，更讓我正式踏入了AI時代，期待以後的作業可以學到更多machine learning的知識