機器學習 Final

- Team name, members and your work division

隊伍名稱：NTU_r04522631_WannaCry

隊伍成員：機械碩二盧玄真(隊長)、電子碩一王廷暉以及應物所碩三王力弘。

隊伍分工：

分工主要依據 final 時程如下表 1 分成三個階段。

表 1 時程分工表

分工時程	分工內容
Stage 1 Final Start ~ Beyond simple baseline	各自分配一個題目，搶拼 simple baseline。 DengAI → 盧玄真 PumpItUp → 王廷暉 Sberbank → 王力弘
Stage 2 Beyond simple baseline ~ Beyond strong baseline	根據上一階段過關的題目 DengAI 提出可能的 model 並分工執行。 cnn、linear regression → 盧玄真、王廷暉 rnn → 王力弘
Stage 3 Beyond strong baseline ~ Final End	撰寫報告以及整理程式碼 → 盧玄真、王廷暉。

- Preprocessing/Feature Engineering

本次 Final 我們主要使用了兩種 model，分別是 linear regression 以及 rnn，Preprocessing/Feature Engineering 的方法如下：

1. linear regression：

   (1) 遺失資料處理

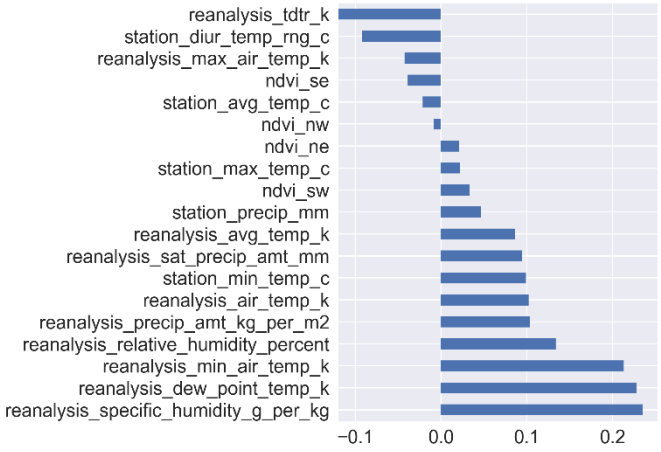
   將所有遺失資料利用 fillna(method =’ffill’)填滿。

   (2) 城市資料分離

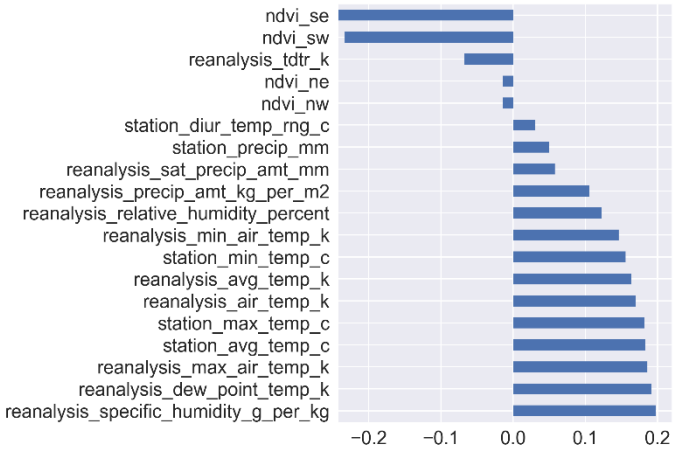
   分別看 sj 以及 iq 城市的 label 會發現兩者的登革熱發生 case 數明顯有落差，因此必須相兩個城市的 model 分開 train。

   (3) 相關度分析

   如下圖為我們對兩個城市當週 feature 與當週 label 分別做相關度的分析。



(a)



(b)

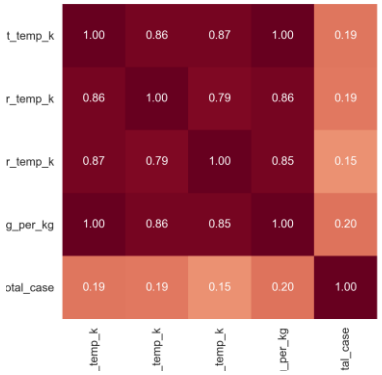
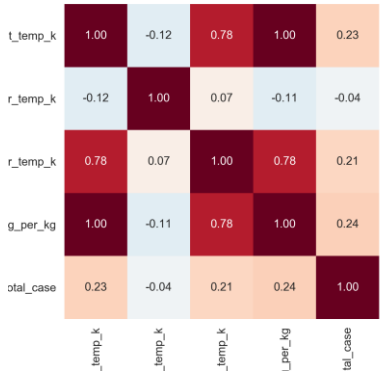
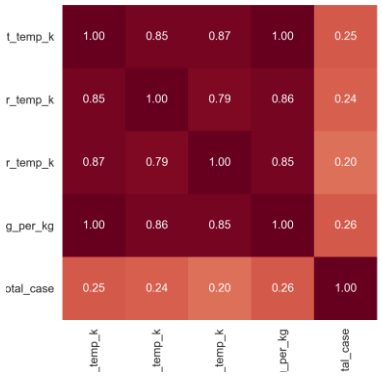
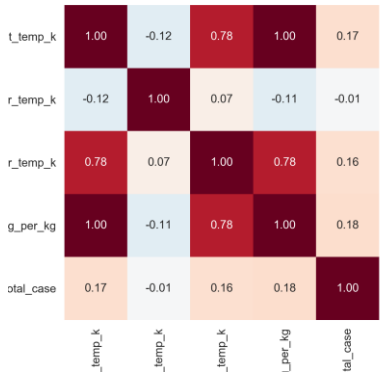
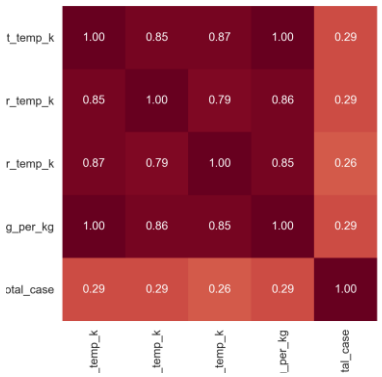
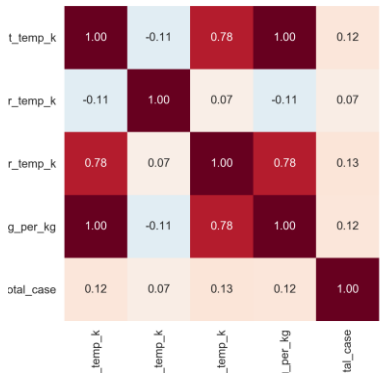
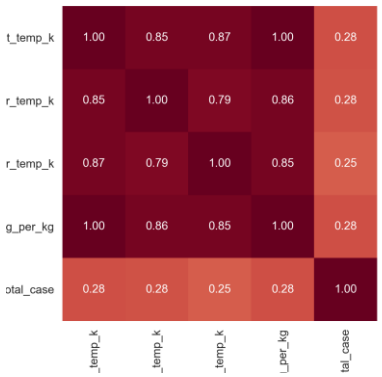
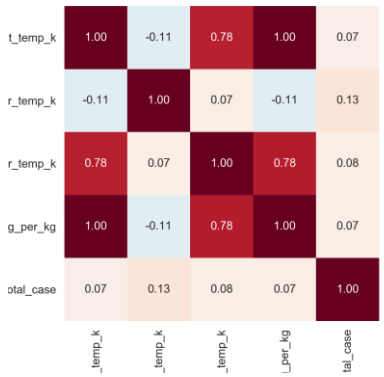
圖 1 兩城市相關度長條圖 (a) iq city (b) sj city

從圖中可以看出登革熱的發生次數明顯與相對濕度(reanalysis_specific_humidity_g_per_kg)、平均露點溫度(reanalysis_dew_point_temp_k)、最高溫(reanalysis_max_air_temp_k)以及最低溫(reanalysis_min_air_temp_k)相關度最大，因此本次 final project 選擇這四個數據做為 feature。

(4) 時間軸平移

接著將 case 數與提前幾週的 feature 做相關度分析，並且將其用 heatmap 畫出整理成表格如下：

表 2 登革熱 case 數量與 feature 相關度隨平移週數關係表

提前週數	sj city	iq city
0 week		
3 week		
7 week		
10 week		

從上表中可以看出兩個城市發生登革熱的 case 數量對提前幾周的相關度變化，其中 sj city 發生登革熱的 case 與 feature 的相關度會隨著提前週數上升直到提前七週相關度達到最高，因此在 Linear regression 的 model 最後選擇此週做為訓練 feature。再看到 iq city，提前幾週 feature 跟 case 數的相關度都是降低的，因此選擇不平移時間。

(5) 特徵標準化

為了讓訓練快速收斂在這裡將特徵做標準化，標準化方法如下：

$$X_{nor} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}$$

2. rnn：

(1) 遺失資料處理

同 linear regression

(2) 城市資料分離

同 linear regression

(3) 相關度分析

同 linear regression

(4) 時間軸平移

由於 rnn 與時間序列極度相關，因此用前 7 週的 feature 來 train，因此除了前七週的以上所選的四個數據以外還會加上前七週的 case 數來做預測。

(6) 特徵標準化

同 linear regression

- Model Description

1. linear regression：

根據我們對資料的觀察，我們發現 label，也就是 city 發生登革熱的數量是沒有固定上限的，而且它的值可以很大，也可以很小，沒有固定範圍，完全取決於 feature 資料的不同而定。因此我們選擇了 linear regression 來做為我們的 model，如下表所示：

```
1  from sklearn import linear_model
2  sjlr = linear_model.LinearRegression()
3  sjlr.fit(sjX_train,sjY_train)
```

linear regression 指的是把我們選取的 feature，分別乘上一個 weight，再把這些值加總起來，再和 label 做 mean_absolute_error，最後使用

gradient descent 之類的方法來求出有最小 mean_absolute_error 時的 weight。

而由上一頁的圖我們可以發現，兩個城市之間的 data 之間沒什麼相關性，因此在這次的實驗中，我們選擇對兩個城市的 data 分別做出各自的 linear regression，並且在 predict 階段，依照 test data 在哪個城市，分別做出 predict 的動作。

在這個 model 裡面，我們可以改變的參數有 alpha 的大小，以及 training 的 feature，在初次我們選擇了三個 feature，相對濕度(reanalysis_specific_humidity_g_per_kg)、平均露點溫度(reanalysis_dew_point_temp_k)、最高溫(reanalysis_max_air_temp_k)以及最低溫(reanalysis_min_air_temp_k)，最後得出的結果在 drivendata 上面為 24.8990，輕鬆通過 simplebaseline，由此可見，這個 model 適合這個題目。

2. rnn：

在機器學習的方法裡，神經網路(neural network)是非常普遍且輕易上手，又可以獲得不錯效果的選擇。在這個題目中,我們選用 keras 套件做為神經網路實作的工具,以下為基本架構,激勵函數(activity rule)的部分選用 relu,學習規則(learning rule)則是選用 adam，loss 設定為題目規定的 mean_absolute_error。

```
1  from keras.models import Sequential
2  from keras.layers import Dense, LSTM
3  model = Sequential()
4  model.add(LSTM(256,activation='tanh',dropout=0.1,input_shape=(sj_trfe
   at_shape[1], sj_trfeat_shape[2])))
5  model.add(Dense(256,activation='relu'))
6  model.add(Dropout(0.1))
7  model.add(Dense(256,activation='relu'))
8  model.add(Dropout(0.1))
9  model.add(Dense(256,activation='relu'))
10 model.add(Dropout(0.1))
11 model.add(Dense(1,activation='relu'))
12 model.summary()
13 adam = Adam(lr=0.001,decay=1e-6,clipvalue=0.5)
14 model.compile(loss='mean_absolute_error', optimizer=adam,)
```

利用簡單的神經網路來預測 testing data 之後，我們獲得 33.26 的成績，離我們的目標 25 還有一段很長的距離。

而下圖是我們用來做 EarlyStopping 的部分，避免 training 到最後都在做無用的步驟，可以用來節省 epoch 的數量，並且記錄最好的 model。

```
1  earlystopping = EarlyStopping(patience = 1000, verbose=1, mode='max')
2  checkpoint = ModelCheckpoint(filepath='best.hdf5',
                                verbose=1,
                                save_best_only=True,
                                save_weights_only=True,
                                mode='max')
3  hist = model.fit(sj_trfeat, sj_trlab,
                    validation_split = 0.1,
                    epochs=nb_epoch,
                    batch_size=batch_size,
                    callbacks=[earlystopping,checkpoint])
```

此外，在選取 feature 的時候我們發現，training data 和 testing data 的時間是連續的，因此我們認為在 training 時候我們可以每次選取該 label 的前七周的 data 來 train 並做 RNN。而在 predict 階段的前幾筆 testing data 雖然沒有前七周的 data 可以用來 predict，但是我們可以取用 training data 的最後幾周資料來當作 testing data，因為他們時間是連續的，所以其實 testing data 的前幾周就是 training data 的最後幾周。

- Experiments and Discussion

linear regression：

(1) 兩城市是否分開

首先我們將其分開 train，再用 validation data 分別測試兩個城市發現，兩個城市的 case 數 MAE 相差甚遠，結果如下

表 3 兩城市個別結果

城市	Validation MAE
sj	27.22
iq	5.26

表 4 兩城市是否分離結果

方法	Validation MAE	MAE
無分開	20.96	26.87
分開 train	20.52	25.71

從上面實驗可以看出兩個城市因為地理環境的關係，參數會有所不同，因此 model 應該要分開 train 才合理。

(2) 時間軸是否平移

表 5 時間軸是否平移結果

方法	Validation MAE	MAE
無平移	20.52	25.71
平移 7 週	19.39	24.89

從上表來看，時間軸平移確實可以讓結果更好，可能的原因也許跟蚊子的生命週期以及疾病擴散時間有關係。一般真正有傳染能力的母蚊生命週期為 4 週，因此當氣候適合蚊子產卵後還要延後幾週才會爆發極大的登革熱感染數。