# HW1 Report

**106062322 江岷錡**

由於本次關注於 Regression 的 prediction，鑑於原始 Dataset 僅有 `cases` 具有較高識別性，我使用 AutoRegression 作為本次作業主要採取 Model。

$$X_t = c + \sum_{i=1}^{p} \varphi_i X_{t-i} + \varepsilon_t$$

Autoregression 為動態處理時間序列的方法，可用 X(1) → X(t-1) 的實際數值，預期 X(t) 的表現，讓整體表現為線性關係。對比上傳統的 Linear Regression 或 Polynominal Regression 而言，不需要額外的 feature （Y 軸、Z軸 ...）進行預測，用自身的X軸便可進行未來預測，符合 Time-Series Data 類型的 Model 型態。本次實驗適用 `statsmodels` 的 `AutoReg` 進行操作。

然則，如何確定 t 的數值（往前看 t 項），則是另一個實作的目標。在我的 Model 中，我先切割原始的 Dataset 為 Training Data 與 Testing Data 。接著，執行一個 for-loop ，套入不同的 t 值 (lags) 進行 autoregression 計算，並用 Testing Data 確認 performance，確認的 metric 以 *MAPE (Mean Absolute Percentage Error)* 為主。也因此，不同的國家，會有不同的 Autoregression Model。

本次實驗的參數設定為：

**t** (lags): 1 → 20

**Training Dates**： 從倒數 57 天 → 倒數第 8 天 (共 50 天)

**Testing Dates**： 從倒數 7 天 → 倒數第 1 天 (共 7 天)

設定 t(lags) 最高為 20 的原因，是為了避免 Overfitting ，讓 Prediction 結果太過 fit training Data 。從實驗數據觀察，基本上並沒有最適合每個國家的 t(lags) 值，也證實本次 Model 的設計方向。

而當我們對每個國家，都找出最佳 **t (lags)** 參數後，便重新以 最近 50 天的 `cases` ，重新 predict 未來 7 天的值。

### Data Pre-Process

由於資料本身讀入時，時序剛好相反，且大部分資料仍待整理，故進行以下步驟：

1. `pandas` 進行資料 `dataframe` 化
2. 翻轉資料
3. 將低於零的值歸零
4. 分割原始資料為 Training Data 與 Testing Data
5. 冠上 `index`

## How to the use model file

### 前言

為了實踐 AutoRegression 的功能，我除了使用 `pandas` / `numpy` 協助整理資料，最後使用 `statsmodels` 所提供的 `AutoReg` 相關 APIs ，進行 `AutoRegression` 的相關設計。

使用我的檔案時，需要安裝 `statsmodels` 。(必須指名 upgrade ，才能升級到包含 `Autoreg` 的版本)

```
pip install statsmodels --upgrade
```

其餘 packages 則為 `pandas` / `numpy` / `matplotlib` 等常見的工具。

### Model

解壓縮後，確保 `106062322_country.ipynb` 檔案與 `download.csv` 檔案在同一層資料夾，打開 `106062322_country.ipynb` ，直接全部執行，即可產生最終 Output File。