**HW3-1 嚴立言**

**目標：**

希望能以當日之前的資料預測出一周後(五個交易日)的趨勢(漲或跌)

**資料視覺化：**

嘗試以交易量(Volume)、震幅(high price – low price)、收盤價變化(close price)繪製散佈圖(scatter plot)，並依照趨勢分不同顏色繪製。

一張含有 地圖 的圖片

自動產生的描述

從圖中可看出這兩群幾乎無法分開，因此推測 logistic regression 得步道好結果。

**資料前處理：**

訓練資料中沒有缺失值，離群值可能反映重大市場變化因此不處理，所以只針對資料 normalize。

**Logistic regression：**

結果：

training set

average train accuracy: 0.5983172981165285

min train accuracy: 0.5899280575539568

max train accuracy: 0.6048699501936912

average valid accuracy: 0.5983203500578851

min valid accuracy: 0.5720620842572062

max valid accuracy: 0.6319290465631929

test set

0.532520325203252

和一開始猜的一樣，效果並不怎麼好

**Neural network：**

結果：

training set accuracy

0.5983170947741364

test set accuracy

0.540650406504065

使用 skilearn 的 MLPClassifier

得出來的結果和 logistic regression 差不多

**LSTM (Long Short-Term Memory)：**

考慮到股票趨勢和時間的相關性，因此選擇用 lstm。以五天為一個區間預測下周的走勢，先用 lstm 算出一條新的價格曲線，再將新曲線的漲跌趨勢和真正的趨勢做比較。

參考資料：<https://ithelp.ithome.com.tw/articles/10195400>

一張含有 地圖, 文字 的圖片

自動產生的描述

結果：

train

0.7719220549158547

test

0.8130081300813008

上圖為 lstm 算出的新價格曲線和真實曲線的對比，以及預測出的趨勢準確度，可以看的出來比起前兩個方法要好很多。

## 比較：

* logistic regression
  + train :0.5983172981165285
  + test :0.532520325203252
* neural network
  + train :0.5983170947741364
  + test :0.540650406504065
* lstm
  + train :0.7719220549158547
  + test :0.8130081300813008

由於 lstm 考慮了時間因素，因此準確率高出其他兩者很多

，對於這種和時間相關的資料，用 lstm 應該都能獲得不錯的結果

**Improvement：**

前兩個由於一開始就認為不管用，因此沒特別做 improvement，而 lstm 一開始就獲得了不錯的結果，因此也沒有做其他的修正

**實用性：**

雖然 lstm 的結果精確度不低，但那只能預測到下周的結果，而且還有很多現實隨機因素沒有考慮，因此仍然不適合用來作為投資參考。