Multivariate Time Series Prediction for Stock Market Data

資科一	資科一	資科二
曹昱維	鄭詠儒	謝政彥
110753201	110753126	109753207

1 Introduction

時間序列資料是由按照時間發生先後順序進行排列的數據點序列，以股票市場資料為例，從 2000 年至 2020 年為止，股票 A 的每日收盤股價就是一種時間序列資料。我們希望透過單一變數的時間序列資料(如股票 A 的每日收盤價)以及多變數的時間序列資料(如所有能源類股的每日收盤價)來訓練各種模型，來比較單變數與多變數對於預測效果的影響，也同時比較相同類型的變數下各種模型的預測效果。

股票市場以波動性、動態性和非線性著稱。由於政治、全球經濟狀況、突發事件、公司財務業績等多重（宏觀和微觀）因素，準確預測股價極具挑戰性。但是，所有這一切也意味著有大量數據可供尋找模式。因此，金融分析師、研究人員和數據科學家不斷探索分析技術來檢測股市趨勢。股票分析基本上可利用基本面分析與技術分析。

ARIMA (p, d, q)模型被廣泛使用在時間序列分析，在了解 ARIMA 模型之前可以了解相關的模型，(1)自我迴歸模型(AR)，它用前期的資料來預測本期的資料，越接近本期的資料，對預測結果的影響力就越大，設定一筆資料會與他過去 p 期的資料相關，(2)移動平均模型(MA)，方法在於本期的隨機誤差會與過去產生的隨機誤差有關，接著設定要計算 q 期移動平均，這些模型的缺點是

只能處理穩定的資料，時間序列資料是否為定態資料，會影響後續預測的結果，因此，ARIMA 模型改進了以上缺點，使用 ADF-test 計算差分次數 d 將時間序列資料處理成定態資料。

LSTM 模型是近期時間序列資料常見的深度學習模型，屬於 RNN (Recurrent Neural Network) 的一種，適合在輸入特徵空間中提取模式，其中輸入數據跨越長序列，可以從多個輸入變量的問題進行建立 many to many 或 many to one 模型，在建模問題方面提供了很大的靈活性，包括可以很好地控制時間序列的幾個參數。

本專案嘗試針對 S&P 500 中的能源類股 APA Corporation 股價，利用 python 相關套件(arima, keras, sklearn,etc.)實作深度學習 RNN 模型 LSTM 以及傳統的時間序列模型 ARIMA 進行預測與比較。

2 Related Work

在早期有發展預測模型用來預測與分析股價，像是 ANN 模型[1]，近年來更是有相對穩定的 ARIMA 模型[2, 3]用來預測金融相關序列資料，而深度學習模型更是盛行，深度學習模型的效能仰賴參數的設定，以 LSTM 模型為例重要的參數有 activation function (sigmoid, tanh, softmax 等等)、optimizer (Adam, Adadelta, RMSprop 等等)、batch size、epoch 數量以及 hidden layers 數量等等[4]。

ARIMA 模型處理數據的非平穩性收集和建模，以類似的方式，作為基於深度學習的算法的代表 – LSTM 模型是儲存和訓練於較長時間內給定的數據特徵，因此[5]對 ARIMA 和 LSTM 做了比較，提供了我們比較依據與方法。

3 Method

3.1 Dataset

數據集為 S&P 500 其中的 APA 股價數據，為一檔美國能源類股，APA Corporation 是一家是從事油氣勘探的美國公司，我們從 Kaggle[6]上抓取 S&P 500 股價，並將其 APA 的股價抓出來，時間序列從 2010 年一月至 2021 年十一月，原始資料集如圖一。

	Ticker	Date	High	Low	Open	Close	Volume	Adj Close
0	A	2010-01-04	22.625179290771484	22.26752471923828	22.45350456237793	22.389127731323242	3815561	20.461841583251953
1	A	2010-01-05	22.33190345764160	22.00286102294922	22.324750900268555	22.145923614501953	4186031	20.239572525024414
2	A	2010-01-06	22.174535751342773	22.00286102294922	22.06723976135254	22.06723976135254	3243779	20.167661666870117
3	A	2010-01-07	22.045780181884766	21.81688117980957	22.017166137695312	22.03862762451172	3095172	20.14151382446289
4	A	2010-01-08	22.06723976135254	21.745351791381836	21.917024612426758	22.03147315979004	3733918	20.134977340698242
5	A	2010-01-11	22.21030044555664	21.93848419189453	22.08869743347168	22.045780181884766	4781579	20.148056030273438
6	A	2010-01-12	21.924177169799805	21.616594314575195	21.859800338745117	21.781116485595703	2871073	19.906173706054688
7	A	2010-01-13	22.017166137695312	21.494993209838867	21.795421600341797	21.952789306640625	3418949	20.063068389892578
8	A	2010-01-14	22.346208572387695	21.81688117980957	21.88125991821289	22.281831741333008	6163782	20.363784790039062
9	A	2010-01-15	22.43204689025879	21.69527816772461	22.33190345764160	21.766809463500977	4626681	19.893098831176758
10	A	2010-01-19	22.052932739257812	21.709585189819336	21.716737747192383	22.03147315979004	3563642	20.134977340698242
11	A	2010-01-20	21.93848419189453	21.595136642456055	21.838340759277344	21.909870147705078	4589075	20.02383804321289
12	A	2010-01-21	22.253219604492188	21.587982177734375	22.174535751342773	21.831186294555664	6081440	19.951932907104492
13	A	2010-01-22	21.709585189819336	20.808298110961914	21.709585189819336	20.865522384643555	4263061	19.06938934326172
14	A	2010-01-25	21.20886993408203	20.9084415435791	21.044349670410156	21.065807342529297	3608518	19.252443313598633

圖一、S&P 500 股價資料集

3.2 ARIMA (Autoregressive Integrated Moving Average)

3.2.1 ARIMA（p，d，q）

ARIMA 模型是 ARMA（p，q）模型的擴展。ARIMA（p，d，q）模型可以表示為：

- p 是 AR(Autoregressive)的落後期數，為過去資料的加權平均，今天的股價會是過去股價的加權平均值
- d 是非季節性差異的數量，使其成為平穩序列所做的差分次數，公式如下：

$$\left(1-\sum_{i=1}^{p}\phi_i L^i\right)(1-L)^d X_t = \left(1+\sum_{i=1}^{q}\theta_i L^i\right)\varepsilon_t$$

其中 L 為 Lag operator，d ∈ ℤ，$d > 0$

資料來源：wikipedia

- q 是 MA(Autoregressive)的落後期數，隨機誤差的加權平均，今天的股價的隨機誤差會與過去產生的隨機誤差有關

3.2.2 原始資料分析

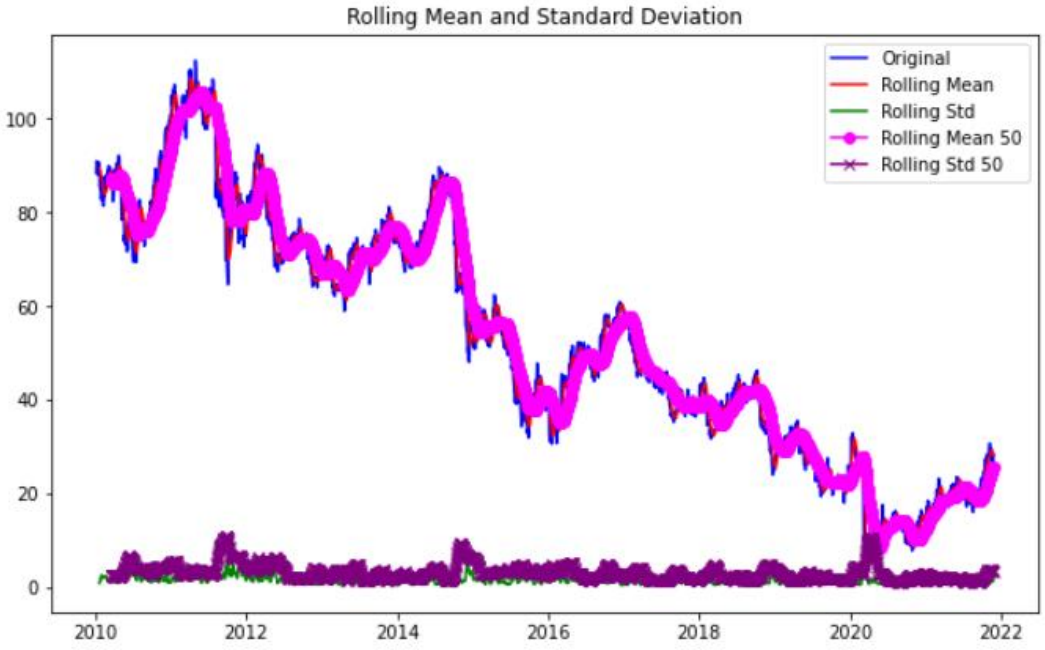
我們使用使用 EDA 對資料作分析

- 將原始資料轉為 Log-rolling 格式，如圖二。

圖二、原始資料與平滑後資料視覺化呈現

- 將原始資料收盤價轉為 rolling 格式，如圖三、圖四。

圖三、原始資料收盤價趨勢圖



圖四、原始資料收盤價平滑平均與標準化視覺化呈現

3.2.3 ARIMA 參數設定：

- 模型參數：

```
model_autoARIMA = auto_arima(train_data, start_p=0, start_q=0,
                      test='adf',    # use adftest to find optimal 'd'
                      max_p=10, max_q=10, # maximum p and q
                      m=1,                 # frequency of series
                      d=None,             # let model determine 'd'
                      seasonal=False,     # No Seasonality
                      start_P=0,
                      D=0,
                      trace=True,
                      error_action='ignore',
                      suppress_warnings=True,
                      stepwise=True)
```

- 建立模型：ARIMA(3,0,2)

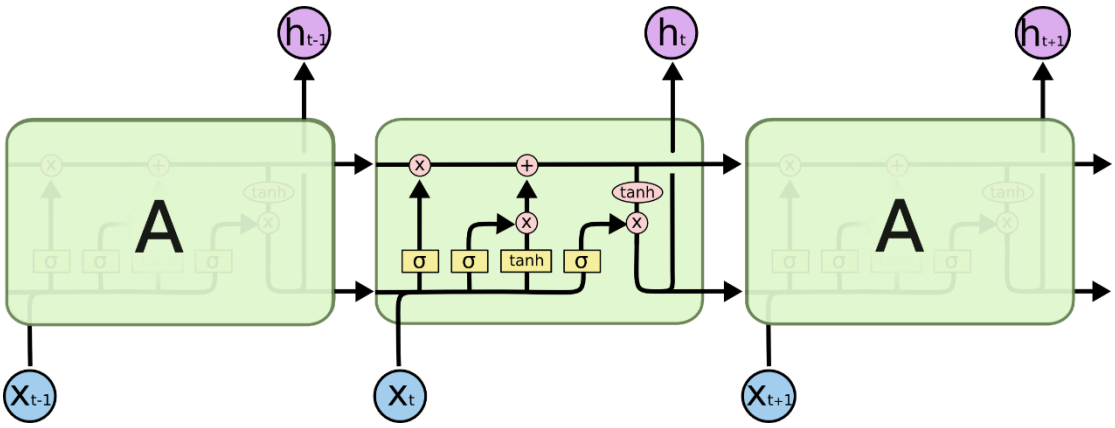
```
model = ARIMA(train_data, order=(3, 0, 2))
```

3.3 LSTM (Long Short-Term Memory)

3.3.1 模型架構

LSTM 是 RNN 架構的其中一種模型，不同於 RNN 的一層架構，主要解決時間序列的問題，是由四種架構構成如圖五：

- Input Gate: feature 輸入時，input gate 會控制是否輸入
- Memory Cell: 將計算出的值儲存，使下個階段能使用
- Output Gate: 控制是否將這次計算出來的值輸出
- Forget Gate: 控制是否將 Memory 清除



圖五、LSTM 模型架構

3.3.2 實驗過程

我們使用 LSTM 中 many-to-one 方式來訓練資料來預測 2021.7.1 ~ 2021.11.26 的收盤股價：

- 關聯係數分析

```
High          0.999711
Low           0.999743
Open          0.999399
Close         1.000000
Volume       -0.479447
Adj Close     0.999291
Name: Close, dtype: float64
```

圖六、Correlation Analysis

- 實驗特徵：Close(收盤價)、High(股票高點)、Volume(成交量)如圖七。

Date	Close	High	Volume
2021-11-19	26.000000	27.010000	12427500
2021-11-22	26.530001	27.360001	11652900
2021-11-23	28.469999	28.500000	10602000
2021-11-24	28.610001	29.209999	8531600
2021-11-26	26.240000	26.549999	8469000

圖七、LSTM 模型取用特徵

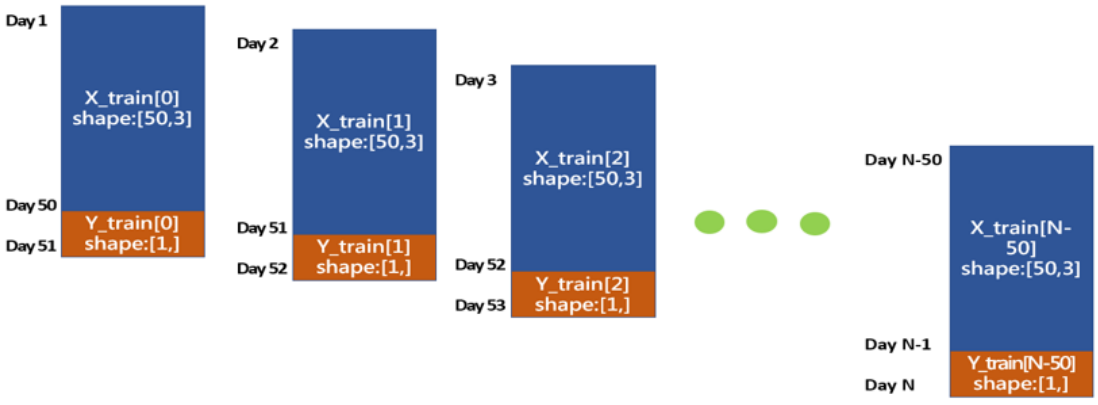
- 訓練資料

我們將資料切分成 train、validation 以及 test 資料做訓練，資料日期：2010.1.4 ~ 2021.11.26 總共 2997 筆，訓練資料切分方式如表一。

資料集	時間	Shape
Training Data	2010.1.4 ~ 2019.12.31	(2516, 3)
Validation Data	2020.1.1 ~ 2020.12.31	(377, 3)
Testing Data	2021.7.1 ~ 2021.11.26	(104, 3)

表一、訓練資料切分方式

Timestep 設定：輸入資料的切分方式為 104 筆資料(Testing Data)中假設每 50 天為一筆輸入資料來預測下一天資料，例如第一天至第五十天預測第五十一天，方式如圖八。



圖八、訓練資料的輸入資料切分

- 交叉驗證

使用時間序列的 Cross Validation 來進行驗證。

- LSTM 模型參數

資料前處理使用了 MinMaxScaler：

```
from sklearn.preprocessing import MinMaxScaler

sc = MinMaxScaler(feature_range = (0, 1))



train = sc.fit_transform(train_data)

val = sc.fit_transform(val_data)

test = sc.transform(test_data)

print(train.shape,val.shape,test.shape)
```

而實驗過稱中透過參數的調整來找出最佳結果，參數有 Activation function: selu、relu、sigmoid，Optimizer: Adam、RMSprop，Batch size、Epoch、Hidden layers 以及 Loss 等等如圖九。

Timestep	Hidden_Layers	Learning_Rate	Batch_Size	epoch	Loss	Activation	Optimizer	Train_Loss	Val_Loss
50	**60,55**	**0.001**	**64**	**100**	**mean_squared_error**	**selu**	**Adam**	**0.001316**	**0.000163**
50	60,55	0.001	64	100	mean_squared_error	selu	RMSprop	0.001900	0.000200
50	70,65	0.001	64	100	mean_squared_error	selu	RMSprop	0.001400	0.000321
50	30, 25	0.001	64	50	mean_squared_error	selu	Adam	0.003080	0.002134
40	30, 25	0.001	64	50	mean_squared_error	selu	Adam	0.002859	0.004156
50	40,35	0.001	64	50	mean_squared_error	selu	RMSprop	0.002792	0.004604
50	40,35	0.001	64	50	mean_squared_error	selu	Adam	0.002809	0.005299
40	30, 25	0.001	64	50	mean_squared_error	selu	Adam	0.003089	0.006579
30	30, 25	0.001	32	50	mean_squared_error	selu	Adam	0.003126	0.007293
30	30, 25	0.001	32	50	mean_squared_error	sigmoid	Adam	0.004847	0.011306
40	30, 25	0.001	32	50	mean_squared_error	selu	Adam	0.003156	0.014207
30	30, 25	0.001	32	50	mean_squared_error	relu	Adam	0.004418	0.015898
30	30, 25	0.001	32	50	mae	selu	Adam	0.038945	0.076571
30	30, 25	0.001	32	50	mae	relu	Adam	0.048553	0.094576
30	30, 25	0.001	32	50	mae	sigmoid	Adam	0.082377	0.156985

圖九、LSTM 實驗參數

4 Results

我們對 APA Corp. 50 天的股票價格進行預測，LSTM 模型的最佳參數設定如圖十。

Timestep	Hidden_Layers	Learning_Rate	Batch_Size	epoch	Loss	Activation	Optimizer
50	60,55	0.001	64	100	mean_squared_error	selu	Adam

圖十、LSTM 最佳參數

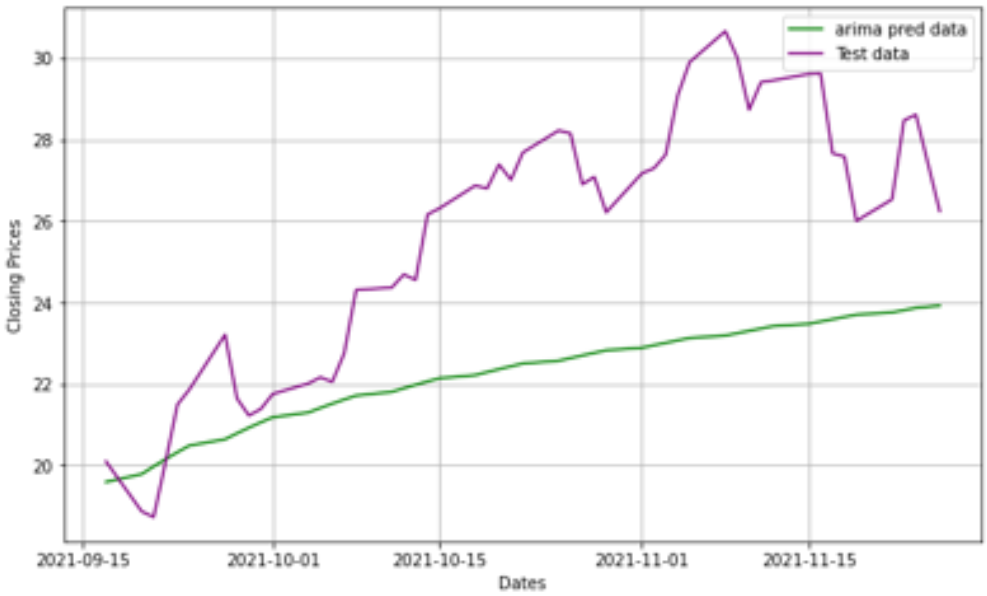
而評估指標計算使用了 RSME、MSE 以及 R-Square Score，結果如表二。

- RMSE(Root Mean Square Error)：均方根誤差
- MSE(Mean Square Error)：均方誤差
- R-Squared：回歸模式之變異值與所有y_i變異量之比例

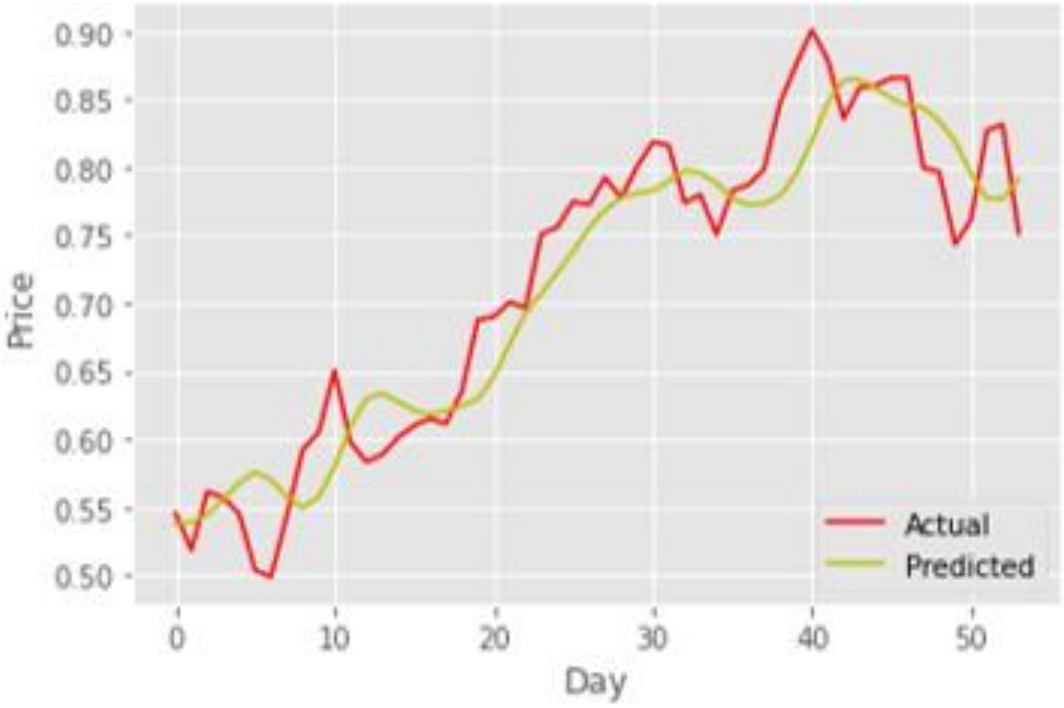
評估指標	ARIMA(3,0,2)	LSTM
RMSE		0.0014981071162395126
MSE	16.362673218826885	0.03870538872352935
R-Squared	-0.5659850824780199	0.8876417258075385

表二、ARIMA 和 LSTM 評估指標結果

預測的結果趨勢圖如圖十一、圖十一二



圖十一、RIMA 預測結果



圖十二、LSTM 預測結果

5 Discussion

從實驗結果可以發現，比起傳統的 ARIMA 模型預測，因 LSTM 的參數調整彈性以及多樣性，預測出來的結果明顯筆 ARIMA 好很多，我們在實驗過程也有遇到幾項問題，如下：

5.1. 時間序列資料的 validation set 設置

- 實驗需要設置 sliding window
- 日期很重要，我們希望拆分數據每個 window 都包含 X 天
- Test set 必須在 Train data 之後

圖十三、Time-based cross-validation

5.2. 參數設定問題

一般 LSTM 多對一(many to one)模型，其參數 return_sequences 設定為 False ，且不可使用 TimeDistribution。然而本專案，雖未使用 TimeDistribution，但 return_sequences 卻設為 True，如果改成 False，反而出現錯誤訊息，值得再深入研究。

6 Future Work

從實驗的結果發現因少量資料丟進模型去跑，y'確實只有一個數字，或許 many to many 可當作本專案後續可以繼續精進的議題。

7 Reference

[1] G.S. Atsalakis and P.V. Kimon, “Forecasting stock market short-term trends using a neuro-fuzzy methodology”, Expert Systems with Applications, vol. 36, no. 7, pp.10696–10707, 2009.

[2] Debadrita Banerjee, “Forecasting of Indian Stock Market using Time-series ARIMA Model”, ICBIM 2014

[3] Ayodele A. Adebiyi., Aderemi O. Adewumi and Charles K. Ayo, “Stock Price Prediction Using the ARIMA Model”, UKSim-AMSS 2014

[4] Anita Yadava, C K Jhaa and Aditi Sharanb, “Optimizing LSTM for time series prediction in Indian stock market”, ICCIDS 2019

[5] Sima Siami-Namini, Neda Tavakoli and Akbar Siami Namin, “A Comparison of ARIMA and LSTM in Forecasting Time Series”, IEEE 2018

[6] S&P 500 stocks price with financial statement https://www.kaggle.com/hanseopark/sp-500-stocks-value-with-financial-statement

[7] Predicting Stock Prices Using Machine Learning https://neptune.ai/blog/predicting-stock-prices-using-machine-learning

[8] Prediction of price for ML with finance stats https://www.kaggle.com/hanseopark/prediction-of-price-for-ml-with-finance-stats/data

[9] Time-Series Forecasting: Predicting Stock Prices Using An LSTM Model https://towardsdatascience.com/lstm-time-series-forecasting-predicting-stock-prices-using-an-lstm-model-6223e9644a2f

[10] Berkshire Hathaway - Stock Time Series Analysis https://www.kaggle.com/kalilurrahman/berkshire-hathaway-stock-time-series-analysis

[11] A Multivariate Time Series Modeling and Forecasting Guide with Python Machine Learning Client for SAP HANA https://blogs.sap.com/2021/05/06/a-multivariate-time-series-modeling-and-forecasting-guide-with-python-machine-learning-client-for-sap-hana/

[12] Understanding LSTM Networks

https://colah.github.io/posts/2015-08-Understanding-LSTMs/

[13] LSTM 深度學習股價預測

https://medium.com/data-scientists-playground/lstm-%E6%B7%B1%E5%BA%A6%E5%AD%B8%E7%BF%92-%E8%82%A1%E5%83%B9%E9%A0%90%E6%B8%AC-cd72af64413a

8 Team Work

(1) 110753201 曹昱維: Slides、Data Preprocessing、ARIMA Model

(2) 109753207 謝政彥: Slides、LSTM Model

(3) 110753126 鄭詠儒: Slides、Presentation、Report、Self-Attention Model(未完成)