

以 LSTM-RNN 預測 ETF 50 股價趨勢 並結合交易策略以獲取最大獲利率

沈沛瑄

國立台灣大學工程科學系
r07525056@ntu.edu.tw

魏廉臻

國立台灣大學工程科學系
d01525010@ntu.edu.tw

張瑞益

國立台灣大學工程科學系
rayichang@ntu.edu.tw

摘要

本研究以台灣50 (ETF50) 「指數股票型證券投資信託基金」的股票指數作為預測目標。運用深度學習中的長短期記憶模型進行研究，將台灣50指數和其成分股占比最大股票之歷史資料及技術指標資料做為輸入模型之變數資料。本研究將收集到的樣本切割成兩部份，3486筆日資料為訓練資料；387筆日資料為測試期資料。除了預測收盤價趨勢以外，同時預測出高低價趨勢通道，並在訓練後進行本研究提出的校正策略，最後結合交易策略使交易能儘量買在最低點並賣在最高點，以獲取最大的獲利率。經實驗過後發現：使用二次交易策略並以60天移動視窗校正後的預測最高最低價進行交易有最良好的獲利率表現，假設以當日實際最高及最低價進行買賣，在理想值為4%獲利率下，我們已經有2.59%的獲利率表現。

關鍵詞：長短期記憶模型、台灣50、校正策略、交易策略。

Abstract

This study uses Taiwan Top50 Exchange Tracker Fund (ETF50) as a forecast target. Use the Long Short-Term Memory (LSTM) model in the Deep Learning for research. We use the historical data and technical indicators of ETF50 and the largest share of ETF50's constituent stocks as input data. Our sample data is mainly separated into two parts, 3486 records of training data and 387 records of testing data. In addition to predicting the closing price trend, we also predict the trend channel formed by the highest and lowest prices. After training, applying our proposed correction strategy to the prediction results. Finally, combined with the trading strategy, the trading action can be bought as low as the possible point and sold as high as the possible point to obtain the maximum profitability. After the experiment, it was found that: Using the secondary trading strategy and trading at the forecasted highest and lowest price corrected by the 60-day mobile window has the best profitability performance. Assuming that using the actual highest and lowest prices the day to trade, we have a profit performance of 2.59% at an ideal value of 4%.

Keywords: LSTM, ETF50, Correction strategy, Trading strategy.

1. 前言

現今在股市裡有三大分析方法，即：技術面、基本面、籌碼面等。各領域都有其自圓其說的選股策略，無法直接評斷哪種方法好或壞，且複雜

程度本就相對不同，各個方法都有其適合的分析週期。在傳統技術分析方面，通常根據證券歷史資料，進行統計分析，然而人為主觀判斷容易影響到分析結果的客觀性。因此本論文希望藉由深度學習演算法讓電腦自主學習到股票歷史價格資料裡的規律，進行股價趨勢的預測，進一步預測出交易訊號，並探討配合何種買賣決策規則，打造較佳股票買賣預測及策略的機制，得到相對較高的獲利率。

台灣50 (ETF50) 為一種「指數股票型證券投資信託基金」，其被動追蹤台灣市值前50大之股價的表現，考慮各股的流通量、流動性及市值等部份，依據各股所占市值比重與各股價指數的乘積加總而成，是一項指數化投資的商品；並且可以像一般股票在集中市場掛牌交易，上下漲跌幅限制為10%，買賣手續費為1.425%，主要差異在於證交稅為1‰，少於一般股票的3‰；採取類似共同基金的模式，由投信公司管理，並發行受益憑證做為資產持有的表徵；其好處在於股價之變化幅度不易受公司企業單獨事件影響產生大幅度的變化、買賣方便交易方式與一般股票相同、分散分險、透明度高、避險工具...等，因此選用ETF50進行實驗。

在投資之前，需先了解適合自己的複雜程度及投資週期，以選擇投資分析的方法。基本面分析，了解一間企業的資產配置狀況、營收或是持有現金部分等，從數據推測企業價值及未來成長，適合中、長期投資人；技術面分析，股票未來走向判讀的參考，基礎建立與歷史數據上，適合短期投資；籌碼面，其實就是紀錄股東權益的變化，大戶的交易狀況，為進出場的指標之一。台股受外資及投信投顧影響甚鉅，因此散戶跟著法人進出場也是常見的策略。買股票除了要買得好以外，賣得快也是很重要的，畢竟投報率會因為時間的流逝而漸漸下降。因為技術指標與時間相關，所以在這裡選擇利用技術分析結合深度學習的長短期記憶模型 (Long Short-Term Memory, LSTM) [1]去做預測。

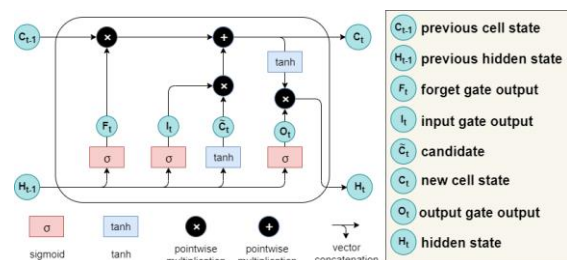


圖 1 LSTM 架構

2. 文獻探討

如圖 1，LSTM 是基於遞迴神經網路（RNN）[2]所改良之架構，RNN 受限於短期記憶的問題，因為他只考慮前面的數據去預測下一個值，當預測到越後面時，越前面的數據對當前數據的影響力逐漸遞減，容易被遺忘，因此，如果嘗試處理一段很長的時間序列，RNN 可能在開始時就會遺漏重要資訊。而 LSTM 其架構具有三個 Gate，用來增加記憶力，避免梯度消失，能解決上述 RNN 短期記憶的問題。

LSTM 之計算步驟分為四個部分[3]：Forget Gate 能決定那些資訊應丟棄或保留，來自當前輸入值 x_t 和先前隱藏狀態 h_{t-1} 同時輸入到 Sigmoid 函式，輸出值 f_t 處於0和1之間，越接近0表示越應該忘記，越接近1表示越應該保留，如式（1）。

$$f_t = \text{sigmoid}(x_t + h_{t-1}) \quad (1)$$

Input Gate 用來更新當前單元狀態 c_t ，將當前輸入值 x_t 和先前隱藏狀態 h_{t-1} 輸入到 Sigmoid 函式，調整輸出值至0跟1之間，得到 i_t ，來決定更新哪些資訊，如式（2）。

$$i_t = \text{sigmoid}(x_t + h_{t-1}) \quad (2)$$

另一方面，當前輸入值 x_t 和先前隱藏狀態 h_{t-1} 傳輸給 tanh 函式，並將數值壓縮在-1和1之間以調節網路，得到 \bar{c}_t ，如式（3）。

$$\bar{c}_t = \tanh(x_t + h_{t-1}) \quad (3)$$

New Cell State 計算新的單元狀態 c_t 。將上述計算所得之 i_t 和 \bar{c}_t 相乘，Sigmoid 輸出值 i_t 將決定在 tanh 輸出值 \bar{c}_t 中哪些資訊是重要的且需要進行保留，並且將 Forget Gate 的輸出值 f_t 和上一個的單元狀態 c_{t-1} 相乘，如果它乘以接近0的值，則表示在新的單元狀態中可能要丟棄這個值，最後相加得到新的單元狀態 c_t ，如式（4）。

$$c_t = i_t \times \bar{c}_t + f_t \times c_{t-1} \quad (4)$$

Output Gate 用來決定下個隱藏狀態的值 h_t ，隱藏狀態中包含了先前輸入的相關資訊。將當前輸入值 x_t 和先前隱藏狀態 h_{t-1} 傳遞給 Sigmoid 函式，得到 Output Gate 輸出值 o_t ，如式（5）。

$$o_t = \text{sigmoid}(x_t + h_{t-1}) \quad (5)$$

接著將計算所得之新的單元狀態 c_t 傳遞給 tanh 函式；最後將 Output Gate 輸出值 o_t 和 tanh 輸出值相乘，以決定新的隱藏狀態 h_t 應攜帶的資訊，如式（6）。

$$h_t = o_t \times \tanh(c_t) \quad (6)$$

最後把新的單元狀態 c_t 和新的隱藏狀態 h_t 傳輸給下個時間序列的單元，新的隱藏狀態 h_t 則作為當

前單元輸出。

在輸入特徵方面，除了輸入基本歷史數據外，加入技術指標有效推測未來價格的變動趨勢，此外，由[11][12]所提出的論文中皆將相關股票數據一併作為特徵輸入，其實驗證實相關股票之趨勢也可能是影響預測目標的因素之一。[8]其實驗結果中則證實特徵越多不一定提升準確率，還是需要足夠的數據量。

在預測模型比較方面，包含傳統金融模型（ARIMA、GARCH）[4] [5]、早期的倒傳遞類神經網路（BPNN）[6]、卷積神經網路（CNN）[7]、RNN 及 LSTM，在上述研究中，[9]的實驗結果顯示，LSTM 確實比傳統金融模型有較好的準確度且優於 RNN，並且再加上 Dropout 確實能有效防止過度擬合的問題。且[10]認為股票價格數據雜訊多、複雜及非線性，容易受到政策、經濟和心理等諸多因素的影響，其實驗針對 LSTM 及 ARIMA，利用 python 中的 ADF 檢測方法進行模型穩定性的檢驗，實驗結果顯示穩定性對預測結果影響不大，但對算法的收斂速度有輕微影響，另一方面以平均絕對誤差（MAE）和均方根誤差（RMSE）來評估模型性能及呈現預測值和實際值之間的誤差，結果表明 LSTM 算法的失誤率比 ARIMA 低66.78%，LSTM 算法在預測中表現更好並且具有更小的誤差。然而，他們也提到 LSTM 算法的缺點是需要花費大量時間來訓練模型並且需要大量數據樣本。

在交易策略方面，除了一般直觀的漲買跌賣策略外，[12]提出了一次及二次交易策略，其方法以買入所花費的手續費和賣出所花費的手續費及證交稅作為買賣點門檻，本研究將採取此策略進行模擬交易。表 1，整理了國內外相關研究之論文，針對輸入模型之特徵、預測模型及交易策略去進行分析比較。

表 1 國內外相關研究比較表

| 文獻 | 輸入特徵 | 預測模型 | 交易策略 |
|------|--|-------------------------------|--------------|
| [8] | 收盤價線圖和5日、10日及20日的移動平均線(MA)之均線圖 | CNN | 漲買跌賣 |
| [9] | 台灣與主要貿易對手通貨兌美元之匯率 | ARIMA GARCH RNN LSTM | 無 |
| [10] | 收盤價 | ARIMA LSTM | 無 |
| [11] | 成交量、最高價、最低價、開盤價、收盤價及相關綜合股票指數之歷史資料 | LSTM | 無 |
| [12] | 台灣50和其排名前20大之成分股的過往5日收盤價、當日成交量、12項技術指標數據 | BPNN | 一次交易 二次交易 |

本研究整合上述研究之結果，使用基本歷史資料及技術指標並加入最相關之成分股作為輸入變數，經由 LSTM 模型進行預測，為了在進一步提升預測準確度，提出了校正策略對預測結果進行校正動作。最後，在交易策略部分參考了二次交易的概念，並結合本研究提出的高低價趨勢通道進行模擬買賣動作，使買點能買在最低，賣點能

賣在最高，以達到提升獲利率之目的。

3. 研究方法

圖 2 為本研究之研究流程架構圖。首先，先收集相關歷史資料，由於股票市場中，該公司當天可能會有重大事件宣布而停止交易，因此對其交易日採取前一天之值進行補值的動作。接下來將資料預處理後的變數輸入進 LSTM 模型進行預測，分別預測出最高、最低及收盤價趨勢，再對其預測結果進行本研究所提出之校正策略。透過預測結果進行買賣點判斷，將買點及賣點應用於高低價趨勢通道，使其買在可能的最低點，賣在可能的最高點，最後計算獲利率進行交易策略之評估。

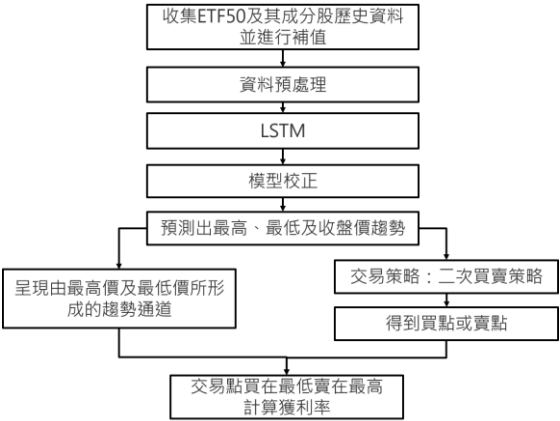


圖 2 研究流程架構圖

研究資料

- 資料來源：台灣證券交易所 (TWSE)[13]
- 股票資料：台灣50、台積電
- 資料區間：2003/06/30 至 2019 /02/18，3873筆日資料
- 歷史數據：每日的開盤價、收盤價、最高價、最低價及成交量
- 技術指標：隨機指標（KD）、移動平均線（MA）、乖離率（Bias）、強弱指標（RSI）、威廉指標（WILLR）、動量指標（MTM）、聚散指標（MACD）、差離值（DIF）
- 訓練資料：總資料的前90%，共3486筆日資料
- 測試資料：總資料的後10%，共387筆日資料

本研究中，輸入模型所使用的變數，如下表 2。當要預測不同趨勢時，如最高價、最低價及收盤價，使用各自的歷史資料。而固定會加入之歷史資料為開盤價以及成交量，並加入[12]所提出之論文中，8種技術指標進行不同時間區間之指標計算，最後得到12個技術指標之變數。因此本研究總共有14個輸入變數。

表 2 輸入變數一覽表

| 變數 | | 說明 |
|-----------|----------------|-------------|
| X_1 | High/low/close | 最高/最低/收盤價 |
| X_2 | Open | 開盤價 |
| X_3 | Deal | 成交量 |
| $X_4、X_5$ | K、D | 9日隨機指標 |
| $X_6、X_7$ | MA6、MA12 | 6日及12日移動平均線 |

| | | |
|-----------------|----------------|------------|
| $X_7、X_8$ | Bias、3-6Bias | 6日及3-6日乖離率 |
| X_9 | RSI6 | 6日強弱指標 |
| X_{10} | WILLR12 | 12日威廉指標 |
| $X_{11}、X_{12}$ | MTM6、MTM(avg6) | 6日動量指標及均線 |
| X_{13} | MACD | 10日聚散指標 |
| X_{14} | DIF | 10日差離值 |

本研究輸入數據分別預測最高價、最低價及收盤價，因此當預測其價格時，加入其歷史資料以外（最高/最低/收盤價）並加入其他歷史資料，如：開盤價、成交量、技術指標以及成分股（台積電）的歷史數據與技術指標，其中加入較具影響力之股票的歷史數據做為特徵值可以進一步提高準確率。

Step-1: 依據所加入的成分股占有比例重新計算權重[12]：

$$W_T = \sum T, W_{i/T} = \frac{W_i}{W_T} \times 100\%$$

W_i ：股票之權重比例

W_T ：總權重

$W_{i/T}$ ：條件百分比

Step-2: 根據歷史數據計算術指標，接著將歷史數據與技術指標進行正規化：

$$m = (x - x_{min}) / (x_{max} - x_{min})$$

m ：正規化後的數值

x ：欲正規化的數值

x_{min} ：該批資料的最小值

x_{Max} ：該批資料的最大值

Step-3: 重新計算各個股票的資料強弱度，將正規化後的數值乘上第一步所計算出來的權重：

$$new_value = m \times W_{i/T}$$

m ：正規化後的數值

$W_{i/T}$ ：股票之權重比例

Step-4: 刪除前幾筆用於計算技術指標時所造成的空值資料，並依據 time_frame 切割相對日資料長度多加一天做為驗證答案，逐筆取出 time_frame+1 個數值做為一筆時間序列，這裡 time_frame 根據股票月線設定為20天。

本研究訓練模型架構為圖 3，以 Keras 框架[14]做為 LSTM 的實作選擇，首先在前面加了兩層 LSTM layer，並都加上了 Dropout 層來防止資料過度擬合（overfitting），最後再加上兩層有不同數目神經元的全連結層來得到只有1維數值的輸出結果。 X_1 至 X_n 為所需輸入的時間序列，經過模型訓練後得到隔一天的預測值 Y。

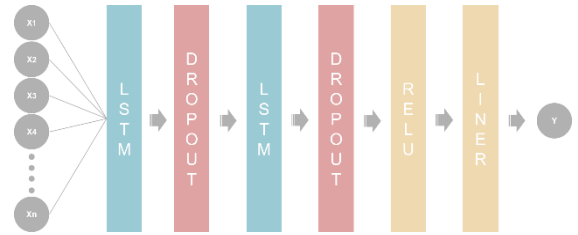


圖 3 訓練模型架構圖

在參數設定方面，目前沒有一個相對較肯定的說法來規範如何調整參數去配置類神經網路是最恰當的。根據 David[15]之研究，其所提出必須透過試誤法(Trial and Error)不斷嘗試，才可能找出最合適之神經元個數。如表 3，為本研究經不斷實驗及調整後得出相對較佳之參數設定。

表 3 LSTM 參數設定一覽表

| 參數 | 說明 | 設定 |
|--------------------------------|----------------|--------|
| time_frame | 記憶區間 | 20天 |
| LSTM_layer_1_neuron | 第一層 LSTM 神經元個數 | 128 |
| dropout1 | 丟棄神經元的比例 | 0.3 |
| LSTM_layer_2_neuron | 第二層 LSTM 神經元個數 | 128 |
| dropout2 | 丟棄神經元的比例 | 0.3 |
| DN_layer_3_neuron | DNN 神經元個數 | 32 |
| DN_layer_3_activation_function | DNN 激活函數 | ReLU |
| DN_layer_4_neuron | DNN 神經元個數 | 1 |
| DN_layer_4_activation_function | DNN 激活函數 | Linear |
| optimizer | 梯度下降法 | Adam |
| Loss function | 損失函數 | MSE |

本研究以均方誤差(MSE, mean-square error)來評估模型，MSE 越小說明預測模型具有更好的精確度。其計算方式是計算預測值與真實值之間差距的平方和，公式如下：

$$MSE = \sum_{i=1}^n (y_i - y_i^p)^2$$

n：測試樣本大小， y_i ：預測值， y_i^p ：實際值

由於模型訓練可能還是稍有誤差，為了再降低誤差值，本研究提出校正模型的方法，以預測值跟實際值去逐一進行每一天的預測值校正。校正方式如下：

R_t ：今日實際之收盤股價指數

P_t ：昨日預測今日之收盤股價指數

T：移動視窗校正天數

今日預測偏差值： $\beta_t = P_t - R_t$

$$\text{校正值} : C = \frac{\sum_{i=t-T+1}^{i=t} (P_i - R_i)}{T}$$

校正判斷：

$$\text{If } C > (R_t \times 10\%)$$

$$P_{t+1} = \begin{cases} P_{t+1} + C, & \beta_t > 0 \\ P_{t+1}, & \beta_t = 0 \\ P_{t+1} - C, & \beta_t < 0 \end{cases}$$

以移動式窗的方式，每T天去進行校正的動作，需要校正的判斷依據為，這T天中每天的前一日預測當日之收盤股價指數與當日實際之收盤股價指數的誤差絕對值總和取平均得到校正值C。若C大於當日實際收盤價 R_t 的漲跌幅10%，就對隔日的預測值校正動作，以當日偏差值 β_t 為依據，若 β_t 大於0，便對隔日預測值 P_{t+1} 加上校正值C；若 β_t 等於0，不進行校正動作；若 β_t 小於0，便對隔日預測值 P_{t+1} 減去校正值C，將誤差過多的數值距離拉近，如圖4之示意圖。但由於經過模型訓練後，漲跌幅

基本上不會超過10%，本研究在實驗過後採用0.5%進行判斷。

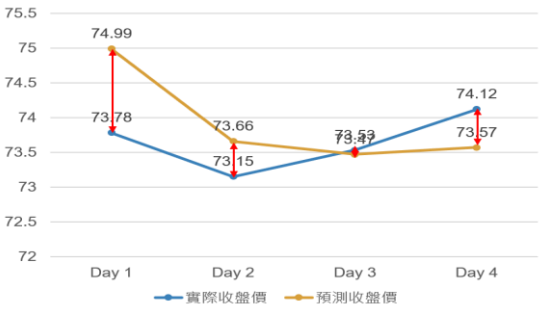


圖 4 校正數值示意圖

[12]提出的一次轉換及改良後的二次轉換策略，以「一日預測之報酬率」來判斷買、賣的時點。當一日的報酬率超過上限門檻值則買；當一日的報酬率低於下限門檻值則賣；若是介於上下限門檻值之間則持平。

$$f(\text{rate}) = \frac{P_{t+1} - R_t}{R_t}, \text{ 預測之報酬率}$$

P_{t+1} ：預測明日之收盤股價指數

R_t ：今日實際之收盤股價指數

上下門檻值設定：

- 買入時所花之成本為股票本金之1.425‰(手續費)
- 賣出時所花之成本為股票本金之2.425‰ (1.425‰的手續費及1‰的證交稅)

買賣訊號判斷規則：

- 若 $f(\text{rate}) > 1.425\text{‰}$ ，則為“買訊”
- 若 $-2.425\text{‰} \leq f(\text{rate}) \leq 1.425\text{‰}$ ，則為“持平”
- 若 $f(\text{rate}) < -2.425\text{‰}$ ，則為“賣訊”

但由於一次轉換的買、賣訊號過於敏感，且過於頻繁的交易會使得交易成本快速地增加。因此[12]提出「二次轉換」的交易策略，如表4，先出現連續兩次賣訊，但目前並無買過股票且本研究暫不考慮放空回補，因此不進行賣出動作。之後的連續出現兩次買訊，則“買入”；連續出現兩次賣訊，則“賣出”；前次已買或已賣之動作而維持『持平』，並不加買或加賣之動作，直到出現上述任一種情況，才轉換為買或賣。

表 4 模擬交易情況(10 日的為例)

| 天 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|------|---|---|---|---|----|---|---|---|---|--------|
| 一次交易 | 賣 | 賣 | 買 | 買 | 持平 | 賣 | 賣 | 買 | 買 | 持平(結算) |
| 二次交易 | - | - | - | 買 | - | - | 賣 | - | 買 | 結算 |

本研究提出不單單只以收盤價去做隔天預測可能買賣的價格，更預測了最高價與最低價，以兩者之預測值形成一個趨勢通道，如圖5，當判定為買點時，使用預測之最低價與預測收盤價之平均值進行買入；判斷為賣點時，使用預測之最高價與預測收盤價之平均值進行賣出。使交易能更有效的提高收益率，並給予可能模型預測不夠精準之緩衝範圍，因此取平均值。

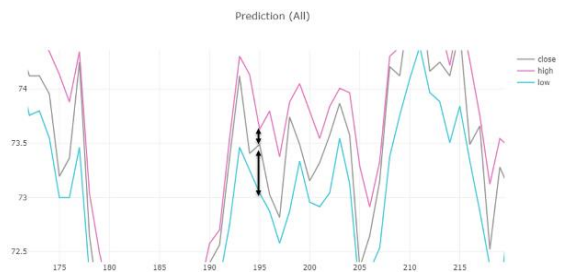


圖 5 高低價趨勢通道示意圖

4. 研究結果

表 5 為校正前與不同天數校正後的 MSE 比較表。可從表中觀察到，雖然差距都不大，但預測最高價及最低價在分別以 60、120 和 240 天移動視窗校正後的 MSE 較好於校正前的 MSE，證實經過校正後有明顯改善訓練模型的表現，而交易策略正是以預測最高及最低價進行買賣價的定奪，因此，預測最高及最低價校正後，可使其更接近實際價格，有機會提高獲利率。

表 5 模型校正前後比較表

| MSE | 校正前 | 校正後(使用不同天數) | | | | |
|-----|------|-------------|------|------|-------|-------|
| | | 5 天 | 20 天 | 60 天 | 120 天 | 240 天 |
| 收盤價 | 0.53 | 4.93 | 0.62 | 0.57 | 0.56 | 0.55 |
| 最高價 | 0.38 | 0.72 | 0.39 | 0.38 | 0.37 | 0.37 |
| 最低價 | 0.50 | 0.62 | 0.46 | 0.48 | 0.47 | 0.49 |

實驗結果為表 6，可看到以預測收盤價進行買賣時，二次轉換確實比一次轉換的交易策略的獲利率提升 2 倍。並且也加入了股市中常用來判斷買賣點的技術指標 KD 值進行比較，也確實二次轉換的表現優於 KD 指標。再來加入本研究提出的以高低價趨勢通道買賣策略，可以明顯看出在以二次轉換為交易策略時，以預測高低價做買賣比以預測收盤價做買賣略有提升。最後，觀察本研究提出對模型進行校正前後實驗結果的表現，明顯可以看出，以 60 天及 120 天進行校正後的獲利率為 2.59% 及 2.58%，比校正前的獲利率 2.2% 也有明顯提升，證實了校正模型的對於獲利率的影響。最後在研究中，使用二次交易策略且以 60 天移動視窗校正後的預測最高最低價進行買賣有最良好的獲利率表現為 2.59%。

表 6 模擬交易獲利率比較表

| 交易策略 | 買賣價 | 校正前獲利率 (%) | 校正後獲利率 (%) | | | | |
|------|-------|------------|------------|------|------|-------|-------|
| | | | 5 天 | 20 天 | 60 天 | 120 天 | 240 天 |
| 一次交易 | 預測收盤價 | 0.9 | -6.3 | -1.2 | -0.9 | -0.8 | -0.6 |
| 二次交易 | 預測收盤價 | 1.8 | 1.8 | 2.0 | 2.0 | 2.1 | 2.0 |

| | | | | | | | |
|-------|-------|------|------|------|------|------|------|
| 二次交易 | 預測高低價 | 2.20 | 2.40 | 2.51 | 2.59 | 2.58 | 2.40 |
| KD 指標 | 預測收盤價 | 0.70 | 2.10 | 0.01 | 0.50 | 0.60 | 0.50 |

相較於假設真的剛好能精準的以當日實際最高及最低價進行買賣，如表 7，在理想值為 4% 獲利率下，我們已經有 2.59% 的獲利率表現。

表 7 假設能以當日實際最高及最低價進行買賣的交易獲利率

| 交易策略 | 買賣價 | 校正前獲利率 (%) | 校正後獲利率 (%) | | | | |
|------|-------|------------|------------|------|------|-------|-------|
| | | | 5 天 | 20 天 | 60 天 | 120 天 | 240 天 |
| 二次交易 | 實際高低價 | 4.0 | 3.8 | 3.8 | 3.8 | 3.9 | 4.2 |

5. 結論

從研究結果可顯示出，在本研究中，分別提出的校正模型方法及預測高低價通道買賣策略中，模型經過校正後確實有降低了 MSE 值，連帶影響模擬交易後的獲利率表現。在模擬交易判斷買賣點時，使用二次轉換交易策略對於獲利率的表現確實優於一次轉換交易策略及 KD 技術指標。進行買賣時，使用預測的最高最低價做買賣，獲利率較優於單單使用預測收盤價做買賣，也實現了使股票買賣者可以買在最低價及賣在最高價的概念，是可以有效提升獲利率的。

誌謝

感謝中研院資訊所何建明教授的寶貴意見，感謝科技部計畫支持 107-2218-E-001-009, 108-2410-H-002-230-MY2 and 108-3116-F-002-003-CC1。

參考文獻

- [1] Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." *Neural computation* 9.8 (1997): 1735-1780.
- [2] Elman, Jeffrey L. "Finding structure in time." *Cognitive science* 14.2 (1990): 179-211.
- [3] Illustrated Guide to LSTM's and GRU's: A step by step explanation (<https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21>)
- [4] Box, G. E. P. And Jenkins, G. M. (1976). *Time Series Analysis, Forecasting and Control*, Holden-Day, San Francisco
- [5] Bollerslev, T. (1986), "Generalized Autoregressive Conditional Heteroscedasticity," *Journal of Econometrics*, 31, 307-27.
- [6] Hecht-Nielsen, Robert. "Theory of the backpropagation neural network." *Neural networks for perception*. Academic Press, 1992. 65-93.
- [7] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems*. 2012.
- [8] 劉昭雨、顏士淨，卷積神經網路在金融技術指標之應用，2017

- [9] 鄭允中, 基於長短期記憶遞迴神經網路之新台幣兌美元匯率預測模型. 臺灣大學資訊工程學研究所學位論文, 2017: p. 1-32.
- [10] Qian, F. and X. Chen. Stock Prediction Based on LSTM under Different Stability. in 2019 IEEE 4th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA). 2019. IEEE.
- [11] Chen, K., Y. Zhou, and F. Dai. A LSTM-based method for stock returns prediction: A case study of China stock market. in 2015 IEEE International Conference on Big Data (Big Data). 2015. IEEE.
- [12] 黃華山; 邱一薰. 類神經網路預測台灣 50 股價指數之研究. 國立彰化師範大學資訊管理學系研究所學位論文, 2005.
- [13] 台灣證券交易所, <https://www.twse.com.tw/zh/>。
- [14] Keras, <https://keras.io/>。
- [15] Davies, P. C., "Design issues in neural network development", NEUROVEST Journal, 5, pp. 21-25, 1994