陳郁晴、李俊賢（2019），『多目標特徵挑選與時間序列預測』，*中華民國資訊管理學報*，第二十六卷，第四期，頁 451-482。

多目標特徵挑選與時間序列預測

陳郁晴
國立中央大學資訊管理學系

李俊賢*
國立中央大學資訊管理學系

摘要

時間序列資料的變化有著眾多變因，在預測上一直是具有挑戰性的問題和研究。最常應用於股市上的股價變化，從時間的推移中找出股票之間的關係，因此本篇設計一多目標時間序列預測模型，應用於股價預測。模型包含兩種模型架構，人工神經網路（Artificial nural networks; ANN）及球型複數神經模糊系統（Sphere complex neuro-fuzzy system; SCNFS）。在 SCNFS 的部分，加入球形複數模糊集（Sphere complex fuzzy sets; SCFS）及箭靶層（Aim object layer），前者產生之歸屬程度為複數空間上之單位球體，使模型可進行多目標運算；後者以多對多的因果關係，降低模型後鑑部的運算負荷。在資料前處理的部分，加入多目標特徵挑選，以亂度熵（Entropy）的概念，從龐大的資料集中挑選出對目標具貢獻的輸入資料。在機器學習的部分，使用混合式機器學習演算法，結合連續型蟻群演算法（Continuous ant colony optimization; CACO）及遞迴最小平方估計法（Recursive least squares estimation; RLSE），有效地訓練模型的參數。本篇共執行三個實驗，實驗一透過單目標預測驗證 ANN-SCNFS 的效能及 CACO-RLSE 的優化效果；實驗二以相同資料集同時進行單目標預測及多目標預測，驗證模型可一次預測多目標之能力；實驗三透過不同標的股進行多目標預測，研究資料集在使用上的彈性。實驗結果表明本研究提出之預測結果大多優於文獻結果，顯示在股票的時間序列預測上有良好的效能。

關鍵詞：多目標特徵挑選、人工神經網路、球型複數模糊集、球型複數神經模糊系統、混合式機器學習

* 本文通訊作者。電子郵件信箱：jamesli@mgt.ncu.edu.tw
2019/04/22 投稿；2019/06/09 修訂；2019/07/04 接受

Chen, Y.C. and Li, C. (2019), 'Multi-target feature selection and time-series forecasting', *Journal of Information Management*, Vol. 26, No. 4, pp. 451-482.

Multi-target Feature Selection and Time-series Forecasting

Yu-Ching Chen
Department of Information Management, National Central University

Chunshien Li*
Department of Information Management, National Central University

Abstract

Purpose ─ Time series forecasting is a challenging research issue. In the past research, most of them were single-target forecasting. However, in the stock market, stocks will effect each other. Therefore, we hope to predict the stock price on multiple targets simultaneously, and find the relationship between them through this research.

Design/methodology/approach ─ In the data preprocessing phase, we use multi-target feature selection to find the useful data for prediction. Then, in order to predict multiple targets, the proposed model in the paper combines artificial neural networks (ANN) and asymmetric sphere complex neuro-fuzzy system (SCNFS). Moreover, we use the hybrid machine learning algorithm CACO-RLSE, which combines continuous ant colony optimization algorithm (CACO) and recursive least squares estimation (RLSE) to train the parameters in the model.

Findings ─ The result shows that the performance of multi-target prediction is better than single-target prediction. In addition, the use of input dataset is flexible, we can use US index to predict US stock price.

Research limitations/implications ─ This study only focused on US stock market information. In the future, we plan to research the different stock market, and get more information to do multi-target time-series forecasting.

* Corresponding author. Email: jamesli@mgt.ncu.edu.tw
2019/04/22 received; 2019/06/09 revised; 2019/07/04 accepted

Practical implications—In this paper, we have proposed a multi-target time-series forecasting model. This method can give the investors some support to make investment decisions.

Originality/value—We use hybrid machine learning algorithm, CACO-RLSE, to improve the efficiency of training parameters. Besides that, we change the structure of original SCNFS, and let it become an asymmetric model structure by using the novel aim-object layer (AOL) in the model.

Keywords: multi-target feature selection, artificial neural networks (ANN), sphere complex fuzzy sets (SCFS), sphere complex neuro-fuzzy system (SCNFS), hybrid machine learning algorithm

壹、緒論

在人工神經網路（Artificial neural networks; ANN）發展初期，許多人致力於時間序列的相關研究。最常應用的資料集為股市資料，源自於股市資料最易取得，也具有眾多的影響因素，因此股價預測的困難度也隨之提升，給予研究人員極大的挑戰。然而，在各個國家的股票市場中，以美國的股市最為發達，不論是金融商品的發行量，還是市場發展程度，美國在世界上均是首屈一指的，因此本篇會從美國的金融市場中挑選出幾個具有重大影響力的股票，並使用本篇所設計的模型針對美國股市進行研究。

過去有許多時間序列的研究，Wang 與 Leu（1996）使用自回歸移動平均模型（Autoregressive integrated moving average model; ARIMA），針對台灣加權股價指數進行預測。Hassan 與 Nath（2005）以隱藏式馬可夫模型（Hidden Markov model; HMM）預測航空股股價，驗證模型效能。Hadavandi、Shavandi 與 Ghanbari（2010）根據模糊邏輯提出遺傳模糊系統（Genetic fuzzy system; GFS）。近幾年多以深度網路為主，最常見為遞迴神經網路（Recurrent neural networks; RNN）和長短期記憶網路（Long short-term memory; LSTM）。Selvin 等（2017）結合 LSTM、RNN 及卷積神經網路（Convolutional neural networks; CNN）滑動視窗，預測印度國家證券交易所上市股。Faustryjak、Jackowska-Strumiłło 與 Majchrowicz（2018）提出將 LSTM 結合 Google 的搜尋趨勢 API，進行股價趨勢的研究。

過去的研究大多為單目標預測，輸入的來源也較為單一，因此本研究設計一多目標預測模型，ANN-SCNFS(AOL)，支援大量輸入資料集的運算。以球型複數模糊集（Sphere complex fuzzy sets; SCFS），支援多目標的運算。藉由多目標特徵挑選從龐大的資料集中保留對目標有益處的資料作為輸入。結合連續型蟻群演算法（Continuous ant colony optimization; CACO）和遞迴最小平方估計法（Recursive least squares estimation; RLSE），以混合式機器學習演算法提升學習效率。

第貳節為文獻探討，會根據本篇的研究方向介紹研究背景的發展歷史，以及對其領域具重大轉折的文獻，進行簡短式的介紹。第參節為研究方法，會詳細說明本研究所使用的方法、流程，以及架構的設計原因。將 ANN 與改良版 SCNFS 進行結合，並以新型混合式機器學習演算法，CACO-RLSE，進行有效率的參數訓練。第肆節為實驗結果，會針對美國的指數及股票使用本篇模型進行單目標及多目標的預測，並與文獻進行模型表現的比較。第伍節為結果與討論，會針對本篇的研究進行各種議題的探討，以及實驗結果的討論。第陸節為結論與建議，會針對本篇的研究做總結，並說明本篇對社會產生的貢獻。

貳、文獻探討

一、人工神經網路

人工神經網路（Artificial neural networks; ANN）起源於 McCulloch 與 Pitts（1943）以人類神經元傳導為概念所提出的神經計算（Neuro-computing），然而當時並無有效的應用。直到 Rosenblatt（1958）成功應用其概念，並以簡單的加減法建立一用於圖形辨識的演算法，稱為感知機（Perceptron）。史丹佛大學教授 Widrow 與其學生 Hoff 於 1960 年提出自適應線性神經元（Adaptive Linear Element; ADALINE）（Widrow & Hoff 1960），由於 ADALINE 的模型架構特性，又被稱為單層神經網路，並應用於文字的辨識，後又與另一學生 Winter 提出多層的 ADALINE（Multiply ADALINE; MADALINE）（Winter & Widrow 1988）。

在 1960 到 1980 年間，ANN 的發展出現了空窗期，源於當時不足的電腦計算能力以及 ANN 的互斥或問題。直到 Werbos（1974）發明倒傳遞演算法（Backpropagation; BP）才得以解決，又可訓練更為複雜的神經網路，因而開啟了 ANN 的快速發展，產生許多 ANN 的應用，其中以模糊類神經網路最為盛行，以人類的模糊思維為概念進行模糊控制。

二、模糊集合及模糊推論系統

Zadeh（1965）提出模糊集（Fuzzy sets）的概念，讓元素與集合之間的歸屬程度從 0 和 1 轉為介於 0 和 1 間的數值，體現真實世界中的灰色地帶，提高錯誤容忍性。相對於傳統集合（Crisp sets），模糊集可產生更多訊息量。Takagi 與 Sugeno（1985）以模糊集提出模糊推論系統（Fuzzy inference system; FIS），以模糊集作為條件的依據，建立模型的前鑑部（IF-part）及後鑑部（THEN-part），藉由模型將輸入資料推測出結果。Jang（1993）根據 FIS 提出適應性類神經模糊系統（Adaptive neuro-fuzzy inference system; ANFIS），以混合式機器學習演算法調整模型參數，結合 Hsia（1997）的最小平方估計法（Least squares estimation; LSE）及 BP，前向傳播中以 LSE 調整參數，反向傳中則由 BP 調整參數。

除一般模糊集外，Ramot 等（2002）提出新的模糊集，複數模糊集合（Complex fuzzy sets; CFS），其歸屬程度為一複數型態的數值，包含實部及虛部，將數值範圍從數線轉為平面，形成半徑為 1 的複數單位圓盤（Unit disc complex plate; UDCP）。比起一般模糊集，CFS 可提升模型的映射能力與效能。另外，Li 與 Chiang（2010）以 ANFIS 為基礎，將模型中的模糊集改用為 CFS，支援多目標運算及降低參數量，命名為複數神經模糊系統（Complex neuro-fuzzy system; CNFS）。

Tu 與 Li（2019）根據 CFS 提出球型複數模糊集合（Sphere complex fuzzy sets; SCFS），為本篇模型的重點之一，以歸屬函數（Membership function）及振幅函數（Amplitude function）轉換輸入資料，產生多個複數的歸屬程度，應付更多目標的需求。與此同時，又以 ANFIS 做為模型的主架構，將模糊集改為 SCFS，並將其命名為球型複數神經模糊系統（Sphere complex neuro-fuzzy system; SCNFS）。

三、機器學習演算法

除模型的設計外，機器學習演算法在應用中也扮演重要角色，因應不同的情況，有許多種不同的演算法。若是單純的ANN模型架構，用導數即可訓練參數，BP 為最具代表性的演算法。然而對於複雜的模型是無法使用 BP，因此也發展出一些無導數的演算法（Derivative-free algorithm）。

Holland（1992）根據達爾文（Darwin）進化論，以進化論的遺傳現象：遺傳、突變、選擇和雜交，提出基因演算法（Genetic algorithm; GA）。Kennedy 與 Eberhart（1995）以鳥群行為為靈感提出粒子群演算法（Particle swarm optimization; PSO），以鳥的飛行慣性、自身及群體內的學習經驗尋找食物。Socha 與 Dorigo（2008）以蟻群為靈感提出連續型蟻群演算法（CACO），加入輪盤法（Roulette wheel selection）參考多個較佳的位置來尋找新位置，並改善原始蟻群演算法（Ant colony optimization; ACO）（Dorigo, Maniezzo and Colorni 1996）無法解決連續型數值的問題。Karaboga（2005）根據蜂群行為模式提出人工蜂群最佳化演算法（Artificial bee colony optimization; ABC），由三種不同職務的蜜蜂，合作尋找花蜜的位置。Li、Yang 與 Nguyen（2012）提出自我學習粒子群最佳化演算法（Self-learning particle swarm optimization; SLPSO），利用四種策略應付不同問題，改良 PSO 單一學習模式，可處理更複雜的問題。Mirjalili 與 Lewis（2016）提出鯨群最佳化演算法（Whale optimization algorithm; WOA），以三種不同的移動策略，在不同疊代下，策略的觸發機率也不同，訓練初期有較大的機率是以隨機的方式尋找新位置，後期則是有較大的機率以趨向最佳解方式更新位置。

四、特徵挑選

在模型的訓練中，使用的資料也是一大重點。股價有眾多的影響因素，但卻只能從經驗中得知與股票具相關性的因素。由於本篇希望擷取更多資料進行預測，因此資料的挑選就變得極其重要，若接收資料過於龐大，對模型會產生極大的負荷。為從眾多因素中選擇有益的資料，本篇加入特徵挑選以進行資料前處理，將冗餘或無關的資料篩選掉，同時可簡化模型、縮短模型訓練時間、降低過度擬合。

Whitney（1971）提出循序向前選擇法（Sequential forward selection; SFS），以前 K 個最近鄰居法則尋找相關的資料，找出最相關的特徵組合，為目前最基本最常使用的特徵選擇演算法。除此之外，評估相關性的方法有許多種，如 Kwak 與 Choi（2002）提出的互資訊（Mutual information; MI），以 Shannon（1948）的亂度指標熵（Entropy）為概念，從亂度中找出資料間的相關性。Tu 與 Li（2017）提出以影響資訊（Influence information）作為相關性指標，針對各個目標挑選有益的特徵，再從這些特徵中挑選出對整體目標具有貢獻的特徵。

參、研究方法

為解決多目標預測問題，本篇以多目標特徵挑選（Multi-target feature selection）作為資料前處理方法，從龐大的輸入資料中經由影響資訊矩陣（Influence information matrix; IIM）挑選出對目標有益的資料。以 SCFS 使模型可進行多目標運算，其多個複數型態的 MD，可大幅降低參數量。以非對稱式 SCNFS 使模型形成多對多的非對稱因果關係，使其更容易接近目標值。在 SCNFS 運算前加入 ANN 以容納更多的輸入特徵，同時可降低 SCNFS 的運算負荷，因此本篇將提出的多目標預測模型命名為 ANN-SCNFS（AOL），模型概念圖如圖 1 所示。

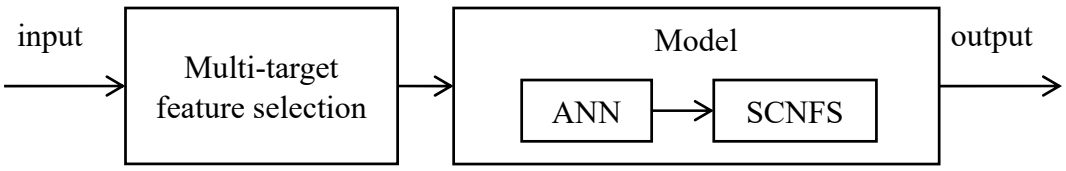

圖 1：模型概念圖

一、多目標特徵挑選

為縮短模型訓練時間，降低過度擬合，同時處理多目標預測問題，本篇採用多目標特徵挑選作為本研究資料處理的第一步，將冗餘的或無關的資料去除掉，保留下來的特徵即為模型的輸入。在挑選時會先分別將輸入資料集與目標資料集放置於 CP 及 TS 中，集合的說明列於表 1。本篇以影響資訊（Influence information）建立 IIM，並以此進行特徵挑選。影響資訊是經由互資訊衍生的評估指標，互資訊的資訊量是相互對稱的，然而在現實世界中卻不一定相等，他人給予的資訊量與我們付出的並不完全相同，因此，將互資訊以發生的機率作為權重

進行計算，其結果即為影響資訊，並以此作為本篇的相關性評估指標。

表 1：多目標特徵挑選符號說明

Symbol	Description
CP	候選特徵池（Candidate feature pool），$\mathrm{CP} = \{f_i, i = 1, 2, \dots, n_{CP}\}$
f_i	第i個候選特徵（The ith candidate feature）
n_{CP}	候選特徵總數
$\mathrm{SP}^{(j)}$	針對第j個目標的挑選特徵池（Selected feature pool for the jth target），$\mathrm{SP}^{(j)} = \left\{x_k^{(j)}, k = 1, 2, \dots, n_{SP^{(j)}}\right\}$
$n_{SP^{(j)}}$	針對第j個目標所挑選出的特徵總數
$x_k^{(j)}$	針對第j個目標所挑選的第k個特徵
Ω	結合所有目標 SP 中的特徵集合
n_Ω	有出現於任一 SP 中的特徵總數
ϕ_k	第k個出現於任一 SP 中的特徵
FP	最終特徵池（Final feature pool），$\mathrm{FP} = \{s_i, i = 1, 2, \dots, n_{FP}\}$
n_{FP}	最終挑選之特徵總數
TS	目標集合（Target sets），$\mathrm{TS} = \left\{t_j, j = 1, 2, \dots, n_{TS}\right\}$
t_j	第j個目標（Target）
n_{TS}	目標總數
$I(X, Y)$	隨機變數X及Y之間的互資訊（Mutual information, MI）
$I_{f_i \to f_j}$	特徵f_i對f_j的影響資訊量（Influence information）
$p_d(\cdot)$	機率密度函數（Probability density function）
H	熵（Entropy）
$g(f_i \to t_j)$	特徵f_i對目標t_j的選取增益（Selection gain）

互資訊是基於亂度熵所計算的指標，進行熵的計算前需對資料做機率密度估計（Bowman & Azzalini 1977），以了解資料的機率密度分佈（Probability density distribution），以其計算資料混亂程度，如(1)式所示。為進行互資訊的計算，需使用(2)式計算條件熵，再以(3)式計算互資訊。影響資訊計算方式如(4)式所示，本篇是以 0 為界線將資料分割為兩個不同的資料區段，並以其發生機率作為權重，進行加權式的互資訊計算，並將特徵間的影響資訊建立 IIM。

$$H(X) = -\int_X p_d(x)\log\big(p_d(x)\big)\,dx, \tag{1}$$

$$H(X|Y) = -\int_Y \int_X p_d(x,y)\log\big(p_d(x|y)\big)\,dxdy, \tag{2}$$

$$I(X,Y) = H(X) - H(X|Y), \tag{3}$$

$$I_{f_i\to f_j} = I\big(f_i, f_j^{+}\big)\cdot\int_0^{\infty} p_d\big(f_j\big) + I\big(f_i, f_j^{-}\big)\cdot\int_{-\infty}^{0} p_d\big(f_j\big). \tag{4}$$

為得知哪些輸入特徵對於目標具影響力，首先會建立各個目標的 SP，基於 IIM 計算特徵對目標的選取增益，如(5)式所示，代表選取此特徵後帶給目標的增益資訊量，以評估特徵是否可被納入 SP 中，若選取增益為一正值，即代表加入此特徵對目標有正面的影響，提供更多的資訊，需被納入 SP 中。其中，$R_{f_i\to SP^{(j)}}$為冗餘資訊量，為影響資訊的負項，由特徵對已被挑入 SP 中之特徵所計算的資訊量，若特徵的冗餘資訊量大於影響資訊，代表其給予的資訊已有其他特徵提供，代表此特徵就不需被納入 SP 中，其公式如(6)式所示。

$$g\big(f_i \to t_j\big) = I_{f_i\to t_j} - R_{f_i\to SP^{(j)}}, \tag{5}$$

$$R_{f_i\to SP^{(j)}} = \frac{\sum_{k=1}^{n_{SP^{(j)}}} I_{f_i\to x_k^{(j)}} + I_{x_k^{(j)}\to f_i}}{2n_{SP^{(j)}}}, \tag{6}$$

其中，$x_k^{(j)}$是針對第j個目標所挑選出的第k個特徵，$x_k^{(j)} \in SP^{(j)}$；$n_{SP^{(j)}}$為針對第j個目標所挑選的特徵總數。為取出對所有目標具貢獻的特徵，聯集所有 SP，取出曾被挑選的特徵，並將其先放入集合Ω中，$\Omega = \{\phi_k, k = 1, 2, \dots, \mathrm{n}_\Omega\}$，再以(7)式計算對整體目標的貢獻程度（Contribution index）進一步篩選，其中，$\omega(\phi_k)$代表特徵ϕ_k的對所有目標的覆蓋率，如(8)式所示；$n_{OL}(\phi_k)$代表特徵ϕ_k於各個 SP 中的出現次數；$g_{sum}(\phi_k)$代表特徵ϕ_k對所有目標的影響資訊總合，如(9)式所示。最後以(10)式平均ω和平均g_{sum}的乘積值作為門檻值，取出其貢獻程度大於門檻值的特徵放入 FP 中，結束多目標特徵挑選的運算。

$$\rho(\phi_k) = \omega(\phi_k)\cdot g_{sum}(\phi_k), \tag{7}$$

$$\omega(\phi_k) = \frac{n_{OL}(\phi_k)}{n_{TS}}, \tag{8}$$

$$g_{sum}(\phi_k) = \sum_{j=1}^{n_{TS}} g\big(\phi_k \to t_j\big), \tag{9}$$

$$\rho_{th} = \overline{\omega}\cdot\overline{g_{sum}}, \tag{10}$$

其中，$k = 1, 2, \ldots, n_\Omega$；$\bar{\omega}$為所有特徵$\phi_k$對目標的平均覆蓋率；$\overline{g_{sum}}$為所有特徵$\phi_k$對目標影響資訊總合之平均。多目標特徵挑選可以總結為下列 5 個步驟：

1. 計算影響資訊矩陣。
2. 計算特徵對各目標的選取增益值，並建立各目標的 SP。
3. 聯集各目標 SP 的特徵，置於集合Ω中，令其特徵為ϕ_k。
4. 以ϕ_k的覆蓋率和對各目標的影響資訊總和計算ϕ_k的貢獻程度。
5. 以平均的覆蓋率及平均的影響資訊總和之乘積作為門檻值，從Ω中取出達到門檻值的特徵，並將其放置於 FP 中。

二、球型複數模糊集

為解決多目標預測問題，本篇模型以 SCFS 取代傳統模糊集，由此產生多個複數型態的歸屬程度（Membership degree; MD），以向量的方式儲存，不僅可以支援多目標預測，還可大幅減少 CACO 的訓練參數量，又因 SCFS 的歸屬程度為位於複數平面上的數值，可提升模型的映射能力與效能。SCFS 是基於半徑為 1 之球空間為概念所建立的方法，以高斯函數計算原始的歸屬程度，如(11)式所示。

$$r = \text{Gaussian}(h; c, \sigma) = \exp\left(\frac{(h-c)^2}{2\sigma^2}\right), \tag{11}$$

其中，c代表中心點，σ代表標準差。再以不同的相位角進行投射計算產生多個 MD，相位角是經由歸屬函數的偏微分計算，以相位角頻率調整參數（φ）進行微調，若以第 1 個相位角θ_1為例，如(12)式所示。

$$\theta_1(h; c, \sigma, \varphi_1) = \frac{\partial r(h;c,\sigma)}{\partial h} \times \varphi_1, \tag{12}$$

其中，φ_1是第一個相位角的相位角頻率調整參數。

根據上述的方式計算 MD 和相位角後，即可將 MD 於參數空間中以相位角藉由$\sin(\cdot)$及$\cos(\cdot)$進行投射計算，將原始的 MD 轉換為多個實數型態的 MD，如(13)式所示，其中，N為模型目標的個數，然而，為了使模糊集從原始 0 到 1 的實數轉變於 UDCP 上的歸屬程度，將(13)式所計算的實數型態 MD 以複數的方式組合起來，形成複數型態的向量，以其作為 SCFS 的輸出，如(14)式所示。

$$\begin{cases} u_N = r \times \sin\theta_{N-1}, \\ u_k = r \times \prod_{i=k}^{N-1} \cos\theta_i \times \sin\theta_{k-1}, k \in [2, N-1], \\ u_1 = r \times \prod_{i=1}^{N-1} \cos\theta_i, \end{cases} \tag{13}$$

$$\mu_k = u_{2k-1} + u_{2k}j, \tag{14}$$

$k = 2, 3, \dots, N-1$，其中，N為模型的目標個數；$j = \sqrt{-1}$。

三、球型複數神經模糊系統

SCNFS 是基於 SCFS 所建立的 FIS，然而本篇所使用的 SCNFS 將原始的 SCNFS（Tu & Li 2019）做一些改良，其流程圖如圖 2 所示，原始的 SCNFS 是使用一對一的規則關係去進行推論，然而本篇所使用的改良版 SCNFS 在模型的正規化層（Normalization layer）後加了一層箭靶層（Aim object layer; AOL），以多對多的因果關係推論結果，同時也可降低後鑑部的規則個數，降低結果層（Consequence layer）的運算負荷。模型架構是分別以輸入資料集和目標資料集決定前鑑部及後鑑部的架構，決定其規則數K及Q。為了論述清楚起見，所使用的符號整理於表 2。

表 2：模型符號說明

Symbol	Description
h_i	模型的第i個輸入，$i = 1, 2, \dots, M$
M	模型的輸入個數
T_i	第i個輸入的模糊集集合，$i = 1, 2, \dots, M$，$\mathrm{T}_i = \{A_{i,1}, A_{i,2}, \dots, A_{i,\lvert T_i \rvert}\}$
$\lvert \mathrm{T}_i \rvert$	第i個輸入的模糊集個數
k	$k = 1, 2, \dots, K$
K	前鑑部（前鑑部層及正規化層）的個數
q	$q = 1, 2, \dots, Q$
Q	後鑑部（箭靶層及結果層）的個數
N	模型所針對的目標個數
$\vec{\beta}^{(k)}$	第k個前鑑部之啟動強度向量，$\vec{\beta}^{(k)} = \left[\beta_1^{(k)}, \beta_2^{(k)}, \dots, \beta_N^{(k)}\right]^{\mathrm{T}}$
$\vec{\rho}^{(k)}$	第k個前鑑部之正規化啟動強度向量，$\vec{\rho}^{(k)} = \left[\rho_1^{(k)}, \rho_2^{(k)}, \dots, \rho_N^{(k)}\right]^{\mathrm{T}}$
$\vec{\lambda}^{(q)}$	第q個箭靶層之輸出向量，$\vec{\lambda}^{(q)} = \left[\lambda_1^{(q)}, \lambda_2^{(q)}, \dots, \lambda_N^{(q)}\right]^{\mathrm{T}}$

$\vec{\hat{y}}^{(q)}$	模型第q個輸出，$\vec{\hat{y}}^{(q)} = \left[\hat{y}_1^{(q)}, \hat{y}_2^{(q)}, \ldots, \hat{y}_N^{(q)}\right]^{\mathrm{T}}$
$\vec{\hat{y}}$	模型的輸出向量，$\vec{\hat{y}} = [\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_N]^{\mathrm{T}}$
$\vec{y}$	模型的目標向量，$\vec{y} = [y_1, y_2, \ldots, y_N]^{\mathrm{T}}$

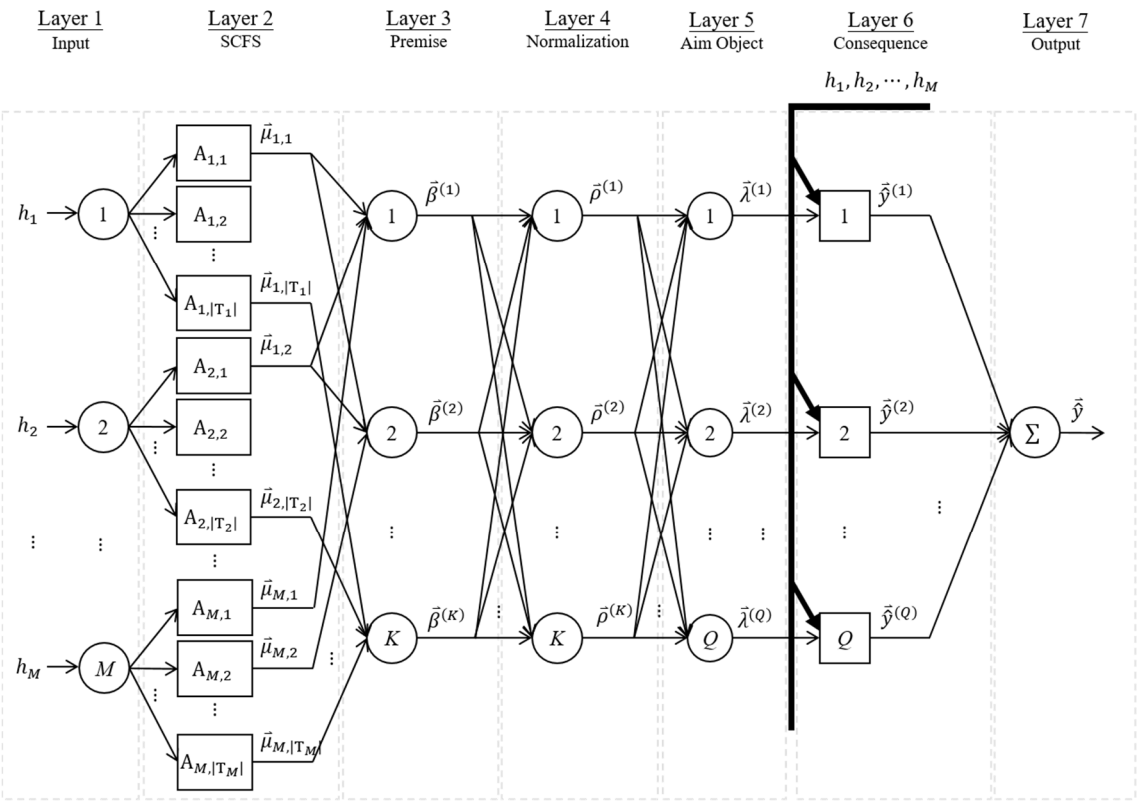

圖 2：SCNFS 模型流程圖

（一）建置模型架構

進行模型運作前需決定模型的神經元架構，為使架構更為客觀，以 Chiu（1994）的減法分群法（Subtractive clustering）分別對輸入資料集與目標資料集分群，決定前鑑部及後鑑部的神經架構。由於 AOL 的接收值及模型的目標值皆為複數型態，因此 AOL 的中心點及標準差需轉換於 UDCP 中。轉換方式是先以(15)式將目標值代入複數高斯函數中，以轉換於 UDCP 中的目標值進行箭靶層中心點及標準差的計算，其中，$r(\cdot)$為高斯函數，其計算公式如(11)式所示，h為需進行轉換的目標值，c與σ分別為目標資料集的平均值及標準差，最後將轉換過的目標

值以前面減法分群決定的規則數代入模糊 C-means 中，決定箭靶中心點，$C_{AOL}^{(q)}$，再依據轉換過的目標值及箭靶中心點計算箭靶標準差，$S_{AOL}^{(q)}$。

$$\text{cGaussian}(h;c,\sigma) = r(h;c,\sigma) \times \exp\left(\frac{\partial r}{\partial h} \cdot \sqrt{-1}\right). \tag{15}$$

（二）模型流程

建構完模型架構後，按照圖 2 即可進行多目標預測，以不同模糊集的組合建立前鑑部規則，並計算資料對規則的符合程度，以調整結果層的輸出值。

Layer 1：輸入層（Input layer）

此階層是用於接收輸入資料（h），將其轉換為符合模型的型式，再放入模型中進行運算，圖 2 中輸入層的神經元個數，M，是代表輸入 SCNFS 的資料個數。

Layer 2：球型複數模糊集層（SCFS layer）

此階層是以模糊集作為輸入資料的條件，進行輸入資料在各個條件下的歸屬程度，MD，因此各個輸入特徵的條件是使用不同的中心點和標準差建立多個模糊集，詳細的計算步驟已於上一節說明。

Layer 3：前鑑部層（Premise layer）

此階層的神經元代表一個規則，由輸入特徵的不同條件組成規則，將 SCFS 層所計算的 MD 根據不同的組合以代數積（Algebraic product）的方式進行模糊交集運算，輸出值代表資料對規則的符合程度，又稱啟動強度（Firing strength）。每一個前鑑部的啟動強度為一向量。第 k 個啟動強度記為 $\vec{\beta}^{(k)} = \left[\beta_1^{(k)}, \beta_2^{(k)}, \dots, \beta_N^{(k)}\right]^{\mathrm{T}}$，$k = 1,2,\dots,K$，其中，$K$代表模型所有的前鑑部數量，$N$代表模型的輸出數量。

Layer 4：正規化層（Normalization layer）

此階層是為了降低啟動強度的離散程度，計算各個規則的啟動強度占所有規則的比例，將啟動強度進行正規化，把各個規則的啟動強度除以所有規則之啟動強度總和，壓縮其數值於[0, 1]之間，如(16)式所示。

$$\vec{\rho}^{(k)} = \frac{\vec{\beta}^{(k)}}{\sum_{i=1}^{K} \vec{\beta}^{(i)}}, \tag{16}$$

$k = 1,2,\dots,K$，其中，$\vec{\rho}^{(k)} = \left[\rho_1^{(k)}, \rho_2^{(k)}, \dots, \rho_N^{(k)}\right]^{\mathrm{T}}$，是第$k$個前鑑部之正規化啟動強度向量（Normalized firing strength vector）；N代表模型所針對的目標個數。

Layer 5：箭靶層（Aim object layer）

此階層中的每一個箭靶會接收前一層所有的輸出，以箭靶中心點（$C_{AOL}^{(q)}$）及

箭靶標準差（$S_{AOL}^{(q)}$）代入複數高斯函數進行轉換，並且將所有轉換過的值進行加總，其結果即為箭靶層的輸出，如(17)式所示。

$$\vec{\lambda}^{(q)} = \sum_{k=1}^{K} w\left(\vec{\rho}^{(k)}; C_{AOL}^{(q)}, S_{AOL}^{(q)}\right), \tag{17}$$

$q = 1, 2, \ldots, Q$，其中，$\vec{\lambda}^{(q)} = \left[\lambda_1^{(q)}, \lambda_2^{(q)}, \ldots, \lambda_N^{(q)}\right]^{\mathrm{T}}$；$w(\cdot)$為箭靶層的神經元之轉移函數，其定義如下，$w\left(\vec{\rho}^{(k)}; C_{AOL}^{(q)}, S_{AOL}^{(q)}\right) = r\left(\vec{\rho}^{(k)}; C_{AOL}^{(q)}, S_{AOL}^{(q)}\right) \times \exp\left(\frac{\partial r}{\partial \vec{\rho}^{(k)}} j\right)$，$j = \sqrt{-1}$。

Layer 6：結果層（Consequence layer）

此階層採用 Takagi-Sugeno 之 FIS 後鑑部函式，是一種線性函數，如(18)式所示，將輸入以其權重進行加總，並將結果與箭靶層的輸出相乘，得到結果層各個規則的輸出，其中，a為機器學習所需訓練之參數，h為模型輸入。

$$\vec{\hat{y}}^{(q)} = \vec{\lambda}^{(q)} \times \left(a_0^{(q)} + a_1^{(q)} h_1 + \cdots + a_M^{(q)} h_M\right) \tag{18}$$

其中，$\vec{\hat{y}}^{(q)} = \left[\hat{y}_1^{(q)}, \hat{y}_2^{(q)}, \ldots, \hat{y}_N^{(q)}\right]^{\mathrm{T}}$。

Layer 7：輸出層（Output layer）

此階層是用以接收結果層的所有輸出並加總，其計算結果即為整個模型的最終輸出，如(19)式所示。

$$\vec{\hat{y}} = \sum_{i=1}^{K} \vec{\hat{y}}^{(i)} \tag{19}$$

其中，$\vec{\hat{y}} = [\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_N]^{\mathrm{T}}$。

四、混合式機器學習演算法

本篇以新混合式機器學習演算法調整參數，CACO-RLSE，結合了 CACO 及 RLSE。CACO 訓練的參數有：ANN 的權重（weight）和偏差值（bias），SCFS 的中心點（c）、標準差（σ）和相位角頻率調整參數（φ）。RLSE 訓練的參數則是結果層中輸入資料權重值（a）。由於模型結果層為線性函數，可以線性回歸的方式快速找到近似解，同時降低 CACO 的訓練參數量，有效地提升準確率及效率。

（一）連續型蟻群演算法（CACO）

Dorigo 等（1996）提出蟻群演算法（Ant colony optimization; ACO），解決旅行推銷員問題（Travelling salesman problem; TSP），利用已知的路徑及拜訪位置決定最短路徑，但卻無法解決連續型數值問題，因此 Socha 與 Dorigo（2008）提出

了 CACO，是一種無導數最佳化演算法，CACO 在更新位置時，會考慮多個位置，以其損失值大小的順序決定每隻螞蟻權重，並以輪盤法來選擇一隻螞蟻，以此螞蟻所定義的搜索範圍去找尋新位置，流程圖如圖 3 所示。

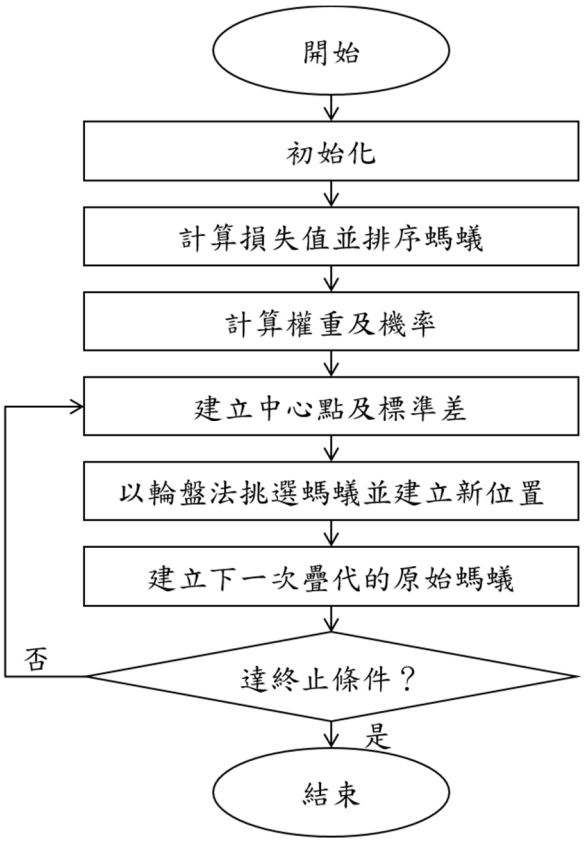

圖 3：CACO 運算流程圖

步驟一：初始化

此步驟需初始化的參數如表 3 所示，並以隨機的方式初始原始螞蟻的位置。

步驟二：計算損失值並排序螞蟻

將k個原始螞蟻的位置代入$f(\cdot)$中，計算出各個螞蟻的損失值，再依照螞蟻的損失值由小到大排列，排序越前面的螞蟻，其表現越好。

表 3：CACO 參數對照表

Symbol	Explanation
k	原始螞蟻個數（Ant archive size）
m	樣本螞蟻個數（Ant sample size），指疊代中建立新螞蟻的數量
$nDim$	維度（Dimension），指螞蟻位置的空間維度，需訓練的參數個數
$f(\cdot)$	成本函數（Cost function），用以計算螞蟻的損失值，評估表現
s	位置（Position）
ξ	蒸發率（Evaporation rate），調整訓練的收斂速度，越小收斂越快
q	學習率（Learning rate），用於調整原始螞蟻被挑選之權重平衡，越大被挑選到的機率越平均
t	疊代次數（Iteration），代表重複運算的次數

步驟三：計算權重及機率

螞蟻權重以常態分配的方式計算，如(20)式，中心點為 1，並以k及q的乘積作為標準差，排序越靠前的螞蟻權重越大，再依(21)式計算螞蟻被挑選到的機率。

$$\omega_l = \frac{1}{qk\sqrt{2\pi}} \times e^{-\frac{(l-1)^2}{2q^2k^2}}, \tag{20}$$

$$p_l = \frac{\omega_l}{\sum_{r=1}^{k} \omega_r}, \tag{21}$$

$l = 1, 2, \ldots, k$。其中，q為學習率；k為原始螞蟻個數。

步驟四：建立中心點及標準差

由於新螞蟻需要在原始螞蟻的周圍去尋找新位置，因此會以原始螞蟻的位置為中心，如(22)式所示，以螞蟻與其他原始螞蟻之間的分散程度決定搜索的寬度，如(23)式所示，由此定義出每隻原始螞蟻的搜尋範圍。

$$\boldsymbol{\mu}^i = \{\mu_1^i, \ \ldots, \mu_k^i\} = \{s_1^i, \ \ldots, s_k^i\}. \tag{22}$$

$$\sigma_l^i = \xi \sum_{e=1}^{k} \frac{\left|s_e^i - s_l^i\right|}{k-1}. \tag{23}$$

其中，$\boldsymbol{\mu}^i$為疊代為第i代時的所有中心點，μ_l^i代表第i次疊代第l個中心點；σ_l^i代表第i次疊代第l個標準差；s_l^i代表第i次疊代第l個螞蟻的位置；ξ為蒸發率。

步驟五：以輪盤法挑選螞蟻並建立新位置

根據螞蟻被挑選的機率（p_l）以輪盤法選擇螞蟻（假設被挑選到的螞蟻是第a

個），根據螞蟻a所定義的搜尋範圍尋找新位置，如(24)式所示。

$$s_l^i = s_a^i + \sigma_a^i \times r, \tag{24}$$

其中，r為標準常態分配隨機值。

步驟六：建立下一次疊代的原始螞蟻

將新建立的螞蟻位置代入$f(\cdot)$中，得到其損失值，再與原始螞蟻依照損失值再進行一次排列，若尚未達結束條件，則取出前k隻螞蟻回到步驟四，若已達結束條件，第一隻螞蟻即為參數訓練後的結果。

（二）遞迴最小平方估計法（RLSE）

RLSE（Stigler 1981）是由 Hsia（1977）的 LSE 演變而來，為一線性回歸方法，從已知的資料中推論出近似解。相較於 RLSE，LSE 需得到完整資料才可計算，當有一筆新資料需重新計算，RLSE 則具有彈性，當有一筆新的資料時可直接更新參數，不需重新計算，還可以權重達到時間序列的概念，適用於時常更新的資料，如(25)式所示。其中，$\boldsymbol{\theta}$為 RLSE 所需訓練之參數，也就是線性函數的參數；$\mathbf{a}$為輸入向量；$\mathbf{y}$為目標向量；$(\mathbf{a}_{\mathrm{k}}^{\mathrm{T}}, \mathbf{y}_{\mathrm{k}}^{\mathrm{T}})$代表第$k$筆資料對。由於本篇是多目標預測，因此$\boldsymbol{\Theta}$為一矩陣。$\mathbf{P}$的初始化如(26)式所示，其中，$\alpha$在本篇設定為$10^8$；$\mathbf{I}$是大小為n × n的單位矩陣；$\boldsymbol{\Theta}_0$則是大小為n × m的零矩陣（n為訓練的參數個數，m為目標個數）。

$$\begin{cases} \mathbf{P}_{k+1} = \mathbf{P}_{\mathrm{k}} - \frac{\mathbf{P}_{\mathrm{k}}\,\mathbf{a}_{\mathrm{k+1}}\,\mathbf{a}_{\mathrm{k+1}}^{\mathrm{T}}\,\mathbf{P}_{\mathrm{k}}}{1+\mathbf{a}_{\mathrm{k+1}}^{\mathrm{T}}\,\mathbf{P}_{\mathrm{k}}\,\mathbf{a}_{\mathrm{k+1}}}, \\ \boldsymbol{\Theta}_{k+1} = \boldsymbol{\Theta}_k + \mathbf{P}_{\mathrm{k+1}}\mathbf{a}_{\mathrm{k+1}}\left(\mathbf{y}_{\mathrm{k+1}}^{\mathrm{T}} - \mathbf{a}_{\mathrm{k+1}}^{\mathrm{T}}\boldsymbol{\Theta}_{\mathrm{k}}\right), \end{cases} \tag{25}$$

$$\mathbf{P}_0 = \alpha\mathbf{I} \tag{26}$$

肆、實驗結果

本篇以 ANN(4)-SCNFS(AOL)進行美國股價預測，結合一層有 4 個神經元的 ANN 及包含 AOL 的 SCNFS，機器學習演算法使用 CACO-RLSE。實驗以均方根誤差（Root-mean-square error; RMSE）評估其表現，如(27)式所示，其中y_i為第i筆目標值，$\hat{y}_i$為第i筆模型輸出值。為公平起見，實驗另外計算兩個評估指標，平均絕對百分比誤差（Mean absolute percentage error; MAPE）和平均絕對誤差（Mean absolute error; MAE），如(28)式和(29)式所示。MAPE 是以平均的誤差率絕對值進行運算，因此其值大多落於 0 到 1 之間，常以百分比的方式表示之，MAE 是以平均的誤差絕對值進行運算，越小代表表現越好。

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}} \tag{27}$$

$$MAPE = \frac{\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|}{n} \tag{28}$$

$$MAE = \frac{\sum_{i=1}^{n}|y_i - \hat{y}_i|}{n} \tag{29}$$

本篇共進行三個實驗，實驗一為單目標預測，輸入資料集與目標資料集使用同一美國指數－道瓊工業平均指數（Dow Jones Industrial Average; DJIA），並與其他模型之實驗結果比較。實驗二為多目標預測，輸入資料集與目標資料集使用相同的四種美國指數－DJIA、標準普爾 500 指數（S&P 500）、納斯達克綜合指數（NASDAQ）、紐約證交所綜合指數（New York Stock Exchange; NYSE），同時進行多目標及單目標的預測，研究多目標預測及單目標預測的表現。實驗三是多目標預測，但輸入資料集與目標資料集的標的股不相同，輸入資料集使用美國指數－DJIA、S&P 500、NASDAQ 和 NYSE，輸出資料集使用具重大影響力的美股－蘋果（Apple）、微軟（Microsoft）、Alphabet 和美國電話電報（AT&T）。

實驗的資料集皆擷取自 Yahoo! Finance，以一年的歷史資料進行模型的訓練及測試。輸入資料是擷取標的股連續 30 筆的最高價（High）、最低價（Low）、開盤價（Open）及收盤價（Close）價差，目標資料集使用第 31 筆收盤價價差，並以 80：20 法則將資料集分割成訓練階段（Training phase）和測試階段（Testing phase），以 500 次疊代訓練模型參數。CACO 的參數設定如表 4 所示，實驗結果皆是以 10 次重複的實驗中取出最佳的結果，並以此與其他文獻比較。

表 4：CACO 參數設定

Parameters	Value
Archive size (k)	5
Sample size (m)	25
Evaporation rate (ξ)	0.8
Learning rate (q)	0.5

一、實驗一

由於大多研究為單目標預測，本篇將使用 ANN(4)-SCNFS(AOL)進行單目標之收盤價預測，再與文獻進行比較。Sadaei 等（2016）提出 ARFIMA-FTS 模型，結合自回歸移動平均部分整合（Auto Regressive Fractional Integrated Moving

Average; ARFIMA）與模糊時間序列（Fuzzy Time Series; FTS），針對 DJIA 進行單目標預測。實驗的模型結構如表 5 所示，其中，n_{CP}代表模型的輸入特徵數，n_{FP}代表特徵挑選後的輸入特徵數；*K*-*Q* 代表 SCNFS 的非對稱模型架構。本實驗的實驗結果列於表 6 中。

表 5：模型結構（實驗一）

Year	n_{CP}	n_{FP}	*K*-*Q*	Training phase	Testing phase
2001	120	9	10-2	2001/2/14 - 2001/10/25	2001/10/26 - 2001/12/28
2002	120	10	10-2	2002/2/13 - 2002/10/24	2002/10/25 - 2002/12/30
2003	120	7	10-2	2003/2/13 - 2003/10/24	2003/10/27 - 2003/12/30
2004	120	12	10-3	2004/2/13 - 2004/10/26	2004/10/27 - 2004/12/30
2005	120	10	10-2	2005/2/14 - 2005/10/26	2005/10/27 - 2005/12/30
2006	120	16	10-2	2006/2/15 - 2006/10/25	2006/10/26 - 2006/12/29
2007	120	11	10-2	2007/2/15 - 2007/10/25	2007/10/26 - 2007/12/28
2008	120	9	10-2	2008/2/13 - 2008/10/24	2008/10/27 - 2008/12/30
2009	120	15	10-2	2009/2/13 - 2009/10/26	2009/10/27 - 2009/12/30

表 6：以 RMSE 表現 DJIA 實驗結果（實驗一）

Model	2001	2002	2003	2004	2005	2006	2007	2008	2009	Avg
Chen 1996 (Sadaei et al. 2016)	104.25	119.33	68.06	73.64	60.71	64.32	171.62	310.52	92.75	118.36
ARIMA (Sadaei et al. 2016)	97.43	121.23	71.23	70.23	58.32	64.43	169.33	306.11	94.39	116.97
Yu 2005-average (Sadaei et al. 2016)	100.54	119.33	65.35	71.50	57.00	63.18	168.76	310.09	91.32	116.34
ETS (Sadaei et al. 2016)	96.80	119.43	68.01	72.33	54.70	63.72	165.04	303.39	95.60	115.45
Yu 2005-distribution (Sadaei et al. 2016)	98.69	119.18	63.66	70.88	54.69	60.87	167.69	308.40	89.78	114.87
Huarng and Yu 2006 (Sadaei et al. 2016)	97.86	116.85	61.32	70.22	52.36	58.37	167.69	306.07	87.45	113.13
Chen and Chen 2011 (Sadaei et al. 2016)	96.39	114.08	61.38	66.75	52.18	55.83	165.48	304.35	85.06	111.28
Chen et al. 2008 (Sadaei et al. 2016)	95.86	114.35	60.32	66.22	50.86	56.62	164.69	303.82	87.45	111.13
ARFIMA (Sadaei et al. 2016)	95.18	115.13	59.43	58.47	50.78	51.23	163.77	315.17	89.23	110.93
Javedani et al. 2014 (Sadaei et al. 2016)	94.80	111.70	59.00	64.10	49.80	55.30	163.10	301.70	84.80	109.37
Sadaei 2016 (Sadaei et al. 2016)	86.67	101.62	**45.04**	55.80	**34.91**	45.14	152.88	293.96	**74.98**	99.00
ANN-SCNFS (Proposed)	**85.34**	**93.89**	55.12	**49.58**	51.77	**40.64**	**123.06**	**255.13**	88.45	**93.67**

小結：

實驗結果如表 6 所示，以 RMSE 作為表現評估指標。本實驗以 ASUS-X550V 進行實驗，以 MATLAB 2018 撰寫實驗程式，硬體的規格如下：處理器為 Intel® Core™ i5-6300HQ，RAM 為 4GB。在模型 500 疊代下，模型機器學習所花費的時間約 170 秒上下。已訓練好之模型預測股票一次所需之計算時間大約為 0.018745 秒。實驗結果顯示，本篇模型的結果大多優於其他文獻，於 2001 年到 2009 年之平均 RMSE 達 93.67，同樣優於其他模型。另外，2007 年與 2008 年的 RMSE 表現比起其他年份高出許多，其緣由為當時的金融海嘯，使價格起伏過大，也因此使模型預測能力降低，同時可說明價格漲跌幅的大小會影響實驗結果的表現。

二、實驗二

為瞭解多目標及單目標的預測能力，本實驗以相同資料集，分別進行單目標及多目標的預測。輸入資料集與目標資料集使用 2018 年的美國指數──DJIA、NASDAQ、S&P 500 和 NYSE。各實驗的模型結構如表 7 所示。其中，Multi-target 為同時使用四個標的股的多目標預測。單目標（Single-target）及多目標（Multi-target）預測之實驗結果比較表如表 8 所示，並分成訓練階段及測試階段進行表現評估，為以示公平使用三種表現評估指標進行結果比較。

表 7：模型結構（實驗二）

Experiment	n_{CP}	n_{FP}	*K-Q*	Training phase	Testing phase
DJIA (Single)	120	14	10-3	2018/2/14 | 2018/10/24 (176 days)	2018/10/25 | 2018/12/28 (44 days)
S&P 500 (Single)	120	13	10-3		
NASDAQ (Single)	120	15	10-2		
NYSE (Single)	120	19	10-2		
Multi-target	480	93	10-10		

表 8：單目標及多目標預測之實驗結果比較（實驗二）

Stock	Method	Training phase			Testing phase		
		RMSE	MAPE	MAE	RMSE	MAPE	MAE
DJIA	Single-target	**187.35**	**0.0057**	**142.25**	356.68	0.0120	293.84
	Multi-target	195.82	0.0062	153.61	**345.34**	**0.0115**	**279.52**
S&P 500	Single-target	**19.72**	**0.0053**	**14.50**	39.19	0.0119	31.05
	Multi-target	20.53	0.0056	15.47	**36.43**	**0.0112**	**29.30**

Stock	Method	Training phase			Testing phase		
		RMSE	MAPE	MAE	RMSE	MAPE	MAE
NASDAQ	Single-target	**69.26**	**0.0070**	**52.53**	127.15	0.0146	101.75
	Multi-target	73.70	0.0076	56.55	**112.04**	**0.0132**	**92.53**
NYSE	Single-target	**80.93**	**0.0050**	**63.45**	142.28	0.0098	116.41
	Multi-target	89.65	0.0058	73.03	**132.49**	**0.0088**	**104.76**

從表 8 中可發現，多目標預測在測試階段的表現皆優於單個股票下之預測結果。除此之外，多目標預測在訓練階段之實驗結果卻是比單目標差，同時可說明模型實際以非訓練期間之資料代入時，其結果並不會與訓練期間之表現差距過大，相較於單目標預測，多目標預測的結果顯得較為穩定。

除了實驗結果的數據外，在圖 4─圖 7 中列出各個美國指數在單目標預測下之學習曲線（Learning curve），並於圖 8 中列出多目標預測的學習曲線。學習曲線是用於表示模型在訓練過程中 RMSE 的變化，由於本篇是以正規化的資料進行實驗，因此其數值也是正規化的結果。因此為公平起見，會將結果返回原始數值，再進行表現評估。另外在圖 9─圖 14 中列出單目標及多目標在不同標的股下的預測結果圖。

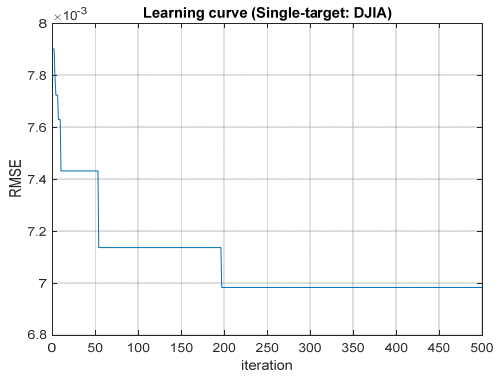

圖 4：DJIA 單目標學習曲線（實驗二）

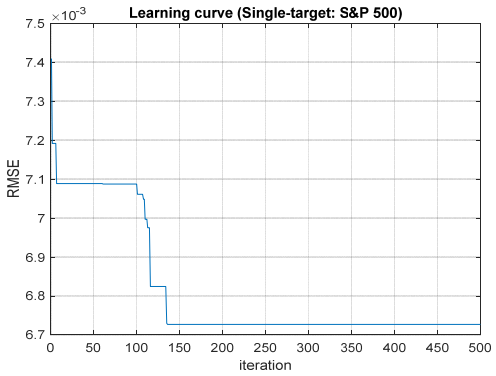

圖 5：S&P 500 單目標學習曲線（實驗二）

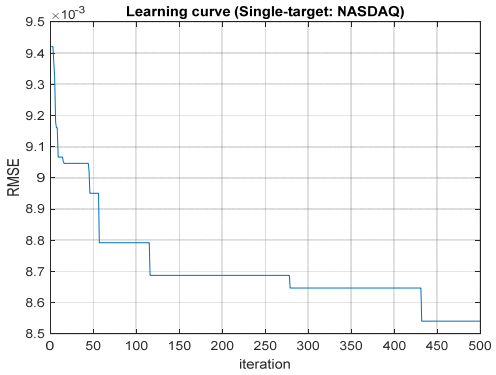

圖 6：NASDAQ 單目標學習曲線（實驗二）

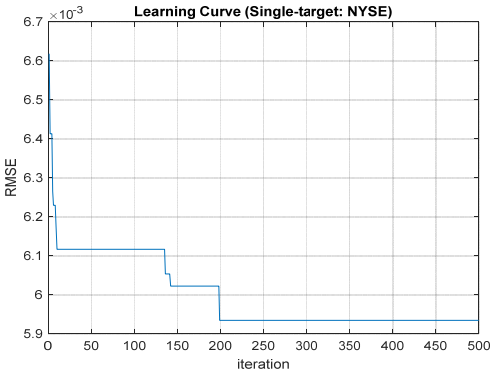

圖 7：NYSE 單目標學習曲線（實驗二）

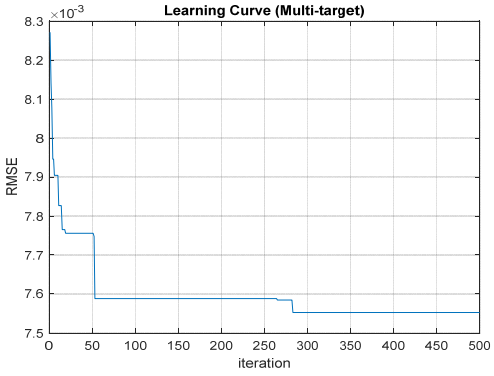

圖 8：多目標學習曲線（實驗二）

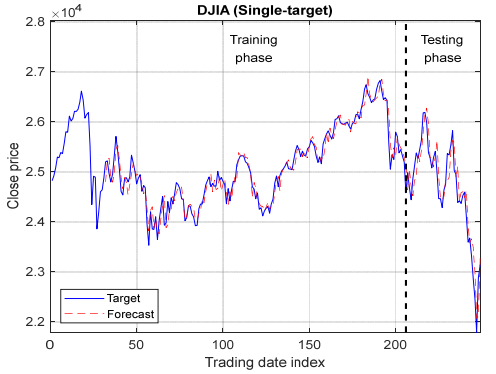

圖 9：DJIA 單目標預測結果（實驗二）

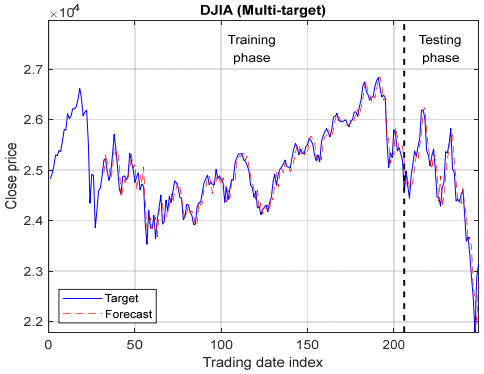

圖 10： DJIA 多目標預測結果（實驗二）

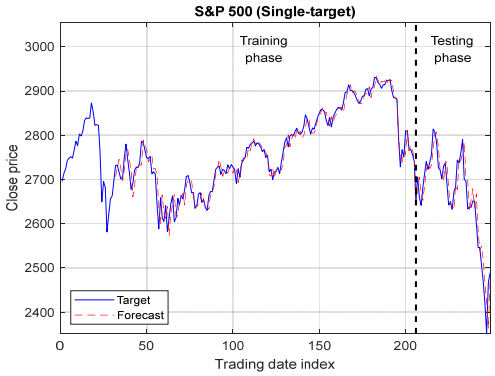

圖 11： S&P 500 單目標預測結果（實驗二）

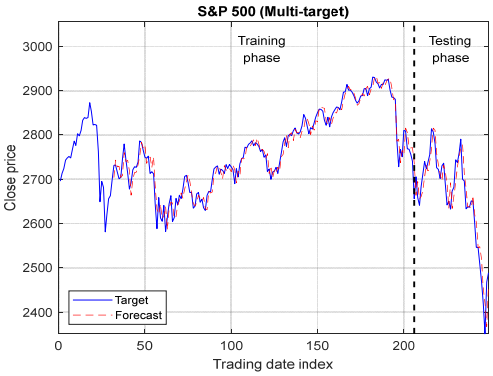

圖 12： S&P 500 多目標預測結果（實驗二）

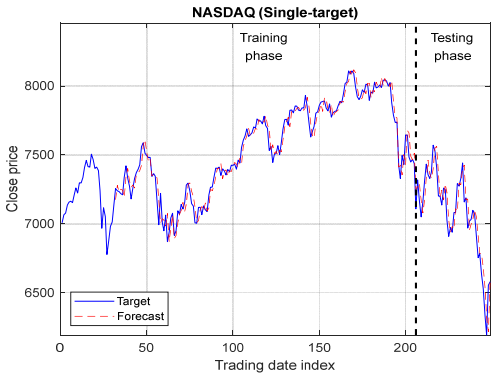

圖 13： NASDAQ 單目標預測結果（實驗二）

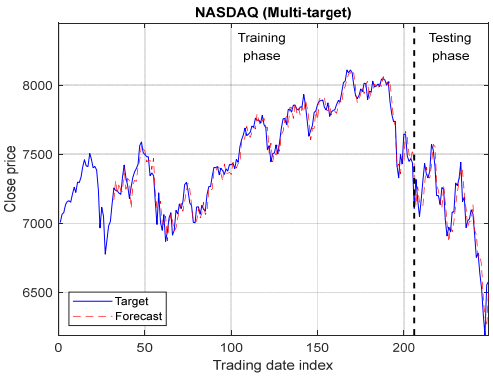

圖 14： NASDAQ 多目標預測結果（實驗二）

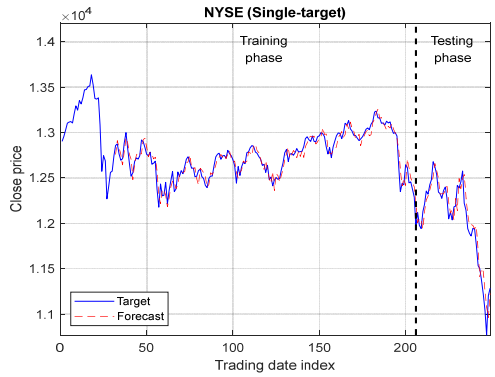

圖 15： NYSE 單目標預測結果（實驗二）

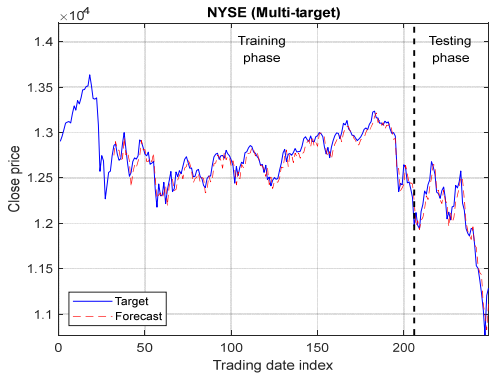

圖 16： NYSE 多目標預測結果（實驗二）

小結：

從實驗結果中可發現，不管是哪一標的股，其訓練階段時單目標預測的結果皆會優於多目標預測，然而，在測試階段時反而是多目標預測的結果較優。由此可說明，多目標預測因為擁有較多的資料，可以提升預測時的穩定度，減少訓練階段及測試階段之間的表現差距，因此實際以其他時間段的資料集放入已訓練好的模型中時，其表現結果並不會與訓練時的表現差距過大。除此之外，多目標預測在使用多個資料的同時，又進行了多個目標的預測，但單目標預測卻需對逐個標的股進行運算，因此多目標預測在時間效率上會較優。

三、實驗三

本實驗與實驗二相同為多目標預測，但又實驗二稍顯不同，此實驗的模型之輸入資料集與目標資料集是使用不同的股票標的。輸入資料集包含四種美國指數－DJIA、S&P 500、NASDAQ 和 NYSE；目標資料集則包含四種具重要地位之美股－Apple、Microsoft、Alphabet 和 AT&T，實驗模型結構如表 9 所示。

此實驗之主要的目的為瞭解輸入資料集與目標資料集在使用上的彈性，以非本身標的股之歷史資料作為輸入的前提下，研究是否還可有效地進行標的股的預測，進而找出指數與美股之間的關係。同時，為瞭解是否有效預測，與多篇單目標預測的文獻進行比較，以 2018 年訓練階段之資料集訓練模型後，代入 2018 年及文獻定義之時間段的資料集，並評估其實驗結果。

表 9：模型結構（實驗三）

n_{CF}	n_{FP}	K	Q	Training phase	Testing phase
480	58	10	11	2018/2/14 - 2018/10/24	2018/10/25 - 2018/12/28

本實驗的學習曲線如圖 17 所示，預測結果如圖 18-圖 21 所示。為瞭解實驗結果之優劣，分別對不同標的股與各文獻進行比較。各個文獻所使用的資料集區間如表 10 所示，可發現各文獻所使用的資料區間皆不相同。為進行實驗結果的比較，本實驗會以 2018 年所訓練之模型代入不同時間段的資料集，並評估其表現。實驗結果如表 11 所示，除了 Microsoft 外，本篇模型的結果皆是優於其他文獻的，由此也可說明在使用不同資料集的前提下，本研究的模型可有效預測。

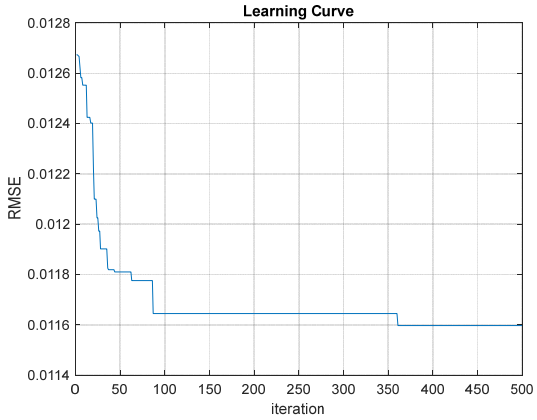

圖 17：學習曲線

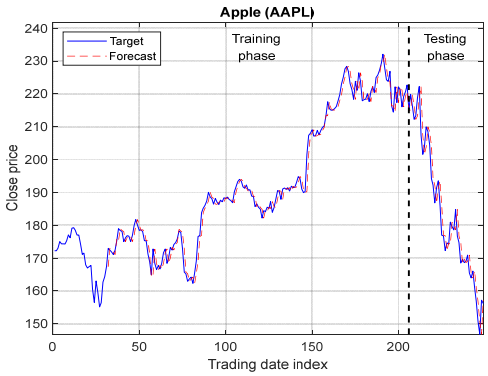

圖 18： Apple 多目標預測結果（實驗三）

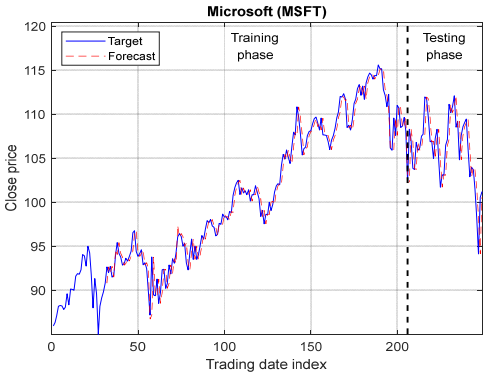

圖 19： Microsoft 多目標預測結果（實驗三）

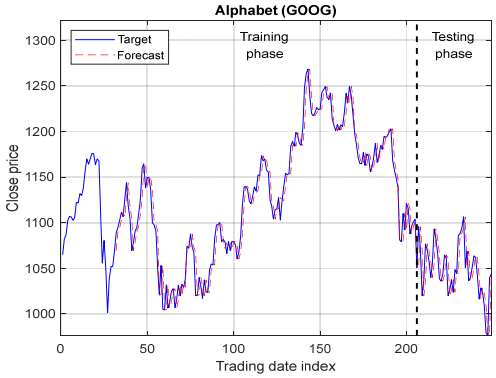

圖 20： Alphabet 多目標預測結果（實驗三）

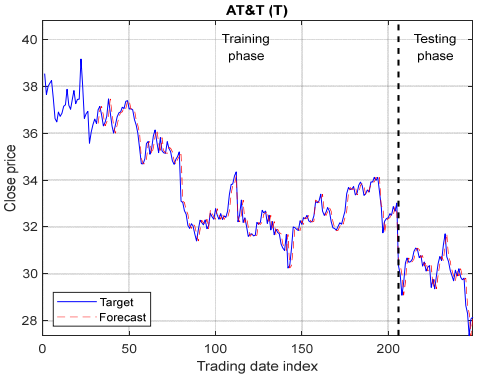

圖 21： AT&T 多目標預測結果（實驗三）

表 10：比較文獻的資料集時間段（實驗三）

Stock	Time period	Time interval
Apple (Khuat et al. 2016)	2009-2013	4 years
Alphabet (Khuat et al. 2016)	2014/3-2015/7	16 months
Apple (Darwish & Hilal 2017)	2004/9/13-2005/1/21	about 4 months
Microsoft (Darwish & Hilal 2017)	2011/1/3-2011/12/30	1 years
Microsoft (Siddique et al. 2017)	2006/5-2016/5	10 years
AT&T (Hegazy, Soliman and Salam 2014)	2009/1-2012/1	3 years

表 11：實驗結果（實驗三）

Stock (symbol)	Method	Time period	RMSE	MAPE	MAE
Apple (AAPL)	Wavelet (Khuat et al. 2016)	2009-2013	25.8086	0.0452	22.2795
	ANN (Khuat et al. 2016)	2009-2013	36.6934	0.0692	33.6165
	DABC-ANN (Khuat et al. 2016)	2009-2013	28.5191	0.0522	25.4332
	PF & C-means (Darwish & Hilal 2017)	2004/9/13-2015/1/21	-	1.2030	-
	ANN-SCNFS (Proposed)	2009-2013	**1.3816**	**0.0146**	**0.8499**
	ANN-SCNFS (Proposed)	2004/9/13-2015/1/21	**0.7053**	**0.0400**	**0.1492**
	ANN-SCNFS (Proposed)	2018/10/25-2018/12/28	5.1157	0.2323	4.1965
Microsoft (MSFT)	ANN (Ticknor 2013)	2004/9/13-2005/1/21	-	**0.0106**	-
	PF & C-means (Darwish & Hilal 2017)	2011/1/3-2011/12/30	**0.2528**	-	-
	HEA (Siddique et al. 2017)	2006/5-2016/5	-	0.0373	-
	VSPSEG (Siddique et al. 2017)	2006/5-2016/5	-	0.0351	-
	CMS-PSO (Siddique et al. 2017)	2006/5-2016/5	-	0.0320	-
	ANN-BP (Siddique et al. 2017)	2006/5-2016/5	-	0.0308	-
	ANN-PSO (Siddique et al. 2017)	2006/5-2016/5	-	**0.0100**	-
	ANN-SCNFS (Proposed)	2004/9/13-2005/1/21	1.0079	0.0111	0.3078
	ANN-SCNFS (Proposed)	2011/1/3-2011/12/30	1.1975	0.0184	0.4740
	ANN-SCNFS (Proposed)	2006/5-2016/5	1.0426	0.0155	0.4805
	ANN-SCNFS (Proposed)	2018/10/25-2018/12/28	2.3440	0.0178	1.8603
Alphabet (GOOG)	Wavelet (Khuat et al. 2016)	2014/3-2015/7	6.0585	0.0083	4.4775
	ANN (Khuat et al. 2016)	2014/3-2015/7	6.5841	0.0094	5.1956
	DABC-ANN (Khuat et al. 2016)	2014/3-2015/7	5.8324	**0.0078**	**4.1769**
	ANN-SCNFS (Proposed)	2014/3-2015/7	**5.7606**	0.0083	4.4871
	ANN-SCNFS (Proposed)	2018/10/25-2018/12/28	23.3016	0.0175	18.3170
AT&T (T)	PSO-LS-SVM (Hegazy et al. 2014)	2009/1-2012/1	0.5395	-	-
	LS-SVM (Hegazy et al. 2014)	2009/1-2012/1	0.6836	-	-
	NN-BP (Hegazy et al. 2014)	2009/1-2012/1	0.6368	-	-

	ANN-SCNFS (Proposed)	2009/1-2012/1	**0.3661**	**0.0100**	**0.2691**
	ANN-SCNFS (Proposed)	2018/10/25-2018/12/28	0.4555	0.0121	0.3583

小結：

實驗結果顯示，在所有目標中 Microsoft 的結果比文獻結果較差，但也相差不遠，因此在模型可同時預測多目標的特性下，目標的預測可大部分優於單目標預測之結果。同時由此實驗可說明，輸入資料集及目標資料集的使用，在本篇模型下是很具有彈性的，縱使本篇只使用 2018 年約 8 個月的資料集進行訓練，在結果上也是大多優於其他的文獻。

伍、結果與討論

為解決多目標預測問題，本篇以 SCNFS 作為模型主架構，藉由 SCFS 產生多組複數模糊集的特性，達到支援多目標預測的目的，並以箭靶層將原始一對一規則關係打破，使模型規則轉變為多對多的因果關係，分別以輸入資料集及目標資料集建立模型的前鑑部及後鑑部，並將輸入資料以模型推論出目標值。然而，支援多目標的同時，又希望降低模型的訓練負荷，因此以多目標特徵挑選作為資料前處理之方法，從輸入資料中挑選出對整體目標具有貢獻的資料作為輸入，可從龐大的資料集中挑選對整體目標具有貢獻的資料。

在多目標特徵挑選與 SCNFS 的結合中，多目標特徵挑選後的資料對於 SCNFS 還是過於龐大，對 SCNFS 會產生極大的負荷，因此在特徵挑選後，加上一層的 ANN 架構，不但能夠使得 SCNFS 所接收的輸入量減少，又可以將多目標特徵挑選後的輸入進行權重式的調整，由此便可接收更為龐大的資料量。

除了模型架構的設計外，本篇使用混合式機器學習演算法，CACO-RLSE，訓練此模型的參數，由於模型參數的特性不同，使用不同的演算法，針對那些無法以導數找出與目標間關係的參數，以無導數的演算法，CACO，進行參數的訓練，反之則使用 RLSE 進行訓練，根據其線性回歸運算，得以快速地找到近似解，由於 CACO 在訓練參數越多的情況下，其訓練效果會降低，因此加上 RLSE 的演算法也可提升 CACO 訓練時的表現。

以單目標進行時間序列預測的實驗一中，本篇的實驗結果大多是優於文獻中的其他模型，在平均下之結果也是較為優良。除此之外，在實驗中的 2007 年及 2008 年之 RMSE 比其他年度的大，結合當時所發生的金融海嘯，可發現股票在波動過於劇烈的情況下，不管是使用哪一種模型，皆會有較差的結果，在預測上也會較為不準確，這是我們應該要去克服的問題，若是可以克服此問題，就可降低股災發生所造成的巨大損失。

實驗二是進行多目標預測，並與實驗一之單目標預測架構進行實驗比較。實驗結果表明，多目標之預測結果優於單目標的，在實驗一中本篇的單目標預測結果又優於其他的文獻。表示多目標預測不只可以有效進行預測，又可以有較好的表現。另外，表 12 和表 13 分別是實驗二中單目標預測及多目標預測重複 10 次實驗的結果，前述的實驗結果是取於其中的最佳結果，從中可發現除了多目標預測在測試階段優於單目標，其標準差也是多目標預測較單目標的小，進一步證明多目標預測在實驗上的穩定度。

表 12：2018 年重複 10 次實驗結果之 RMSE 結果（實驗二，單目標預測）

No.	DJIA		S&P 500		NASDAQ		NYSE	
	Train	Test	Train	Test	Train	Test	Train	Test
1	186.52	380.32	19.35	41.00	69.07	153.41	79.52	148.60
2	190.22	368.09	19.94	43.70	69.75	150.44	82.66	150.33
3	188.47	401.17	19.40	41.77	68.89	173.35	83.02	144.96
4	188.08	381.05	19.72	39.19	71.18	156.27	80.86	152.64
5	191.71	397.23	20.66	48.33	69.53	160.47	80.93	142.28
6	195.54	384.18	19.87	44.08	68.79	157.01	77.76	156.33
7	193.05	401.32	19.83	47.93	68.21	137.79	82.53	150.17
8	192.05	387.17	19.98	40.40	68.09	135.06	80.23	147.46
9	187.51	383.58	20.25	46.47	69.26	127.15	82.77	157.15
10	187.35	356.68	20.28	46.70	68.47	156.62	82.77	157.51
Avg.	190.05	384.08	19.93	43.96	69.12	150.76	81.30	150.74
Std.	2.95	14.15	0.40	3.30	0.90	13.69	1.76	5.19

表 13：2018 年重複 10 次實驗結果之 RMSE 結果（實驗二，多目標預測）

No.	DJIA		S&P 500		NASDAQ		NYSE	
	Train	Test	Train	Test	Train	Test	Train	Test
1	208.07	383.09	21.48	40.43	73.86	132.50	86.63	147.86
2	206.50	368.08	21.64	38.86	72.65	128.22	87.50	149.64
3	197.98	353.48	20.74	36.92	73.44	119.33	86.75	139.93
4	195.82	345.34	20.53	36.43	73.70	112.04	89.65	132.49
5	201.49	356.17	21.04	36.63	73.19	120.52	84.81	141.46
6	196.97	346.64	20.72	36.51	73.42	122.83	83.37	138.54
7	204.82	371.85	21.02	39.52	72.08	136.01	85.21	151.81
8	207.17	377.92	21.29	39.95	73.47	129.79	85.91	152.73

9	201.25	371.26	21.19	39.60	74.85	132.34	85.69	151.55
10	196.52	362.53	20.52	38.07	74.29	125.77	83.01	143.98
Avg.	201.66	363.64	21.02	38.29	73.49	125.93	85.85	145.00
Std.	4.73	12.96	0.39	1.57	0.78	7.31	1.95	6.80

實驗三是以不同標的股作為模型之輸入及目標資料集，並由此進行多目標預測。實驗結果表明，本篇在使用不同標的股的前提下，其結果大多還是優於其他文獻的結果。不只可說明本篇模型在輸入資料集及目標資料集在使用上的彈性，同時可證明本篇模型上預測上有不錯的效能，這也顯現出多目標特徵挑選的優點，可從龐大的資料集中選擇對目標有益的資料，因此在未來的實驗中也可嘗試以更多的資料進行訓練，可能會有更好的結果。

陸、結論與建議

為了找出股價變動的規律及股價間的影響關係，本篇結合了 ANN 和 SCNFS。其中，SCNFS 加入了 SCFS 及箭靶層，使模型在可支援多目標運算的同時，又可降低模型後鑑部的運算負荷。在資料前處理的部分，結合了多目標特徵挑選，使模型可接收更為龐大的資料集，從中找出對目標具貢獻的輸入資料。根據實驗結果，顯示本篇的多目標預測結果有良好的表現，且在效率上會優於逐個目標預測的模型。本篇的主要貢獻如下：

1. 在模型架構方面，以多目標特徵挑選篩選輸入資料，並加入 ANN，將特徵挑選後的大量資料，透過權重調整，從中萃取少數代表性的資料，降低 SCNFS 因輸入個數增加產生的訓練時間。
2. 在 SCNFS 中以 SCFS 產生多個歸屬程度的方式，支援多目標的運算，並加入箭靶層，使模型不再是一對一的規則型態，而是形成多對多的因果關係，有效降低結果層的訓練時間。
3. 在參數訓練上，使用了混合式機器學習演算法，CACO-RLSE，透過分而治之的方式，針對不同階層使用不同演算法進行訓練，使模型能有效訓練。

誌謝

這項研究工作得到了科技部經費支持 MOST 105-2221-E-008-091 和 MOST 104-2221-E-008-116, TAIWAN，表示感謝之意。另外，在本文的審稿期間，兩位匿名的審查委員提供作者許多寶貴的建議並給予指正，在此對審查委員表達感謝

之意。

參考文獻

Bowman, A.W. and Azzalini, A. (1997), *Applied Smoothing Techniques for Data Analysis: The Kernel Approach with S-Plus Illustrations* (Vol. 18), OUP Oxford.

Chiu, S.L. (1994), 'Fuzzy model identification based on cluster estimation', *Journal of Intelligent & Fuzzy Systems*, Vol. 2, No. 3, pp. 267-278.

Darwish, A.H. and Hilal, A.A. (2017, December), 'Prediction of stock market using C-means clustering and particle filter', *International Journal of Computer Applications*, Vol. 179, No. 2, pp. 12-19.

Dorigo, M., Maniezzo, V. and Colorni, A. (1996), 'Ant system: Optimization by a colony of cooperating agents', *IEEE Transactions on Systems, man, and cybernetics, Part B: Cybernetics*, Vol. 26, No. 1, pp. 29-41.

Faustryjak, D., Jackowska-Strumiłło, L. and Majchrowicz, M. (2018, May), 'Forward forecast of stock prices using LSTM neural networks with statistical analysis of published messages', in *2018 International Interdisciplinary PhD Workshop (IIPhDW)*(pp. 288-292). IEEE.

Hadavandi, E., Shavandi, H. and Ghanbari, A. (2010), 'Integration of genetic fuzzy systems and artificial neural networks for stock price forecasting', *Knowledge-Based Systems*, Vol. 23, No. 8, pp. 800-808.

Hassan, M.R. and Nath, B. (2005, September), 'Stock market forecasting using hidden Markov model: A new approach', in *5th International Conference on Intelligent Systems Design and Applications (ISDA'05)*(pp. 192-196). IEEE.

Hegazy, O., Soliman, O.S. and Salam, M.A. (2014), 'A machine learning model for stock market prediction', *International Journal of Computer Science and Telecommunications*, Vol. 4, No. 12.

Holland, J.H. (1992), *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*, MIT press, Cambridge, MA, USA.

Hsia, T.C. (1977), *System Identification: Least-Squares Methods* (1st ed.), Lexington Books.

Jang, J.S. (1993), 'ANFIS: Adaptive-network-based fuzzy inference system', *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 23, No. 3, pp. 665-685.

Karaboga, D. (2005), 'An idea based on honey bee swarm for numerical optimization', *Technical report-tr06, Erciyes University, Engineering Faculty, Computer Engineering Department*, Vol. 200.

Kennedy, J. and Eberhart, R. (1995), 'Particle swarm optimization', in *Proceedings of the IEEE International Conference on Neural Networks*, Vol. 4, pp. 1942-1948.

Khuat, T.T., Le, Q.C., Nguyen, B.L. and Le, M.H. (2016), 'Forecasting stock price using wavelet neural network optimized by directed artificial bee colony algorithm', *Journal of Telecommunications and Information Technology*, No. 2, pp. 43-52.

Kwak, N. and Choi, C.H. (2002), 'Input feature selection by mutual information based on Parzen window', *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 24, No.12, pp. 1667-1671.

Li, C. and Chiang, T.W. (2010, March), 'Complex neuro-fuzzy self-learning approach to function approximation', in *Asian Conference on Intelligent Information and Database Systems* (pp. 289-299), Springer, Berlin, Heidelberg.

Li, C., Yang, S. and Nguyen, T.T. (2012), 'A self-learning particle swarm optimizer for global optimization problems', *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, Vol. 42, No. 3, pp. 627-646.

McCulloch, W.S. and Pitts, W. (1943), 'A logical calculus of the ideas immanent in nervous activity', *The Bulletin of Mathematical Biophysics*, Vol. 5, No. 4, pp. 115-133.

Mirjalili, S., and Lewis, A. (2016), 'The whale optimization algorithm', *Advances in Engineering Software*, Vol. 95, pp. 51-67.

Ramot, D., Milo, R., Friedman, M. and Kandel, A. (2002), 'Complex fuzzy sets', *IEEE Transactions on Fuzzy Systems*, Vol. 10, No. 2, pp. 171-186.

Rosenblatt, F. (1958), 'The perceptron: A probabilistic model for information storage and organization in the brain', *Psychological Review*, Vol. 65, No. 6, pp. 386-408.

Sadaei, H.J., Enayatifar, R., Guimarães, F.G., Mahmud, M. and Alzamil, Z.A. (2016), 'Combining ARFIMA models and fuzzy time series for the forecast of long memory time series', *Neurocomputing*, Vol. 175, Part A, pp. 782-796.

Selvin, S., Vinayakumar, R., Gopalakrishnan, E.A., Menon, V.K. and Soman, K.P. (2017, September), 'Stock price prediction using LSTM, RNN and CNN-sliding window model', in *2017 IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI)*(pp. 1643-1647), IEEE.

Shannon, C.E. (1948), 'A mathematical theory of communication', *Bell System Technical Journal*, Vol. 27, No. 3, pp. 379-423.

Siddique, M., DebdulalPanda, Das, S. and Mohapatra, S.K. (2017), 'A hybrid forecasting model for stock value prediction using soft computing technique', *International Journal of Pure and Applied Mathematics*, Vol. 117, No. 19, pp. 357-363.

Socha, K. and Dorigo, M. (2008), 'Ant colony optimization for continuous domains', *European Journal of Operational Research*, Vol. 185, No. 3, pp. 1155-1173.

Stigler, S.M. (1981), 'Gauss and the invention of least squares', *The Annals of Statistics*, Vol. 9, No. 3, pp. 465-474.

Takagi, T. and Sugeno, M. (1985), 'Fuzzy identification of systems and its applications to modeling and control', in *Readings in Fuzzy Sets for Intelligent Systems* (pp. 116-132), Morgan Kaufmann.

Ticknor, J.L. (2013), 'A Bayesian regularized artificial neural network for stock market forecasting', *Expert Systems with Applications*, Vol. 40, No. 14, pp. 5501-5506.

Tu, C.H. and Li, C. (2017, April), 'A novel entropy-based approach to feature selection', in *Asian Conference on Intelligent Information and Database Systems* (pp. 445-454), Springer, Cham.

Tu, C.H. and Li, C. (2019), 'Multitarget prediction-A new approach using sphere complex fuzzy sets', *Engineering Applications of Artificial Intelligence*, Vol. 79, pp. 45-57.

Wang, J.H., and Leu, J.Y. (1996, June), 'Stock market trend prediction using ARIMA-based neural networks', in *Proceedings of International Conference on Neural Networks (ICNN'96)*(Vol. 4, pp. 2160-2165). IEEE.

Werbos, P. (1974), 'Beyond regression: New tools for prediction and analysis in the behavioral sciences', *Ph. D. dissertation*, Harvard University.

Whitney, A.W. (1971), 'A direct method of nonparametric measurement selection', *IEEE Transactions on Computers*, Vol. C-20, No. 9, pp. 1100-1103.

Widrow, B. and Hoff, M.E. (1960), 'Adaptive switching circuits' (No. TR-1553-1), Stanford Univ Ca Stanford Electronics Labs.

Winter, C.R. and Widrow, B. (1988, July), 'Madaline Rule II: A training algorithm for neural networks', in *Secondf Annual International Conference on Neural Networks* (Vol. 1, pp. 401-408).

Zadeh, L.A. (1965), 'Fuzzy sets', *Information and Control*, Vol. 8, No. 3, pp. 338-353.