許敦盛、李俊賢（2019），『卷積神經模糊方法於多目標時間序列預測研究』，
中華民國資訊管理學報，第二十六卷，第四期，頁 483-512。

卷積神經模糊方法於多目標時間序列預測研究

許敦盛
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

本研究針對時間序列提出多目標預測模型，結合卷積神經網路（Convolutional neural networks; CNN）與球型複數模糊神經系統（Sphere complex neural fuzzy system; SCNFS）。球型複數模糊集（Sphere complex fuzzy sets; SCFSs）可產生複數型態的歸屬程度，使 SCNFS 能根據目標數產生多個輸出值。在訓練資料進入模型前，使用多目標特徵選取，從中挑選有影響力之特徵。CNN 置於 SCNFS 前，以容納更多有影響力之特徵，並識別特徵中趨勢的變化，萃取出有用資訊。參數訓練方面，使用高斯分布型鯨群最佳化演算法（Gaussian distribution based whale optimization algorithm; GD-WOA）及遞迴最小平方估計法（Recursive least squares estimator; RLSE）之複合式機器學習演算法，GD-WOA 針對 CNN 權重參數及球型複數模糊集參數進行訓練；RLSE 針對模糊神經系統之後鑑部參數進行訓練。為驗證本研究所提之方法，三個實驗以股價指數數據作為資料集，進行不同目標數的時間序列預測。實驗結果與過往文獻相比，有較佳的預測結果，顯示本研究所提之方法，在時間序列預測上有良好效能。

關鍵詞：多目標預測、球型複數模糊集、卷積神經網路、模糊推論系統、複合式機器學習演算法

* 本文通訊作者。電子郵件信箱：jamesli@mgt.ncu.edu.tw
2019/06/21 投稿；2019/07/27 修訂；2019/08/23 接受

Hsu, D.S. and Li, C. (2019), 'CNN-based neural fuzzy approach to multi-target prediction using sphere complex fuzzy sets', *Journal of Information Management*, Vol. 26, No. 4, pp. 483-512.

CNN-Based Neural Fuzzy Approach to Multi-Target Prediction Using Sphere Complex Fuzzy Sets

Dun-Sheng Hsu
Department of Information Management, National Central University

Chunshien Li*
Department of Information Management, National Central University

Abstract

Purpose－Time series prediction is a challenging issue. Most prediction methods in literature implement prediction for a single target at a time only. In this paper, our purpose is to design a novel approach for multi-target simultaneous prediction.

Design/methodology/approach－In this paper, the proposed approach integrate a convolutional neural networks (CNN) and a novel neural fuzzy system using sphere complex fuzzy sets (SCFSs). The proposed predictive model, denoted as CNN-SCNFS, is presented, where SCFSs can produce complex-valued membership degrees, providing more membership information than conventional fuzzy set can. The SCFS characteristics can enable the proposed model to bring about multi-output property for applications with multiple targets. A novel feature selection method is used for selecting useful cross-target data for the modeling of CNN-SCNFS. In the proposed CNN-SCNFS approach, a CNN is placed in front of the SCNFS, and the utility of CNN is to accommodate more useful features and to capture sudden changes of input data to the model. For machine learning, a hybrid method using the divide-and-conquer principle, called the GD-WOA-RLSE algorithm, is applied for parameter learning of the model. The Gaussian distribution based whale optimization algorithm (GD-WOA) hereby presented adapts the parameters of CNN and of SCFSs in the model while the well-known recursive least squares estimator (RLSE) updates the parameters of the Takagi-Sugeno consequence layer of the CNN-SCFNS.

* Corresponding author. Email: jamesli@mgt.ncu.edu.tw
2019/06/21 received; 2019/07/27 revised; 2019/08/23 accepted

Findings ─ Several real-world data sets of stock markets are used to test the proposed approach performing multi-target prediction. The results indicate that the proposed method has shown prominent performance through comparison with other methods.

Research limitations/implications ─ In this study, the setting of proposed CNN architecture is determined by trial and error which may influent the performance of prediction. In the future work, we are intended to developed the model based on the input data to achieve self-organize property.

Practical implications ─ Currently, data is growing faster than ever before. The data analysis become more important. If there are multiple targets we want to predict, we can use the proposed model to predict these targets simultaneously instead of using the model with single output property.

Originality/value ─ In this paper, a novel approach for multi-target prediction is proposed which can well perform prediction for multiple targets in one model simultaneously.

Keywords: multi-target prediction, sphere complex fuzzy sets, convolution neural networks, fuzzy inference system, hybrid learning

壹、緒論

在數據遽增的時代，資料分析方法成為重要的研究議題，在相關研究與應用中，時間序列分析是主要的方法之一。時間序列分析方法是從一組具有時間順序的資料中，以數理或統計模型進行分析，找出隱藏於資料背後的模式或關聯性。對於時間序列問題，以財經預測最為困難，源於眾多因素影響著趨勢的變動，如經濟相關指標，國內外政治形勢及投資者的預期心理等。傳統的時間序列分析方法有自回歸條件異變異數模型（Autoregressive conditional heteroscedasticity; ARCH）（Engle 1982）、廣義 ARCH 模型（Generalized ARCH; GARCH）（Bollerslev 1986）、自回歸滑動平均模型（Autoregressive moving average; ARMA）和自回歸積分移動平均模型（Autoregressive integrated moving average; ARIMA）（Box & Jenkins 1990）等。隨著計算能力的提升及資料存儲的成本下降，計算智能（Computational intelligence）開始成為新型態的分析方法並應用於相關研究上，如人工神經網路（Artificial neural networks; ANN）（Guresen et al. 2011; Kim & Han 2000; Kimoto et al. 1990）、自適應類神經模糊推論系統（Adaptive neuro-fuzzy inference system; ANFIS）（Atsalakis & Valavanis 2009; Bagheri et al. 2014; Boyacioglu & Avci 2010）、混合式模型（Wang et al. 2012; Wei 2016; Zhang 2003）與深度學習（Deep learning）方法（Fischer & Krauss 2018; Rout et al. 2017; Selvin et al. 2017）等。其中，以模糊規則為基礎的 ANFIS 模型有著良好預測表現，是多數時間序列預測研究使用的方法。

本研究所提之模型，是以模糊推論系統為基礎進行設計。模糊推論系統結合兩個不同理論：「模糊理論」及「類神經網路」。透過模糊理論，輸入變數間的歸屬關係能以數值化的方式描述。在類神經網路中，每個節點皆為一計算單元（Process unit）。透過訓練方法，可調整節點參數，使模型輸出趨近於目標值，藉此達成自適應性。然而，對於模糊推論系統而言，越多的模型輸入會導致後鑑部的參數，在訓練時間上的增加。因此，為接收更多有影響力之資訊作為模型輸入，且不增加模型輸入的數量下，本研究將卷積神經網絡（Convolutional neural networks; CNN）置於模糊推論系統前。CNN 運用多層的特徵檢測（Feature detectors），從模型輸入中萃取資訊。其架構包含一個輸入、輸出層及多個隱藏層，隱藏層通常由卷積層（Convolutional layer）、池化層（Pooling layer）和全連接層（Fully-connected layer）組成。CNN 常用於字體辨識、影像辨識和自然語言分析等。Selvin 等（2017）指出 CNN 架構能夠識別趨勢的變化，於週期及模式不固定的股票市場資料上，比起以記憶資料順序關係的遞迴式神經網路（Recurrent neural networks; RNN）及長短期記憶網路（Long short-term memory; LSTM）之深度學習方法，有較好的預測表現。

模糊集方面，使用了新型態的複數模糊集—球型複數模糊集（Sphere complex fuzzy sets; SCFSs）（Tu & Li 2019）於本研究模型中。1965 年，模糊集（Fuzzy sets）（Type-1 模糊集）的概念首先被提出（Zadeh 1965）。對於模糊集，每個元素的歸屬程度皆介於實數區間[0,1]，藉此表現出人對於一個事物的語意描述。隨後，模糊理論之研究持續發展。複數模糊集（Complex fuzzy sets; CFSs）（Ramot et al. 2003; Ramot et al. 2002）是傳統模糊集的延伸，在複數模糊集中，元素的歸屬程度是在複數平面的單位圓盤範圍中。透過複數的歸屬程度，複數模糊集比 Type-1 模糊集具有更好乘載歸屬資訊的能力，相關應用陸續被提出（Li & Chiang 2013; Ma et al. 2012; Man et al. 2007）。基於複數模糊集，球型複數模糊集採用球坐標系（Spherical coordinate system）的概念，能靈活調整歸屬程度的輸出數量。比起傳統模糊集，球型複數模糊集可容納更多的歸屬資訊，若將其應用於模型中，可使模型實現多輸出的能力。

機器學習演算法方面，常見的演算法為基於梯度下降（Gradient descent）的倒傳遞法（Backpropagation; BP）（Rumelhart et al. 1986），該方法是根據誤差結果，透過導數由後往前調整神經網路各層的權重參數。相對於此，無需導數的最佳化演算法（Derivative-free optimization）是根據參數對目標函數的評估結果，調整每次迭代參數的更新準則，如基因演算法（Genetic algorithm; GA）（Goldberg 1989）、粒子群最佳化演算法（Particle swarm optimization; PSO）（Kennedy & Eberhart 1995）以及鯨群最佳化演算法（Whale optimization algorithm; WOA）（Mirjalili & Lewis 2016）等。由於不相依於模型，此類型演算法適用於複雜架構的模型上。此外，混合策略的最佳化方法比單一最佳化方法，具有更高的訓練能力（Li & Wu 2011; Wu et al. 2015）。因此，本研究使用了複合式機器學習演算法，對本研究所提之模型進行參數訓練，利用高斯分布型鯨群最佳化演算法（Gaussian distribution based whale optimization algorithm; GD-WOA）（Wang & Li 2019）調整 CNN 及球型複數模糊集之參數。接著使用遞迴最小平方估計法（Recursive least squares estimator; RLSE）（Jang & Sun 1997），透過線性回歸方法，對模糊神經系統之後鑑部參數進行調整。

在過去研究中，多數模型只進行單一目標預測。然而，資料分析的需求越趨複雜化，當預測目標數增加時，這些方法就須批次進行預測，較無效率。本研究提出一個多目標預測模型 CNN-SCNFS，結合 CNN 與球型複數類神經模糊系統（Sphere complex neural fuzzy system; SCNFS），在選擇模型輸入上，使用以資訊熵為基礎的過濾方法（Tu & Li 2017），挑選出有影響力之特徵。訓練模型方面，使用複合式機器學習演算法 GD-WOA-RLSE，透過分而治之的方式，優化模型參數。為測試本研究所提之方法，分別以股價指數作為資料集，進行單目標、雙目標及四目標時間序列預測。

本論文其餘的部分如下，第貳節描述研究方法，介紹球型複數模糊集、以資訊熵為基礎的特徵挑選、本研究所提之模型架構及複合式機器學習演算法流程。第參節以三個實例，驗證本研究所提之方法的預測能力，並與其他文獻方法進行比較。第肆節將對實驗結果進行討論。最後為本研究之結論。

貳、研究方法

一、球型複數模糊集

球型複數模糊集是將球坐標系及複數的概念結合。對於球坐標系，三維空間中的任意一點位置，可由三個參數來表示，包含徑向距離r、天頂角θ、方位角φ。透過直角坐標系的轉換，可得到該點在三軸上映射之位置。相似的概念被應用於球型複數模糊集理論中，其歸屬程度的計算方法分為三個階段，第一階段為計算振幅值r及多個相位值θ，在本研究中，分別使用高斯型態的歸屬函數（Gaussian type membership function;GMF）以及其偏微分方程式，如下式：

$$r = \mathrm{GMF}(h;c,\sigma) = \exp\left[-\frac{1}{2}\left(\frac{h-c}{\sigma}\right)^2\right] \tag{1}$$

$$\theta_1 = \mathrm{GMF}'(h;c,\sigma) = \exp\left[-\frac{1}{2}\left(\frac{h-c}{\sigma}\right)^2\right]\cdot\left[\frac{-(h-c)}{\sigma^2}\right] \tag{2}$$

$$\theta_2 = \mathrm{GMF}''(h;c,\sigma) = \exp\left[-\frac{1}{2}\left(\frac{h-c}{\sigma}\right)^2\right]\cdot\left[\left(\frac{-(h-c)}{\sigma^2}\right)^2 - \frac{1}{\sigma^2}\right] \tag{3}$$

$$\theta_3 = \mathrm{GMF}'''(h;c,\sigma) = \exp\left[-\frac{1}{2}\left(\frac{h-c}{\sigma}\right)^2\right]\cdot\left[\left(\frac{-(h-c)}{\sigma^2}\right)^3 - \frac{3(h-c)}{\sigma^2}\right] \tag{4}$$

其中，h為輸入值；c為高斯型態之歸屬函數的中心值；σ為高斯型態之歸屬函數的寬度。第二階段為計算歸屬程度，透過直角坐標系的轉換方式，計算出不同的歸屬程度u，如下式：

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} r\cdot\sin\theta_1\cdot\cos\theta_2 \\ r\cdot\sin\theta_1\cdot\sin\theta_2 \\ r\cdot\sin\theta_2\cdot\cos\theta_3 \\ r\cdot\sin\theta_2\cdot\sin\theta_3 \end{bmatrix} \tag{5}$$

第三階段為複數化，將不同歸屬程度結合複數的概念，產生出球型複數模糊集之歸屬程度$\vec{\mu}$，如下式所示，其中$j=\sqrt{-1}$：

$$\vec{\mu} = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} = \begin{bmatrix} u_1 + ju_2 \\ u_3 + ju_4 \end{bmatrix} \tag{6}$$

球型複數模糊集可根據目標數，在不增加參數數量的情況下，一次產生多個複數歸屬程度，比起複數模糊集與 Type-1 模糊集，能傳遞更多的歸屬資訊。

二、多目標特徵選取

對於多目標特徵選取的輸入資料集，模型輸入及目標分別代表候選特徵池（Candidate feature pool; CP）及目標集合（Target set; TS），如下矩陣所示：

$$\begin{bmatrix} f_{1,1} & f_{2,1} & \cdots & f_{|CP|,1} & t_{1,1} & t_{2,1} & \cdots & t_{|TS|,1} \\ f_{1,2} & f_{2,2} & \cdots & f_{|CP|,2} & t_{1,2} & t_{2,1} & \cdots & t_{|TS|,2} \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ f_{1,S} & f_{2,S} & \cdots & f_{|CP|,S} & t_{1,S} & t_{2,S} & \cdots & t_{|TS|,S} \end{bmatrix} \tag{7}$$

其中，$i = 1,2,\dots,|CP|$，$|CP|$為候選特徵個數；$k = 1,2,\dots,|TS|$，$|TS|$為目標個數；$j = 1,2,\dots,S$，S為資料集筆數；$f_{i,j}$為第i個候選特徵的第j筆資料；$t_{k,j}$為第 k 個預測目標的第j筆資料。以下將介紹多目標特徵挑選之流程步驟：

步驟一：計算影響資訊矩陣

影響資訊矩陣（Influence information matrix; IIM）是以夏農資訊熵（Shannon information entropy）（Shannon 1948）為基礎，計算互資訊（Mutual information），再透過互資訊，算出特徵對特徵、特徵對目標之間的影響資訊量，並將結果以矩陣的形式存儲。互資訊的計算方法為資訊熵減去條件資訊熵，如下所示：

$$H(Y) = \int p_d(y)\log(\frac{1}{p_d(y)})dy \tag{8}$$

$$H(Y|X) = \iint p_d(y|x)p_d(x)\log(\frac{1}{p_d(y|x)})dydx \tag{9}$$

$$I(X,Y) = H(Y) - H_X(Y) \tag{10}$$

其中，$H(Y)$為隨機變數Y之資訊熵；$p_d(y)$為機率密度分布下，事件 y 發生的機率；$H(Y|X)$為已知隨機變數X情況下，隨機變數Y之條件資訊熵；$p_d(y|x)$為已知事件x發生的條件下，事件y發生的機率；$p_d(x)$為機率密度分布下，事件x發生的機率。

為獲取更準確的資訊，此方法會再分成兩種條件計算互資訊，分別是在隨機

變數X為正的情況下，與隨機變數Y產生之互資訊，以及在隨機變數X為負的情況下，與隨機變數Y產生之互資訊。隨後，乘上對應的機率密度分布期望值，再將兩個結果相加，就可得到影響資訊，如下所示：

$$I(X^+,Y) = H(Y) - H_{X^+}(Y) \tag{11}$$

$$I(X^-,Y) = H(Y) - H_{X^-}(Y) \tag{12}$$

$$I_{X\to Y} = I(X^+,Y)\int_0^\infty p_d(x)dx + I(X^-,Y)\int_{-\infty}^0 p_d(x)dx \tag{13}$$

根據目標數，會有$|TS|$個影響資訊矩陣，以下為影響資訊矩陣之示意圖：

	$\boldsymbol{f_1}$	$\boldsymbol{f_2}$	$\cdots$	$\boldsymbol{f_{\lvert CP\rvert}}$	$\boldsymbol{t_k}$
$\boldsymbol{f_1}$	0	$I_{f_1\to f_2}$	$\cdots$	$I_{f_1\to f_{\lvert CP\rvert}}$	$I_{f_1\to t_k}$
$\boldsymbol{f_2}$	$I_{f_2\to f_1}$	0	$\cdots$	$I_{f_2\to f_{\lvert CP\rvert}}$	$I_{f_2\to t_k}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$\boldsymbol{f_{\lvert CP\rvert}}$	$I_{f_{\lvert CP\rvert}\to f_1}$	$I_{f_{\lvert CP\rvert}\to f_2}$	$\cdots$	0	$I_{f_{\lvert CP\rvert}\to t_k}$
$\boldsymbol{t_k}$	$I_{t_k\to f_1}$	$I_{t_k\to f_2}$	$\cdots$	$I_{t_k\to f_{\lvert CP\rvert}}$	0

圖 1：影響資訊矩陣之示意圖（第k個目標）

步驟二：計算選取增益

特徵對目標的選取增益（Selection gain），是根據影響資訊進行計算。當選取增益值為正時，該特徵就會被放入相對目標的選取特徵池（Selected feature pool; SP）中。選取增益計算方法如下式：

$$g(f_i \to t_j) = I_{f_i\to t_j} - R_{f_i - SP^{(j)}} \tag{14}$$

其中，$i = 1,2,\dots,|CP|$；$j = 1,2,\dots,|TS|$；$g(f_i \to t_j)$為f_i對t_j的選取增益;$I_{f_i\to t_j}$為f_i對t_j的影響資訊；$R_{f_i\to SP^{(j)}}$為f_i對第j個選取特徵池的冗餘資訊（Redundant information），對於新選取的特徵，冗餘資訊是由已被選進特徵池的特徵所提供，故將此重複的部分扣除。

步驟三：計算已選取特徵之覆蓋率與總選取增益

每個選取特徵池經過上一步驟後，會有若干被選取之特徵。然而，這些特徵是對應單一目標而被選取的。為評估這些特徵對整體目標之貢獻，需計算每個特徵，在所有選取特徵池的覆蓋率及總選取增益。為方便計算，聯集所有選取特徵

池內的特徵，並將結果放入集合Ω內，$\Omega = \{\phi_k, k = 1,2,\dots,|\Omega|\}$，$|\Omega|$為被選取特徵的總個數。覆蓋率計算方法如下式：

$$\omega(\phi_k) = \frac{n_{OL}(\phi_k)}{|TS|} \tag{15}$$

其中，$\omega(\phi_k)$為第k個被選取特徵的覆蓋率；$n_{OL}(\phi_k)$為第k個被選取特徵，在這些選取特徵池中出現的次數（Number of overlapping）。總選取增益的計算方法如下式：

$$g_{sum}(\phi_k) = \sum_{j=1}^{|TS|} g(\phi_k \to t_j) \tag{16}$$

其中，$g_{sum}(\phi_k)$是第k個被選取特徵，對每個目標之選取增益的加總。

步驟四：挑選有效貢獻之特徵

透過上一步驟中的覆蓋率及總選取增益，可計算特徵對整體目標的有效貢獻量，計算方法如下式：

$$\rho(\phi_k) = \omega(\phi_k) \cdot g_{sum}(\phi_k) \tag{17}$$

其中，$\rho(\phi_k)$為第k個被選取特徵，對整體目標的有效貢獻量。最後，將有效貢獻量大於零之特徵，由多至少排序後，放入最終選取池（Final selected feature pool; FP）內，作為多目標特徵選取的最終結果，共有$|FP|$個特徵。

二、CNN-SCNFS 預測模型

本研究所提之模型分為兩個部分，前部分為 CNN，後部分為 SCNFS，以下為 CNN-SCNFS 預測模型介紹。

（一）CNN 架構

為探討卷積層於模型中的效用，本研究之CNN架構皆以卷積層為主，激勵函數則使用雙曲正切函數tanh，其輸出區間介於$[-1,1]$，對於漲跌值之資料較為合適。最後一層則為線性輸出的全連接層。卷積層的運作方式如下：

$$a_{i,j} = f\left(\sum_{m=1}^{M}\sum_{n=1}^{N} w_{m,n} x_{i+m-1,j+n-1}\right) \tag{18}$$

$$\tanh(z) = \frac{2}{1+e^{-2z}} - 1 \tag{19}$$

其中，x為輸入矩陣；$w_{m,n}$為卷積核（Kernel）第m列的第n個權重；$a_{i,j}$為特徵圖

（Feature map）的第i列的第j個元素；f為激勵函數。多目標特徵挑選後，會將特徵向量進行重構（Reshape）轉化為方陣，以符合 CNN 之輸入格式。

（二）SCNFS 架構

為減少主觀設定，SCNFS 應用結構學習的概念，其建構方法是透過模型輸入來設定，以達到資料驅動的目的。SCNFS 架構共分為六個階層，順序如下：輸入層、球型複數模糊集層、前鑑部層、正規化層、後鑑部層與輸出層。

在輸入層，會將CNN之輸出作為模型輸入並傳進下一層。在球型複數模糊集層的建立方法上，本研究使用 Chiu（1994）提出的減法分群法（Subtractive clustering），將模型輸入進行分群，算出各群的中心值及其寬度，以此建立每個模型輸入的歸屬函數進而算出歸屬程度。球型複數模糊集之歸屬函數如下式：

$$\vec{\mu}_{i,j} = A_{i,j}(h_i; c_{i,j}, \sigma_{i,j}) \tag{20}$$

其中，$i = 1,2,\dots,M$，M為模型輸入個數；$j = 1,2,\dots,|T_i|$，$|T_i|$為第i個模型輸入之語譯變數個數；h_i為第i個模型輸入；$c_{i,j}$為第i個模型輸入的第j個中心值；$\sigma_{i,j}$為第i個模型輸入的第j個寬度；$A_{i,j}$為第i個模型輸入的第j個球型複數模糊集的歸屬函數；$\vec{\mu}_{i,j}$為第i個模型輸入的第j個球型複數模糊集的歸屬程度。

對於規則式的前鑑部層、正規化層、後鑑部層，其建立方法是將所有模型輸入分群後的中心個數，經由笛卡兒乘積（Cartesian product）算出若干組合，每一組合代表一條獨立規則。規則的前鑑部是計算每一模型輸入對不同規則的關聯性，稱為啟動強度（Firing strength），計算方法如式(21)，其中，$k = 1,2,\dots,K$，K為規則個數；$\vec{\mu}_i^{(k)}$為第k條規則的第i個模型輸入對應之歸屬程度。隨後，為降低啟動強度的離散程度，加入正規化層，將啟動強度限制在區間$[0,1]$，計算方法如式(22)。

$$\vec{\beta}^{(k)} = \prod_{i=1}^{M} \vec{\mu}_i^{(k)} \tag{21}$$

$$\vec{\lambda}^{(k)} = \frac{\vec{\beta}^{(k)}}{\sum_{k=1}^{K} \vec{\beta}^{(k)}} \tag{22}$$

接著將正規化後的啟動強度，乘上對應規則之後鑑部後，可得到模型輸入對該條規則之反應。最後，經由輸出層加總所有規則反應後，可得到模型輸出，如下式：

$$\vec{y} = \sum_{k=1}^{K} \vec{\lambda}^{(k)} \times (a_0^{(k)} + a_1^{(k)} h_1 + \cdots + a_M^{(k)} h_M) \tag{23}$$

四、複合式機器學習演算法

本研究使用複合式機器學習演算法，包含 GD-WOA 及 RLSE，分別負責模型中不同的模型參數，GD-WOA 訓練的參數為 CNN 的卷積核權重，及球型複數模糊集之歸屬函數的中心值及寬度。其訓練方法是將這些模型參數，視為多維空間的最佳化問題，經由搜尋最佳位置來調整參數；RLSE 訓練的參數為 SCNFS 之後鑑部參數，使用模型輸入及目標，進行線性回歸來調整參數。複合式機器學習演算法之訓練流程如圖 2 所示。以下分別為 GD-WOA 及 RLSE 的詳細介紹。

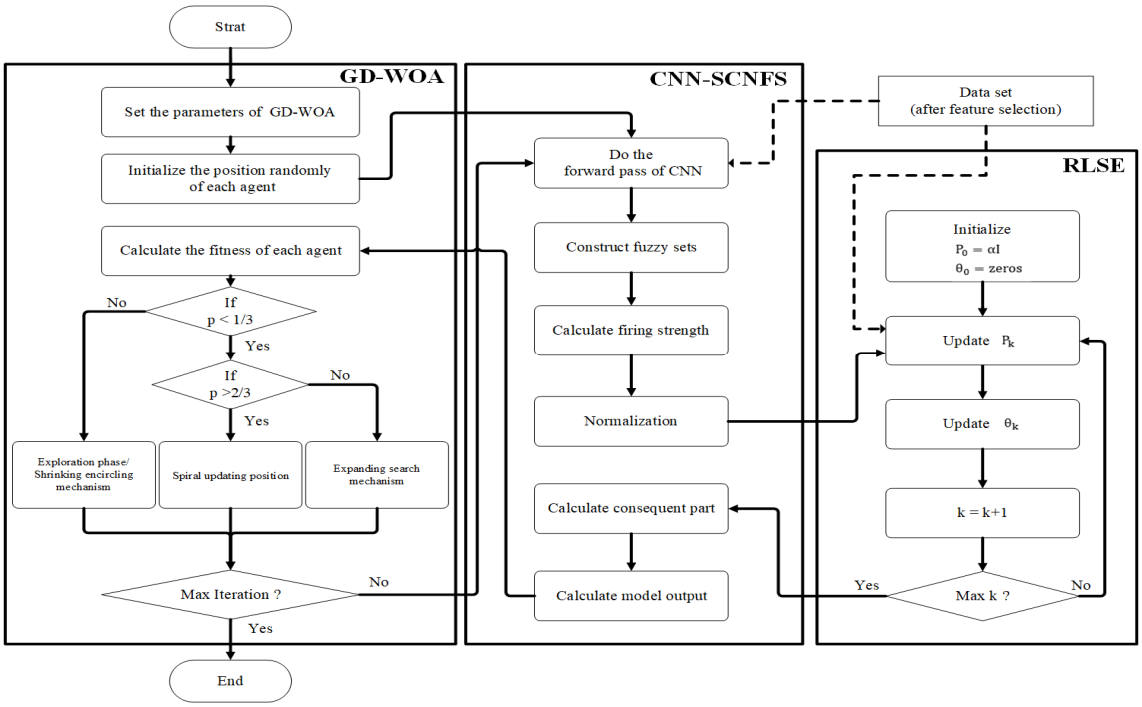

圖 2：複合式機器學習演算法 GD-WOA-RLSE 之流程圖

（一）GD-WOA

GD-WOA 是以 WOA 進行改良，在最佳化問題上，有良好的搜尋能力及穩定性，其搜尋最佳位置的過程分為四種優化策略，分別是探索階段（Exploration phase）、收縮環繞機制（Shrinking encircling mechanism）、螺旋更新位置（Spiral updating position）及擴大搜索機制（Expanding search mechanism）。以下為 GD-WOA 四種優化策略之介紹，對於探索階段，更新鯨魚位置之方法如下式：

$$\vec{D} = |\vec{C} \cdot \vec{X}_{rand} - \vec{X}_i(t)| \tag{24}$$

$$\vec{X}_i(t+1) = \vec{X}_{rand} - \vec{A} \cdot \vec{D} \tag{25}$$

其中，$\vec{X}_i(t)$為第t次迭代的第i隻鯨魚位置；$\vec{X}_{rand}$為當前群中隨機選擇出的位置向量；$\vec{A}$為隨機亂數，亂數範圍在迭代開始時，是介於$[-2,2]$的區間內，隨著迭代遞減而逐漸趨近於 0。$\vec{C}$則是介於$[0,2]$的隨機亂數。對於收縮環繞機制，更新鯨魚位置之方法如下式：

$$\vec{D}' = \vec{C} \cdot \vec{X}_{\text{gauss}} - \vec{X}_i(t) \tag{26}$$

$$\vec{X}_i(t+1) = \vec{X}^* + \vec{A} \cdot \vec{D}' + \vec{D}_{\text{good}} \tag{27}$$

其中，$\vec{X}_{\text{gauss}}$是以群中表現最佳之鯨魚位置為中心，建立高斯分布從中產生的一個新位置；$\vec{D}_{\text{good}}$是以群中表現次佳之鯨魚位置，加上隨機亂數產生的一個新位置；$\vec{X}^*$為當前群中最佳鯨魚的位置向量。對於螺旋更新位置機制，更新鯨魚位置之方法如下式：

$$\vec{D}'' = |\vec{X}_{\text{gauss}} - \vec{X}_i(t)| \tag{28}$$

$$\vec{X}_i(t+1) = \vec{X}^* + \vec{D}'' \cdot e^{bl} \cdot \cos(2\pi l) + \vec{D}_{\text{good}} \tag{29}$$

其中，b為一個常數，用來定義螺旋路徑的形狀；l為介於$[-1,1]$之間的亂數，目的是增添路徑旋轉角度的隨機性。對於新增的擴大搜索機制，更新鯨魚位置之方法如下式，其中，ξ為一個介於$[0,1]$的隨機亂數：

$$\vec{D}''' = \left(1 + \frac{\xi}{2}\right)^6 \cdot \left(\vec{X}_{\text{gauss}} - \vec{X}_i(t)\right) \tag{30}$$

$$\vec{X}_i(t+1) = \vec{X}_i(t) + \vec{D}''' \tag{31}$$

（二）RLSE

對於線性函數，RLSE 主要由以下兩式來進行參數優化：

$$\begin{cases} \mathbf{P}_{k+1} = \mathbf{P}_k - \frac{\mathbf{P}_k \mathbf{a}_{k+1} \mathbf{a}_{k+1}^T \mathbf{P}_k}{1 + \mathbf{a}^T \mathbf{P}_k \mathbf{a}} \\ \mathbf{\Theta}_{k+1} = \mathbf{\Theta}_k + \mathbf{P}_{k+1} \mathbf{a}_{k+1} (\mathbf{y}_{k+1} - \mathbf{a}_{k+1}^T \mathbf{\Theta}_k) \end{cases} \tag{32}$$

其中，$\mathbf{a}$為一個$m \times n$的輸入矩陣；$\mathbf{\Theta}$為一個$n \times 1$的參數矩陣；$\mathbf{y}$為一個$m \times 1$的目標矩陣。在使用 RLSE 前，必須先設定初始值$\mathbf{\Theta}_0$及$\mathbf{P}_0$，$\mathbf{\Theta}_0$為一個全為 0 的$n \times m$矩陣，$\mathbf{P}_0$的設定為αI，α為一個相當大的正數值；I為單位矩陣。

參、實驗實作與結果

本研究實驗使用之設備為 Intel Core i7-4900MQ、8GB RAM 的筆記型電腦，計算軟體為 Matlab 2018b，共有三個實驗。實驗一分別以 SCNFS 及不同 CNN-SCNFS 等模型，進行單目標預測，主要驗證 CNN 的加入以及不同的模型架構，對模型預測能力之影響。實驗二及實驗三為多目標預測，實驗二以不同區域的兩個股票市場指數作為目標；實驗三以同樣區域的四個股票市場指數作為目標。主要驗證球型複數模糊集的使用，對於模型在多目標問題上的可行性及預測能力。在各項實驗中，每個實驗皆重複進行二十次。此外，本研究在三個實驗中，加入倒傳遞類神經網路（Backpropagation neural network; BPNN）作為演算法訓練模型時間的比較對象，其訓練參數量設定為與本研究提出之方法相近（本研究所提之模型訓練參數量為 45，BPNN 為 50）。最後，測試階段的實驗結果與其他文獻比較呈現於相關表格中。

實驗資料是從 Yahoo Finance 獲得，每一股價指數包含開盤價、最高價、最低價及收盤價等四種資料。實驗會將資料進行一階差分（Differencing）來達到穩態（Stationarity），只專注在波動的變化。在資料集的格式上，使用四種資料之連續三十筆差分值，依照開盤價、最高價、最低價及收盤價的次序擺放，作為輸入值；並將第三十一筆收盤價的差分值，作為預測目標值。

在機器學習演算法的設定方面，迭代次數為 100 次；鯨魚數量為 30 隻；BPNN 之訓練方法則依照上述的計算次數，迭代次數設定為 3000。學習指標為根均方誤差（Root mean square error; RMSE），當其值愈小時，可代表預測的結果愈準確。在實驗二中，也另外加入其他檢定指標如平均絕對誤差（Mean absolute error; MAE）、平均絕對值誤差率（Mean absolute percentage error; MAPE）及平均平方根百分比誤差（Root mean square percentage; RMSPE）來做為模型表現之評估。其中，MAE、MAPE 及 RMSPE 其值愈小時，可代表預測的結果愈準確。各指標計算方法如下，其中，y_i為第i筆實際目標值；$\hat{y}_i$為第i筆模型之輸出值。

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{n}} \tag{33}$$

$$\text{MAE} = \frac{\sum_{i=1}^{n}|y_i-\hat{y}_i|}{n} \tag{34}$$

$$\text{MAPE} = \frac{\sum_{i=1}^{n}\left|\frac{y_i-\hat{y}_i}{y_i}\right|}{n} \tag{35}$$

$$\text{RMSPE} = \sqrt{\frac{\sum_{i=1}^{n}(\frac{y_i - \hat{y}_i}{y_i})^2}{n}} \tag{36}$$

一、 實驗一

實驗一的預測目標為 2000 年至 2004 年台灣加權股價指數（Taiwan Stock Exchange Capitalization Weighted Stock Index; TAIEX）之收盤價格，資料集設定如表 1 所示。本實驗以十種不同模型架構進行單目標預測，如表 2 所列，其中，CNN(L,U)代表該 CNN 架構有 L 層卷積層，每層有 U 個卷積核，卷積核大小為 2×2。最後，會從中挑選最佳預測表現之架構與表 3 中的文獻方法比較，並作為後續實驗之模型。

表 1：資料集設定（實驗一）

Data set	Data size (original)	Data pairs (Training/ Test/ Total)	Ratio (%) of data pairs (Training/ Test)
TAIEX (2000)	246*	172/ 43/ 215	80/ 20
TAIEX (2001)	245	171/ 43/ 214	80/ 20
TAIEX (2002)	248	171/ 43/ 214	80/ 20
TAIEX (2003)	249	174/ 44/ 218	80/ 20
TAIEX (2004)	250	175/ 45/ 219	80/ 20

* Yahoo Finance 提供之資料筆數，實際交易日為 271 天。

表 2：不同模型之預測能力及訓練時間（RMSE）(時間單位：秒)

Method		2000	2001	2002	2003	2004	Average
BPNN	Best	144.76	113.53	61.75	50.06	58.64	85.75
	Worst	176.85	127.10	76.96	58.56	72.14	102.32
	Mean	158.97	119.84	68.40	54.43	66.51	93.63
	Std	8.97	3.99	4.36	2.40	3.66	4.68
	CPU time (s)	0.0002	0.0004	0.0002	0.0003	0.0004	0.0003
SCNFS	Best	142.03	114.37	59.60	54.72	57.09	85.56
	Worst	254.12	114.38	105.76	88.92	71.94	127.02
	Mean	161.34	114.37	67.51	63.66	60.39	93.45
	Std	24.72	0.00	10.51	8.01	3.33	9.31

	CPU time (s)	6.1147	6.0905	6.4913	6.1603	6.3012	6.2316
CNN(1,1)-SCNFS	Best	144.18	109.63	61.10	50.78	55.59	84.26
	Worst	168.18	121.90	75.42	61.31	78.12	100.99
	Mean	152.35	116.49	69.09	53.94	63.45	91.06
	Std	6.15	3.67	3.11	2.58	4.44	3.99
	CPU time (s)	7.5377	6.9528	7.2746	7.0062	7.0109	7.1564
NN(1,2)-SCNFS	Best	140.52	111.18	62.07	50.64	56.49	84.18
	Worst	166.32	123.96	71.71	59.40	70.49	98.38
	Mean	151.65	117.41	68.13	54.41	62.95	90.91
	Std	7.30	2.95	2.94	2.14	4.02	3.87
	CPU time (s)	8.4286	7.6844	7.6080	7.5773	7.4383	7.7473
CNN(1,3)-SCNFS	Best	143.17	105.52	60.92	50.82	55.73	83.23
	Worst	171.20	126.47	71.58	61.89	69.22	100.07
	Mean	152.61	115.65	66.09	56.08	62.79	90.64
	Std	8.08	4.93	2.59	3.19	3.29	4.42
	CPU time (s)	8.6936	9.2970	8.9646	8.6663	8.8607	8.8964
CNN(2,1)-SCNFS	Best	139.18	107.41	61.38	50.59	52.47	82.21
	Worst	171.87	124.36	74.40	57.09	70.83	99.71
	Mean	155.37	116.33	68.73	53.34	63.74	91.50
	Std	6.76	4.66	2.89	1.72	4.10	4.03
	CPU time (s)	7.7065	7.5183	7.1117	7.5714	7.2051	7.4226
CNN(2,2)-SCNFS	Best	139.02	110.05	63.40	52.17	58.58	84.64
	Worst	166.48	123.40	78.31	63.92	75.26	101.47
	Mean	150.62	116.12	70.05	55.68	65.25	91.54
	Std	7.53	2.85	3.96	2.92	4.29	4.31
	CPU time (s)	8.0105	8.1007	8.1835	8.4964	8.1241	8.1830
CNN(2,3)-SCNFS	Best	135.97	109.21	62.98	48.91	57.52	82.92
	Worst	165.12	129.35	76.57	68.31	76.83	103.24
	Mean	154.22	117.61	68.20	54.91	64.11	91.81
	Std	7.48	5.10	3.23	4.00	4.25	4.81
	CPU time (s)	8.2566	8.8840	8.7596	8.2744	8.3059	8.4961
CNN(3,1)-SCNFS	Best	138.12	110.44	61.20	50.92	53.81	82.90
	Worst	167.75	128.25	70.18	58.73	69.28	98.84

	Mean	151.50	117.48	66.78	53.51	60.75	90.00
	Std	8.02	4.50	2.14	2.22	3.77	4.13
	CPU time (s)	7.3503	7.6008	7.4241	7.5957	7.3381	7.4618
CNN(3,2)-SCNFS (proposed)	Best	126.34	108.47	59.40	49.74	53.94	79.58
	Worst	166.06	124.98	75.09	63.17	83.61	102.58
	Mean	153.28	115.73	69.34	55.92	64.05	91.66
	Std	9.00	4.88	3.53	3.31	6.74	5.49
	CPU time (s)	8.3723	7.8009	8.8033	8.2972	8.1944	8.2936
CNN(3,3)-SCNFS	Best	136.70	108.36	60.58	50.55	52.89	81.82
	Worst	183.40	122.31	76.99	60.36	80.93	104.80
	Mean	158.46	116.20	68.48	55.05	63.70	92.38
	Std	10.22	4.12	4.22	2.30	6.80	5.53
	CPU time (s)	9.0506	9.4201	9.1487	9.3922	8.9193	9.1862

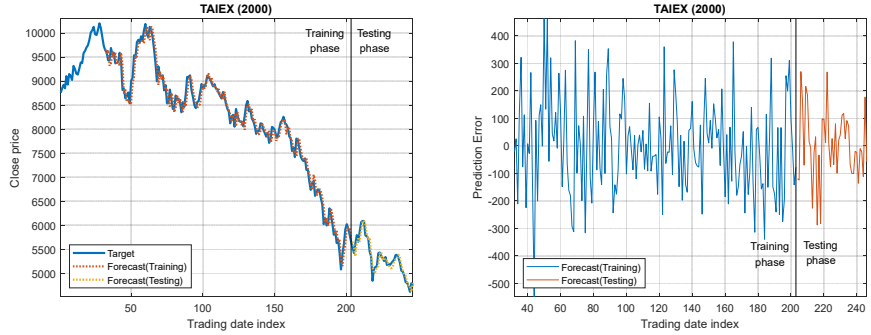

圖 3：模型預測結果與誤差（實驗一）（TAIEX-2000）

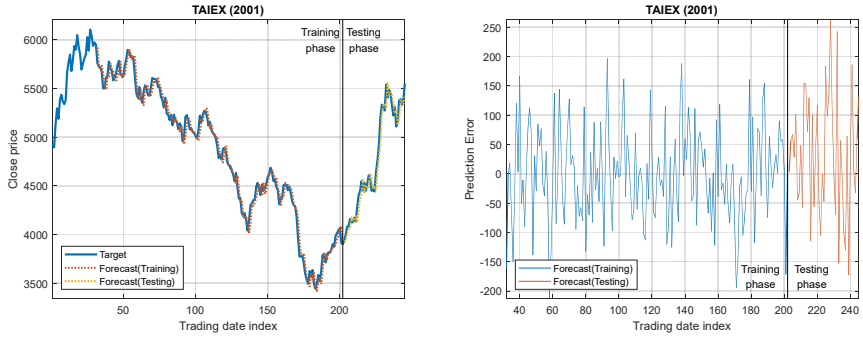

圖 4：模型預測結果與誤差（實驗一）（TAIEX-2001）

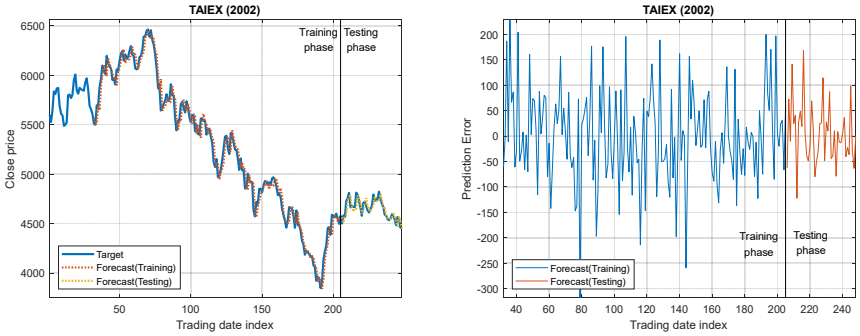

圖 5：模型預測結果與誤差（實驗一）（TAIEX-2002）

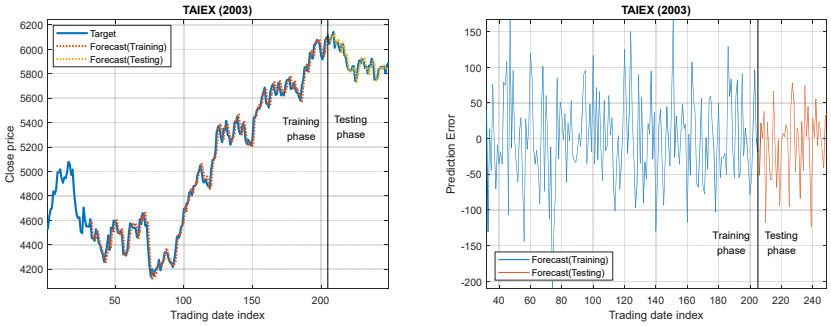

圖 6：模型預測結果與誤差（實驗一）（TAIEX-2003）

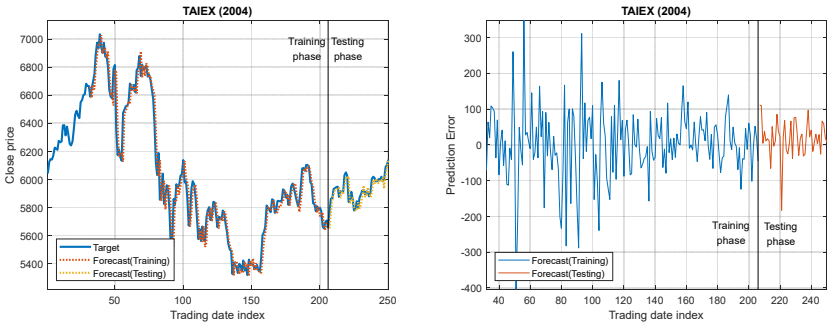

圖 7：模型預測結果與誤差（實驗一）（TAIEX-2004）

表 3：模型預測表現比較（實驗一）（RMSE）

Method	2000	2001	2002	2003	2004	Average
Chen(1996)	176.00	148.00	101.00	74.00	84.00	116.60
Huarng(2001)	152.00	130.00	84.00	56.00	116.00	107.60
Huarng et al.(2007)	154.42	124.02	93.48	65.51	72.35	101.96

Yu and Huarng(2008)	131.00	130.00	80.00	58.00	67.00	93.20
Chen and Chang(2010)	129.42	113.33	66.82	53.51	60.48	84.71
Chen and Chen(2011)	123.62	115.33	71.01	58.06	57.73	85.15
Chen et al.(2012)	119.98	114.47	67.17	52.49	52.27	81.28
Chen et al.(2013)	131.25	112.55	65.77	52.23	54.17	83.19
Egrioglu et al.(2013)	255.00	130.00	84.00	56.00	116.00	128.20
Chen and Chen(2015)	124.52	114.66	64.71	52.84	52.96	81.94
Cai et al.(2015)	131.53	112.59	60.33	51.54	**50.33**	81.26
Cheng et al.(2016)	125.62	113.04	62.94	51.46	54.25	81.46
Ye et al.(2016)	125.42	113.22	63.99	52.99	52.40	84.88
Cai et al.(2013)	126.68	115.79	65.56	57.40	56.10	87.38
Yolcu and Alpaslan(2018)	**113.38**	110.89	**58.68**	50.07	51.60	**76.92**
BPNN	144.76	113.53	61.75	50.06	58.64	85.75
CNN(3,2)-SCNFS (proposed)	126.34	**108.47**	59.40	**49.74**	53.94	79.58

二、 實驗二

實驗二的預測目標為自 2006 年 4 月 12 日至 2010 年 4 月 1 日止巴西股市指數（Bovespa Index; BVSP）每日之收盤價格，及自 2006 年 3 月 13 日至 2010 年 4 月 1 日止日經平均指數（Tokyo Nikkei-225 Index; Nikkei）每日之收盤價格，實驗比照 Kao 等（2013）之設定，將資料集後兩百筆作為測試資料集，如表 4 所示。圖 8 為模型預測結果與誤差。最後，與文獻（Kao et al. 2013）之實驗方法的比較於表 6 中呈現。

表 4：資料集設定（實驗二）

Data set	Data size (original)	Data pairs (Training/ Test/ Total)	Ratio (%) of data pairs (Training/ Test)
BVSP (2006/4/12 ~ 2010/4/1)	1000	769/ 200/ 969	79/ 21
N225 (2006/3/13 ~ 2010/4/1)			

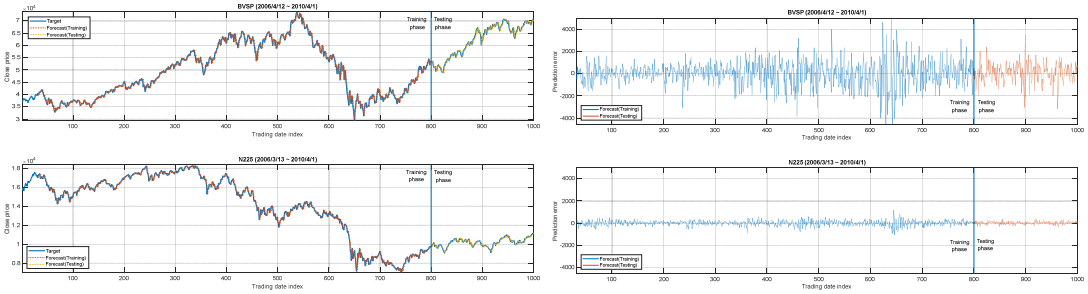

圖 8：模型預測結果與誤差（實驗二）

表 5：本研究提出方法之預測表現（實驗二）

	BVSP				Nikkei 225			
	RMSE	MAE	MAPE	RMSPE	RMSE	MAE	MAPE	RMSPE
Best	874.4982	647.6479	1.049%	1.426%	135.5853	103.8964	1.026%	1.346%
Worst	908.8371	684.9844	1.113%	1.484%	142.5481	115.6294	1.142%	1.414%
Mean	892.4666	660.1552	1.070%	1.456%	135.1342	106.8471	1.055%	1.340%
Std	10.4514	11.6460	0.020%	0.018%	2.6661	2.9252	0.029%	0.027%

表 6：本研究提出之方法與 BPNN 的訓練時間比較（實驗二）（時間單位：秒）

Method	BVSP	Nikkei 225
BPNN	0.0012	0.0015
CNN(3,2)-SCNFS	31.2560*	

*本研究所提之方法（一個模型同時針對兩個目標進行預測），故顯示單一訓練時間。

表 7：不同模型預測表現比較（實驗二）

Method	BVSP				Nikkei 225			
	RMSE	MAE	MAPE	RMSPE	RMSE	MAE	MAPE	RMSPE
ARIMA	882.0765	652.8783	1.072%	1.439%	140.3164	111.2556	1.130%	1.380%
SVR	880.0151	649.6584	1.053%	1436%	137.6541	109.3261	1.081%	1.368%
ANFIS	882.0765	651.6157	1.057%	1.437%	135.5629	107.6568	1.066%	1.352%
Wavelet-SVR	878.7561	642.7191	1.044%	1.430%	133.2766	104.8005	1.014%	1.321%
Wavelet-MARS	881.0468	651.4920	1.055%	1.437%	138.8247	110.5228	1.093%	1.381%

Wavelet-MARS-SVR	876.9092	**641.1467**	**1.043%**	1.428%	**132.3721**	**103.4128**	**1.012%**	**1.303%**
BPNN	890.4214	669.6366	1.086%	1.451%	138.4238	108.2607	1.071%	1.377%
CNN(3,2)-SCNFS*	**874.4982**	647.6479	1.049%	**1.426%**	135.5853	103.8964	1.026%	1.346%

*本研究所提之方法（一個模型同時針對兩個目標進行預測）。

三、實驗三

實驗三的預測目標為 1995 年至 1999 年之道瓊工業指數（Dow Jones Industrial Average; DJIA）、那斯達克綜合指數（National Association of Securities Dealers Automated Quotations; NASDAQ）、標準普爾 500 指數（Standard & Poor's 500; S&P500）及美國費城半導體指數（PHLX Semiconductor Sector Index; SOX）之收盤價格，表 8 為資料集設定。最後，會與表 11 中的文獻方法比較。

表 8：資料集設定（實驗三）

Data set	Data size (original)	Data pairs（Training/ Test/ Total）	Ratio (%) of data pairs (Training/ Test)
DJI, NASDAQ, S&P500, SOX (1995)	252	177/ 44/ 221	80/ 20
DJI, NASDAQ, S&P500, SOX (1996)	254	178/ 45/ 223	80/ 20
DJI, NASDAQ, S&P500, SOX (1997)	253	178/ 44/ 222	80/ 20
DJI, NASDAQ, S&P500, SOX (1998)	252	177/ 44/ 221	80/ 20
DJI, NASDAQ, S&P500, SOX (1999)	252	177/ 44/ 221	80/ 20

表 9：本研究提出方法之預測表現（實驗三）（RMSE）

Stock		1995	1996	1997	1998	1999	Average
DJI	Best	29.23	43.14	80.75	89.75	79.74	64.522
	Worst	31.75	49.54	99.37	103.42	104.54	77.724
	Mean	29.98	47.43	88.61	98.07	92.09	71.236
	Std	0.89	1.40	4.74	4.11	5.95	3.418
NASDAQ	Best	10.01	9.99	19.81	28.60	48.96	23.474

	Worst	11.06	10.29	22.93	32.01	51.18	25.494
	Mean	10.71	10.13	21.79	30.05	51.06	24.748
	Std	0.30	0.27	2.14	1.14	1.53	1.076
S&P500	Best	3.13	5.08	9.66	12.30	10.40	8.114
	Worst	3.46	6.45	11.85	15.70	12.57	10.006
	Mean	3.30	5.57	10.55	13.38	11.70	8.900
	Std	0.12	0.25	0.62	0.72	0.64	0.470
SOX	Best	6.05	4.48	9.19	7.33	16.32	8.674
	Worst	6.78	6.02	10.27	8.35	17.86	9.856
	Mean	6.57	4.51	9.50	7.59	17.00	9.034
	Std	0.24	0.37	0.23	0.25	0.36	0.290

表 10：本研究提出之方法與 BPNN 的訓練時間比較（實驗三）（時間單位：秒）

Method	Stock	1995	1996	1997	1998	1999	Average
BPNN	DJI	0.0005	0.0005	0.0004	0.0004	0.0004	0.0004
	NASDAQ	0.0004	0.0004	0.0004	0.0004	0.0005	0.0004
	S&P500	0.0004	0.0004	0.0004	0.0004	0.0004	0.0004
	SOX	0.0004	0.0004	0.0004	0.0004	0.0004	0.0004
CNN(3,2)-SCNFS*	4 targets	14.0548	15.3140	14.3327	14.0279	14.1888	14.3836

*本研究所提之方法（一個模型同時針對四個目標進行預測），故顯示單一訓練時間。

表 11：模型預測表現比較（實驗三）（RMSE）

Stock	Method	1995	1996	1997	1998	1999	Average
DJI	Ye et al.(2016)	30.98	47.83	**79.14**	91.66	84.26	66.774
	Cai et al.(2013)	30.93	46.89	82.60	94.51	82.09	67.404
	Cai et al.(2015)	32.43	46.73	84.01	96.46	87.68	69.462
	LR(Tu & Li, 2019)	30.97	47.4	83.53	93.02	80.97	67.177
	SVM(Tu & Li, 2019)	31.14	49.51	85.14	94.79	85.75	69.264
	SCFS(Tu & Li, 2019)	30.25	44.53	82.19	92.25	79.27	65.858
	BPNN	31.14	44.20	80.72	95.84	**79.50**	66.280
	CNN(3,2)-SCNFS*	**29.23**	**43.14**	80.75	**89.75**	79.74	**64.522**
NASDAQ	Ye et al.(2016)	10.54	9.76	22.86	**21.07**	**45.41**	**21.928**

	Cai et al.(2013)	10.87	11.21	20.64	31.23	49.27	24.644
	Cai et al.(2015)	11.58	9.64	**18.44**	29.68	46.65	23.198
	LR(Tu & Li, 2019)	11.11	9.56	19.69	31.39	48.10	23.969
	SVM(Tu & Li, 2019)	11.12	9.64	20.02	30.45	47.63	23.774
	SCFS(Tu & Li, 2019)	10.82	9.69	18.93	30.87	48.69	23.800
	BPNN	10.71	**9.17**	20.84	31.65	48.11	24.096
	CNN(3,2)-SCNFS*	**10.01**	9.99	19.81	28.60	48.96	23.474
S&P500	Ye et al.(2016)	**2.73**	6.80	10.44	**11.44**	**9.50**	8.182
	Cai et al.(2013)	3.67	5.64	**9.64**	13.04	11.38	8.674
	Cai et al.(2015)	3.48	5.70	10.12	13.56	10.87	8.746
	LR(Tu & Li, 2019)	3.36	5.66	9.99	12.89	10.40	8.457
	SVM(Tu & Li, 2019)	3.37	5.87	10.08	12.99	10.56	8.572
	SCFS(Tu & Li, 2019)	3.30	5.29	9.86	13.40	10.23	8.416
	BPNN	3.26	5.38	9.65	14.46	11.05	8.760
	CNN(3,2)-SCNFS*	3.13	**5.08**	9.66	12.30	10.40	**8.114**
SOX	LR(Tu & Li, 2019)	6.68	4.68	**9.10**	16.10	16.37	10.587
	SVM(Tu & Li, 2019)	6.68	4.85	9.90	15.81	16.67	10.784
	SCFS(Tu & Li, 2019)	6.73	4.72	**9.10**	16.43	16.68	10.732
	BPNN	6.96	**4.36**	9.56	7.40	**15.30**	8.716
	CNN(3,2)-SCNFS*	**6.05**	4.48	9.19	**7.33**	16.32	**8.674**

*本研究所提之方法（一個模型同時針對四個目標進行預測）。

四、統計檢定

為驗證本研究所提之方法優於傳統模型 BPNN，本研究使用雙樣本中位數差異檢定（Wilcoxon Signed-Rank Test）來進行測試，該檢定是由統計學家 Frank Wilcoxon 所提出，為無母數的統計方法，是利用雙樣本在同一個變項的分佈狀況來進行檢驗。該檢定是針對兩種不同模型，在預測能力差異的常用檢定之一。表 12 至表 14 為使用三個實驗中，本研究所提之方法 CNN(3,2)-SCNFS 與 BPNN 的二十次實驗結果（RMSE）計算 z 檢定統計量，括號內為 p-value。

從檢定結果可觀察到，在實驗一的 2001 年、2004 年及五年平均 TAIEX 之資料集上，其檢定結果顯示本研究所提之方法與 BPNN 有顯著差異。在實驗二中的檢定結果中，皆顯示本研究所提之方法與 BPNN 有顯著差異，驗證本研究所提之方法在 BVSP 及 Nikkei 225 的預測表現上，優於 BPNN。在實驗三的二十個資料

集中，有十二個資料集顯示兩個模型的預測表現，有顯著差異；在四個指數的五年平均檢定結果中，兩個模型於 NASDAQ 及 S&P500 資料集的預測表現上，有顯著差異。綜合上述，本研究所提之方法在時間序列預測問題上，擁有較佳表現。

表 12：Wilcoxon Signed-Rank Test 之檢定結果（實驗一）

Stock	2000	2001	2002	2003	2004	Average
TAIEX	-1.568(.058)	-2.837*(.002)	-1.120(.131)	-0.746(.226)	-2.426*(.007)	-2.427*(.015)

*p-value < 0.05。

表 13：Wilcoxon Signed-Rank Test 之檢定結果（實驗二）

Stock	
BVSP	-3.509*(.000)
Nikkei 225	-3.919*(.000)

*p-value < 0.05。

表 14：Wilcoxon Signed-Rank Test 之檢定結果（實驗三）

Stock	1995	1996	1997	1998	1999	Average
DJI	-3.920*(.000)	-1.904*(.028)	-1.867*(.030)	-0.410(.340)	-0.485(.312)	-0.971(.332)
NASDAQ	-2.800*(.002)	-2.277*(.011)	-0.448(.326)	-3.883*(.000)	-1.307(.095)	-3.248*(.001)
S&P 500	-2.352*(.009)	-0.634(.264)	-1.306(.095)	-3.883*(.000)	-1.045(.147)	-2.987*(.002)
SOX	-3.920*(.000)	-3.024*(.001)	-3.285*(.000)	-1.008(.156)	-3.920*(.000)	-0.224(.826)

*p-value < 0.05。

肆、實驗結果與討論

本研究提出一個新的預測方法－結合新型態球型複數模糊集。球型複數模糊集比傳統模糊集有著較高的延展性，其複數型態的歸屬程度，可容納較多的歸屬資訊。本研究將其應用於模糊推論系統上，使模型可根據需求產生多輸出值，以進行多目標預測。在訓練資料進入模型前，進行多目標特徵選取，選出對預測目標具有影響力之特徵作為模型輸入。然而，過多的模型輸入對於模糊推論系統，會導致訓練後鑑部之參數上，計算時間的增加。因此，本研究將 CNN 加入模型中，透過此架構容納更多有影響力之特徵，並識別特徵中趨勢的變化，萃取有用資訊。在機器學習算法方面，使用複合式機器學習算法，GD-WOA 訓練 CNN 的

權重及球型複數模糊集的參數；RLSE 訓練後鑑部之參數。透過分而治之的方法，除了能減少個別演算法所負責模型參數之數量，也針對不同的架構，使用合適的演算法，增進學習效率。為驗證性能，本研究進行三個不同目標數的時間序列預測實驗，分別為單目標、雙目標和四目標。實驗結果顯示，本研究所提之方法在時間序列預測上有良好效能。

實驗一為單目標預測，使用的資料集分別為 2000 年至 2004 年，共五年之 TAIEX 資料。從表 2 的實驗結果可觀察到，九個不同 CNN-SCNFS 模型之預測能力皆比 SCNFS 良好，顯示加入 CNN 架構於 SCNFS，能提升模型的預測能力。對於不同 CNN 架構，在卷積層的數量方面，含有三層卷積層之 CNN-SCNFS，其最佳 RMSE 為 79.58，較兩層卷積層之 CNN-SCNFS 的 82.20 及一層卷積層之 CNN-SCNFS 的 83.23 佳。這顯示，多層 CNN 架構能有效萃取特徵，提升模型的預測能力；在卷積核的數量方面，則無法發現明顯差異及規則。此外，在九個不同測試模型中，CNN(3,2)-SCNFS 之預測表現最佳，其預測結果與誤差如圖 3 至圖 7 所示。因此，後續實驗皆以 CNN(3,2)-SCNFS 作為與其他文獻方法比較之模型。在表 3 中，與其他文獻提出之方法相比下，本研究所提之方法表現優於多數比較方法，與 Yolcu 與 Alpaslan（2018）提出之方法，則擁有相當的預測表現。透過實驗一，本研究所提之方法的基礎可行性得到驗證，同時，也驗證加入 CNN 架構，能提高模型預測能力。

實驗二及實驗三之目的為驗證模型對多目標預測能力。實驗二為雙目標預測，分別以 BVSP 及 Nikkei 225 作為預測目標。從圖 8 的結果顯示，本研究所提之方法在雙目標預測上，同樣具有良好的預測能力。從表 5 可觀察到，對於 BVSP，本研究所提之方法在 RMSE 及 RMSPE 的預測表現上，優於所有比較方法。對於 Nikkei 225，本研究所提之方法優於傳統方法 ARIMA、SVR、ANFIS，與其他混合方法相比，也擁有相當的表現。實驗三為四目標預測，分別以 DIJA、NASDAQ、S&P500 及 SOX 進行預測。當目標數提升至四目標時，模型同樣保持良好的預測能力。從表 7 的實驗結果顯示，本研究所提之方法在五年的平均表現上，除 NASDAQ 外，皆優於只進行單一目標預測的模型。對於 Tu 與 Li（2019）提出的多目標預測模型 SCFS，本研究所提之方法在四個股票的平均表現上，皆優於其提出之方法。

對於 BPNN 與本研究所提之方法，在模型訓練速度的比較上，BPNN 皆具有良好優勢，如表 2、6 及 10 所示。然而，在預測結果的表現上，在實驗一中，本研究所提方法之最佳結果及平均結果，皆優於 BPNN，如表 2 所示。在實驗二中，本研究所提之方法在四項評估指標的比較中，皆優於 BPNN，如表 7 所示。在實驗三中，本研究所提之方法在四個資料集的五年平均表現，同樣優於 BPNN。總結上述，針對時間序列的預測問題，相較 BPNN，本研究所提方法之

預測表現較佳。

綜合上述三個實驗的結果，可驗證本研究所提之方法，在對應不同目標數的時間序列預測上，擁有良好的預測能力及適應性。

伍、結論

在數據量遽增的時代下，資料分析的重要程度越趨提升。面對如此龐大的資訊量，設計一個高效能的計算模型勢在必行。本研究以資訊熵為基礎的多目標特徵挑選，從資料中挑選有影響力之特徵作為模型輸入，並提出一個多目標預測模型 CNN-SCNFS，進行時間序列的多目標預測，該模型結合球型複數模糊集、CNN 及模糊推論系統等理論。訓練參數方面，使用複合式機器學習演算法 GD-WOA-RLSE，分別針對CNN、球型複數模糊集及規則後鑑部之參數進行學習。經由實驗證明，可統整本研究之貢獻如下：一、加入CNN架構，可容納更多有影響力之特徵並從中萃取出有用資訊，除了提升模型預測能力，也防止因過多模型輸入而使後鑑部參數在訓練上計算資源的增加。二、球型複數模糊集可使模糊推論系統在不增加訓練參數的情況下，根據目標數產生多個複數型態的歸屬程度，提高模型的靈活性及適應性。與過去預測單目標之方法相比，能夠預測多目標之模型將更具有競爭力。三、使用複合式最佳化演算法 GD-WOA-RLSE 針對模型訓練參數進行訓練，透過分而治之的概念，增加模型的學習效能。綜合上述，本研究所提之方法在股票的時間序列預測上有良好的效能。

儘管如此，本研究所提之方法仍有部分細節能夠在未來延伸。以下提出未來可延續的研究方向：一、本研究使用之資料集包含開盤價、最高價、最低價及收盤價，然而，任何數據都可能帶有未被發現但有用的資訊。因此，未來研究可考量其他資料集作為模型輸入，如經濟或技術指標等，使模型有更好的預測精準度。二、本研究使用CNN之架構設定是基於經驗法則。若能根據資料的特性，利用相關的機器學習演算法決定架構設定，除了能減少人為主觀設定，達到自組織（Self-organize）的特性外，合適的架構更能有效提升模型的預測表現。三、本研究中，SCNFS 是以規則為基礎的模型，一前鑑部建立一後鑑部。若前鑑部數量過多，建立的後鑑部數量也會隨之龐大。如此會增加需要訓練的參數數量，在模型訓練上，是個龐大的負擔。因此，未來可朝向將前鑑部與後鑑部的建立方式分離，使用目標資料集建構後鑑部，降低後鑑部與前鑑部之間的關聯性，形成非對稱式模型，使模型建立更為彈性。

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。

參考文獻

Atsalakis, G.S. and Valavanis, K.P. (2009), 'Forecasting stock market short-term trends using a neuro-fuzzy based methodology', *Expert Systems with Applications*, Vol. 36, No. 7, pp. 10696-10707.

Bagheri, A., Peyhani, H.M. and Akbari, M. (2014), 'Financial forecasting using ANFIS networks with quantum-behaved particle swarm optimization', *Expert Systems with Applications*, Vol. 41, No. 14, pp. 6235-6250.

Bollerslev, T. (1986), 'Generalized autoregressive conditional heteroskedasticity', *Journal of Econometrics*, Vol. 31, No. 3, pp. 307-327.

Box, G.E.P. and Jenkins, G. (1990), *Time Series Analysis, Forecasting and Control*, Holden-Day, Inc.

Boyacioglu, M.A. and Avci, D. (2010), 'An adaptive network-based fuzzy inference system (ANFIS) for the prediction of stock market return: The case of the Istanbul stock exchange', *Expert Systems with Applications*, Vol. 37, No. 12, pp. 7908-7912.

Cai, Q., Zhang, D., Wu, B. and Leung, S.C.H. (2013), 'A novel stock forecasting model based on fuzzy time series and genetic algorithm', *Procedia Computer Science*, Vol. 18, pp. 1155-1162.

Cai, Q., Zhang, D., Zheng, W. and Leung, S.C.H. (2015), 'A new fuzzy time series forecasting model combined with ant colony optimization and auto-regression', *Knowledge-Based Systems*, Vol. 74 , pp. 61-68.

Chen, S.-M. (1996), 'Forecasting enrollments based on fuzzy time series', *Fuzzy Sets and Systems*, Vol. 81, No. 3, pp. 311-319.

Chen, S.-M. and Chang, Y.-C. (2010), 'Multi-variable fuzzy forecasting based on fuzzy clustering and fuzzy rule interpolation techniques', *Information Sciences*, Vol. 180, No. 24, pp. 4772-4783.

Chen, S-M. and Chen, C-D. (2011), 'TAIEX forecasting based on fuzzy time series and fuzzy variation groups', *IEEE Transactions on Fuzzy Systems*, Vol. 19, No. 1, pp. 1-12.

Chen, S-M. and Chen, S-W. (2015), 'Fuzzy forecasting based on two-factors second-order fuzzy-trend logical relationship groups and the probabilities of trends of fuzzy logical relationships', *IEEE Transactions on Cybernetics*, Vol. 45, No. 3, pp. 391-403.

Chen, S-M., Chu, H-P. and Sheu, T-W. (2012), 'TAIEX forecasting using fuzzy time series and automatically generated weights of multiple factors', *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, Vol. 42, No. 6, pp. 1485-1495.

Chen, S-M., Manalu, G.M.T., Pan, J-S. and Liu, H-C. (2013), 'Fuzzy forecasting based on two-factors second-order fuzzy-trend logical relationship groups and particle swarm optimization techniques', *IEEE Transactions on Cybernetics*, Vol. 43, No. 3, pp. 1102-1117.

Cheng, S.-H., Chen, S.-M. and Jian, W.-S. (2016), 'Fuzzy time series forecasting based on fuzzy logical relationships and similarity measures', *Information Sciences*, Vol. 327, pp. 272-287.

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

Egrioglu, E., Aladag, C.H. and Yolcu, U. (2013), 'Fuzzy time series forecasting with a novel hybrid approach combining fuzzy c-means and neural networks', *Expert Systems with Applications*, Vol. 40, No. 3, pp. 854-857.

Engle, R.F. (1982), 'Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation', *Econometrica*, Vol. 50, No. 4, pp. 987-1007.

Fischer, T. and Krauss, C. (2018), 'Deep learning with long short-term memory networks for financial market predictions', *European Journal of Operational Research*, Vol. 270, No. 2, pp. 654-669.

Goldberg, D.E. (1989), *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley Longman Publishing Co., Inc.

Guresen, E., Kayakutlu, G. and Daim, T.U. (2011), 'Using artificial neural network models in stock market index prediction', *Expert Systems with Applications*, Vol. 38, No. 8, pp. 10389-10397.

Huarng, K. (2001), 'Effective lengths of intervals to improve forecasting in fuzzy time series', *Fuzzy Sets and Systems*, Vol. 123, No. 3, pp. 387-394.

Huarng, K-H., Yu, T.H-K. and Hsu, Y.W. (2007), 'A multivariate heuristic model for fuzzy time-series forecasting', *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, Vol. 37, No. 4, pp. 836-846.

Jang, J.-S.R. and Sun, C.-T. (1997), 'Neuro-fuzzy and soft computing: A computational approach to learning and machine intelligence', Upper Saddle River, NJ, USA: Prentice-Hall, Inc. Vol. 42, No. 10, pp. 1482-1484.

Kao, L.-J., Chiu, C.-C., Lu, C.-J. and Chang, C.-H. (2013), 'A hybrid approach by

integrating wavelet-based feature extraction with MARS and SVR for stock index forecasting', *Decision Support Systems*, Vol. 54, No. 3, pp. 1228-1244.

Kennedy, J. and Eberhart, R. (1995, 27 Nov.-1 Dec. 1995), 'Particle swarm optimization', Paper presented at the Proceedings of ICNN'95-International Conference on Neural Networks. pp. 1942-1948,

Kim, K.-j. and Han, I. (2000), 'Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index', *Expert Systems with Applications*, Vol. 19, No. 2, pp. 125-132.

Kimoto, T., Asakawa, K., Yoda, M. and Takeoka, M. (1990), 'Stock market prediction system with modular neural networks', Paper presented at the IJCNN.

Li, C. and Chiang, T.W. (2013), 'Complex neuro-fuzzy ARIMA forecasting—a new approach using complex fuzzy sets', *IEEE Transactions on Fuzzy Systems*, Vol. 21, No. 3, pp. 567-584.

Li, C. and Wu, T. (2011), 'Adaptive fuzzy approach to function approximation with PSO and RLSE', *Expert Systems with Applications*, Vol. 38, No. 10, pp. 13266-13273.

Ma, J., Zhang, G. and Lu, J. (2012), 'A method for multiple periodic factor prediction problems using complex fuzzy sets', *IEEE Transactions on Fuzzy Systems*, Vol. 20, No. 1, pp. 32-45.

Man, J.Y., Chen, Z. and Dick, S. (2007, 24-27 June 2007), 'Towards inductive learning of complex fuzzy inference systems', Paper presented at the NAFIPS 2007-2007 Annual Meeting of the North American Fuzzy Information Processing Society.

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

Ramot, D., Friedman, M., Langholz, G. and Kandel, A. (2003), 'Complex fuzzy logic', *IEEE Transactions on Fuzzy Systems*, Vol. 11, No. 4, pp. 450-461.

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.

Rout, A.K., Dash, P.K., Dash, R. and Bisoi, R. (2017), 'Forecasting financial time series using a low complexity recurrent neural network and evolutionary learning approach', *Journal of King Saud University - Computer and Information Sciences*, Vol. 29, No. 4, pp. 536-552.

Rumelhart, D.E., Hinton, G.E. and Williams, R.J. (1986), 'Learning representations by back-propagating errors', *Nature*, *323*(6088), 533-536.

Selvin, S., Vinayakumar, R., Gopalakrishnan, E.A., Menon, V.K. and Soman, K.P. (2017), 'Stock price prediction using LSTM, RNN and CNN-sliding window model', Paper

presented at the 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI),

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

Tu, C.-H. and Li, C. (2017), '*A novel entropy-based approach to feature selection*', Paper presented at the Intelligent Information and Database Systems, Cham, pp. 445-454.

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.-J., Wang, J.-Z., Zhang, Z.-G. and Guo, S.-P. (2012), 'Stock index forecasting based on a hybrid model', *Omega*, Vol. 40, No. 6, pp. 758-766.

Wang, P. and Li, C. (2019), 'Gaussian distribution based whale optimization algorithm for high-dimensional optimization', *to be submitted for publication*.

Wei, L.-Y. (2016), 'A hybrid ANFIS model based on empirical mode decomposition for stock time series forecasting', *Applied Soft Computing*, Vol. 42, pp. 368-376.

Wu, J., Long, J. and Liu, M. (2015), 'Evolving RBF neural networks for rainfall prediction using hybrid particle swarm optimization and genetic algorithm', *Neurocomputing*, Vol. 148, pp. 136-142.

Ye, F., Zhang, L., Zhang, D., Fujita, H. and Gong, Z. (2016), 'A novel forecasting method based on multi-order fuzzy time series and technical analysis', *Information Sciences*, Vol. 367-368, pp. 41-57.

Yolcu, O.C. and Alpaslan, F. (2018), 'Prediction of TAIEX based on hybrid fuzzy time series model with single optimization process', *Applied Soft Computing*, Vol. 66, pp. 18-33.

Yu, T.H-K. and Huarng, K-H. (2008), 'A bivariate fuzzy time series model to forecast the TAIEX', *Expert Systems with Applications*, Vol. 34, No. 4, pp. 2945-2952.

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

Zhang, G.P. (2003), 'Time series forecasting using a hybrid ARIMA and neural network model', *Neurocomputing*, Vol. 50, pp. 159-175.