

台灣加權股價指數之波動預測

— 以模糊類神經及群體智慧為方法

Volatility prediction of TAIEX using Neuro-Fuzzy and Swarm Intelligence Approach

李俊賢	盧冠宇	王業富	呂學彥	林青屏
國立中央大學資 訊管理系副教授	國立中央大學資 訊管理系學生	國立中央大學資 訊管理系學生	國立中央大學資 訊管理系學生	國立中央大學資 訊管理系學生
jamesli@mgt.ncu. edu.tw	james668888@ho tmail.com	a89525993@yaho o.com.tw	abao753951@hot mail.com	jj78428509@hot mail.com

摘要

本研究利用不同的演算法來建立一個智慧型模糊類神經系統，用以預測台灣加權股價指數 (Taiwan stock exchange capitalization weighted stock index, TAIEX) 之波動 (Volatility)，再設計出四種不同的方法作訓練和預測並比較其實驗結果。本研究之智慧型系統能夠自行產生模糊規則 (IF-then rules)、有效率的訓練系統中的前鑑部參數和後鑑部參數，以準確的預測指數波動。本研究使用之演算法包含分群演算法 (Clustering method, CM)、多族群粒子演算法 (Multi-group particle swarm optimization, MGPSO) 及遞迴最小平方方法 (Recursive least-squares estimator, RLSE)。經由實驗比較，用分群演算法及 MGPSO-RLSE 演算法所設計之系統 (4 條規則) 比用經驗設計之模糊類神經系統 (9 條規則) 表現了比較好的學習收斂效率及預測準確性。

關鍵詞：模糊類神經系統，多族群粒子演算法，遞迴最小平方方法，台灣加權股價指數，時間序列波動預測。

Abstract

In this paper, we study intelligent neuro-fuzzy system (NFS) approaches to forecast the volatility of TAIEX (Taiwan stock exchange capitalization weighted stock index). For performance comparison, we design four NFS predictors using four different learning methods,

respectively. For fuzzy If-Then rules, the first two are by experience-based design while the last two are automatically generated by a clustering method (CM). A new Multi-group particle swarm optimization (MGPSO) is designed and used. For fast parameter learning, the MGPSO is also combined with the famous recursive least-squares estimator (RLSE) to form the MGPSO-RLSE method to adapt free NFS parameters. The performances by the four approaches are summarized and discussed in this paper. According to the experiment results, all the NFS predictors perform excellent performance for the volatility forecast of TAIEX. Among them, the cluster-based NFS predictor with the MGPSO-RLSE hybrid learning method shows the best performance in terms of forecast accuracy and learning convergence.

Keywords : neuro-fuzzy system, multi-group PSO, RLSE, TAIEX, time series volatility forecasting.

1.緒論

模糊類神經系統 (Neuro- Fuzzy System, NFS) [13] 就像人類的大腦，而模糊邏輯 (Fuzzy logic) 則像是人類的心智，藉由兩者間的相輔相成，便能描述人類的思考行為，而兩者皆具有歸納能力及容錯能力，即使遇到異端值，本身架構也不會有嚴重的差異，本研究利用此動機，發展一個具有學習功能的模糊類神經系統 [7]。

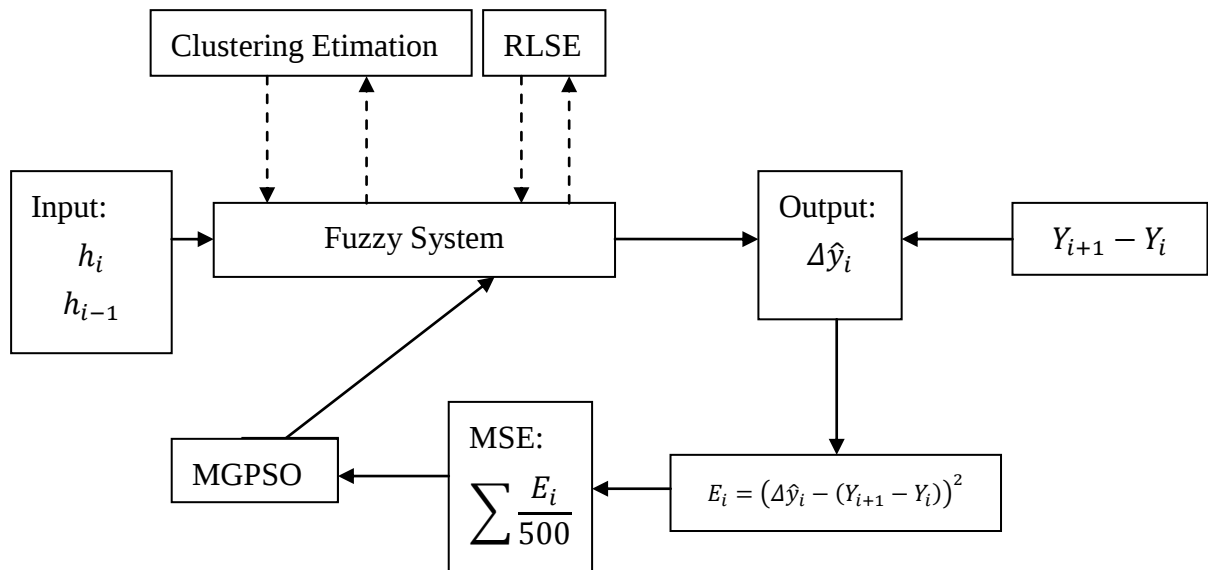


圖 1 模糊類神經系統之簡易流程圖

在 NFS 的系統當中，本研究利用 Takagi-Sugeno (T-S)[6][16]的 NFS，T-S 的 NFS 與傳統的 NFS 相當類似，唯一的差別在於 T-S 的 NFS 在後鑑部利用參數集來增加 NFS 預測結果[5][9]的準確率。

以往的研究當中，預測的準確度不盡理想。本研究進一步的對 NFS 改良，提出混合學習的方法，用分群演算法(Clustering Method, CM)、多族群粒子群演算法 (Multi-group particle swarm optimization, MGPSO)[3]和遞迴最小平方方法 (Recursive least-squares estimator, RLSE) [8][10]和來共同解決問題；分群演算法(CM)可以使系統本身根據資料的延展性來判斷 NFS 要有幾條規則，並自動產生出相對應的規則數。相較於過去的單群粒子群演算法，多族群的粒子群演算法在參數收斂的速度上快上許多，粒子也不容易陷入局部最佳解的困境。而遞迴最小平方方法(RLSE)模型可以加快解決問題的速度。

本研究利用台指作為實驗標的，台指全名即為台灣加權股票指數(Taiwan stock exchange capitalization weighted stock index, TAIEX)，是反應整體市場股票價值變動的指標[2]。台灣證券交易所採柏謝加權算式(Passche Formula)計算，(指數 = 當期總發行市值 ÷ 基值 × 100)，基期指數設為 100，其採樣樣本除了特別股、全額交割股之外，其於上市股票均包括在內。

在後半段，本研究首先介紹模糊類神經系統運作的方式，包括其設計的目的及其中所運用到的演算法都會一一介紹到，之後再將其分成四種方式來訓練，並附上圖表比較其結果。

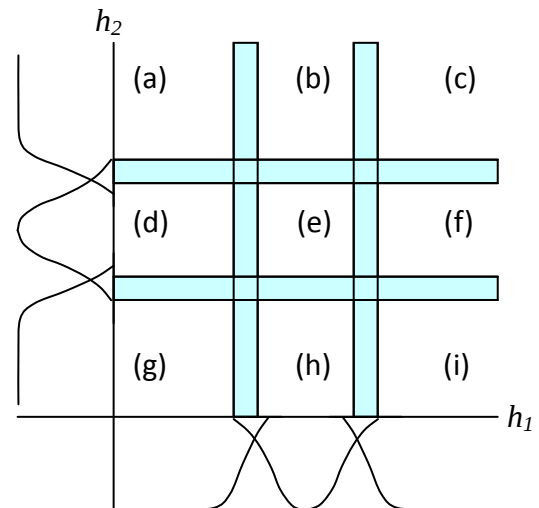


圖 2 九條規則示意圖，其中(a)~(i)為九條規則之隸屬範圍，中間藍色為重疊區域。

2. 模糊類神經系統(Neuro-Fuzzy System, NFS)運作方式介紹

在本研究中，使用自行設計之模糊類神經系統，並透過多族群粒子群演算法、遞迴最小平方方法及分群演算法，來加強本研究模糊類神經系統預測之準確程度及其運作效率，其流程圖大致如圖 1。

本研究之模糊類神經系統的運作方式主要是將一連串台指波動時間序列 $H(t)$ 中的資料輸入系統中，

$$H(t) = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_i \end{bmatrix} \quad (1)$$

h_i 為第 i 天的波動資料，在本研究中共有 800 筆資料，取前 500 筆做訓練，後 300 筆拿來做預測。

系統是採取一步預測，希望能藉由 h_i 、 h_{i-1} 的值，預測出 h_{i+1} 的值。而在訓練的過程中，我們將 h_i 、 h_{i-1} 輸入模糊類神經系統中，輸出值 $\Delta\hat{y}$ 就是第 $i+1$ 天的波動與第 i 天波動的差值，再將輸出的資料與現實的 $Y_{i+1} - Y_i$ 來做比較，之後再比較兩者計算 MSE 值，看是否有符合預測之目標，並透過多族群粒子群演算法改變前鑑部和後鑑部之參數，使其不斷的訓練已達到預測的功能。

$$MSE = \sum (\Delta\hat{y}_i - (Y_{i+1} - Y_i))^2 \quad (2)$$

在模類神經類系統的規則設定方面，有以傳統九條規則為主的模糊類神經系統，主要是由左開型、右開型、高斯型三種模糊集來構成九條規則，如圖 2 所示。而除了九條規則外，我們也有利用分群演算法來推算規則數的模糊類神經系統。

2.1 分群演算法(Clustering Method, CM)

分群演算法主要是讓模糊類神經系統能夠透過公式，自行產生規則，不同於先前所使用九條規則的模糊類神經系統，我們讓模糊類神經系統自行產生恰當的規則，相較於制式的使用九條規則，規則的產生方式及原因較為合理，同時也可省去設置前鑑部之參數，加速系統的運行效率。

其中分群的依據來自於系統的輸入值，系統主要是輸入 h_i 、 h_{i-1} ，並預測 ΔY 的值，因此我們的輸入值為 h_i 、 h_{i-1} 所構成的二維輸入值，所以透過此二維圖形上輸入值的分布來分群。首先利用下列公式(3)算出各個粒子初始的潛在值。

$$P_1 = \sum_{j=1}^n e^{-a\|x_i - x_j\|^2} \quad (3)$$

$$\beta = \frac{4}{r_b^2} \quad (4)$$

其中 r_a 是一個判斷常數，主要是用來設定一個半徑的範圍，一個粒子的潛在值高低決定於其周遭粒子的密集程度，離它越近對於潛在

值的影響就越大，越遠影響則就越小，而在這半徑之外的粒子其影響可以說是微乎其微。當各粒子的潛在值決定之後，我們便選擇潛在值最高的粒子當做第一個分群中心。接著改變各粒子的潛在值，利用下面的公式(5)，其中 x_1^* 為第一個分群中心在二維圖形中的位置， x_k^* 為第 k 個理想分群中心在二維圖形中的位置。

$$P_i \leftarrow P_i - P_1^* e^{-\beta\|x_i - x_1^*\|^2} \quad (5)$$

$$\beta = \frac{4}{r_b^2} \quad (6)$$

公式中的 r_a 和 r_b 類似，但為了避免第一個分群中心中的粒子成為下一個粒子，因此通常 r_b 設的值要略大於 r_a ，根據文獻參考，理想的值是設為 1.5 倍的 r_a 。而在改變潛在值後，依然選取潛在值最高的粒子為第二個分群中心的理想粒子，經過下面的判斷式來決定是某要列為分群中心。

If $P_k^* > \bar{\epsilon}P_1^*$
 讓 x_k^* 成為分群中心，並且繼續尋找下一個分群中心。

Else if $P_k^* < \bar{\epsilon}P_1^*$
 不將 x_k^* 設為分群中心，並且終止尋找分群中心的程序。

Else
 讓 $d_{min} = x_k^*$ 和目前所有分群中心中，距離最短的一個。

If $\frac{d_{min}}{r_a} + \frac{P_k^*}{P_1^*} \geq 1$
 讓 x_k^* 成為分群中心，並且繼續尋找下一個分群中心。

Else
 不將 x_k^* 設為分群中心，為了避免再找到它，將 x_k^* 的潛在值設為 0，並尋找潛在值次高的點 x_k^* 做為新一個分群中心的選擇，再利用判斷式判定一次。

End if

End if

上述判斷式中， P_1 為第一個分群中心的潛在值， P_k^* 為第 k 個分群中心的潛在值，其中 $\bar{\epsilon}$ 是一個判斷值，若潛在值比 $\bar{\epsilon}$ 倍的 P_k 倍高，便能成為分群中心；而 $\bar{\epsilon}$ 則為另外一個判斷值，若潛在值比 P_k 的 $\underline{\epsilon}$ 倍低，則不讓它成為分群中心。根據理論將 $\bar{\epsilon}$ 設為 0.5， $\underline{\epsilon}$ 設為 0.15。而中間的未區分的部分，則用另一個判斷標準 $\frac{d_{min}}{r_a} + \frac{P_k^*}{P_1^*} \geq 1$ 來推斷，其公式的前半段主要是看新的分群

中心和之前的中心距離是否足夠大，愈大則判斷式大於 1 的機會愈高；而後半段則是看新的分群中心的潛在值和第一個核心的潛在值的比值，也是看新的潛在值是否足夠大，愈大則判斷式大於 1 的機會愈高。若整個判斷式大於 1，就能夠成為分群中心，讓就算潛在值沒大於 ε 倍的 P_k ，也能夠因為距離其它分群中心夠遠而成為新的分群中心。

再利用下面的公式(7)中，用來計算各粒子相對於各個分群中心的歸屬程度，判斷其屬於哪一個分群中心。不同於以往分群演算法中有特定公式能算隸屬程度，本研究利用高斯函數來產生隸屬程度，核心為分群中心， σ 則利用每條規則裡面的粒子來計算。之後便可計算各個規則的隸屬程度 β ，以帶入 T-S if-then rules 中的後鑑部裡(8)，來計算各個規則的輸出值，其中 a_0, a_1, a_2 為係數。

$$u_i = e^{-\alpha \|y - y_i^*\|^2} \quad (7)$$

$$y_{i+1} = u_i * (a_0 + a_1 h_i + a_2 h_{i-1}) \quad (8)$$

而在計算 a_0, a_1, a_2 三個係數時，本研究分別使用兩種方式來計算，一種是直接利用多族群粒子群演算法來疊代，另一種則是使用遞迴最小平方法。

2.2 遞迴最小平方法 (Recursive least-squares estimator, RLSE)

本研究利用遞迴最小平方方法的方程式，在資料進入模糊類神經系統前，利用 500 筆台指資料做訓練，利用訓練出來的參數來當作模糊類神經系統中後鑑部參數。

$$\begin{bmatrix} 1 & h_i & h_{i-1} \end{bmatrix} \times \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = [Y_{i+1}] \quad (9)$$

台指的 500 筆訓練資料，我們利用第 i 天的值和第 $i-1$ 天的值帶入模糊類神經系統中規則 1 到規則 9，計算出來的 9 個輸出值加總，結果為預測之 $Y_{i+1} - Y_i$ ；利用每一次的訓練，藉由我們的方程式，係數會隨著時間做改變，最後會得到後鑑部的參數，使得 MSE 值最小。

以下是遞迴最小平方方法的公式推演，而由它的前身 Least-squares estimator，可以得到下列公式：

$$\theta_k = (A^T A)^{-1} A^T Y \quad (10)$$

假設現在是第 K 筆資料，列(row)維度的 A 和 Y 的下標設為 K ， K 也被視為測量時間內所

有資料對的順序變數，假設現在有一筆新的資料對要進入變數，則他的順序變數為 $(m+1)$ 第 $m+1$ 筆，換句話說，我們要利用這筆新資料對 $(a^T: y)$ 來更新我們的 θ_k 變成 θ_{k+1} 。現在我們可以明顯地表示 θ_{k+1} ：

$$\theta_{k+1} = \left(\begin{bmatrix} A \\ a^T \end{bmatrix}^T \begin{bmatrix} A \\ a^T \end{bmatrix} \right)^{-1} \begin{bmatrix} A \\ a^T \end{bmatrix} \begin{bmatrix} Y \\ y \end{bmatrix} \quad (11)$$

因此， θ_{k+1} 能被表示成原有的 θ_k 加上一個新的資料對 $(a^T: y)$ ，這個新的資料對又稱為 adaptation gain vector，經過推導之後：

$$P_{k+1} = P_k - \frac{P_k a_{k+1} a_{k+1}^T P_k}{1 + a_{k+1}^T P_k a_{k+1}} \quad (12)$$

$$\theta_{k+1} = \theta_k + \frac{P_{k+1} a_{k+1} (y_{k+1} - a_{k+1}^T \theta_k)}{(y_{k+1} - a_{k+1}^T \theta_k)} \quad (13)$$

並且令：

$$P_0 = aI \quad (14)$$

a 為一很大的數字， I 是單位矩陣， θ_0 為 0 矩陣。

2.3 多族群的粒子群演算法 (Multi-group particle swarm optimization, MGPSPSO)

本研究使用多族群粒子群演算法來改變前鑑部或後鑑部的參數，而此不同於以往單群粒子群演算法的地方，主要在於將粒子群演算法的群數提高，以提升搜尋的準確度，並且改變粒子移動速度公式中，除了各粒子最佳解 $Pbest_i$ 、分群最佳解 $Gbest_i$ 外，還加上了所有分群最佳化中最好的，也就是全域最佳解 $Wbest_i$ ，使得不同族群的粒子彼此之間也能相互影響。

在一開始先隨機產生分群的粒子，每群的粒子數是固定的，而各個粒子的位置即為 x_i ，其中包含了模糊類神經系統所需用到的所有係數，並隨機產生各粒子的速度 V_i 。而改變粒子速度的因素，除了自身經驗中最好的位置 $Pbest_i$ 之外，還有群體最好位置 $Gbest_i$ ，以及所有粒子中最好的 $Wbest_i$ ，透過計算粒子自身位置和上述三個點的距離向量，我們便可將速度改變。

將第 i 個粒子的初始值設為各自的 $Pbest_i$ ；再從各群的粒子中，選出係數帶入系統後，MSE 值最小的粒子即為 $Gbest_i$ ；最後再由 $Gbest_i$ 中選出最好的並設為 $Wbest_i$ ，再透過下面的公式改變各粒子的初速度，並改變粒子位置，使它們能夠不斷的在區域內搜尋最佳解。

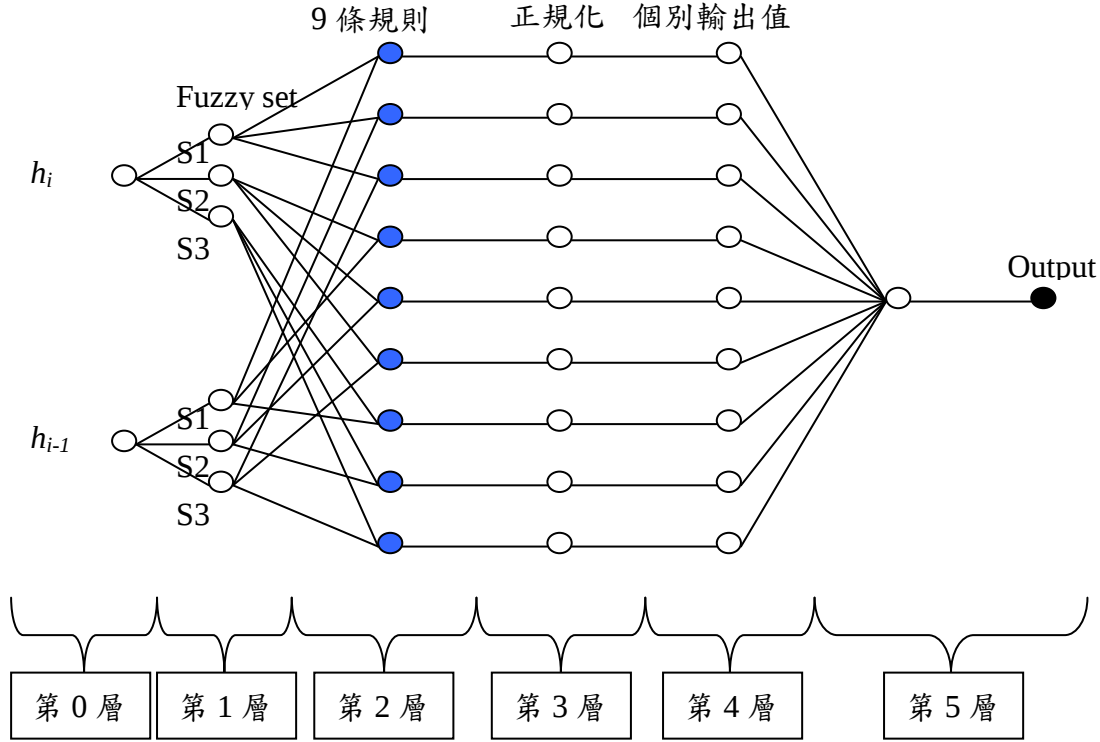


圖 3 模糊類神經系統分層構造

$$V_i(t) = V_i(t-1)W + c_1 \times \text{rand}_1() \times (Pbest_i - X_i) + c_2 \times \text{rand}_2() \times (Gbest_i - X_i) + c_3 \times \text{rand}_3() \times (Wbest_i - X_i) \quad (15)$$

$$X_i(t) = V_i(t) + X_i(t-1) \quad (16)$$

在改變各粒子速度跟位置之後，再次將各粒子代入模糊類神經系統中並與 MSE 值作比對，並重新改變 $Pbest_i$ 、 $Gbest_i$ 及 $Wbest_i$ ，不斷的迭代，直到其 MSE 值小於我們的要求或是當迭代次數到達上限為止。

上述公式(15)中 rand 為隨機產生的亂數矩陣，其平均值為 0 標準差為 1，而 W 為權重，據我們的了解，在以往的研究中，通常將其設為 0.9~0.4 之間的線性遞減，以使粒子能兼具探索及開發能力，因此我們也利用公式(17)讓 W 隨著時間而遞減， T_{max} 為預設之迭代次數。

$$W(t) = \frac{(0.9-0.4) \times (T_{max}-t)}{T_{max}} + 0.9 \quad (17)$$

3.模糊類神經系統架構介紹

本研究使用 NFS 來當作預測的系統，在 If-then 規則方面，則採用 T-S 的規則來幫助計算輸出值，並透過 500 筆的台灣加權股票指數波動資料進行訓練，並且預測後 300 筆的波動，觀測是否達成預測之目的。

本研究模糊類神經系統的訓練目的，主要

是希望藉由輸入 h_i 和 h_{i-1} 的數值，來預測實際第 $i+1$ 天和第 i 天的波動差 ΔY 。因此在一開始，本研究依序將 500 筆的台指波動之時間序列 $H(t)$ ，一次輸入兩筆資料進入模糊類神經系統中，而兩筆資料各別經過 3 個模糊集計算其隸屬程度，再兩兩相乘而成為本研究所設計之九條規則。計算出 9 個規則的輸出值 β_i ，再將其正規化，算出九條規則的之個別輸出值 r_i ，此即我們所設定之模糊類神經系統的前鑑部參數。而後鑑部參數為 $a_0+a_1h_i+a_2h_{i-1}$ ，將九條規則的 r_i 分別乘上 $a_0+a_1h_i+a_2h_{i-1}$ ，最後加總即為整個模糊類神經系統的輸出值。

而本研究之 NFS 之設計方式總計分為六層，如圖 3 所示，下面就依序說明各層的功能：

第 0 層：

輸入層，主要便是輸入我們所需要之時間序列的輸入值，在此次研究中是利用連續兩天的資料來進行一步預測，因此我們便輸入 h_i 及 h_{i-1} 。

第 1 層：

模糊集層，此層的目的是將先前的輸入值，代入我們設定好之模糊集，再算出其個別的隸屬程度，此研究所設定之模糊集分別為右開型、左開型及高斯型，其公式分別如下圖 4 所示。

其中 C 為模糊集之核心， σ 為模糊集之延展程度，在 NFS 我們均將它們設為參數，讓 MGPSON 去改變並且訓練它們。

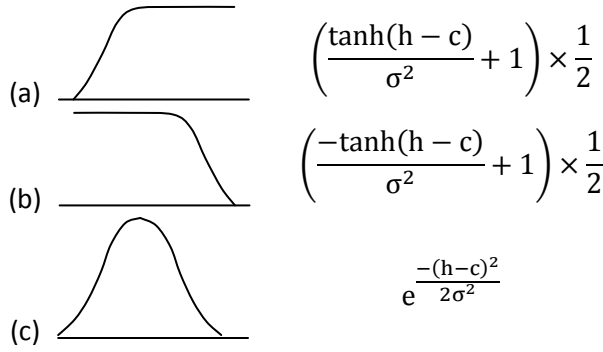


圖 4(a)~(c)各為右開型左開型及高斯型模糊集

第 2 層：

規則層，在此層中我們將上層兩個輸入值之三個模糊集隸屬程度，分別兩兩相乘，便成為九條規則的輸出值 β_i 。

第 3 層：

正規化層，此層將上述九條規則的輸出值 β_i 正規化，而正規化之公式如下。

$$r_i = \frac{\beta_i}{\sum \beta_i} \quad (18)$$

第 4 層：

結果層，此層將上層所算出的 r_i 乘上後鑑部參數的 $a_0 + a_1 h_i + a_2 h_{i-1}$ 即為九條規則的實際輸出值。

$$y_i(t) = r_i (a_0 + a_1 h_i + a_2 h_{i-1}) \quad (19)$$

第 5 層：

輸出層，最後再將九條規則的輸出值 $y_i(t)$ 相加，便是我們整個 NFS 的輸出值，整體而言我們模糊類神經系統的輸出值如下面公式所示。

$$Y_i(t) = \frac{\beta_i y_i(t)}{\sum \beta_i} \quad (20)$$

4. 實驗結果

4.1 實驗步驟

- Step1：先將台灣加權股票指數轉換成 log 指數，利用 h_{i+1} 除以 h_i 取 log 之後得到一筆 log 指數的陣列，其數值以每 50 筆為一個單位取標準差之後乘上 $\sqrt{252}$ ，就會得到以 50 為一個工作視窗的單位時間序列資料。
- Step2：將時間序列中的資料，以兩筆為一單位的方式，輸入本研究的模糊類神經系統中。

Step3：在輸入資料到模糊類神經系統中後，依 Example 不同，分別利用不同方法來決定前鑑部及後鑑部內的參數應該要如何設定。本模糊類神經系統是透過多族群粒子群演算法來進行迭代，系統中一共有 5 群各 100 個粒子，每個粒子均包含整個模糊類神經系統中所需的參數。

Step4：決定好參數設定模式後，將輸入值代入模糊類神經系統中進行運算，其輸出值為 $\Delta \hat{y}_i$ 。

Step5：將其輸出值與實際第 $i+1$ 天和第 i 天的差值 $Y_{i+1} - Y_i$ 做比較，並且透過公式產生 MSE 值。

Step6：產生 MSE 值後，將其代回多族群粒子群演算法中，視為每個粒子的適應值，以較小者為佳。

Step7：之後透過多族群粒子群演算法來改變每個粒子中的參數，再重新進行預測值的計算，直到 MSE 值達到我們所要求的標準為止。

4.2 實驗範例

而在實驗的部分，我們一共使用了 4 種方式來訓練我們的模糊類神經系統。

Example 1 採用由高斯型、左開型、右開型三種模糊及所構成的九條規則，其前鑑部模糊集的核心及 σ 、後鑑部的參數，所有的參數皆利用多族群粒子群演算法來進行迭代以及訓練。

Example 2 採用由高斯型、左開型、右開型三種模糊及所構成的九條規則，其前鑑部模糊集的核心及 σ 利用多族群粒子群演算法來進行迭代以及訓練，後鑑部的參數則使用最小遞迴平方法來進行演算。

Example 3 採用分群演算法來分群並且決定規則數目，在前鑑部方面，每條規則皆使用高斯型模糊集來計算其歸屬程度，並利用其分群核心以及各群的粒子分佈來決定高斯函數的核心及 σ ，後鑑部的參數依舊使用多族群粒子群演算法來進行迭代與訓練。

Example 4 採用分群演算法來分群並且決定規則數目，在前鑑部方面，每條規則皆使用高斯型模糊集來計算其歸屬程度，並利用多族群粒子群演算法來進行迭代以及訓練其核心與 σ ，後鑑部的參數則使用最小遞迴平方法來進行演算。

表 1 模糊類神經系統之 4 種訓練方式及數據

	訓練方法			MSE		模糊集參數	
	CM	IF-part	THEN-part	訓練數據	預測數據	Inputs	規則數
Example 1	X	MGPSO	MGPSO	5.3432×10^{-4}	3.9207×10^{-4}	h_i h_{i-1}	9
Example 2	X	MGPSO	RLSE	4.9995×10^{-4}	3.6962×10^{-4}	h_i h_{i-1}	9
Example 3	O	Fixed	MGPSO	5.3161×10^{-4}	3.9049×10^{-4}	h_i h_{i-1}	4
Example 4	O	MGPSO	RLSE	4.5601×10^{-4}	3.4913×10^{-4}	h_i h_{i-1}	4

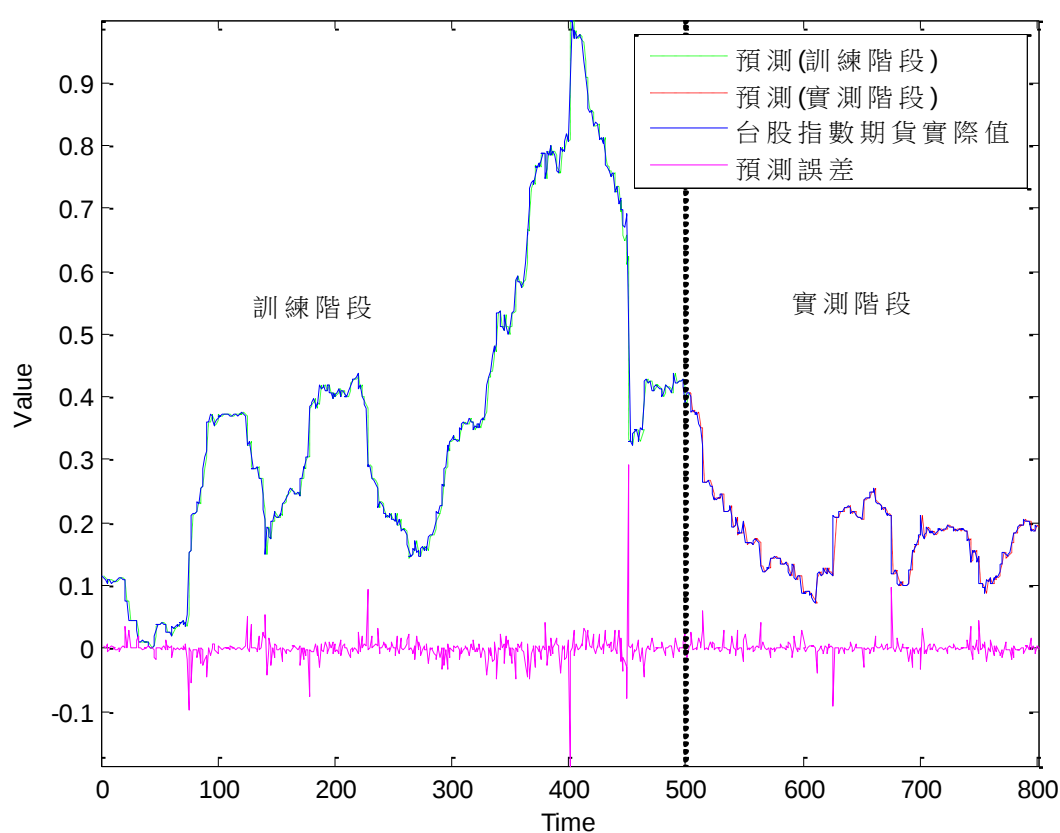


圖 5 Example 4 之預測圖形(利用分群演算法分群後，前鑑部利用多族群粒子群演算法，後鑑部利用最小平方估計法訓練)

其詳細實驗方是及結果如表 1 所示，再經過不斷的實驗過後，我們也發現參數數目的多寡對於多族群粒子群演算法的迭代速度有一定的影響，因此假如在訓練時有利用其它方式來決定參數的話，其訓練時會較為快速使誤差

值降至標準以下，圖 5 為 Example 4 的運行結果以及詳細的參數表(表 2、表 3)，可以看見我們的訓練能有效改變模糊類神經系統之參數，使其能到達到我們所期望之預測目標。

表 2 Case4 訓練前參數表

If-Part			
Fuzzy sets parameters	Center(x , y)	spread(σ_x , σ_y)	
Cluster1 (Rule1)	0.6656, -0.0598	0.7110, -0.7924	
Cluster2 (Rule2)	-0.6469, -0.2165	0.7340, 0.2296	
Cluster3 (Rule3)	-0.5226, 0.1755	-0.4071, -0.6338	
Cluster4 (Rule4)	1.7135, 0.2653	0.2355, 1.2122	
Then-Part			
Parameters	a_0	a_1	a_2
Rule1	0.0022	0.0685	-0.0726
Rule2	0.0023	0.0267	-0.0264
Rule3	0.0012	0.1476	-0.1218
Rule4	0.0069	0.0580	-0.0744

表 3 Case4 訓練後參數表

If-Part			
Fuzzy sets parameters	Center(x , y)	spread(σ_x , σ_y)	
Cluster1 (Rule1)	0.6798, 0.0115	1.2185, -2.2087	
Cluster2 (Rule2)	-0.9018, -0.2287	0.5006, -0.4935	
Cluster3 (Rule3)	-0.3024, -0.5459	-0.1670, -0.9634	
Cluster4 (Rule4)	1.8313, 0.1963	-0.5938, 0.8120	
Then-Part			
Parameters	a_0	a_1	a_2
Rule1	2.5263	-3.9628	0.1720
Rule2	5.2662	1.0611	0.6651
Rule3	0.0010	0.0265	-0.0286
Rule4	-0.7976	0.2314	-4.0283

5. 討論

本研究此次的類神經模糊類神經系統已可透過時間序列的訓練來達到預測之目的，而除了股票市場外，還有選擇權市場，選擇權 (Option) 是一種權利契約，買方支付權利金後，便有權在未來 (到期日) 依約定之履約價格，買入或賣出一定數量的約定標的物。選擇權的價格又稱為權利金，買方支付權利金換取未來報酬權利；而賣方收取權利金彌補未來可能支出義務，這雙方買權、賣權的合理價格即可用，B-S 模型，Black-Scholes 期權定價模型 (Black-Scholes Option Pricing Model) 來計算選擇權目前的價值，此模型的誕生為金融衍生商品奠定了合理定價的基礎，使得各種以市價價格變動的衍生金融工具得以合理定價，也讓國際金融市場更富有效率。至今，該模型以及它

的一些變型已被期權交易商、投資銀行、金融管理者、保險人等廣泛使用。

此模型假設如下：

1. 標的股票價格服從對數常態分配
2. 在期權有效期內，無風險利率和金融資產收益變量是不變的
3. 不計算稅收及交易成本，所有證券完全可分割
4. 期權是歐式期權，在期權到期之前不可實施
5. 證券交易是持續的
6. 不存在無風險套利機會
7. 金融資產在期權有效期內無紅利及其他所得

而模型的公式如下面所示：

$$c(0) = S_0 N(d_1) - e^{-rT} KN(d_2) \quad (21)$$

$$d_1 = \frac{\ln(S_0 \div K) + \left(r + \frac{1}{2}\sigma^2\right)T}{\sigma\sqrt{T}} \quad (22)$$

$$d_2 = d_1 - \sigma\sqrt{T} \quad (23)$$

c：買權價格

T：年化之時間

r：無風險利率

σ ：股票市場標的波動

S：股票市場標的價格

K：股票選擇權履約價格

因此透過 B-S 模型，只要擁有預測後的股票指數波動時間序列，便能程式產生預測後之買權價格時間序列，但本研究尚未將其應用在模糊類神經系統中，或許日後研究可以朝此方發展。

6. 結論

在過去，已有許多利用模糊類神經系統及多族群粒子群演算法來預測時間序列的例子，但本研究除了使用多族群粒子群演算法來增加搜尋的準確率外，還使用了遞迴最小平方法來提升運算的效率，並且使用分群演算法來提升規則建立的可靠性及合理性。

在此次的實驗結果中，雖然本研究利用的多族群粒子群演算法可以提升預測的準確率，但粒子數變多也導致疊代速度較慢，但本研究同時利用最小遞迴平法來提升系統在計算後鑑部系數時的速度，而且本研究也發現了利用分群演算法所創造的規則數，雖然少於九條，但其預測的能力並不低於傳統九條規則的模糊類神經系統，且規則數較少時，參數相對也少，程式運作上面也較為快速，因此結合出來的模糊類神經系統比以往只用單群粒子群演算法來疊代的模糊類神經系統為佳，其效率跟準確度都相對提升了。

參考文獻

- [1] 修春波，“時間序列一步預測方法”，*計算機應用研究*，第四期，2010。
- [2] 張傳章，*期貨與選擇權*，雙葉書廊，2006。
- [3] Ahmadi, A., Karray, F., Kamel, M., “Multiple Cooperating Swarms for Data Clustering,” *Swarm Intelligence Symposium*, 2007. SIS 2007. IEEE, pp.206-212.
- [4] Alam, S., Dobbie, G., Riddle, P., Naeem, M.A., “Particle Swarm Optimization Based Hierarchical Agglomerative Clustering,” *Web Intelligence and Intelligent Agent Technology (WI-IAT)*, 2010 *IEEE/WIC/ACM International Conference*, pp.64-68.
- [5] Boo, J.Y., “Stock Price forecasting using PSO-trained neural networks,” *Evolutionary Computation*, 2007. CEC 2007. *IEEE Congress*, pp.2879-2885.
- [6] Dalhoumi, L., Djemel, M., Chtourou, M., “Fuzzy predictive control based on Takagi-Sugeno model for nonlinear systems,” *Systems Signals and Devices (SSD)*, 2010 *7th International Multi-Conference*, pp.1-7.
- [7] Feng, X.Y., Wan, L.M., Liang, Y.C., Sun, Y.F., Lee, H.P., Wang, Y., “A modified particle swarm optimization for combining forecasting, Machine Learning and Cybernetics,” *Proceedings of 2004 International Conference*, vol.4, pp.2384-2389.
- [8] Gozzo, F., “Recursive least-squares sequence estimation,” *IBM Journal of Research and Development*, pp.131-156.
- [9] Huang, F.Y., “Integration of an Improved Particle Swarm Algorithm and Fuzzy Neural Network for Shanghai Stock Market Prediction,” *Power Electronics and Intelligent Transportation System, 2008. PEITS '08. Workshop*, pp.242-247.
- [10] Zhou, J., Zhu, Y.M., Li, X.R., You, Z.S., “Exactly initialized recursive least squares,” *Decision and Control, 2001. Proceedings of the 40th IEEE Conference*, vol.4, pp.3318-3323.
- [11] Jin, J.L., Cheng, L., Li, Y.Q., Wang, W.S., “Fuzzy clustering method based on set pair analysis and Optimal Fuzzy Equivalent Matrix for flood classification,” *Control and Decision Conference, 2009. CCDC '09. Chinese*, pp.5923-5927.
- [12] Chen, J.Y., Qin, Z., Jia, J., “A PSO-based subtractive clustering technique for designing RBF neural networks,” *Evolutionary Computation*, 2008. CEC 2008. pp.2047-2052.
- [13] Jang, J.S., Sun, C.T., Mizutani, E., *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*, Prentice Hall, 1996.

- [14] Zhang, L., Gao, L., Shao, X.U., "A PSO-FUZZY Group decision-making Support System in vehicle performance evaluation," **Bio-Inspired Computing, 2009. BIC-TA '09. Fourth International Conference**, pp.1-5.
- [15] Mai, V.S., Phan, X.M., "Fuzzy model predictive control using Takagi-Sugeno model," **Control, Automation and Systems, 2008. ICCAS 2008. International Conference**, pp.632- 637.
- [16] Mannle, M., "Parameter optimization for Takagi-Sugeno fuzzy models-lessons learnt," **Systems, Man, and Cybernetics, 2001 IEEE International Conference**, vol.1, pp.111-116.
- [17] Duan, Q., Wu, R., Dong, J., "Multiple swarms immune clonal quantum-behaved particle swarm optimization algorithm and the wavelet in the application of forecasting foundation settlement," **Informatics in Control, Automation and Robotics (CAR), 2010 2nd International Asia Conference** , vol.3, pp.109-112.
- [18] Stephen, L., "Fuzzy Model Identification Based on Cluster Estimation," **Journal of Intelligent and Fuzzy Systems** 2, pp.267-278,1994.
- [19] Takagi, T., Sugeno, M., "Fuzzy identification of systems and its application to modeling and control," **EEE Trans. Syst., Man Cybern**, vol.15, no.1, pp.116-132, 1985.
- [20] Jin, Y.C., Jiang, J.P., Zhu, J., "Neural network based fuzzy identification and its application to modeling and control of complex systems," **Systems, Man and Cybernetics, IEEE Transactions**, pp.990-997.
- [21] Luo, Y.X., Che, X.Y., "Chaos Immune Particle Swarm Optimization Algorithm with Hybrid Discrete Variables and its Application to Mechanical Optimization," **Intelligent Information Technology Application Workshops, 2009. IITAW '09. Third International Symposium**, pp.190-193.
- [22] Liao, Y., Wang, D., Ding F., "Data Filtering Based Recursive Least Squares Parameter Estimation for ARMAX Models," **Communications and Mobile Computing, 2009. CMC '09. WRI International Conference**, pp.331-335.