行政院國家科學委員會專題研究計畫 成果報告

基於證據累加的叢集整合技術之強韌化與功能延伸
研究成果報告(精簡版)

計 畫 類 別 ：個別型
計 畫 編 號 ：NSC 98-2221-E-009-146-
執 行 期 間 ：98 年 08 月 01 日至 99 年 07 月 31 日
執 行 單 位 ：國立交通大學資訊工程學系（所）

計畫主持人：王才沛

計畫參與人員：碩士班研究生-兼任助理人員：蘇偉誌
碩士班研究生-兼任助理人員：徐崇桂
碩士班研究生-兼任助理人員：林俞邦
碩士班研究生-兼任助理人員：魏良佑
碩士班研究生-兼任助理人員：林俞丞
碩士班研究生-兼任助理人員：邱俊予
碩士班研究生-兼任助理人員：蘇裕傑

報 告 附 件 ：出席國際會議研究心得報告及發表論文

公 開 資 訊 ：本計畫可公開查詢

中 華 民 國 99 年 10 月 31 日

行政院國家科學委員會補助專題研究計畫 ■ 成 果 報 告 □期中進度報告

基於證據累加的叢集整合技術之強韌化與功能延伸

計畫類別：■ 個別型計畫　　□ 整合型計畫
計畫編號：NSC－98-2221-E-009-146-
執行期間：2009 年　8 月　1 日至　2010 年　7 月　31 日

計畫主持人：王才沛
共同主持人：
計畫參與人員：蘇偉誌、林俞邦、徐崇桂、邱俊予、魏良佑、蘇裕傑、林俞丞

成果報告類型(依經費核定清單規定繳交)：■精簡報告　□完整報告

本成果報告包括以下應繳交之附件：
□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
■出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、列管計畫及下列情形者外，得立即公開查詢
□涉及專利或其他智慧財產權，□一年□二年後可公開查詢

執行單位：國立交通大學資訊工程學系

中　華　民　國　　99　年　　10　月　　31　日

行政院國家科學委員會專題研究計畫成果報告

基於證據累加的叢集整合技術之強韌化與功能延伸

Robustification and Functionality Extension of Evidence-Accumulation-Based Cluster Ensembles

計畫編號：NSC－98-2221-E-009-146-
執行期間：2009年8月1日 至 2010年7月31日

主持人：王才沛 國立交通大學資訊工程學系(所)

中文摘要

叢集化是一個可以在沒有分類資訊的資料當中，將相關的資料點區分成叢集的方法。叢集化演算法的種類很多，但並沒有一個方法可以對所有的資料與叢集性質都產生好的結果。叢集整合 (cluster ensemble) 技術是近年的一個新趨勢，其做法是對同一組資料產生多個不同的叢集化結果，再結合這些個別結果來產生一個具有共識的、更穩定也更能代表實際資料分佈的分群。叢集整合的優點最近已逐漸被證實，也有愈來愈多的應用出現在不同的領域。

本計畫的整體目標是以證據累加叢集法(evidence-accumulation clustering) --也就是基於co-association 矩陣的叢集整合方法--為基礎，研討改善其強韌性的方法，以及進行模糊與清晰叢集整合的比較。一方面，我們將證據累加叢集法與強韌叢集法做結合來改進其應用到雜訊環境與未知叢集數量的問題時的效能。另一分面，我們也發現更適合與強韌叢及法結合的模糊叢集整合也比清晰叢集整合具有更好的收斂性質。

關鍵詞：叢集整合、證據累加、共識叢集、強韌叢集法

Abstract

Clustering is a process that groups unlabeled data points into clusters. There are a large variety of clustering methods, but none can generate good clustering results for all types of data and cluster characteristics. Cluster ensemble is a new trend in recent years. Its approach is to generate multiple clustering results out of the same data set, and then combine the individual clustering results to form a consensus partition of the data that is more stable and more representative of the actual data distribution. As the benefits of cluster ensemble are gradually recognized in recent years, there are a growing number of applications in various fields.

The overall purpose of this two-year project is to start with evidence-accumulation clustering, that is, the clustering ensemble methods based on co-association matrices, and investigate methods that can improve its robustness as well as comparing the performances of crisp and fuzzy cluster ensembles. First, we combine evidence-accumulation clustering with robust clustering algorithms to improve its performance in problems that involve noisy data and unknown numbers of clusters. On the other hand, we also find that fuzzy cluster ensembles, which work more naturally with robust clustering methods, also exhibit better convergence characteristics

compared to crisp clustering ensembles.

Keywords：cluster ensemble, evidence accumulation, consensus clustering, robust clustering methods

一、簡介與文獻探討

叢集化 (clustering) 代表的是在一組沒有已知分類的資料當中找出其中的叢集 (cluster) 的過程。近幾年的一個趨勢是使用叢集整合 (cluster ensemble) 技術，對同一組資料產生多個不同的叢集化結果，再結合這些個別結果來產生一個具有共識的、更穩定也更能代表實際資料分佈的分群。這一類的方法也常被稱為共識叢集法 (consensus clustering)。完整的叢集整合演算法包括三個主要的部份：用來產生個別分群的方法、如何將數個個別分群用一個整合的資料結構來代表、以及如何由這個整合的資料結構來得出最終的分群。其中第二項與第三項又具有較高的關連性，因為所用的資料結構將決定可以用來得到最終分群的演算法。　早期關於叢集整合的一篇重要論文是 Strehl 與 Ghosh 在 2002 所發表 [1]，這篇論文說明了"知識再利用" (knowledge reuse) 用於叢集的概念，在此"知識"所代表的即是個別分群。

叢集整合的先決條件是要能對同一組資料產生多個不同的分群結果。在這方面一個具代表性的方法可算是使用同一個叢集演算法但是不同的隨機初始化 [2,3]；這方式並使用在許多相關研究中。另一個常見的方法是將高維度的特徵向量 (feature vectors) 投影到多個隨機的低維度次空間 (subspace) [4]，類似的方法的還包括隨機的選擇部份的特徵值來進行叢集化 [5]。這一類方法對高維度資料的叢集化特別有用，因為可以減少所需的運算量。其他的方法還包括將資料隨機分為幾個較小的資料組，並且對這些資料組作叢集化 [6]。當大量資料必須以線上 (on-line) 的方式叢集化時，也可以透過改變各資料點被處理的順序而得到不同的分群結果再做整合。在 [7] 中使用的即屬於這一類的方法。

關於叢集整合的資料結構，最常見的即是 co-association 矩陣。代表性的方法包括 [1] 所提出的 Cluster-based Similarity Partitioning Algorithm (CSPA) 以及 [2,8] 的證據累加叢集法 (Evidence-accumulation clustering; EAC)。在其他使用圖形 (graph) 為資料結構的方法包括 [1] 的 Meta-CLustering Algorithm (MCLA)，與 HyperGraph Partitioning Algorithm (HGPA)。HGPA 將叢集整合問題視為一個分割超圖形 (hypergraph) 的問題。MCLA 則建立一個圖形來代表所有個別分群中的叢集的兩兩相似度。另一個基於 bipartite 圖形的方法 (Hybrid Bipartite Graph Formulation; HBGF) 中 [9]，每個節點可以代表一個資料點或一個原叢集，而每個邊界連接的兩個節點分別代表一個資料點與一個原叢集。另一類非常不同的整合資料結構為，當使用基於離型的叢集演算法來得到個別分群時，以所有個別叢集的離型的集合作為叢集整合的資料結構 [7]。

對於得到最終分群的方法，[1] 所使用的是圖形切割 (graph cut) 的方法來得到最終的叢集。這種方式的限制包括必須事先指定叢集的個數，以及產生類似大小的最終叢集的傾向 [5]，因此不適用於真正叢集數目未知或大小可能有較大差異的情形。在已知叢集數目的條件下，k 均值 (k-means) 與相關的方法如模糊 c 均值 (fuzzy c-means; FCM) 和 k 中值 (k-modes) [2]，以及期望值最大化 (expectation-maximization; EM) [10] 都被使用過。當真正叢集數目未知時，階層聚合（單一連結 (single-link; SL) 或平均連結 (average-link; AL) 或完全連結 (complete-link; CL)）是最常用的方法 [2,4]；最終分群是由最大生命期條件 (maximum-lifetime criterion) 來決定，以得到階層聚合過程中最穩定的一個分群。

叢集整合的一個延伸是軟性叢集整合 (soft cluster ensembles)，或稱為模糊叢集整合。根據 [5]，軟性叢集整合指的是使用數個軟性的、機率式的個別分群結果作為叢集整合技術的輸入。假如叢集整合是以圖形來表示，則使用軟性的分群結果代表該提圖形的邊界是有不同比重的。在資料點與個別叢集之間的歸屬具有某種程度的不確定時，使用軟性叢集整

合可以更真實的代表該資料點的特性。兩個使用了軟性叢集整合的研究 [4,5] 都是使用來得到軟性的個別分群。在 [5]，作者比較了在 CSPA，MCLA 及 HBGF 使用原方法（即使用清晰的分群 (crisp partition) 為輸入）與使用軟性分群的效果，並且發現使用軟性分群可以得到較佳的最終分群結果。[4] 提到由軟性分群算出 co-association 矩陣的方法，但並未將結果與對應的使用清晰分群的結果做比較。[11] 亦討論了使用模糊 k 均值產生個別分群及得到最終模糊分群的模糊叢集整合方法，並分析了在不同階段進行去模糊化 (defuzzification) 的影響。我們在這部分的主要問題是分析一個之前未被處理的問題，也就是個別分群的個數對清晰與模糊叢集整合是否有不同的影響。

只有少數研究有處理雜訊/離群值對叢集整合的影響。[2] 的做法是在進行叢集化之前就先透過其他方法來辨認並去除離群值，但所用的方法假設了離群值只佔整體資料的少部分。[12] 是利用 co-association 矩陣辨認離群值的方法，但仍須由使用者指定所要去除的可能是離群值的資料點的比例，而這在真實應用中多是未知的。我們在此考慮使用強韌叢集演算法產生的分群來建立所謂的強韌叢集整合，藉以結合已有的技術來得到更實用可靠的，可在具有未知雜訊/離群值比例時得到較準確結果的叢集整合。

二、研究內容

A. 強韌叢集整合

現階段我們已針對使用強韌叢集法 (robust clustering) 的叢集整合進行初步分析。對於有雜訊或離群值的資料，若欲在叢集化過程中排除降低或排除這些雜訊或離群值對叢集化結果的影響，一般有兩個方式可以採用：第一個方式是在進行叢集化之前，先使用一個目的是在判斷哪些點是雜訊或離群值的演算法，將可能的雜訊或離群值去除後，再進行原先的叢集化演算法。原有的 EAC 演算法即是使用這個方式。

判斷哪些資料點是雜訊或離群值的第二個方式是使用強韌叢集法，也就是在進行叢集化的同時判斷各資料點是雜訊或離群值的可能性，並且隨著叢集化的結果逐漸更新，這些可能性也隨著更新。這是我們在本年度進行的計畫中所要分析的方法。已下列出我們使用雜訊叢集法 (noise clustering; NC) [13] 的一些實驗結果。雜訊叢集法在模糊 C 均值 (fuzzy c-means; FCM) 中另加一個虛擬的"雜訊叢集"。可能是雜訊的資料點將在這個雜訊叢集中有較高的歸屬程度 (membership)，因而降低了在其他"真正"的叢集當中的。我們的做法是只使用雜訊叢集之外的"真正"的叢集當中的歸屬程度來計算 co-association 矩陣。基本的雜訊叢集法由一個參數（資料點到雜訊叢集之間的距離）來控制，然而我們也可以允許每一個真正的叢集使用不同的距離。

在使用如 FCM 或類似的軟性叢集演算法 (soft clustering algorithms) 所產生的叢集來計算 co-association 矩陣時，計算每個 co-association 矩陣元素的公式為

$$s_{ij}^{*}=\frac{1}{H}\sum_{1\le h\le H}\left[\sum_{c=1}^{k_h}u_{ci}^{(h)}u_{cj}^{(h)}\right] \tag{1}$$

其中 H 是叢集整合中不同分群的總數，k_h 代表第 h ($1\le h\le H$)個分群 P_h 當中的叢集個數，而 $u_{ci}^{(h)}$ 代表在 P_h 當中，資料點 $\boldsymbol{x}_i$ 在第 c 個叢集當中的歸屬程度。FCM 的歸屬程度公式為

$$u_{cj}=\frac{1}{\sum_{q=1}^{k}\left(\frac{d_{ci}^{2}}{d_{qi}^{2}}\right)^{\frac{1}{m-1}}} \tag{2}$$

而 NC 的歸屬程度公式為

$$u_{cj}=\frac{1}{\sum_{q=1}^{k}(\frac{d_{ci}^2}{d_{qi}^2})^{\frac{1}{m-1}}+(\frac{d_{qi}^2}{\delta_c^2})^{\frac{1}{m-1}}} \tag{3}$$

在此 d_{ci} 是 $\boldsymbol{x}_i$ 到第 c 個叢集的距離。我們可看出 NC 算出的歸屬程度較小，而且減少的程度隨 d_{ci} 增加而增加。若有一資料點 $\boldsymbol{x}_i$ 離所有叢集皆很遠，則其所有歸屬程度皆較小，也將造成所有 s_{ij} 與 s_{ji} 都較小。由此推論，我們可以利用 s_{ij} 的大小來判斷 $\boldsymbol{x}_i$ 是否為離群值。一個簡單的判斷方式是依據

$$v_i=\max_j s_{ij} \tag{4}$$

值愈小則愈可能是離群值。

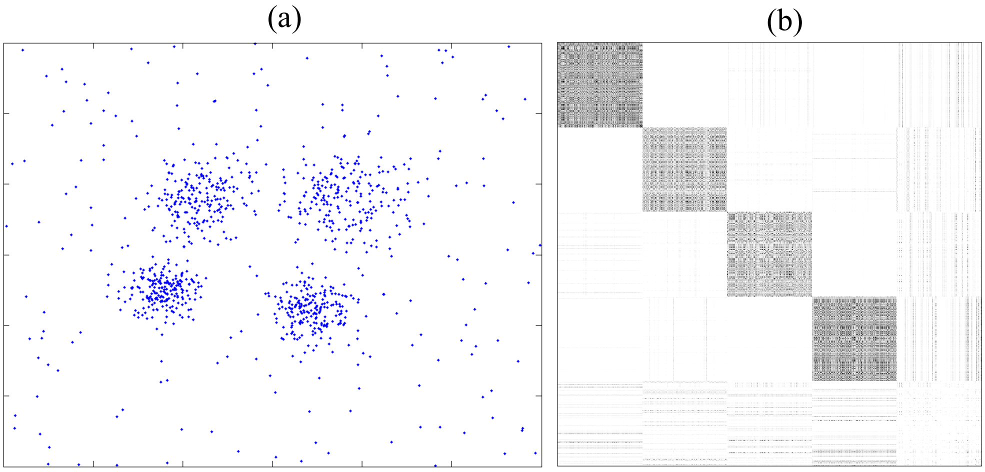
B. 清晰與模糊叢集整合的收斂速度比較

使用 EAC 的模糊叢集整合其基本步驟已在上段中描述。清晰叢集整合的差別（也就是原始 EAC 的做法）只是在於方程式(1)中的 u 的值皆為 0 或 1，取決於各資料點所屬的叢集。在這部分我們主要是測試叢集準確度（我們以 Hungarian Algorithm 作為評估準確度的方法）和叢集整合中個別分群的個數（即 H）的關係。因此我們以 EAC 的原演算法，但計算隨著 H 增加時叢集準確度的變化。

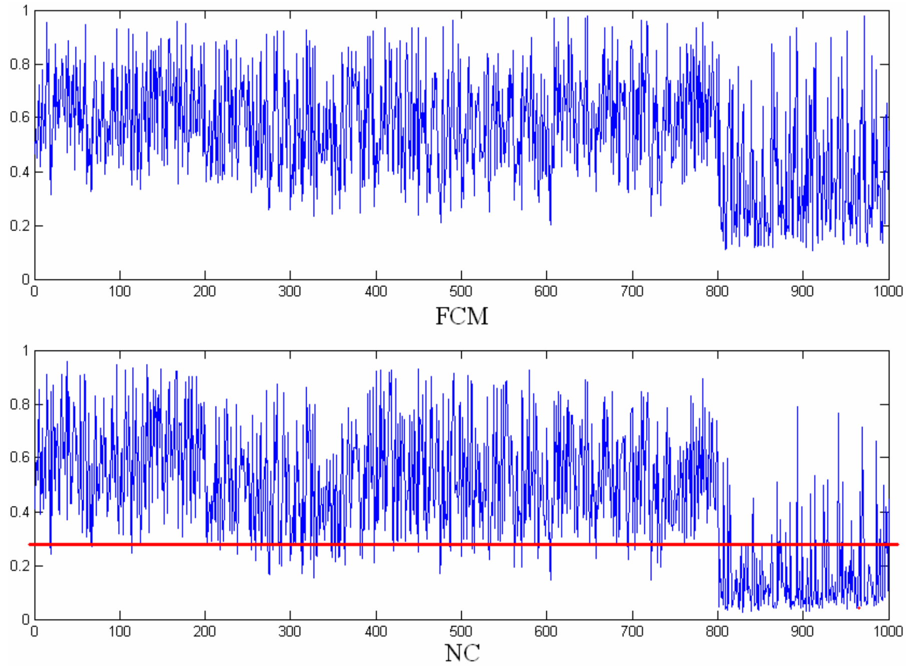
三、結果與討論

A. 強韌叢集整合

下圖一是一個我們使用 NC 於 co-association 矩陣的例子。資料包含四個高斯分佈與外加的離群值（圖一(a)），而圖一(b)是所得到的 co-association 矩陣，其四個明顯的方形即資料中的四個叢集，而右方與下方明顯值較低的部分則是外加的離群值。圖二中是根據 FCM 與 NC 的歸屬程度算出的 co-association 矩陣再依據(4)算出的 v_i。我們可以看出使用 NC 可以比單純 FCM 更清楚的將實際資料與離群值分開（圖中紅線代表一可能的臨界值）。

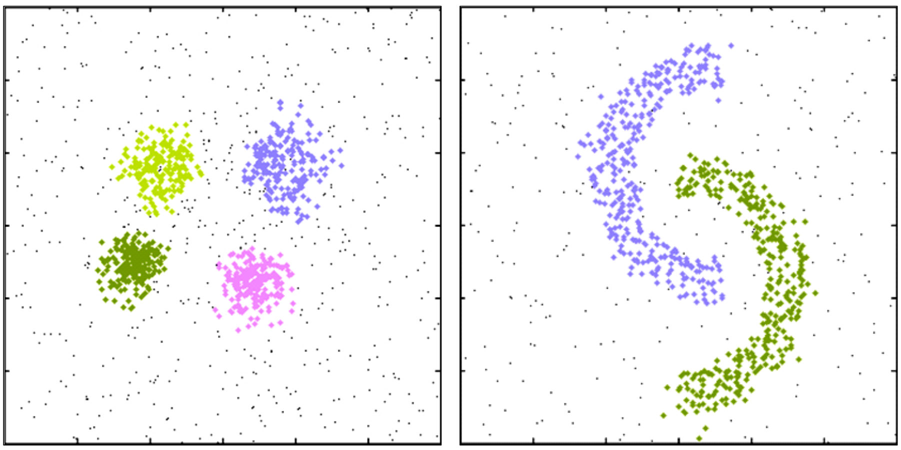


圖一：強韌叢集整合與 co-association 矩陣的例子。(a)含離群值的模擬資料；(b)使用 NC 得到的 co-association 矩陣。



圖二：使用 FCM（上圖）與 NC（下圖）來計算 v_i（公式(4)）的比較。

我們在圖三中顯示在兩個不同資料組中，用以上方法去除離群值再進行階層聚合所得到的最終分群結果。

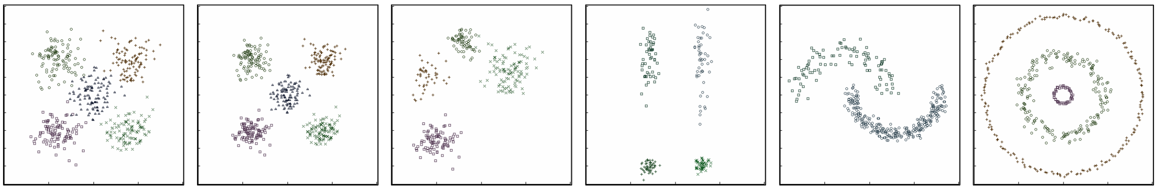


圖三：使用強韌叢集整合的叢集化結果（兩組模擬資料）。不同顏色代表最終分群中的每個叢集，而小黑點則是被判斷為離群值而未包括在最終分群中的點。

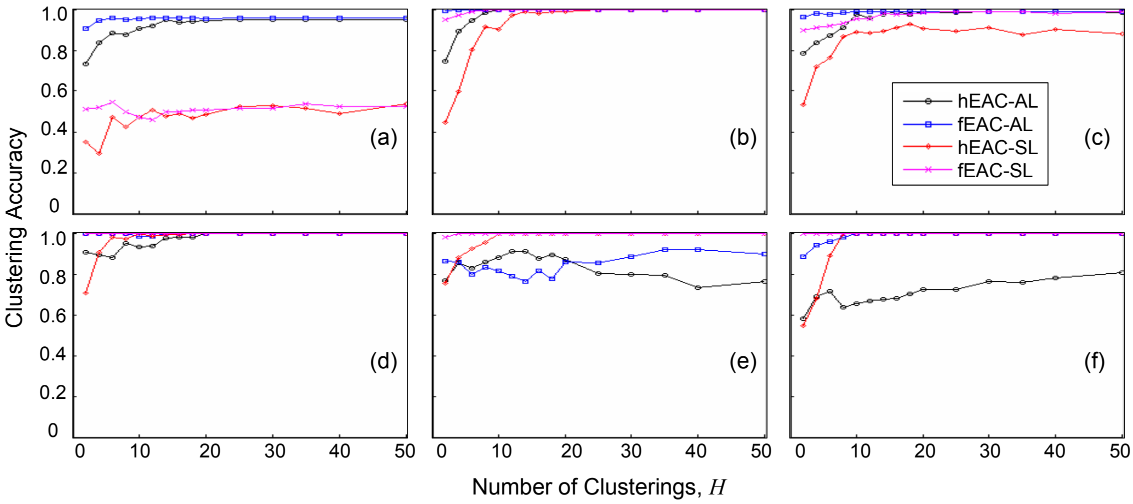
除了 NC 之外，我們也使用了可能性模糊 C 均值 (Possibilistic fuzzy c-means; PFCM)[14]，使用其所算出的兩種歸屬程度（模糊性及可能性）的乘積來計算 co-association 矩陣的方法，其結果與使用 NC 的結果類似，但 NC 的效果稍佳。

B. 清晰與模糊叢集整合的收斂速度比較

這部分我們使用的是與原EAC論文相似的合成資料。下圖四是所使用的資料組圖，而下圖五則是叢集準確率隨個別分群個數H的變化圖。其中hEAC與fEAC分別代表清晰與模糊叢集整合。我們可以清楚看出fEAC總是比hEAC更快收斂。



圖四：六個合成資料組的圖。



圖五：叢集準確率隨個別分群個數 H 的變化圖。

四、參考文獻

[1] A. Strehl and J. Ghosh "Cluster ensembles -- a knowledge reuse framework for combining multiple partitions", *J. Machine Learning Research*, vol. 3, pp. 583-617, 2002.

[2] A.L.N. Fred and A.K. Jain, "Combining multiple clusterings using evidence accumulation", *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 27, pp. 835-850, 2005.

[3] H. Luo, F. Kong, and Y. Li, "Combining multiple clusterings via k-modes algorithm", *LNAI*, vol. 4093, pp. 308 – 315, 2006.

[4] X.Z. Fern and C.E. Brodley, "Random projection for high dimensional clustering: A cluster ensemble approach", *Proc. 20th Int'l Conf. Machine Learning* (ICML), 2003.

[5] K. Punera and J. Ghosh, "Soft Cluster Ensembles", in *Advances in Fuzzy Clustering and its Applications*, Ed. J. Valente de Oliveira and W. Pedrycz (Wiley, 2007).

[6] B. Minaei-Bidgoli, A. Topchy and W. F. Punch, “Ensembles of partitions via data resampling”, *Proc. 2004 Int'l. Conf. Information Technology*, pp. 188-192, 2004.

[7] P. Viswanath and K. Jayasurya, "A fast and efficient ensemble clustering method", *Proc. 2006 Int'l Conf. Pattern Recognition* (ICPR), pp. 720-723, 2006.

[8] A.L.N. Fred and A.K. Jain, "Data clusterings using evidence accumulation", *Proc. 2002 Int'l Conf. Pattern Recognition* (ICPR), pp. 276 – 280, 2002.

[9] X.Z. Fern and C.E. Brodley, "Solving cluster ensemble problems by bipartite graph partitioning", *Proc. 21th Int'l Conf. Machine Learning* (ICML), *ACM International Conference Proceeding Series*, vol. 69, p. 36, 2004.

[10] A. Topchy, A.K. Jain, and W. Punch, "A mixture model for clustering ensembles", *Proc. SIAM Int'l Conf. Data Mining*, pp. 379-390, 2004.

[11] R. Avogadri and G. Valentini , "Ensemble clustering with a fuzzy approach", *Supervised and Unsupervised Ensemble Methods and their Applications*, pp. 49-69, Springer, 2008.

[12] Y. Hong, S. Kwong, Y. Chang, and Q. Ren, "Unsupervised data pruning for clustering of noisy data", *Knowledge-Based Systems*, vol. 21, pp. 612-616, 2008.

[13] R.N. Dave, "Characterization and detection of noise in clustering", *Pattern Recognition Letters*, vol. 12, pp. 657-664, 1991.

[14] N.R. Pal, K. Pal, J.M. Keller, and J.C. Bezdek, "A possibilistic fuzzy c-means", *IEEE Trans. Fuzzy Systems*, vol. 13, pp. 517-530, 2005.

國科會補助專題研究計畫項下出席國際學術會議心得報告

日期：99 年 10 月 31 日

計畫編號	NSC 98 － 2221 － E － 009 － 146 －		
計畫名稱	基於證據累加的叢集整合技術之強韌化與功能延伸		
出國人員姓名	王才沛	服務機構及職稱	國立交通大學資訊工程系助理教授
會議時間	98 年 8 月 20 日至 98 年 8 月 24 日	會議地點	韓國 濟州島
會議名稱	(中文)2009 國際模糊系統會議 (英文)2009 IEEE International Conference on Fuzzy Systems		
發表論文題目	(中文)比較清晰與模糊證據累加叢集法 (英文) Comparing Hard and Fuzzy C-Means for Evidence-Accumulation Clustering		

一、參加會議經過

2009/8/21 抵達濟州島

2009/8/22 返回

二、與會心得

This is a beautiful place for a conference. And this is the most important international conference in the field of fuzzy set theory, fuzzy systems, and applications. However, maybe probably due to travel issue, there are not a lot of people in the conference. Due to another conference commitment on 8/23 (CVGIP in Taiwan) I could only attend this conference for a short time.

This is the first time I'm a session chair for the whole session. I feel that I could do better in, say, time management. It was also a little awkward to transition between the roles as the session chair and a presenter. I learned that even such rarely discussed tasks of basic academic services take some experience and preparation.

三、考察參觀活動(無是項活動者略)

無

四、建議

無

五、攜回資料名稱及內容

會議論文集

六、其他

附件：於會議發表之論文

Comparing Hard and Fuzzy C-Means for Evidence-Accumulation Clustering

Tsaipei Wang, *Member*, *IEEE*

Abstract—**There exist a multitude of fuzzy clustering algorithms with well understood properties and benefits in various applications. However, there has been very little analysis on using fuzzy clustering algorithms to generate the base clusterings in cluster ensembles. This paper focuses on the comparison of using hard and fuzzy c-means algorithms in the well known evidence-accumulation framework of cluster ensembles. Our new findings include the observations that the fuzzy c-means requires much fewer base clusterings for the cluster ensemble to converge, and is more tolerant of outliers in the data. Some insights are provided regarding the observed phenomena in our experiments.**

I. Introduction

CLUSTERING is an unsupervised process of identifying underlying groups or structures in a set of patterns without the use of class labels. There have been a large set of clustering algorithms (see [1], [2] for reviews). Different methods, however, have their limitations in terms of data characteristics that can be processed and types of clusters that can be found. As it is unlikely that a good "universal" clustering method can be found, a recent trend is the use of cluster ensembles which, generally speaking, represent methods that combine the information from multiple clusterings (i.e., partitions of the data into clusters) in order to obtain a new, and hopefully better, clustering. The basic assumption of cluster ensemble here is that the combined clustering is likely to be more robust, more stable, and more representative of the structures/groupings of the data. This approach is inspired by classifier ensembles, and is identified in [3] as one of three major frontiers of clustering techniques in recent years. Some representative works in this area include [4]-[10]. Many results in these papers have indicated improved clustering results compared to the results of single clustering runs.

Ensemble clustering techniques consist of three main components:

The method used to obtain the individual clusterings (also called base clusterings in this paper), including how diversity (i.e., the differences among them) is introduced. For the individual clusterings, k-means related methods are most common, as well as EM. Diversity can be introduced through random initialization for both methods above, such as in [5] and [11]. Diversity through the projection to random subspaces is suitable for high-dimensional data [6]. Different order of data presentation is a source of diversity for on-line clustering [12]. We can also obtain each base clustering from a different subset of the data. This is a natural choice when we want to obtain a combined clustering using data from multiple sites without first putting them all in one place [13].

The representation that combines the information from multiple base clusterings. The most common form is a co-association matrix, where pairwise similarities among the patterns are derived from the individual clusterings. Other methods include hypergraphs [4] and bipartite graphs [7]. The collection of all the prototypes in all the base clusterings is very compact and especially useful when the data set is very large [13].

The extraction of the final (combined) clustering from the representation above. Here the applicable methods depend on the form of the representation. For example, graph partitioning algorithms are used for graph-based representations. For the co-association matrix, k-means based methods [14], spectral clustering [15], graph partitioning [4], and hierarchical agglomeration [5][6] have all been used.

Cluster ensembles based on the evidence-accumulation clustering (EAC) [5] have attracted a lot of attention recently, probably because it is easy to understand conceptually and to implement, and also because it is applicable to problems where the true number of clusters is unknown. The original EAC builds a co-association matrix using outcomes of multiple randomly initialized HCM runs with mostly over-specified numbers of clusters, and extracts the combined clustering using hierarchical agglomeration with single or average linkage. EAC has been extended to clustering patterns with mixed categorical and numerical features [15]. The stability of EAC with single-linkage is analyzed in [17]. However, one known drawback of EAC, as well as other methods that use co-association matrices, is the large number of base clusterings required to achieve reliable results. For example, experiments in [5] indicate that 50 or more base clusterings are usually needed to reliably identify the true number of clusters.

Most cluster ensemble techniques, including the original EAC, are based on crisp (hard) clusterings, and therefore are not able to incorporate ambiguities in the data. Some existing works that do use soft clusterings, such as [6] and [8], do not actually compare the clustering results using hard or soft base clusterings. There have also been a few applications of soft cluster ensembles (ensembles with soft base clusterings), such as [18] and [19], but these contain no comparison between corresponding crisp and soft cluster ensembles. Given the rich research findings arising from the analysis of, say, hard c-means (HCM) vs. fuzzy c-means (FCM) [16], we believe that similar analysis for clustering ensembles should

Tsaipei Wang is with the Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan (e-mail: wangts@cs.nctu.edu.tw).

be very beneficial as well. Research works in this direction have appeared only very recently. Yang, Lv, and Wang [20] studied co-association matrix based soft cluster ensembles generated with three different fuzzy similarity measures. Punera and Ghosh [11] compared the performance of ensembles of crisp or soft base clusterings obtained using EM; a crisp clustering is obtained by assigning each pattern to the most likely cluster in the corresponding soft clustering. The combination methods tested include Cluster-based Similarity Partitioning Algorithm (CSPA) [4], Meta-CLustering Algorithm (MCLA) [4], and Hybrid Bipartite Graph Formulation (HBGF) [7]. The evaluation is based on normalized mutual information (NMI) with the ground-truth partition. The conclusion in [11] is that using soft clusterings does improve the correctness of the final clusterings. Avogadri and Valentini [14] used FCM to generate the base clusterings; the corresponding hard clusterings are obtained by either alpha-cut or assigning each pattern to the most likely cluster. Their results also indicated better accuracy of fuzzy versus hard base clusterings, but only for a synthetic data set of 3 well-separated hyperspherical clusters. Both [11] and [14] study only the cases where the true number of clusters is known and used as the number of clusters for all the base clusterings as well as the combined clustering.

In this paper, we focus on the performance comparison of EAC using either HCM or FCM as the base clustering generator. These two versions are subsequently denoted as hEAC and fEAC in this paper. We specifically analyze two aspects that have not been analyzed in the literature: The speed of convergence in terms of the number of base clusterings needed to produce a stable combined clustering, and how hEAC and fEAC are affected by noise points in the data. For the remainder in this paper, we start with the description of both the crisp and soft versions of EAC, followed by the description of our experiments and results, and the conclusion.

II. Evidence-Accumulation Clustering

A. Crisp Evidence-Accumulation Clustering

Our description here follows the algorithm in [5]. Assume that the set $X=\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$ contains n patterns (data points). Let $P=\{C_1, C_2, \ldots, C_k\}$ be a crisp clustering (partition) of X. Here k is the number of clusters in P, and the clusters, C_1, C_2, …, C_k, are disjoint non-empty subsets of X, and the union of all the clusters in P is the same as X. For a data set X it is possible to obtain many different P. Let a cluster ensemble consists of N clusterings of X: P_1, P_2, …, P_N. As we allow each individual clustering to have a different number of clusters, we use k_q to represent the number of clusters in P_q ($1\leq q\leq N$).

A nxn co-association matrix, $\boldsymbol{S}^{(q)} = [s_{ij}^{(q)}]$, is computed for each clustering P_q. To determine its elements, let $c_i^{(q)}$ represent the cluster index of $\boldsymbol{x}_i$ in P_q. Then $s_{ij}^{(q)}$ is given by the following formula:

$$s_{ij}^{(q)} = \begin{cases} 1, & c_i^{(q)} = c_j^{(q)} \\ 0, & otherwise. \end{cases} \quad . \tag{1}$$

For the cluster ensemble, the overall co-association matrix, denoted as $\boldsymbol{S}^* = [s_{ij}^*]$, is simply the average of all the $\boldsymbol{S}^{(q)}$:

$$s_{ij}^* = \frac{1}{N} \sum_{1\leq q\leq N} s_{ij}^{(q)} \ . \tag{2}$$

The co-association matrix is a similarity matrix that can then be fed into various algorithms for relational data clustering. Hierarchical agglomeration with single or average linkage is the method of choice in EAC because it does not require a pre-specified number of clusters. Such algorithms generate a hierarchy of clusterings. When the number of clusters is unknown, the maximum-lifetime criterion is used to select a clustering from the hierarchy. We use k_f to represent the number of clusters in this selected clustering.

B. Soft Evidence-Accumulation Clustering

Similar to [11], here the term "soft" refers only to the base clusterings in an ensemble, meaning that these individual clusterings are soft. However, the combined clustering may still be crisp. This is the case in this paper as we follow the method in [5] and use hierarchical agglomeration to generate the combined clustering.

A soft clustering is represented by a partition (or membership) matrix $\boldsymbol{U} = [u_{ti}]$, where u_{ti} is the membership of $\boldsymbol{x}_i$ in the t^{th} cluster. For a probabilistic clustering, the membership matrix satisfies the condition

$$\forall i, \quad \sum_{t=1}^{k} u_{ti} = 1 \ . \tag{3}$$

Here k is the number of clusters. Such a clustering can be obtained with EM, which is the method used in [6] and [11], or FCM, which is the method used in [14] and in this paper.

A straightforward extension of (1) for computing co-association matrix from a membership matrix is [6]

$$s_{ij}^{(q)} = \sum_{t=1}^{k_q} u_{ti}^{(q)} u_{tj}^{(q)} \ , \tag{4}$$

or when put in matrix form,

$$\boldsymbol{S}^{(q)} = \boldsymbol{U}^{(q)} \left(\boldsymbol{U}^{(q)}\right)^T \ . \tag{5}$$

The superscripts (q) in $\boldsymbol{U}^{(q)}$ and $u_{ti}^{(q)}$ indicate that the memberships are for the q^{th} clustering within the ensemble. This form of aggregating the memberships is just the algebraic product form of t-norms in fuzzy set theory. It is used in [6] and [14], and is the method used in our experiments. Other forms of t-norms can also be used. An example is the minimum, as used in [13]:

$$s_{ij}^{(q)} = \sum_{t=1}^{k_q} \min[u_{ti}^{(q)}, u_{tj}^{(q)}] \ . \tag{6}$$

Once we have obtained the co-association matrix, the process of extracting the combined clustering is the same as the original (crisp) EAC.

III. Experiment Setup

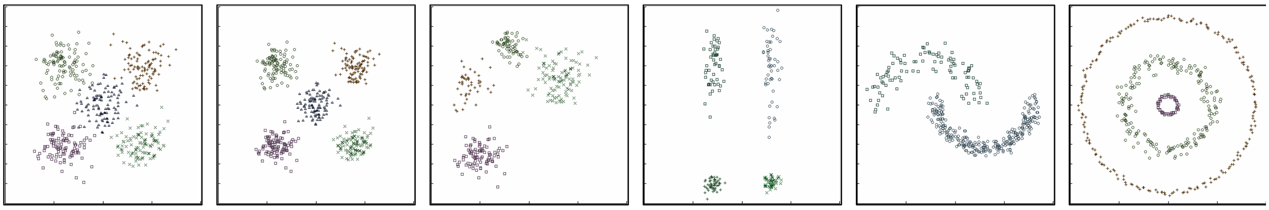


Figure 1. The 6 synthetic data sets used in our experiments. From left to right: spherical-touching (S1); spherical-separated (S2); spherical-unbalanced (S3); cigar (S4); half rings (S5); 3 rings (S6). Different colors represent the ground-truth cluster labels.

For each cluster ensemble, the co-association matrix is derived from N clusterings. The values of N range from 2 to 50. Each clustering is generated with a HCM or FCM run, with the number of clusters k in each run randomly selected from an interval [k_{min}, k_{max}]. The actual values of k_{min} and k_{max} are data-set dependent. However, we use a large range (at least a factor of three) to ensure that the results are not too sensitive to the choice of an optimal k. The HCM or FCM runs are initialized using a randomly selected subset of the data points as the initial prototypes. We use a fuzzification factor of 1.5 for the FCM runs. The experimental results in this paper are averaged over 20 ensembles.

The quality of the final clustering is evaluated by matching the final clustering with the ground-truth cluster labels of the patterns. For this purpose we use the Hungarian algorithm to find the optimal assignment (the one that results in the largest number of correctly labeled patterns) between the two sets of cluster labels. We then use the ratio of correctly labeled patterns using this optimal assignment as the clustering accuracy measure.

Several synthetic data sets are generated for the testing purpose. These data sets (shown in Fig. 1) are two-dimensional for easy visualization. In each plot, the underlying cluster labels used to generated the individual clusters are represented by separate colors. The "spherical-touching" and the "spherical-separated" data sets both have five spherical clusters with 100 points each. The "spherical-unbalanced" data sets have four touching spherical clusters of different sizes ranging from 45 to 105 points. The other three data sets are designed to be similar to those in [5]. The "cigar" data set has 4 clusters, two elongated and two compact, with 50 points each. The "half-rings" data set has two unbalanced half rings with 100 and 300 points, respectively. The "3-rings" data set has three concentric circles of 50, 200, and 200 points, respectively, from inside out. We also use S1 to S6 to refer to these 6 synthetic data sets.

In addition to the synthetic data sets, several real-world data sets from the UCI Machine Learning Repository [21] are also used in this paper:

- *Iris*: This well-known data set has 150 patterns with 4 features each, and 3 classes that represent iris families;

- *Wisconsin Breast Cancer*: 683 patterns with 9 features each, and two classes that represent benign and malignant diagnoses;

- Pen-Based Recognition of Handwritten Digits (*Pen-digits*): from the 10992 patterns with 16 features each, we only use the first 100 patterns in each of the 10 classes.

Table I gives a summary of the data sets, including the intervals [k_{min}, k_{max}] used. Here L is the data dimensionality (number of features) and k^* is the "natural" (ground-truth) number of clusters, taken as the number of classes for the real data sets, and as the number of clusters used for generating a synthetic data set.

IV. Results and Discussion

In this section we try to compare the performance of hEAC and fEAC for identifying the correct clusters when the clusters. Both single-linkage (SL) and average-linkage (AL) methods are used for cluster merging. For each ensemble, the final clustering is selected according to the maximum-lifetime criterion. In Fig. 2 we plot k_f, the number of clusters in the final clustering versus N, averaged over 20 ensembles. Results for both AL and SL are shown. In Fig. 2(e) and 2(f), the AL results are not visible because the numbers of clusters are more than 12. In addition, in order to validate the final clusterings, we also plot the clustering accuracy in Fig. 3. This is because just having the correct number of clusters (i.e., with k_f equal to k^*) does not necessarily imply a correct clustering.

From Fig. 2 and Fig. 3, we can see that the best performer in each case, in terms of both accurate clustering results and fast convergence, is either fEAC-AL (for data sets S1-S3) or fEAC-SL (for data sets S4-S6). fEAC-SL is better for S4-S6, which contain well separated non-spherical clusters, and fEAC-AL is better for the touching clusters in S1 and S3. In all these cases, this best performer (fEAC-AL or fEAC-SL) is better than its crisp counterpart for N<10 and always converges to its optimal accuracy with smaller N. Fig. 4 is similar to Fig. 3 except that for each data set, we select the combined clusterings with the number of clusters equal to k^*. Other than generally better clustering accuracy compared to Fig. 3, which is no surprise, we can also clearly see faster convergence of the best-performing fEAC version relative to

TABLE I

Summary of Data Sets

Data Set	Num. Patterns	Num. Features	k^*	[k_{min}, k_{max}]
Spherical-touching	500	2	5	[3, 12]
Spherical-separated	500	2	5	[3, 12]
Spherical-unbalanced	300	2	4	[3, 12]
Cigar	200	2	4	[5, 20]
Half rings	400	2	2	[20, 80]
3 rings	450	2	3	[20, 80]
Iris	150	4	3	[2, 10]
Breast cancer	683	9	2	[2, 10]
Pen-digits	1000	16	10	[10, 50]

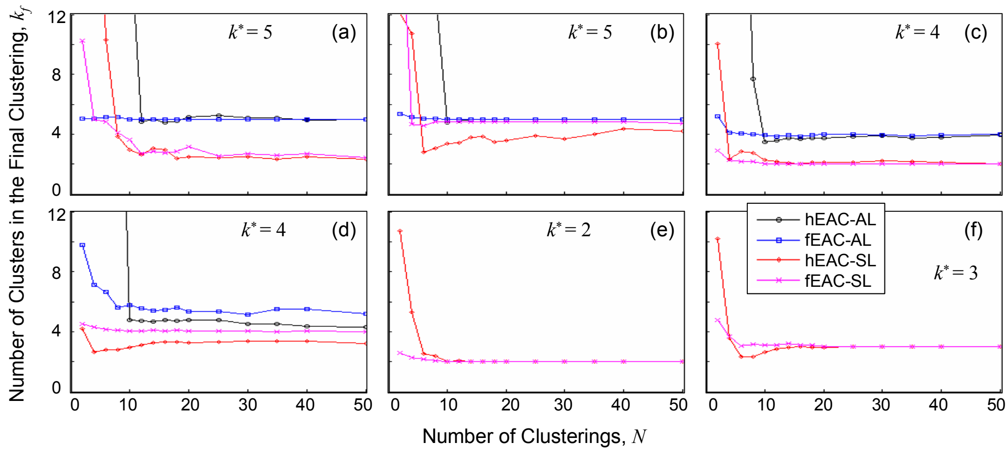


Figure 2. The number of clusters in the final clustering vs. the number of clusterings in each ensemble. (a)-(f) are plots for the synthetic data set S1-S6, respectively. The value of k^* is also indicated in each plot.

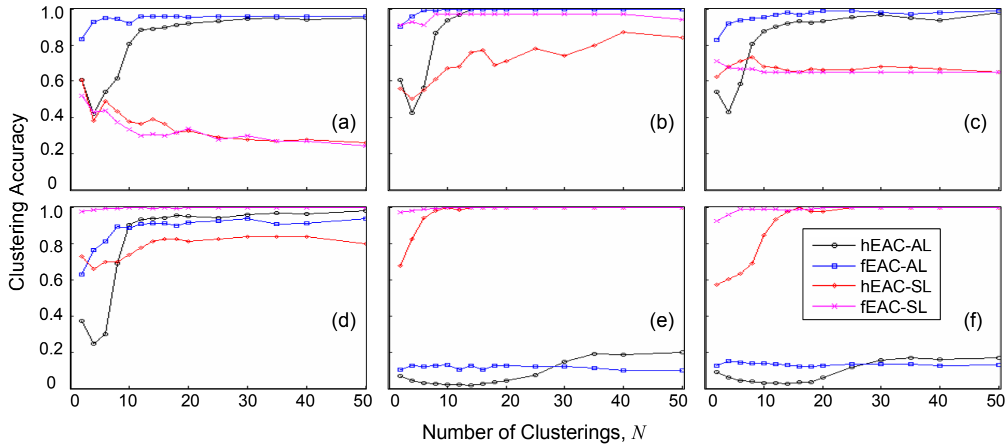


Figure 3. The clustering accuracy vs. the number of clusterings in each ensemble. The number of clusters in the final clustering is selected using the maximum-lifetime criterion. (a)-(f) are plots for the synthetic data set S1-S6, respectively.

its hEAC counterpart.

For a more quantitative comparison regarding the dependence of the performance on N, let us now consider the value of N needed for the clustering accuracy to reach 95% of its maximum value in each case. For the 12 cases here (6 data sets with combined clustering selection based on maximum lifetime or known k^*), the median N needed is 2 for the fuzzy versions and 11 for the corresponding crisp versions. It is evident then that fEAC converges much faster than hEAC with respect to N. Similar observation can be made for k_f when the maximum lifetime criterion is used. This is a clear indication that FCM is an attractive option for generating the base clusterings when the total number of base clusterings is constrained by, say, available system resource.

Fig. 5 displays the clustering performance (k_f and accuracy when using the maximum lifetime criterion, and accuracy when selecting the combined clustering with k^* clusters) versus N for the three real data sets. The overall best performer is clearly fEAC-AL, although hEAC-AL is better for the Pen-digits data set when using the maximum lifetime criterion. More importantly, faster convergence versus N for fEAC compared to hEAC is still evident here.

Overall, fEAC is better than hEAC when N is very small, and the main reason is because fEAC has much faster convergence. On the other hand, the performance of fEAC and hEAC for larger N is comparable for most cases. In order to provide insight to these observations, we show in Fig. 6 the co-association matrices of using hEAC (left column) and fEAC (right column) with three different N. The data set is half rings. We can see that while the co-association matrix becomes progressively fuzzier for hEAC as N increases, the co-association matrix for fEAC does not change much. The most significant difference here is between the co-association matrices between hEAC and fEAC at N=1. With only a single clustering, the co-association matrix is very crisp for hEAC but quite fuzzy already for fEAC. This directly results from the fact that with a clustering, HCM clusters are disjoint and FCM clusters overlap with one another. When N is large, the co-association matrices for hEAC and fEAC are more similar, as the "fuzziness" here is more of the result of averaging over many clusterings. Therefore, we can infer that the overlapping between FCM clusters has an effect of "fuzzifying" the co-association matrix similar to averaging over several clusterings. The faster convergence of fEAC is

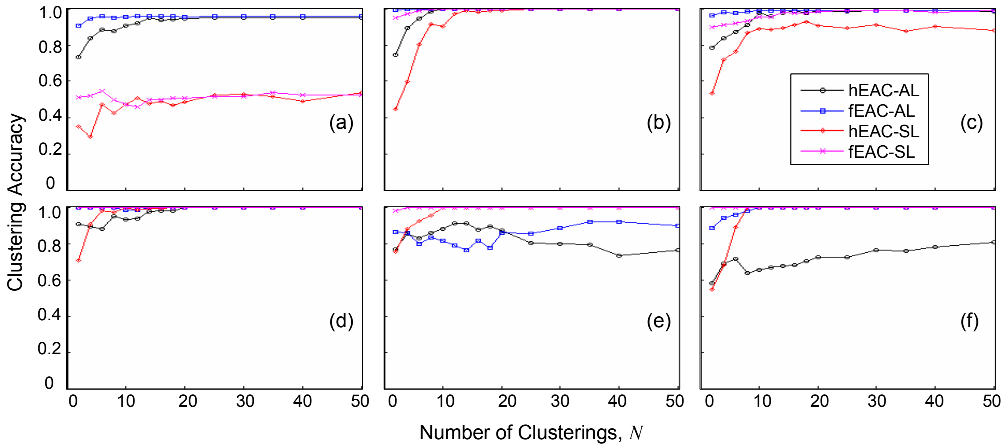


Figure 4. The clustering accuracy vs. the number of clusterings in each ensemble. The number of clusters in the final clustering is set to be the same as k^*. (a)-(f) are plots for the synthetic data set S1-S6, respectively.

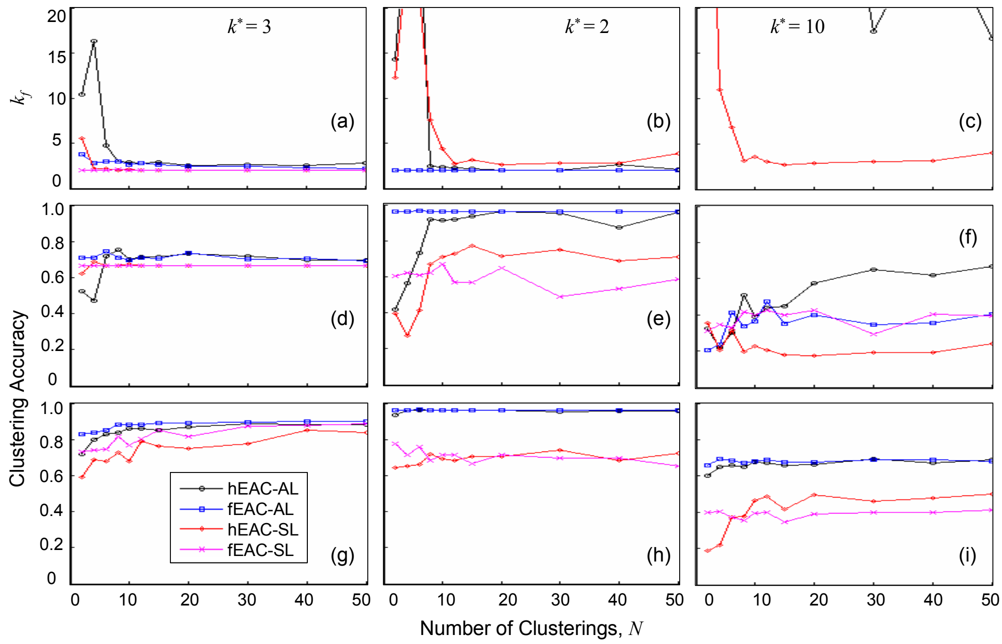


Figure 5. The clustering performance vs. N. for the three real-world data sets. (a)(d)(g) Iris; (b)(e)(f) Wisconsin breast cancer; (c)(f)(i) Pen-digits. (a)-(c) The number of clusters in the final clustering using the maximum lifetime criterion. (d)-(f) Clustering accuracy using the maximum lifetime criterion. (g)-(i) Clustering accuracy when the number of clusters in the final clustering is set to be the same as k^*.

then the result of similarly fuzzy co-association matrices at different N.

In addition to convergence speed, we are also interested in analyzing how the clustering performance is affected by noise. For this purpose, we add 40 randomly distributed noise points to each of the 6 synthetic data sets and re-do the experiments. It is expected that clustering accuracy will be degraded for the corresponding noiseless and noisy data sets. We are instead more interested in seeing how the performances of hEAC and fEAC are degraded differently. The plots for this purpose are in Fig. 7. Here the horizontal and vertical axes represent the clustering accuracy difference between fEAC and hEAC for noiseless and noisy data sets, respectively. Each point corresponds to a combination of data set, SL/AL, N, and the two modes of combined clustering selection (maximum lifetime and pre-specified number of clusters). The majority of the points are above the diagonal, meaning that the degradation for fEAC due to the added noise points is less than that for hEAC. There, we can conclude that fEAC is more tolerant of noise than hEAC, even for large N values when both have converged for noiseless data.

V. CONCLUSIONS

In summary, we have presented in this paper experimental comparison between HCM and FCM as the base clustering generator for EAC. We find that the most notable difference is that evidence-accumulation based cluster ensembles based on FCM converge much faster than those based on HCM, and fEAC has much higher accuracy compared with hEAC when the numbers of clusterings are low. We believe this enhances the usefulness of EAC because it provides a possible solution to the original drawback of EAC that a large number of clusterings is required for convergence. We also provide the insight into this observation by directly comparing the co-association matrices of hEAC and fEAC for different numbers of base clusterings. In addition, our experiments also show that fEAC is more tolerant of noise than hEAC.

There are still many research problems related to soft cluster ensembles that can be pursued. A more comprehensive study (such as one by varying the fuzzification factor) can lead to a more clear understanding of the phenomena associated with fuzzy cluster ensembles. Another direction is the use of non-probabilistic partitions (such as those obtained with possibilistic c-means [22] and its derivatives) as the base clusterings in the ensemble. Overall, we believe that the combination of fuzzy approaches and cluster ensembles will be of great values in future research and applications.

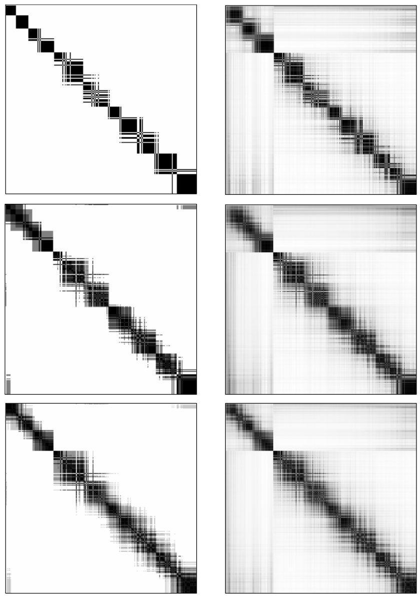


Figure 6. Co-association matrices (half-rings data set) at different *N*. Left and right columns are obtained using HCM and FCM, respectively. *N* = 1, 4, and 40 for the top, middle, and bottom row, respectively.

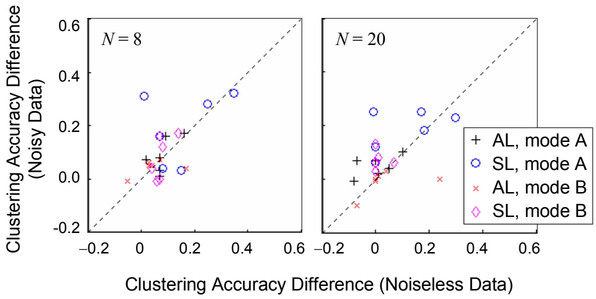


Figure 7. Clustering accuracy difference (accuracy of fEAC minus accuracy of hEAC) with noiseless data (horizontal axis) vs. noisy data (vertical axis). Modes A and B refer to the use of maximum-lifetime criterion and the use of known k^* for selecting the combined clustering, respectively.

REFERENCES

[1] A.K. Jain, M.N. Murty, and P.J. Flynn, "Data Clustering: A Review", *ACM Computing Surveys*, vol. 31, pp. 264-323, 1999.

[2] S. Theodoridis and K. Koutroumbas, *Pattern Recognition* (3rd Ed.), San Diego, CA: Academic Press, 2006.

[3] A.K. Jain and M.H.C. Law, "Data clustering: A user's dilemma", *Lecture Notes on Computer Science*, vol. 3776, pp. 1–10, 2005.

[4] A. Strehl and J. Ghosh "Cluster ensembles -- a knowledge reuse framework for combining multiple partitions", *J. Machine Learning Research*, vol. 3, pp. 583-617, 2002.

[5] A.L.N. Fred and A.K. Jain, "Combining multiple clusterings using evidence accumulation", *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 27, pp. 835-850, 2005.

[6] X.Z. Fern and C.E. Brodley, "Random projection for high dimensional clustering: A cluster ensemble approach", *Proc. 20th Int'l Conf. Machine Learning* (ICML), 2003.

[7] X.Z. Fern and C.E. Brodley, "Solving cluster ensemble problems by bipartite graph partitioning", *Proc. 21th Int'l Conf. Machine Learning* (ICML), *ACM International Conference Proceeding Series*, vol. 69, p. 36, 2004.

[8] A. Topchy, A.K. Jain, and W. Punch, "A mixture model for clustering ensembles", *Proc. SIAM Int'l Conf. Data Mining*, pp. 379-390, 2004.

[9] A. Topchy, A.K. Jain, and W. Punch, "Clustering ensembles: Models of consensus and weak partitions", *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 27, pp. 1866-1881, 2005.

[10] B. Minaei-Bidgoli, A. Topchy and W. F. Punch, "Ensembles of partitions via data resampling", *Proc. 2004 Int'l. Conf. Information Technology*, pp. 188-192, 2004.

[11] K. Punera and J. Ghosh, "Soft Cluster Ensembles", in *Advances in Fuzzy Clustering and its Applications*, Ed. J. Valente de Oliveira and W. Pedrycz, Wiley, 2007.

[12] P. Viswanath and K. Jayasurya, "A fast and efficient ensemble clustering method", *Proc. 2006 Int'l Conf. Pattern Recognition* (ICPR), pp. 720-723, 2006.

[13] P. Hore, L. Hall, and D. Goldgof, "A Cluster Ensemble Framework for Large Data sets", *Proc. 2006 IEEE Int'l. Conf. System, Man, and Cybernetics*, pp. 3342-3347, 2006.

[14] R. Avogadri and G. Valentini , "Ensemble clustering with a fuzzy approach", *Supervised and Unsupervised Ensemble Methods and their Applications*, pp. 49-69, Springer, 2008.

[15] H. Luo, F. Kong, and Y. Li, "Clustering mixed data based on evidence accumulation", *Lecture Notes on Computer Science*, vol. 4093, pp. 348-355, 2006.

[16] J.C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, New York: Plenum, 1981.

[17] L.I. Kuncheva and D.P. Vetrov, "Evaluation of stability of k-means cluster ensembles with respect to random initialization", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, pp. 1798-1808, 2006.

[18] Z. Yu, Z. Deng, H.-S. Wong, and X. Wang, "Fuzzy cluster ensemble and its application on 3D head model classification", *Proc. 2008 IEEE Int'l Joint Conf. Neural Networks*, (IJCNN), pp. 569-576, 2008.

[19] J. Gllavata, E. Qeli, and B. Freisleben, "Detecting text in videos using fuzzy clustering ensembles", *Proc. 8th IEEE Int'l Symposium on Multimedia*, pp. 283-290, 2006.

[20] L. Yang, H. Lv, and W. Wang, "Soft cluster ensemble based on fuzzy similarity measure", *Proc. IMACS Multiconf Comp Eng Systems Appl*, pp. 1994–1997, 2006.

[21] A. Asuncion and D.J. Newman, *UCI Machine Learning Repository* [http://www.ics.uci.edu/~mlearn/MLRepository.html]. Irvine, CA: University of California, School of Information and Computer Science, 2007.

[22] R. Krishnapuram and J.M. Keller, "A possibilistic approach to clustering", *IEEE. Trans. Fuzzy Systems*, vol. 1, pp. 98-110, 1993.

21ps03-vidmar

無衍生研發成果推廣資料

98 年度專題研究計畫研究成果彙整表

計畫主持人：王才沛	計畫編號：98-2221-E-009-146-

計畫名稱：基於證據累加的叢集整合技術之強韌化與功能延伸

成果項目			量化			單位	備註（質化說明：如數個計畫共同成果、成果列為該期刊之封面故事...等）
			實際已達成數（被接受或已發表）	預期總達成數(含實際已達成數)	本計畫實際貢獻百分比		
國內	論文著作	期刊論文	0	0	100%	篇	
		研究報告/技術報告	0	0	100%		
		研討會論文	0	0	100%		
		專書	0	0	100%		
	專利	申請中件數	0	0	100%	件	
		已獲得件數	0	0	100%		
	技術移轉	件數	0	0	100%	件	
		權利金	0	0	100%	千元	
	參與計畫人力（本國籍）	碩士生	7	3	100%	人次	
		博士生	0	0	100%		
		博士後研究員	0	0	100%		
		專任助理	0	0	100%		
國外	論文著作	期刊論文	0	0	100%	篇	
		研究報告/技術報告	0	0	100%		
		研討會論文	1	1	100%		
		專書	0	0	100%	章/本	
	專利	申請中件數	0	0	100%	件	
		已獲得件數	0	0	100%		
	技術移轉	件數	0	0	100%	件	
		權利金	0	0	100%	千元	
	參與計畫人力（外國籍）	碩士生	0	0	100%	人次	
		博士生	0	0	100%		
		博士後研究員	0	0	100%		
		專任助理	0	0	100%		

其他成果 (無法以量化表達之成果如辦理學術活動、獲得獎項、重要國際合作、研究成果國際影響力及其他協助產業技術發展之具體效益事項等，請以文字敘述填列。)	無

	成果項目	量化	名稱或內容性質簡述
科教處計畫加填項目	測驗工具(含質性與量性)	0	
	課程/模組	0	
	電腦及網路系統或工具	0	
	教材	0	
	舉辦之活動/競賽	0	
	研討會/工作坊	0	
	電子報、網站	0	
	計畫成果推廣之參與（閱聽）人數	0	

國科會補助專題研究計畫成果報告自評表

請就研究內容與原計畫相符程度、達成預期目標情況、研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性）、是否適合在學術期刊發表或申請專利、主要發現或其他有關價值等，作一綜合評估。

1. 請就研究內容與原計畫相符程度、達成預期目標情況作一綜合評估

■達成目標

□未達成目標（請說明，以 100 字為限）

　　□實驗失敗

　　□因故實驗中斷

　　□其他原因

說明：

2. 研究成果在學術期刊發表或申請專利等情形：

論文：■已發表 □未發表之文稿 □撰寫中 □無

專利：□已獲得 □申請中 ■無

技轉：□已技轉 □洽談中 ■無

其他：（以 100 字為限）

已發表於國際會議部分為部分成果，其他部分仍撰寫中。

3. 請依學術成就、技術創新、社會影響等方面，評估研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性）（以 500 字為限）

作為一個較新的領域，叢集整合的研究主要在於其特性與演算法的分析，但在各領域中也逐漸被應用於實用問題，而且主要是對一般叢集演算法較為困難的問題，也需要較大的計算量。我們研究的強韌叢集整合以及模糊叢集整合分析結果可以在這方面提升準確率與效率，因此對於許多相關問題皆有助益。