國立交通大學

資訊科學與工程研究所

碩士論文

以區域性鄰集為基礎之相似度轉換方法應用於分群演算法

A Locality Based Similarity Transformation Method
for Clustering Algorithms

研 究 生：陳彥嘉

指導教授：胡毓志

中華民國一百零一年九月

以區域性鄰集為基礎之相似度轉換方法應用於分群演算法

A Locality Based Similarity Transformation Method for Clustering Algorithms

研 究 生：陳彥嘉　　Student：Yen-Chia Chen

指導教授：胡毓志　　Advisor：Yuh-Jyh Hu

國 立 交 通 大 學

資訊科學與工程研究所

碩 士 論 文

A Thesis
Submitted to Institute of Computer Science and Engineering
College of Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in

Computer Science

September 2012

Hsinchu, Taiwan, Republic of China

中華民國一百零一年九月

以區域性鄰集為基礎之相似度轉換方法應用於分群演算法

學生：陳彥嘉　　指導教授：胡毓志

國立交通大學資訊科學與工程研究所　　碩士班

摘要

如何選擇一個合適的相似度函式在分群演算法中是一項相當重要的問題，相似度函式會直接影響分群結果。我們提出一種以區域性鄰集為基礎之相似度轉換法，藉由觀察區域性鄰集的分布以調整資料的相似度。透過相似度的轉換，我們能夠調整資料的分布情形，凸顯資料間的邊界，以利於分群演算法尋找具有意義的集群。將此方法應用至非監督式或半監督式分群演算法，我們預期可以尋找出多種分布型態的集群。我們的實驗結果說明：1. 利用以區域最近鄰為基礎之相似度轉換法，在無配對限制的幫助下，能夠有效的處理多種凹凸形狀的資料分布，而配對限制的加入亦能進一步提升整體的準確率；2. 以區域最近鄰為基礎之相似度轉換方法亦能應用於資料維度的縮減。實驗顯示維度的縮減不僅能減少計算量以改善分群演算法的整體速度，同時在多項實驗中仍能維持分群的正確性。

關鍵字：分群演算法、相似度轉換、凹凸形狀資料分布、維度縮減

A Locality Based Similarity Transformation Method
for Clustering Algorithms

Student：Yen-Chia Chen Advisor：Dr. Yuh-Jyh Hu

Institute of Computer Science and Engineering
College of Computer Science
National Chiao Tung University

ABSTRACT

An appropriate similarity function is crucial to clustering algorithms because it affects the clustering result directly. We propose a locality based similarity transformation method, which transforms the similarity between two data points based on the distribution of their neighbors in vicinity. The blurry boundary between clusters can be better revealed after transformation. By applying the locality based similarity transformation method to unsupervised or semi-supervised clustering, we can discover clusters more easily even if they are of irregular contours. Our experimental results demonstrate that: (1) the proposed locality based similarity transformation method can improve clustering methods in finding arbitrarily shaped clusters without any prior knowledge, (2) prior knowledge represented as pairwise constraints can be incorporated to further improve the performance of clustering, and (3) a dimension reduction method based on multiple dimension scaling can be combined with the transformation procedure not only to reduce the feature space but also the computation cost in transformation.

Keywords：Clustering algorithms, Similarity transformation, Non-convex shaped clusters, Dimensional reduction

致謝

在就讀碩士班的過程中，首要感謝的是我的指導教授胡毓志老師。在這兩年中，老師不僅提供我研究上的方向與意見，讓我能夠快速的進入狀況學習做研究的方法，同時也給予許多寶貴的意見在生活或處世態度上，這些對我在日後的道路上都有極大的幫助。最後也感謝老師花了許多心思在校閱我的論文內容和驗證方法、實驗。

我要感謝實驗室的學長姐和學弟妹、我的室友以及在交大認識的朋友陪我度過了兩年時光，不論是在修課或做研究時的互相幫助，或是在閒暇之餘陪我聊天、一起出遊、參加比賽，這些都讓我在兩年中過得相當充實且充滿回憶。

最後我要感謝我的爸媽和姊姊一路上給予的關心和支持，使我在這些年之中能夠過得安穩順利，不需擔憂任何問題。

目錄

表目錄

圖目錄

第一章，緒論

1.1 研究動機

分群演算法的目標是將資料集細分成多個擁有相同性質的子分群，目前已被廣泛的應用於許多領域中，例如：機器學習、影像分析、文字探勘、生物資訊等方面[1]。機器學習的主要目標是使用一種自動學習的演算法，從中獲得部分資訊後，再利用這些資訊對未知的資料進行預測分析，可以被應用至檢測信用卡欺詐、市場分析等方面。而在影像分析的領域上，分群演算法常被用來辨識影像的特徵點，或是用於分群大量影像，進而提升影像檢索的效率。除了上述二種領域外，亦可應用於文字探勘，其主要目標在於從非結構化的文字中，萃取出有用的資訊或知識。生物資訊的資料來源通常是以文字表示，其應用可包含在文字探勘的範圍內，多數學者使用分群演算法進行微陣列基因晶片的資料分析、分析蛋白質序列的排序方式，從中獲取更有價值的資訊。

分群演算法屬於非監督式學習法，多數情況下只能依靠使用者的知識提出假設，設法找出假設下的最佳解。最常見的假設是：屬於相同分群的資料具有類似的性質，而不同分群的資料具有顯著性的差異。然而，我們發現當資料的分布形態屬於非凸狀圖型(non-convex)時，將違反上述的假設而無法得到正確的分群結果。非監督式分群演算法的困難處在於如何提出合適的假設以及相似度函式的定義，不適當的假設或相似度函式都是影響分群準確率的原因。對於分群演算法的假設和相似度函式的定義並非全然依靠使用者，有時我們能夠從資料中取得少許資訊，例如：配對限制(pairwise constraints)或少數已被標記類別的資料，這些資訊都能稍稍透露資料的型態。因此，有學者提出一種新的概念，稱為半監督式分群演算法[2]。半監督式分群演算法藉由觀察少數資訊的分布形態和相似關係，修正提出之假設或重新定義相似度函式，使得演算法的設計是以滿足少數資訊的分布形態和相似關係做為前提。

將半監督式分群演算法應用於真實世界中，取得配對限制或少數已被標記類別的資料其成本通常是相對困難，且演算法的準確性將受限於預先蒐集而來的知識(prior knowledge)。配對限制的數量的多寡，或是在資料分布中是否位在關鍵性位置等，將成為影響準確性的因素。因此，為了同時解決非監督式分群演算法中如何選擇相似度函式的問題，以及半監督式分群演算法中重新定義的相似度函式其分群準確性受限於配對限制的問題，我們提出一種以區域性(locality based)為基礎的相似度轉換方法，它是屬於非監督式的學習方法。此方法藉由觀察任意二個資料點形成之鏈結其兩端點的區域型鄰集(local neighbors)分布情形，重新描述資料的相似關係，使得分群演算法找出的分群結果可以更加接近真實的分群情形。

1.2 目標

此篇論文提出一種屬於非監督式學習法的相似度轉換方法，它是以區域性鄰集為基礎，重新調整資料點彼此的相似度。新的相似關係受制於區域性鄰集的定義，而用來尋找區域性鄰集的方法則有相當多種。此篇論文中我們以 K 最近鄰(K-nn)的概念出發，延伸出一種稱為可互相包含最近鄰(mutual included nearest neighbors, MI-nn)的新方法。它不僅能保有 K 最近鄰的特性，同時改善 K 最近鄰之效能受限於參數 K 的缺點。我們將調整後的相似關係應用至分群演算法，例如：K-means 演算法等，預期能夠找出更加接近真實的分群情形。

此篇論文提出的方法預期能解決：1. 在非監督式演算法中，如何選擇相似度函式的問題；2. 在半監督式分群演算法中，如何利用重新定義的相似度函式減輕分群結果被預先蒐集而來的知識(prior knowledge)所限制。

1.3 論文架構

論文架構主要分為五個章節。第一章說明研究動機、概念想法與目標。第二章簡介應用的領域，並討論多種現存的研究方法與文獻。第三章介紹此篇論文設計的方法，包含方法的各種假設、設計、演算法流程以及許多應用於方法中的統計學方法。第四章記錄實驗評量方式與實驗設計、討論。第五章總結此篇論文，說明此篇論文所提出的方法之貢獻，並與現存的其他方法做完整的比較、討論。

第二章，相關方法與文獻

2.1 非監督式分群演算法(unsupervised clustering)

分群演算法是一種非監督式學習法，我們通常只能預測分群的數目，資料真實的分群情況卻是未知的。它能夠將輸入的資料依據某種假設，自動的分成數個子分群，最常見的假設是：屬於相同分群的資料具有類似的性質，因此同個分群之間的資料相似度必須越大越好；不同分群的資料具有顯著性的差異，因此不同分群之間的資料相似度必須越小越好。描述相似度的公式有相當多種類，包含歐氏距離(Euclidean distance)、歐式距離平方(Squared Euclidean distance)等，如表 2-1 所示。其中歐氏距離是最常被應用在分群演算法的相似度函式。本章節所介紹的方法，多數都是採用歐氏距離描述相似度。依據假設，我們能夠定義一組目標函數(objective function)來滿足假設的情況，緊接著設計出一套專屬的演算法，試著找出此目標函數的最佳解。然而，由目標函數找出的最佳解所形成的分群未必是正確解，它是隨著不同的目標函數而有所區別，僅能說明在此目標函數之下，最佳解是假設下的正確解。

表 2-1. 相似度函式

Euclidean distance	$\sqrt{\sum_i (a_i - b_i)^2}$
Squared Euclidean distance	$\sum_i (a_i - b_i)^2$
Minkowski distance	$\left(\sum_i \lvert a_i - b_i \rvert^p\right)^{\frac{1}{p}}$
Manhattan distance	$\sum_i \lvert a_i - b_i \rvert$
Mahalanobis distance	$\sqrt{(a-b)^T S^{-1}(a-b)}$

一個好的分群演算法往往需要滿足幾項需求[1]，在此我們特別針對以下三項做探討與改進：

1. 尋找任意形狀的群集(Discovery of clusters with arbitrary shape)
2. 高維度(High dimensionality)
3. 遵循某些限制條件下的分群方法(Constraint-based clustering)

較常見的非監督式分群演算法包括切割式(partitional clustering)、階層式(hierarchical clustering)和密度基礎式(density based clustering)。

2.1.1 切割式(partitional clustering)

切割式分群法的目標是將完整的資料集，依據使用者給予的分群數 K 切割成符合目標函數假設的 K 個子分群，是一種由上至下(top-down)的演算法。切割式分群法常見的有 K-means 演算法[3]和 K-medoids 演算法[4]。

K-means 演算法是以中心為基礎進行分群，其目標是將資料個數為 N 的資料集細分成 K 個擁有高度相似度的子群，藉由最佳化資料與中心的相似度，自多個隨機出發的初始中心之中尋找一組最佳解。使用 K-means 演算法通常需要事先預測分群數 K 並選擇一種相似度函式，不恰當的分群數量或相似度函式都會引導演算法至錯誤的結果，而最常使用的相似度函式則是歐式距離。K-means 演算法首先在資料總數為 N 的資料集中任選 K 個資料做為初始中心，接著將剩餘的 N-K 個資料分配至能夠最佳化目標函數的分群，隨後更新分群中心。不斷重複上述步驟直到中心停止改變，我們可以得到一組分群結果。由於 K-means 演算法對初始中心相當敏感，不同的初始中心未必會收斂至相同的分群結果，我們必須反覆挑選不同的初始中心，從中挑選一組能夠最佳化目標函數的分群結果做為最佳解。

K-medoids 演算法和 K-means 十分相似，兩者的差別在於 K-means 使用中心(center)代表分群，而 K-medoids 使用實心(medoid)表示。中心的位置可以不在任何一個資料點上，通常以相同分群內所有資料的平均值表示中心；實心的位置則

必定在資料點上，通常以相同分群內最接近中心的資料點表示實心。K-medoids 演算法將表示分群的方式從中心修改成實心，使演算法要最佳化的目標是成對資料的相似度，能夠相對的不受離群值(outlier)的影響，大幅降低對離群值的敏感性，亦不會有空群的情況發生。

K-means 或 K-medoids 演算法的優點在於能夠快速收斂、易於建構、通常能找到區域最佳解(local optimum)；然而，兩者都不善於處理各分群資料大小不均衡、密度不均衡的凸狀圖形或凹狀圖形。

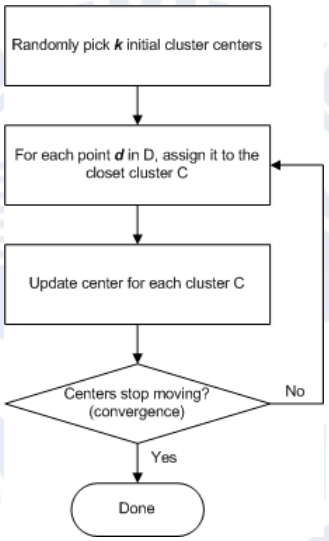


圖 2-1. K-means 演算法流程

2.1.2 階層式(hierarchical clustering)

階層式分群法的目標是建立一棵多層級的樹狀架構(dendrogram)，透過此樹狀架構表示資料彼此的相似關係，可依照使用者的需求彈性產生不同分群數的分群結果。階層式分群法主要可以分成二類：聚合法(agglomerative)和分裂法(divisive)。

聚合法是屬於由下至上(bottom-up)的演算法，每一筆資料最初都被視為單獨的分群，藉由不斷合併兩個最為相似的分群，直到所有資料形成完整的一群。大多數的階層式演算法是採用聚合法；分裂法是屬於由上至下(top-down)的演算法，

起初將所有資料視為一個分群，依據相似度不斷將大分群拆解成多個彼此相似度低的小分群，直到每筆資料都形成獨立分群或分群總數已達到預先設定的目標。

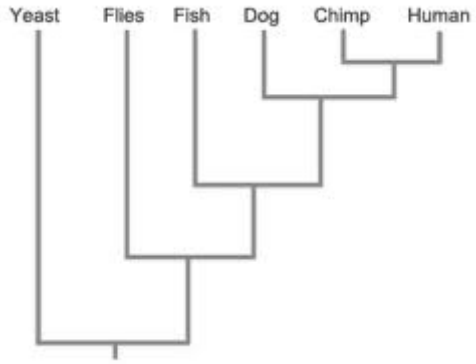


圖 2-2. 多層級樹狀架構(dendrogram)

Chameleon[5]是階層式分群法的代表之一，是一種採用動態模型的分群法。它使用 K 最近鄰圖 (K-nearest neighbor graph)動態調整相鄰半徑(neighborhood radius)以描述資料彼此關係，任何一筆資料的相鄰半徑取決於資料所在的區域。在密度高的區域，相鄰半徑變得較狹小；在密度低的區域，相鄰半徑變得更為寬鬆。Chameleon 首先建構出 K 最近鄰圖，接著將圖形切割成多個子分群，使用 RI(relative interconnectivity)和 RC(relative closeness)評估各個子分群的相似度，反覆合併至預先設定的分群數。Chameleon 能產生更自然的分群結果，善於處理任意形狀的群集，但對於參數的敏感性較高。Chameleon 包含的參數相當多種，然而分群的過程屬於非監督式學習，對於參數的設定往往只能透過反覆測試(trial and error)，進而尋找出一組使用者接受的結果。

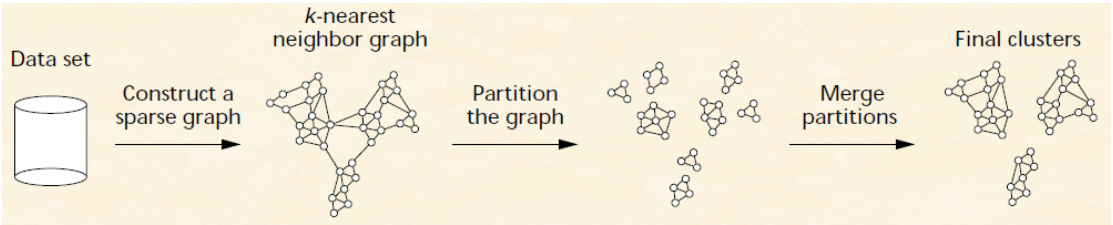


圖 2-3 Chameleon 演算法流程[5]

階層式分群法的困難處在於如何選擇聚合或分裂的位置。選擇聚合或分裂的位置相當關鍵，當資料被聚合或分裂形成新的分群，下一步聚合或分裂的位置將發生在新形成的分群上。若是在聚合或分裂的過程中，其中一個階段做出較差的選擇，最終將引導至一個錯誤的分群結果[1]。

2.1.3 密度基礎式(density based clustering)

為了分辨出任意形狀的群集，我們觀察資料所在空間的密度，直覺的假設屬於相同分群的資料會聚集在一個密度較高的區域內，而不同的分群則是由一塊密度較低的區域分隔。DBSCAN[6]是密度基礎式分群法中最典型的一種，其目標是分辨任意形狀的群集。DBSCAN 定義下方五項名詞：

1. ε 最近鄰(ε-neighbor-hood)：點 p 周圍半徑 ε 內的所有資料點都稱作 p 的 **ε 最近鄰**。
2. 核心點(core object)：若是點 p 的 ε 最近鄰超過 MinPts 個，則點 p 成為分群的**核心點**。
3. 直接密度可達(direct density-reachable)：若 p 是 q 的 **ε 最近鄰**，且 q 為核心點，則說 q **直接密度可達** p。
4. 密度可達(density-reachable)：若存在一個鏈結 $p_1,p_2,\ldots\ldots p_n$，$p_1=q$，$p_n=p$，且 p_i **直接密度可達** p_{i+1}，則說 q **密度可達** p。
5. 密度相連(density-connected)：若存在一點 r，使 r **密度可達** p 和 q，則說 p 與 q 是**密度相連**。

DBSCAN 透過觀察每筆資料的 ε 最近鄰來尋找分群。若是某個資料點 d 的 ε 最近鄰超過 MinPts 個，則建立一個以 d 為核心點的新分群。DBSCAN 加入核心點密度可達的其他點至分群中，擴張分群所能涵蓋的區域。分群擴張的過程可能涉及不同分群合併，不斷重複擴張的步驟直到所有點都被分群完成。

DBSCAN 的優點包含：自動找出分群數目、可以處理任意形狀的群集、較不受離群值的影響且能夠標記出離群值。缺點則有兩項：1. 當資料維度較高時，

高維度空間下的資料分布必定是非常散亂，尋找一個適當的參數 ε 將成為難題；2. DBSCAN 仍然無法解決不同分群的密度有極大差異的情況。

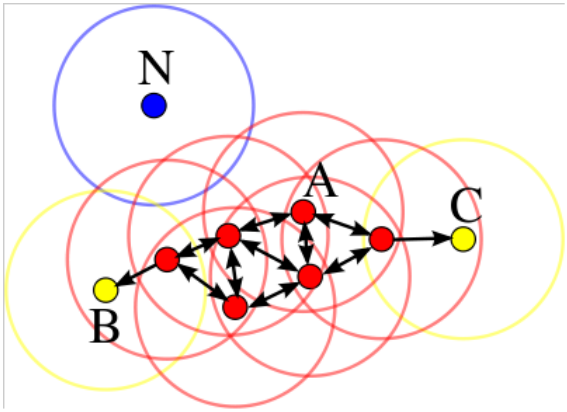


圖 2-4. DBSCAN 示意圖

說明：A 代表核心點。相對於 A，B 和 C 是密度相連，且 B 和 C 屬於相同分群。N 表示離群值。

2.2 半監督式分群演算法(semi-supervised clustering)

分群演算法是在未知資料分群型態的情況下進行，只能依據特定的假設將資料分成多個子分群。在真實世界的應用上，若是能透過領域專家(domain expert)的幫助從中獲得少數資訊，將這些資訊與非監督式演算法結合，可以大幅提升分群演算法的準確性。這樣的合併方式稱為半監督式分群法。

專家所提供的資訊可以分為二類：配對限制(pairwise constraints)和少數已被標記過的資料，兩者的獲取成本通常是高且困難的。配對限制可以用來說明成對資料之間的關係，主要可以再分為二類：MUST-link 和 CANNOT-link：

1. MUST-link：若某兩筆資料 x、y 的關係可使用 MUST-link 描述，則 x 和 y 屬於相同分群。
2. CANNOT-link：若某兩筆資料 x、y 的關係可使用 CANNOT-link 描述，則 x 和 y 必定分布在不同分群。

MUST-link 擁有遞移性(transitive)，我們可以藉著建構遞移包(transitive closure)產生更多的 MUST-link。例如：x 和 y 形成 MUST-link 且 y 和 z 形成 MUST-link，根據遞移性，x 和 z 必定形成 MUST-link；雖然 CANNOT-link 並不

具備遞移性，例如：x 和 y、y 和 z 的關係皆是 CANNOT-link，並不保證 x 和 z 的關係是 CANNOT-link。我們仍可以利用特殊方法建構出更多的 CANNOT-link。假設 M、N 分別是兩個獨立的 MUST-link 連通單元(connected component)，若是在 M、N 之間存在一個 CANNOT-link，那麼由 M、N 所形成的任意配對關係都會具備 CANNOT-link 性質。將配對限制應用至分群演算法，多數學者都會遵循上方提出的特性延展配對限制的數量。

如何將配對限制整合至分群演算法中，主要可以分為二類：search-based 和 similarity-based[2]。

2.2.1 Search-based

Search-based 方法修改屬於非監督式的分群演算法，使分群演算法在最佳化目標函數的過程中，利用配對限制將分群引導至更接近真實的分群結果。修改分群演算法的方式主要有下列三種方式：

1. 利用配對限制建立初始分群[7]。
2. 分群的過程不可違反配對限制，亦即將資料被分配至任何一個分群，都必須盡量滿足所有配對限制[8]。
3. 更改衡量分群結果的目標函數，使得在尋找最佳化目標函式數值的分群過程中，同時能最小化違反配對限制的資料數量[9, 10]。

2002 年 Basu 等的研究中[7] 修改 K-means 演算法抽取初始中心的方法。K-means 演算法是隨機抽出初始中心，由 Basu 等提出的方法則是將少數已標記過的資料作為種子集(seed set)，對種子集進行分群產生種子分群(seed clustering)，從中挑選出更合適的初始中心位置。完成挑選初始中心後，剩餘的資料則是依據 K-means 演算法分配分群的方式完成分群，其中，種子集中的資料其分群情況在完成分群後不再更動。建立種子分群可以引導分群結果至更接近少數已標記過資料的分布情形，同時能降低分群結果陷入區域最佳解的機率。

2001 年 Wagstaff 等[8] 提出 constrains K-means 方法，它整合配對限制至原

始的 K-means 演算法中，藉由修改分群分配的方法，直接影響分群的過程與結果。當資料被分類至任何一個分群時，都不可違背配對限制。分群結果和配對限制有極大的相關，提供的配對限制越多，準確性越高。然而，當使用者提供的配對限制中存在錯誤，容易引導演算法至錯誤的結果，與其他方法相比彈性較低、更加依賴配對限制。

1999 年 Demiriz 等的研究中[9] 結合非監督式和監督式演算法。他們使用非監督式分群演算法來最佳化一種用來評估監督式演算法準確率的分數，提出一組全新的目標函式：$\min \quad \beta \times \text{Cluster\_dispersion} + \alpha \times \text{Cluster\_impurity}$。當 α 為零，產生的結果相當於非監督式演算法；當 β 為零，產生的結果相當於監督式演算法。Cluster dispersion 通常使用 DBI (Davis-Bouldin index)或 MSE (mean square error)，採用 MSE 則和 K-means 演算法的目標函數相同，而 Cluster impurity 則是使用 Gini index。Demiriz 等提出的方法使用非監督式分群法建構分群，藉此計算 cluster dispersion；使用已標記過的資料做為訓練集建立分類器，以未標記過的資料做為測試集，藉此計算 cluster impurity，終極目的都是最佳化目標函式。

2004 年 Basu 等的研究中[10] 將違反配對限制的代價函式合併至目標函式之中，定義代價函式為：$w\,\mathbb{I}[\ell_i \neq \ell_j] + \bar{w}\,\mathbb{I}[\ell_i = \ell_j]$，其中 w 和 $\bar{w}$ 分別是違反 MUST-link 和 CANNOT-link 的權重。違反配對限制會獲得一定代價，造成目標函數值增加，因此，在最佳化目標函式尋找分群最佳解的過程中，同時能最小化違反配對限制的資料數量，達到與 Cop K-means[8]相同的目標。

2.2.2 Similarity-based

Similarity-based 方法通常搭配一個以相似度函式為基礎的分群演算法，相似度函式會被重新訓練，使其盡可能不違反配對限制。在訓練的過程中，屬於 MUST-link 的成對資料將被重新描述得更加相似，使這組成對資料越有可能被分類至同個分群；相反的，屬於 CANNOT-link 的成對資料會被描述得更加相異，越有可能被歸類至不同分群。重新訓練相似度函式之目的是盡可能滿足數量較為

稀少的配對限制，然而多數資料的分群仍然是未知的，若是進一步將 similarity-based 方法與 search-based 方法結合，成為一種二階段模型，更能增進分群的準確性[11, 12, 13, 14, 15,16]。

2002 年 Xing 等的研究中[11] 將如何學習合適的相似度矩陣視為一種凸性最佳化問題(convex optimization problem)。他們透過半正定規劃(semidefinite programming)學習馬式距離的共變數矩陣，目標是最小化已標記資料中屬於相同分群的資料與其中心的距離總和，使得經過相似度轉換後的新屬性能夠遵循配對限制，改善整體的分群準確率。使用新的共變數矩陣作距離轉換後，額外與 Cop K-means[8]結合，實驗結果說明能增進分群的準確性。

2003 Bar-Hillel 等的研究中[12] 使用 RCA(relevant component analysis)建構出馬式距離的共變數矩陣。RCA 能重新描述資料的特徵空間，利用已標記過的資料來估量各個維度的重要性。透過線性轉換將較重要的維度給予更高的權重，較不重要的維度給予更低的權重。RCA 類似 PCA(principal component analysis)和 LDA(linear discriminant analysis)，目標是尋找合適的投影方向，將一群已分類的資料投影至更低維度空間，使得相同分群的資料更為集中，而不同分群的資料更為遠離。RCA 已被證明可以用來最佳化分群與其中心的距離總和。

2003 年 Basu 等[13] 提出一種將 similarity-based 和 search-based 兩種方法合併的演算法，預期能同時享有此二種方法的優點。合併的方式如下：1. 使用 2002 年 Basu 等[8] 提出的方法，透過已標記過的資料建構出初始的分群；2. 採用 2002 年 Xing 等[11] 提出的方法，修改馬式距離的共變數矩陣，重新描述相似度；3. 使用由 2004 年 Basu 等[10] 提出的方法，整合代價函式至目標函數中。Similarity-based 往往需要充足的配對限制或已被標記過的資料才能有更接近真實的分群結果，與 search-based 方法合併能夠大幅提升分群的準確率。

2004 年 Basu 等[14] 提出一種基於隱藏式馬可夫隨機領域(hidden Markov random field)的機率模型之半監督式分群法，將配對限制與 K-means 演算法結合，

同時允許多樣的相似度函式，例如：餘弦相似度(cosine similarity)和 KL 距離(Kullback-Leibler Divergence)。藉由修改 K-means 的目標函式，定義一組違反配對限制的代價函式與目標函式結合，進而影響分群的過程和結果。

2002 年 Klein 等[15] 提出傳播限制(propagation constraints)的概念。傳播限制主要的意涵在於：若是存在兩點 x、y 相似，任何一點 z 與 x 相似，同時會與 y 相似；若是存在兩點 x、y 不相似，任何一點 z 與 x 相似，則 z 與 y 不相似。 觀察配對限制所包含的資料，將關係屬於 MUST-link 的兩個資料其相似度設為最低，通常以 0 表示，恣意修改相似關係將無法滿足賦距空間(metric space)中的三角不等式，因此改以最短路徑重新計算相似度。Klein 等[15] 說明以最短路徑表示仍然可以擁有賦距空間的特性；將關係屬於 CANNOT-link 的兩個資料其相似度設為最高，通常以資料集中的最大值加 1 表示，與 MUST-link 不同，改以一種用來描述資料關係的計量分數，例如 complete-link，重新描述修改後的相似關係。上述的方式能夠依照增殖配對限制的目標重新描述資料彼此的相似關係，實驗指出，採用修改後的相似關係，能夠顯著的增進完整連結聚合演算法(complete-linkage agglomerative algorithm)的準確性。

2006 年 Weinberger 等提出一種稱為 LMNN 的方法[16]，藉由觀察少數已被標記的資料中任一筆資料的 K 最近鄰，使用類似 SVM(support vector machine)的做法，重新訓練相似度函式，目標是讓任何一筆資料的 K 最近鄰擁有相同分群，而不同的分群資料彼此會有較大的區隔。K 最近鄰的效能與採用的相似度函式有強烈的相依性，根據已被標記的資料重新學習的相似度函式能夠明顯改善分群準確率。Weinberger 等[16]提出的方法，其原始目標是應用於分類(classification)領域，輸入的監督資訊是資料類別(class)而非配對限制，資料類別所提供的資訊強度明顯高於配對限制。與 Xing 等[11]、Bar-Hillel 等[12]和 Basu 等[13]提出的方法最大的差異在於最佳化的目標並非是少數已被標記過的資料彼此的相似度，而是彼此的鄰居關係。

第三章，方法

在此章節，我們提出一種以區域性為基礎(locality based)的相似度轉換方法，透過觀察資料集所形成的完整圖中，任意二個資料點形成之鏈結其兩端點的區域型鄰集分布，重新描述此鏈結的相似度權重。鏈結主要可以分為二種類型：1. 連接不同分群的鏈結(inter-cluster link)；2. 連接相同分群的鏈結(intra-cluster link)。相似度轉換的目標在於讓連接不同分群的鏈結權重被相對的提高，而連接相同分群的鏈結權重被相對的降低。

3.1 相似度轉換公式與區域性鄰集

以完整圖(complete graph)描述資料集中所有資料點彼此的相似關係，圖中任意二個資料點之間必定有一個鏈結存在，我們目前使用歐氏距離作為兩個資料點的相似度分數，並賦予每一個鏈結獨立的權重值 A。定義相似度轉換公式為：

$$\mathrm{dist}(r,s) = \mathrm{dist}(r,s) \times A_{r,s}, \forall\, r,s \in V$$

將權重值 A 之初始值設定為 1，使得在更新權重 A 之前，相似度轉換方法相當於原始的相似度函式。

權重值 A 的用途在於分辨鏈結的類型，一個擁有較高權重值的鏈結更有可能屬於連接不同分群的鏈結(inter-cluster link)，反之則屬於連接相同分群的鏈結(intra-cluster link)。權重值 A 是影響相似度轉換的重要因素，新的相似關係受制於權重值 A 的定義方式。在此我們透過觀察鏈結兩端點的區域性鄰集，各端點與其區域性鄰集成員的相似關係定義鏈結權重。鏈結權重的計算公式定義為[1]：

$$A_{r,s} = \frac{1}{2}\left[\sum_{x \in L-nn(r)} \frac{\mathrm{dist}(r,s)}{\mathrm{dist}(r,x)} + \sum_{x \in L-nn(s)} \frac{\mathrm{dist}(r,s)}{\mathrm{dist}(s,x)}\right]$$

我們的假設是：1. 相鄰的資料點有較高的機率屬於相同的群集；2. 密度越高的區域越有可能有群集存在；將鏈結的相似度放置於分子，而區域性鄰集中成員與

[1] L-nn(r)表示資料點 r 的區域性鄰集集合

端點的相似度則放置於分母。當區域最近鄰中成員的分布情形如圖 3-1(a)時，區域最近鄰成員與資料的距離都明顯小於兩資料點 r、s 的距離時，可從 $\frac{\text{dist(r,s)}}{\text{dist(r,x)}}$ 得到一組較大的數值，權重值會被相對的調高，可以說明資料點的周圍形成兩個密度較高的區域，滿足我們提出的第二種鏈結權重假設，因此我們推測兩個資料點有更高的機率屬於不相同的分群；若是區域最近鄰中成員的分布情形如圖 3-1(b)時，區域最近鄰成員與資料的距離都和兩資料點 r、s 之距離相近時，可從 $\frac{\text{dist(r,s)}}{\text{dist(r,x)}}$ 得到一組較小的數值，權重值會被相對的調低，能夠說明兩資料點同時位於一個密度較鬆散的區域內，因此我們推測兩個資料點應該屬於相同的分群。

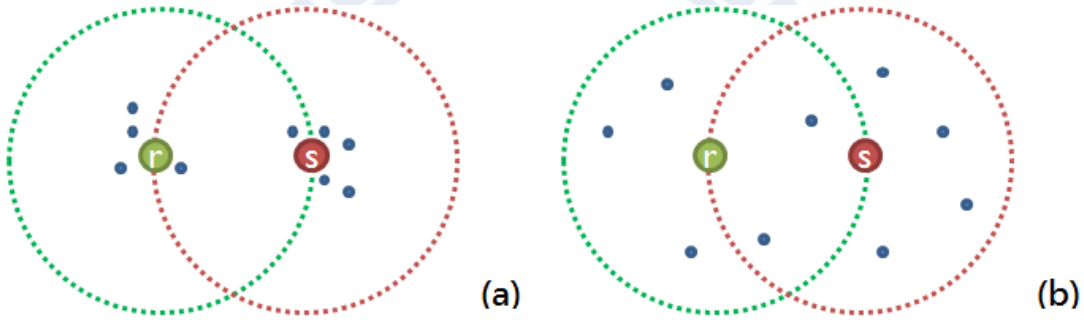


圖 3-1. 各種區域性鄰集分布情形

新的鏈結權重的計算受制於區域性鄰集的定義，而用以尋找區域性鄰集的方法有相當多種，我們提出兩種定義：1. 依照 K 最近鄰作為定義新權重的基礎；2. 依照鏈結兩端點間的距離作為定義新權重的基礎。

3.1.1 依照 K 最近鄰做為定義新權重的基礎

使用 K 最近鄰演算法來尋找區域性鄰集是相當直覺的方法，與端點最接近的 K 個資料點將自動形成一個區域性鄰集集合。定義鏈結權重的計算公式為[2]：

$$A_{r,s} = \frac{1}{2}\left[\sum_{x\in K-nn(r)} \frac{\text{dist}(r,s)}{\text{dist}(r,x)} + \sum_{x\in K-nn(s)} \frac{\text{dist}(r,s)}{\text{dist}(s,x)}\right]$$

以圖 3-2 為例，尋找 3 最近鄰做為區域性鄰集。圖 3-2 (a)中兩端點 r、s 的 3 最近

[2] K-nn(r)表示資料點 r 依照 k 最近鄰作為定義新權重的基礎之區域性鄰集集合

鄰集合與端點的距離都明顯小於鏈結的長度，能夠明顯找出兩個分群；圖 3-2(b)中兩端點 r、s 的 3-最近鄰中，分別有兩個資料點與端點的距離都和鏈結長度接近，能說明 r、s 有較高的機率屬於相同分群。我們計算圖 3-2(a)(b)的鏈結權重，分別是：9 和 5.4，比較權重的大小已能相對的區分鏈結的類型；再將新求出的權重值帶入至相似度轉換公式，可以得到一組新的距離，分別是：27 和 16.2，圖 3-2(a)中的距離已被相對的增加。目前設計的相似度轉換公式能讓圖 3-2(a)中較有可能是連接不同分群的鏈結的距離相對於圖 3-2(b)調整的更大，雖然在進行相似度轉換之前，點 r 和點 s 在圖 3-2(a)和圖 3-2(b)中的距離是相同的，但透過觀察鏈結兩端點周圍的 K 最近鄰集合分布情況，重新調整鏈結權重，改變實際距離使其更接近預期的分群情況。

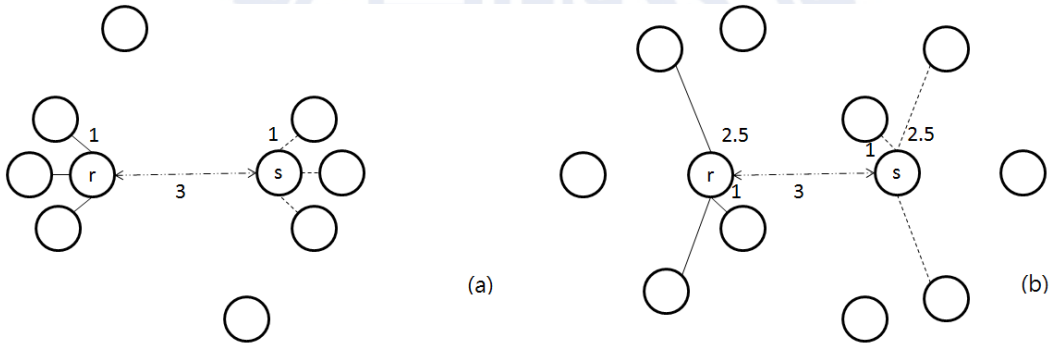


圖 3-2. 依照 K 最近鄰做為定義新權重的基礎之 3-nn 示意圖

說明：實線表示點 r 的 3-最近鄰集合，虛線表示點 s 的 3-最近鄰集合

然而，K 最近鄰演算法的成效是隨著輸入參數 K 的變化而有所不同，同樣以圖 3-2 為例，當我們改成尋找 1 最近鄰作為區域性鄰集而非 3 最近鄰時，圖 3-2(a)和圖 3-2(b)的鏈結權重同樣都為 3，經過相似度轉換後兩者的距離仍然相同，難以區別此兩組不同的資料分布情形。在多數的情況下，我們往往只能利用反覆測試(trial and error)尋找一個較合適的值，因此我們提出一個依照鏈結兩端點間的距離關係作為參考的區域性鄰集，稱為可互相包含最近鄰(mutual included nearest neighbors, MI-nn)，盡可能減少參數值對演算法效能帶來的影響。

3.1.2 依照鏈結兩端點間的距離做為定義新權重的基礎

首先，考慮圖上存在一個鏈結由任兩點 r、s 連接而成，定義此鏈結上點 r 的可互相包含最近鄰(MI-nn)為：

$$\mathrm{MI-nn(r)} = \{\mathrm{x} \mid \mathrm{dist(x,s)} < dist(\mathrm{r,s}) \wedge \mathrm{dist(x,r)} < dist(x,s)\}$$

可互相包含最近鄰的精神在於將鏈結兩端點間的距離視為相鄰半徑，以此半徑尋找每組鏈結其兩端點的區域性最近鄰。當二個資料點距離較接近，相鄰半徑較小，能夠包含到的區域性鄰集集合也越小，使得計算出的權重相對更低，能夠滿足我們提出的第一種鏈結權重假設；當二個資料點距離較遠離，相鄰半徑較大，能夠包含到的區域性鄰集集合也越大，更能從較大的區域觀察資料整體的分布，進一步動態區別兩種不同的資料分布情形。

我們重新定義鏈結權重的計算公式為[3]：

$$\mathrm{A_{r,s}} = \frac{1}{2}\left[\sum_{\mathrm{x}\in \mathrm{MI-nn(r)}} \frac{\mathrm{dist(r,s)}}{\mathrm{dist(r,x)}} + \sum_{\mathrm{x}\in \mathrm{MI-nn(s)}} \frac{\mathrm{dist(r,s)}}{\mathrm{dist(s,x)}}\right]$$

以圖 3-3 為例，尋找可互相包含最近鄰作為區域性鄰集。我們計算圖 3-3(a)(b)的鏈結權重，分別是：10.07 和 6.47，；再將新求出的權重值帶入至相似度轉換公式，可以得到一組新的距離，分別是：30.21 和 19.41，圖 3-3(a)中的距離同樣被

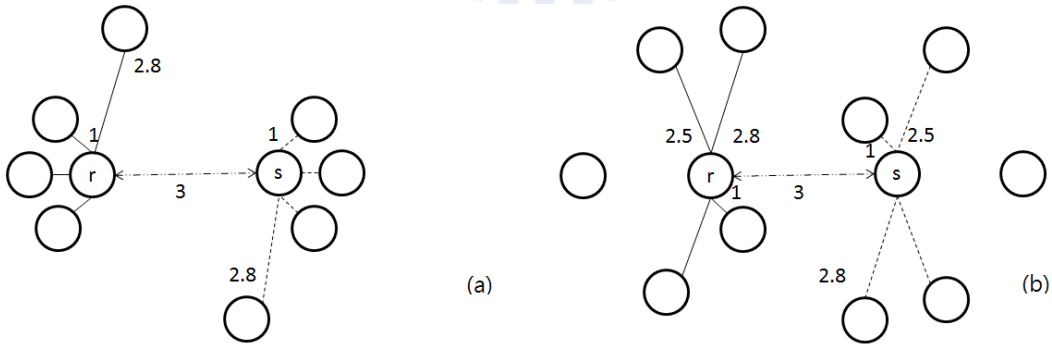


圖 3-3. 依照 MI-nn 做為定義新權重的基礎之示意圖

說明：實線表示點 r 的可互相包含最近鄰集合，虛線表示點 s 的可互相包含的最近鄰集合。

[3] MI-nn(r)表示資料點 r 依照鏈結兩端點間的距離關係作為定義之區域性鄰集集合

相對的增加。與以 k 最近鄰作為區域性鄰集的方法相比，我們同樣能夠保持相同的期望，而最大的差別在於省略一個可能會影響結果的參數 k。

目前設計用來描述鏈結權重的公式，是一種簡單且非常直覺的方法。然而，當圖上所有成對資料形成的鏈結經過相似度轉換後，資料間彼此的關係將無法滿足賦距空間(metric space)的條件。賦距空間必須滿足三個條件：1. 識別性(identity of indiscernible)，$\text{dist}(x,y) = 0 \leftrightarrow \text{dist}(y,x) = 0$；2. 對稱性(symmetry)，$\text{dist}(x,y) = \text{dist}(y,x)$；3. 三角不等式(triangular inequality)，$\text{dist}(x,y) \leq \text{dist}(x,z) + \text{dist}(z,y)$。轉換後的相似度將無法滿足三角不等式。若是原始資料的相似關係符合賦距空間的條件，我們可知任一組成對資料間的直接距離就是此組資料的最短路徑。因此，在經過相似度轉換後，改以最短路徑重新描述所有成對資料彼此的相似關係，將可儘量減少違反賦距空間條件的鏈結組合。

依照定義的鏈結權重公式求出圖上的所有鏈結權重，並利用相似度轉換公式得出一組全新的相似度矩陣後，帶入至非度量性多元尺度法，目標是得到一組與原始資料維度相同且滿足賦距空間條件的全新屬性。

3.2 非度量性多元尺度法(non-metric multidimensional scaling)

多元尺度法[17]是屬於非屬性基礎的方法(non-attribute-based approaches)，與因素分析(Factor Analysis)或區別分析(Discriminant Analysis)等屬性基礎的方法(attribute-based approaches)不同，只需擁有資料的相似關係即可達成目標。多元尺度法之主要目標是根據資料的相似關係，在一個人為選擇的特定維度空間內，使得資料在此空間內的實際歐式距離可以與相似關係保持一致。當相似關係不具備賦距空間的條件，改以非度量性多元尺度法，依順序尺度(ordinal)來比較資料的關係。Kruskal[18]提出一種演算法和壓力係數，壓力係數是一種數值用以檢驗在新維度空間內所找出的歐式距離關係與相似關係的一致性。圖 3-4 是由 Kruskal 提出的非度量性多元尺度法方法，目標是尋找一組能夠最小化壓力係數的空間分佈。根據圖 3-4 的步驟，當壓力係數達到收斂時，可找出一組資料分布情形。我

$\boldsymbol{\delta}_{rs}$：original dissimilarities between pairs of points

$\mathbf{d}_{rs}$：distance between pairs of points in the space

$\hat{\mathbf{d}}_{rs}$：$\hat{\mathbf{d}}$ is a measure of how well the distance **d** "matches" dissimilarity $\boldsymbol{\delta}$

1. Specify the number of ordination dimensions to be used.
2. Choose an initial configuration, $\mathbf{x}_0$.
3. Normalize the configuration to have its center at the origin and unit mean square distance from the origin.
4. Find $\{\mathbf{d}_{rs}\}$ from the configuration.
5. Fit $\{\hat{\mathbf{d}}_{rs}\}$.It was seen that the monotonic least squares regression of $\{\mathbf{d}_{rs}\}$ on $\{\boldsymbol{\delta}_{rs}\}$ partitioned $\{\boldsymbol{\delta}_{rs}\}$ into blocks in which the values of $\hat{\mathbf{d}}_{rs}$ were constant, and equal to the mean of the corresponding $\mathbf{d}_{rs}$ values.
6. Find the gradient $\frac{\partial S}{\partial x}$ and the new step length ***sl***.
7. Find the new configuration：$\mathbf{x}_{n+1} = \mathbf{x}_n - \boldsymbol{sl}\frac{\frac{\partial S}{\partial x}}{\left|\frac{\partial S}{\partial x}\right|}$

Go to step 4 until stress is invariant to translation.

圖 3-4. Kruskal's iterative technique[18]，尋找一組能夠最小化壓力係數的空間分佈。

表 3-1. Kruskal's stress 計算公式與對應之合適程度[19]

$$S = \sqrt{\frac{\sum_{r,s}\left(d_{r,s} - \hat{d}_{r,s}\right)^2}{\sum_{r,s} d_{r,s}^2}}$$

Kruskal's stress	Quality
0.200	Poor
0.100	Fair
0.050	Good
0.025	Excellent
0.000	Perfect

們使用壓力係數來觀察新找出的資料分布是否與相似度矩陣的一致，不同的壓力係數有各自代表的合適程度，壓力係數越小表示與相似矩陣越一致。壓力係數的公式與合適程度如表 3-1。

當資料依照 3.1 節提出的鏈結權重公式完成相似度轉換後，彼此的相似度關係已經被更改，原始的資料屬性便失去意義，且資料彼此的關係將不再具備賦距空間的條件。我們的目標是設法將新產生的相似關係，在所選擇的維度空間內建立一組全新的屬性關係，同時滿足賦距空間的條件。選用非度量性多元尺度法能夠輕易達成當前的目標。

如何選擇非度量性多元尺度法所建構的空間維度是一項重要的問題，我們通常選擇建構於相對容易視覺化的二維空間。當選擇建構的空間維度小於原始的資料維度時，尋找新屬性的過程則等同維度縮減，與主成分分析法(principle component analysis)類似。維度縮減的優點主要有下列二項：1. 可藉由降低資料維度來大幅減少計算量；2. 可將資料投影到更低維度的子空間(subspace)，幫助使用者容易形象化欲分析的資料。以非度量性多元尺度法做為維度縮減的方法，在維度縮減的過程中，我們必須觀察新產生的屬性關係其壓力係數是否能滿足表 3-1 的合適程度，壓力係數必須盡可能越小。當壓力係數不小於 0.05 時，說明此組新的屬性無法完整的描述相似矩陣上的關係，我們可以從中得知將欲建構的空間維度數設定得過低。透過反覆測試可以尋找一個更恰當的維度數目。

除了將空間維度縮減至可滿足壓力係數的條件下，我們亦可考慮建構於原始空間維度數相等的屬性空間，這樣的方法可以視為將資料依據相似度轉換後的結果對應於相同維度的空間內做重新排列，使得資料分布情形可以更真實地反映資料相互關係。我們首先將建構的資料維度數設定和原始維度數相同，原因在於維度縮減勢必會損失部分資訊，且一個恰當的維度數目是難以尋找，更為耗費時間；接著在不影響效能的情況下，我們再嘗試縮減資料維度。

3.3 尋找相似度轉換的最佳解

透過相似度轉換公式與非度量性多元尺度法的幫助，我們能夠學習出一組新的屬性關係。然而，僅對資料集做一次的相似度轉換，有時無法說明新的屬性是否能真實反映資料間的相似關係。倘若相似度轉換無法滿足真實的相似關係，我們對新屬性再次進行相似度轉換，透過連續迭代轉換，期望資料之間的關係能更接近真實。對於不同的資料集，所需要的相似度轉換次數未必相同，因此相似度轉換所需要的次數成為重要的問題。

相似度轉換的過程可被視為一種非監督式學習，在轉換的過程並無參考其它外部資訊，無需透過外部資訊的幫助即可尋找相似度轉換所需的次數。我們結合分群演算法與相似度轉換，每當完成相似度轉換後，隨即將新產生的屬性代入至分群演算法產生新的分群結果，協助我們判斷一個較為合理的相似度轉換次數。如何選擇適當的轉換次數，我們的假設是：當轉換前和轉換後的屬性所分別產生的二組分群結果呈現高度相似時，表示相似關係並無受轉換公式的影響產生過多的變化，可推測已經找到一組穩定的相似關係，無需再繼續做更多次的相似度轉換，因此，選擇轉換前的屬性關係做為相似度轉換的最佳解。我們提出一種符合上述假設的停止條件：比較前後兩次相似度轉換產生的新屬性其分群結果，當用來評估兩分群結果相似程度的分數高於門檻值時，表示已找出一個恰當的相似度轉換次數。

用來評估分群相似度的方法有許多種，例如：Rand Index[19]或 Adjusted Rand Index[20]，然而，多數方法所求出的分數往往只具有相對的比較意義，並無法直接說明相似的程度。我們的目標是在迭代的過程中找出一個停止點，因此需要一種具有實質意義的分數來訂定符合停止條件的門檻值。.

Cohen’s Kappa 一致性係數[21]是一種統計學中測量信度的方法，用來表現重覆測量的一致性，求出的分數將具有實質的意義，最常見的應用在於醫學臨床診斷之一致性判斷或社會科學的研究上。適用 Cohen’s Kappa 一致性係數的情況

主要有下列兩種：

1. 人際信度(inter-rater reliability)：評估兩種檢驗方法的結果是否一致

2. 人內信度(intra-rater reliability)：評估相同檢驗方法重覆量測的結果是否一致

Cohen's Kappa 一致性係數的公式如下：

$$\mathcal{K}appa = \frac{P_0 - P_c}{1 - P_c}$$

P_0 為觀測一致性(observed agreement)，表示前後兩種測量結果一致的百分比；P_c 為期望一致性(chance agreement)，表示前後兩種測量結果預期相同的機率。參考表 3-2，使用列聯表(contingency table)描述分類結果，則 P_0 和 P_c 的公式如下：

$$P_0 = \frac{\sum_{i=1}^{N} O_{ii}}{N}, P_c = \frac{1}{N}\sum_{i=1}^{N} \frac{C_i \times R_i}{N}$$

Cohen's Kappa 係數將以百分比表示，範圍介於-1 至 1 之間，通常會落在 0 至 1 之間。1977 年由 Landis 和 Koch [22]提出五種不同數值範圍所分別對應的一致性等級，如表 3-3 所示。

表 3-2. 使用列聯表描述分類結果

Class Label	$\mathbf{v_1}$	…	$\mathbf{v_N}$	Sums
$\mathbf{u_1}$	$\mathbf{O_{11}}$	…	$\mathbf{O_{1N}}$	$\mathbf{R_1}$
⋮	⋮		⋮	⋮
$\mathbf{u_N}$	$\mathbf{O_{N1}}$	…	$\mathbf{O_{NN}}$	$\mathbf{R_N}$
Sums	$\mathbf{C_1}$	…	$\mathbf{C_N}$	N

我們使用 Cohen's Kappa 一致性係數比較前後兩次分群結果相似程度，然而 Cohen's Kappa 一致性係數主要應用的領域卻是比較分類結果的一致性，其類別都具有實質意義。以醫學診斷為例，假設兩位醫師先後診斷八位病患，而每位病

患都有兩種可能的診斷結果，分為良性或惡性。以表 3-4(a)說明兩位醫師的診斷結果，將診斷結果使用列聯表呈現，我們可以算出兩位醫師診斷的觀測一致性 P_0 是 0.625、期望一致性 P_c 是 0.5，則 Cohen's Kappa 一致性性係數為 0.25。將 Cohen's Kappa 一致性係數應用於分類領域上，每一個評估者(rater)被賦予的類別都具有實質意義，我們不能恣意修改分類結果，例如將醫師 1 的診斷結果中所有屬於惡性的病患替換成良性、良性替換成惡性，新的觀測一致性 P_0 為 0.375、期望一致性 P_c 為 0.5，使得 Cohen's Kappa 一致性性係數變更成-0.25。恣意修改分類結果等同於錯誤分類，且會造成 Cohen's Kappa 一致性性係數顯著不同。我們將上述問題延伸至分群，醫師的診斷結果並非用於良性與惡性病患之分類，而是將相同病症的病患分成多個分群，表 3-4(b)將診斷結果一致的病患以相同的數字標記。對照表 3-4(a)，我們發現醫師 1 的診斷結果中被標記為 1 的病患屬於良性、0 則是屬於惡性，而在醫師 2 的診斷結果中屬於良性的記號是 0、惡性則是 1。二位醫師產生不一致的診斷結果，正如同將表 3-4(a)中醫師 1 的診斷結果顛倒。我們已知分群結果所賦予的分群記號並無實質意義，若是直接使用分群記號做為類別計算 Cohen's Kappa 一致性係數，會顯著影響 Cohen's Kappa 一致性係數的數值。因此在 2005 年由 Reilly 等[23]學者提出一種新的方法延伸 Cohen's Kappa 一致性係數，使其能應用於分群領域。

2005 年由 Reilly 等[23]學者提出一種延伸自 Cohen's Kappa 一致性係數的方法，命名為 K_{max}，目的是修正 Cohen's Kappa 一致性係數使其能夠應用在分群領

表 3-3. Cohen's Kappa 一致性係數等級[22]

Kappa	Rank of agreement
0.00 ~ 0.20	Slight
0.21 ~ 0.40	Fair
0.41 ~ 0.60	Moderate
0.61 ~ 0.80	Substantial
0.81 ~ 1.00	Almost perfect agreement

表 3-4(a). 分類結果應用於 Cohen’s Kappa 一致性係數

病患	A	B	C	D	E	F	G	H
醫師 1	良	良	惡	惡	惡	惡	良	良
醫師 2	良	良	良	惡	惡	良	良	惡

表 3-4(b).分群結果應用於 Cohen’s Kappa 一致性係數

病患	A	B	C	D	E	F	G	H
醫師 1	1	1	0	0	0	0	1	1
醫師 2	0	0	0	1	1	0	0	1

域。由於分群結果所賦予的類別記號並無實質意義，我們可以隨意改變分群的類別編號而不影響分群結果。K_{max} 方法的目標在於最佳化分群一致性，我們從中挑選出一組可以最大化 Cohen’s Kappa 一致性係數的類別排列方式，並以此時計算出的 Cohen’s Kappa 一致性係數做為 K_{max} 的最佳解。

沿用表 3-4(a)的例子，我們任意賦予診斷結果一個類別編號，屬於相同類別之診斷結果以相同的數字標記。在只有二位醫師的情況下，每位醫師都有 2 階乘種排列方式，共計 4 種類別排列方式，如表 3-5 所示。

由於情況 1 和情況 4、情況 2 和情況 3 屬於相同的排列方式，我們僅說明情況 1 和情況 2 對於 Cohen’s Kappa 的影響。表 3-6 將表 3-5 中的情況 1 和情況 2 其類別排列方式以列聯表呈現。列聯表中每個欄位分別代表的意義是：

a. 表示被醫師 1 和醫師 2 歸類為第 0 群的病患數量

b. 表示被醫師 1 歸類為第 0 群，但被醫師 2 歸類為第 1 群的病患數量

c. 表示被醫師 1 歸類為第 1 群，但被醫師 2 歸類為第 0 群的病患數量

d. 表示被醫師 1 和醫師 2 歸類為第 1 群的病患數量

在表 3-6(a)中，二位醫師都將良性病患其編號設定為 0，惡性病患的編號設定為 1；而在表 3-6(b)中，醫師 1 仍然將良性病患的編號設定為 0，惡性病患的編號設為 1，但醫師 2 對病患的編號設定卻與醫師 1 不同。考慮兩種情況的觀測一致性 (observed agreement, P_0)，即加總列聯表中位於對角線的數值。我們發現表 3-6(a)的觀測一致性明顯大於表 3-6(b)，因此可求得較高的 Cohen’s Kappa 一致性係數，

說明表 3-5 的情況 1 是讓分群結果最為一致的排列方法，能夠充分解釋二位醫師對於診斷的共識。

我們使用 K_{max} 方法做為判斷相似度轉換的迭代過程是否停止的條件，將門檻值(threshold)設定在 0.9 以上，表示兩組分群結果呈現的一致性必須滿足表 3-3 中的一致性等級，達到”Almost perfect agreement”的水準之上。

完成迭代的停止條件設計，我們將相似度轉換公式與非度量性多元尺度法結

表 3-5. 類別記號可能的排列方式

	醫師 1		醫師 2	
	良	惡	良	惡
情況 1	0	1	0	1
情況 2	0	1	1	0
情況 3	1	0	0	1
情況 4	1	0	1	0

表 3-6(a). 隨意排序分群結果之示意圖，情況 1

情況 1		醫師 2		
		0(良性)	1(惡性)	總數
醫師 1	0(良性)	[a] 3	[b] 1	4
	1(惡性)	[c] 2	[d] 2	4
	總數	5	3	8

表 3-6(b). 隨意排序分群結果之示意圖，情況 2

情況 2		醫師 2		
		0(惡性)	1(良性)	總數
醫師 1	0(良性)	[a] 1	[b] 3	4
	1(惡性)	[c] 2	[d] 2	4
	總數	3	5	8

合。其中，用於判斷一個合適的相似度轉換次數的分群演算法，我們採用 K-means 或 K-medoids 演算法。由於目標是比較前後兩次分別產生的分群結果的相似程度來判斷是否需要繼續向下迭代，分群結果必須盡可能不受離群值的影響。K-medoids 演算法與 K-means 演算法相比，對離群值的敏感性較低，因此選用 K-medoids 作為分群演算法；然而，少數情況下，即便選用 K-medoids 演算法仍然無可避免受到離群值的影響。

K-medoids 演算法的目標是由多個不同的初始實心出發，從中挑選一組能夠最佳化目標函數的分群結果做為最佳解。若是離群值被挑選至初始實心集合中，難以藉由 K-medoids 演算法的實心迭代步驟修正分群結果，依然會產生不平衡的分群。在某些情況下，錯誤的分群結果也許會擁有最佳的目標函數值。儘管離群質的數量稀少，仍然有機率被挑選至初始實心集合中，若是不幸被挑進初始實心集合便會造成不平衡的分群。離群值的數量必定是相對稀少，因此我們認為不平衡的分群結果是較難出現在較為頻繁的分群集合之中。為了避免受到離群值的影響，我們提出一種改進方法，稱為 ferq K-medoids，它是修改 K-medoids 演算法中尋找最佳解的方法，將最佳化目標函數的步驟替換成尋找最頻繁出現的分群結果。由於挑選初始實心的方式採用全隨機，自不同的初始實心出發，倘若有較多次數能收斂至相同的實心，足以說明分群結果並非偶然。目標函數扮演的角色並非完全失去舞台，仍然有其存在的必要。雖然我們不再參考目標函數以挑選最佳分群結果，但在資料分類的過程中，依然是遵循目標函數，將資料分類至能夠最佳化目標函數的分群，使得分群結果能符合預期的假設。

圖 3-5 是尋找相似度轉換最佳解的虛擬碼(pseudo code)。其中，非度量性多元尺度法使用 R statistics[24]提供的 isoMDS 函式，isoMDS 實作由 Kruskal[18]提出的演算法，時間複雜度是 $O(n^2)$ ，n 是資料總數；分群演算法則是採用修改後的 K-mediuds 演算法，時間複雜度是 $O(nkt)$，n 是資料總數，k 是分群總數，t 是迭代次數。

Algorithm: Similarity Transform

Input: Original data **Output:** Transformed data

Method:

1. $\mathbf{D_0} \leftarrow$ Original data
2. $\mathbf{w} \leftarrow 0$
3. Finding clustering result, $\mathbf{C_w}$, by freq K-medoids.
4. **repeat**

4a. $\mathbf{w} \leftarrow \mathbf{w}+1$

4b. Find all pairs of link weight, **A**, and transform.

4c. Use Kruskal's nMDS to generate new attribute, $\mathbf{D_w}$.

4d. Find the clustering result, $\mathbf{C_w}$, by freq K-medoids.

4e. Calculate $\mathbf{K_{max}}$ between $\mathbf{C_{w-1}}$ and $\mathbf{C_w}$.

4. **until** $\mathbf{K_{max}} > 0.9$
5. Transformed data $\leftarrow \mathbf{D_{w-1}}$

Algptithm: freq K-medoids

Input：Data

Method:

1. Let C_1, C_2… C_k be the initial cluster center.
2. **repeat**

2a. Assign data point to cluster **h** so that objective function value is minimized

2b. $\mathbf{h} = \arg\min\left(\sum_{r\in V}(r - C_r)^2\right)$

2c. For each cluster **h**, update its medoids.

2. **until medoids remain unchanged**

圖 3-5. 尋找相似度轉換的最佳解流程和 freq K-medoids 方法

3.4 結合半監督式分群演算法

將使用者提供的配對限制(paiewise constraints)整合至分群演算法中，這種方式稱為半監督式分群演算法。我們將經由相似度轉換方法產生的新屬性與屬於 search-based 的半監督式分群法結合，進一步提升分群的準確性，同時利用配對限制彌補相似度轉換法所採用的假設其可能遺漏的部分資訊。Cop K-means[8]是 search-based 半監督式分群法的代表之一，它是改良自 K-means 演算法，在尋找分群的過程中，資料被分類至任一分群時皆不可違背配對限制，進而影響分群的過程與結果。然而，這樣的做法完全依賴配對限制，若是使用者提供的配對限制之中存在少許錯誤，將會引導分群演算法至錯誤的結果。參考 2002 年由 Kleinberg 和 Tardos[25] 提出的代價函式，由於 Kleinberg 和 Tardos 提出的方法僅考慮 MUST-link 類型，我們進一步加入 CANNOT-link，延伸設計一組基於參考分群中心的代價函式，將其與目標函式合併，使得在最佳化目標函式的過程中，同時能最小化違反配對限制的資料數量，並判斷使用者所提供的配對限制是否合適。

我們首先定義下方用來描述代價函式的名詞與其表示方法。1. 令$\mathcal{M}$表示 MUST-link 所形成的集合，$\mathcal{C}$表示 CANNOT-link 所形成的集合；2. 存在任意兩點 r、s，連接此二點的鏈結以link(r, s)表示；3. 令 CSET 表示中心所形成的集合；4. L(r)、L(s)分別表示所屬的分群；5. C_r、C_s分別表示點 r 與點 s 所在分群對應的中心。

考慮成對資料與配對限制的關係，基於參考分群中心的代價函式必須考慮下列兩種情況，分別是：違反 MUST-link 和違反 CANNOT-link。

- 定義違反 MUST-link 的代價函式：

$$\text{Cost}_{\mathcal{M}} = \frac{1}{2}[\text{dist}(r, C_s) + \text{dist}(s, C_r)],$$

$$\text{if link}(r, s) \in \mathcal{M} \wedge L(r) \neq L(s)$$

考慮違反 MUST-link 的情形，亦即任兩筆資料彼此關係屬於 MUST-link 卻被分類至不同的分群中。若是要滿足正確的配對限制，最直接的方法是改變其中一筆

資料所屬分群，強迫將兩筆資料歸類至相同分群。因此，違反配對限制所需的代價被設定為：鏈結上的任一點 r，與另一點 s 所屬的中心 C_s，此兩點的距離關係 $dist(r,C_s)$。代價函式等同於計算符合正確配對限制的目標函數值。我們將同時考慮鏈結上的兩點，最後取兩者的平均值。

- 定義違反 CANNOT-link 的代價函式：

$$\mathrm{Cost}_{\mathcal{C}} = \frac{1}{2}\left[\min_{\forall x \in \mathrm{CSET}, x \neq C_r} \mathrm{dist}(r,x) + \min_{\forall x \in \mathrm{CSET}, x \neq C_s} \mathrm{dist}(s,x)\right],$$
$$\text{if } \mathrm{link}(r,s) \in \mathcal{C} \wedge \mathrm{L}(r) = \mathrm{L}(s)$$

考慮違反 CANNOT-link 的情形，亦即任兩筆資料彼此關係是屬於 CANNOT-link 卻被分類至相同的分群中。與違反 MUST-link 的情況相同，代價函式的目的在於計算符合正確配對限制的目標函數值。若是要滿足正確的配對限制關係，最直接的方法是將鏈結上的其中一點移動至剩餘分群中最接近的一群，強迫將兩筆資料歸類至不同分群。因此，違反配對限制所需的代價將被設定為：鏈結上的任一點 r，考慮扣除 C_r 後從 CSET 中剩餘的質心 C_x，挑選出兩點距離關係 $dist(r,C_x)$ 最小者。相同的，我們仍需考慮鏈結上的兩點，最後取兩者的平均值。

完成違反 MUST-link 和違反 CANNOT-link 之代價函式的定義，我們將代價函式合併至原始的目標函數中：

$$\mathrm{Obj}' = \sum_{r \in V} (r - C_r)^2 + \sum_{\mathrm{link}(r,s) \in \mathcal{M}} \mathrm{Cost}_{\mathcal{M}} + \sum_{\mathrm{link}(r,s) \in \mathcal{C}} \mathrm{Cost}_{\mathcal{C}}$$

我們將此方法命名為：freq K-medoids with cost function，如圖 3-6。雖然在 3.3 節中將 K-medoids 方法稍做修改為 freq K-medoids，不再以尋找最佳化目標函數的結果做為最佳解，改用最頻繁出現的結果當作最佳解，但在資料分類的過程中，採用合併代價函數後的目標函數，依然可依據配對限制提供的資訊將資料分類至更為正確的分群，最小化違反配對限制的數量。

Algorithm: freq K-medoids with cost function

Input：Data, $\mathcal{M}\leftarrow$ MUST-link constraints, $\mathcal{C}\leftarrow$ CANNOT-link constraints

Method:

1. Let C_1, C_2… C_k be the initial cluster center.
2. **repeat until convergence**

2a. Assign data point to cluster **h** so that objective function value is minimized.

2b. $\mathbf{h} = \arg\min\left(\sum_{r\in V}(r - C_r)^2 + \sum_{link(r,s)\in\mathcal{M}} Cost_{\mathcal{M}} + \sum_{link(r,s)\in\mathcal{C}} Cost_{\mathcal{C}}\right)$

2c. For each cluster **h**, update its medoids.

2. **until medoids remain unchenged**

圖 3-6. freq K-medoids with cost function 方法

第四章，實驗與討論

在此章節中，我們將對此篇論文提出的以區域最近鄰為基礎的相似度轉換方法，進行下方的幾項實驗：

1. 非監督式分群演算法比較：

   將相似度轉換方法與非監督式分群演算法結合，使用在 3.3 節提出的 freq K-medoids 演算法。實驗分為兩個部分：第一，探討 k-nn 和 MI-nn 兩種尋找區域性鄰集方法用於相似度轉換，其分群的準確率與方法優劣比較。第二，與現行存在的方法比較分群的準確性。

2. 半監督式分群演算法比較：

   將相似度轉換方法與半監督式分群演算法結合，使用 3.4 節提出的 freq K-medoids with cost function 演算法，和現行存在的方法比較分群的準確性。

3. 資料維度個數對於相似度轉換的影響：

   比較經過相似度轉換後，使用非度量性多元尺度方法尋找新的投影空間的過程，挑選不同空間維度數的影響程度。

4.1 資料與前處理

此章節中使用的資料集分成兩種：1. 7 組人工設計的二維資料集；2. 10 組來自 UCI 資料庫[26]蒐集的真實世界資料集。

自真實世界中收集而成的資料往往是不完整的，例如：資料遺漏某些屬性、資料集包含重複的資料或屬性、相同的資料卻擁有不同的類別，這些都是影響分群演算法準確性的因素，因此，對資料做前處理是必要的。前處理的步驟分為兩個部分：1. 資料整合；2. 資料正規化。資料整合的目的在於移除資料不一致與資料重複性，我們移除資料集中重複出現的資料、屬性，並移除所有屬性都相同而類別卻不同的資料，確保前處理後的資料集中，每筆資料都是唯一存在且所屬的類別不互相矛盾。由於資料集中每種屬性所代表的意義並不相同，資料正規化

有其必要性。資料正規化能將屬性重新描述至一個特定區間內，同時保有原始的相對關係。資料正規化的方法有許多種，在此我們選擇極值正規化(min-max normalization)[1]。極值正規化對原始資料屬性進行線性轉換，假設 $\min_A$ 和 $\max_A$ 分別是屬性 A 的極小值與極大值，極值正規化的公式為：

$$v' = \frac{v - \min_A}{\max_A - \min_A}(new\_max_A - new\_min_A) + new\_min_A$$

將任一筆資料的屬性 A 重新投射至區間$[new_min_A, new_max_A]$中，我們將區間的範圍設定在 0 與 1 之間。

圖 4-1 是此章節中實驗所使用的 7 組人工設計資料集之分布情形；表 4-1 是 10 組 UCI 真實世界資料集經過前處理後的詳細資料。

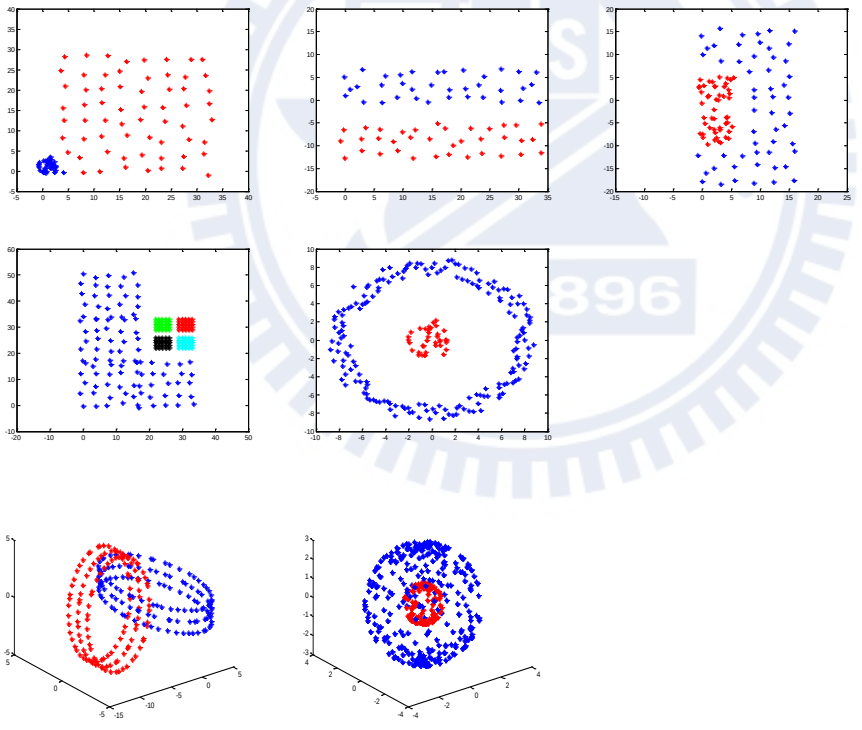


圖 4-1. 人工設計資料集分布

說明：由左至右，由上至下，資料集的名稱分別是：d1、d2、d3、d4、d5、d6、d7。

表 4-1. UCI 資料集

	Num of instances	Num of dimensions	Num of classes
Iris	147	4	3
Wine	178	13	3
Glass	213	9	6
Balance	625	4	3
Ionosphere	351	34	2
Breast cancer	449	9	2
WDBC	569	30	2
Soybean	47	21	4
Segmentation	210	19	7
Pima diabetes	768	8	2

4.2 評量方法

Rand index(RI)[18]是一種用來衡量兩個分群相似程度的分數，常用於統計或資料探勘等領域上，在此章節中，我們選擇 RI 做為評量的標準。

RI 觀察分群結果中成對資料彼此的關係，衡量兩分群的相似程度。令 $S = \{s_1, s_2, s_3, \ldots\ldots, s_n\}$ 表示個數為 n 的資料集；令 $X = \{X_1, X_2, X_3, \ldots\ldots, X_P\}$ 與 $Y = \{Y_1, Y_2, Y_3, \ldots\ldots, Y_q\}$ 分別代表兩組不同的分群結果，兩者都是 S 的子集合，分群個數分別是 p 和 q。自S中任意抽取兩筆資料所形成的成對組合(s_i, s_j)，觀察兩個不同分群結果中的關係，主要可分為下列四種：

1. 在 X 中被歸類至相同分群且在 Y 中被歸類至相同分群
2. 在 X 中被歸類至相同分群卻在 Y 中被歸類至不同分群
3. 在 X 中被歸類至不同分群卻在 Y 中被歸類至相同分群
4. 在 X 中被歸類至不同分群且在 Y 中被歸類至不同分群

以 A、B、C、D 分別表示上述四種情形出現的次數，成對組合的總數是 $\binom{n}{2} = A + B + C + D$，則 RI 的公式可被定義為：

$$RI = \frac{A + D}{A + B + C + D} = \frac{A + D}{\binom{n}{2}}$$

理想的 RI 值將介於 0 至 1 之間，分數越高表示兩個分群的相似程度越高，當兩種分群結果完全吻合時，RI 值為 1。以數學意義的觀點來看，RI 相當於計算整體準確度(overall accuracy)。

實驗的設計方式，對於任何一組資料集，我們都將資料集隨機切割成兩個部分，分為測試集和訓練集。測試集的主要用途則在於檢測分群的準確性，以 RI 當做評量標準；訓練集則是用於當相似度轉換方法與半監督式分群演算法結合時，其所需的配對限制將是自訓練集中隨機抽取一定數量的鏈結而成。對於不同資料集的實驗，我們都會產生 20 組不同的訓練集和測試集，取此 20 組測試集的平均值做為最終的答案。

4.3 實驗

4.3.1 非監督式分群演算法比較

資料集透過相似度轉換方法能夠將彼此的相似關係重新描述，而新的相似關係受制於區域性鄰集的定義。此篇論文中提出了兩種尋找區域性鄰集的方法，分別是 K 最近鄰(K-nn)和可互相包含最近鄰(MI-nn)。在此小節中，我們首先比較 K 最近鄰(K-nn)和可互相包含最近鄰(MI-nn)兩種尋找區域性鄰集方法用於相似度轉換，其分群的準確率與方法優劣比較；接著在與現行存在的非監督式分群演算方法比較分群的準確性。

4.3.1.1 以 K 最近鄰(K-nn)和可互相包函最近鄰(MI-nn)為基礎之相似度轉換方法比較

我們分別使用 K 最近鄰(K-nn)和可互相包函最近鄰(MI-nn)做為尋找區域性鄰集的方法，藉由 3.3 節中提出的相似度轉換流程重新找出屬性關係。相似度轉換的目標在於辨別任意形狀的資料分布，我們先對七組人工資料集進行實驗，其中 d1 至 d5 是分布在二維空間，而 d6 和 d7 則是分布在三維空間內。觀察七組人工設計資料集，d1、d2 是屬於凸狀(convex)圖形，但是 d1 存在分群大小不均衡的情況，而 d1 和 d2 同時存在分群密度不均衡的情況；d3 至 d7 則是屬於凹狀(concave)圖形，其中 d4、d5 和 d7 同樣存在分群大小不均衡的情況，而 d3 和 d4 亦同時存在分群密度不均衡的情況。

以 K 最近鄰為基礎的相似度轉換方法在執行前必須設定參數 K，參數 K 的設定非常敏感，此部分的實驗我們分別嘗試 K=1、3、10，三種不同的數值。

我們依照 3.3 節提出的流程尋找相似度轉換的最佳解，針對每一組資料集，尋找適當的相似度轉換次數，如表 4-2。完成相似度轉換後，將產生的新屬性代入至 freq K-medoids 演算法中，並使用 20 組隨機抽取的測試集評估分群準確性，實驗結果如圖 4-2 和表 4-3。實驗結果顯示以 K 最近鄰為基礎之相似度轉換方法

確實會受到參數 K 的影響，少部分資料集雖然在某個特別的參數 K 可以正確分群，但所有資料集未必適用於同個參數 K，尋找合適的參數 K 成為難題。

表 4-2. 7 組人工資料集，使用以 MI-nn、1-nn、3-nn、10-nn 為基礎之相似度轉換次數

Dataset	d1	d2	d3	d4	d5	d6	d7
MI-nn	1	1	1	4	1	1	1
1-nn	1	3	0	2	1	5	1
3-nn	2	2	0	4	4	2	1
10-nn	1	2	0	3	1	2	1

最後，我們觀察 d6 和 d7 兩組分布於三維空間的資料集，資料分布經由可互相包函最近鄰(MI-nn)為基礎之相似度轉換的變化情況，如圖 4-3。

d6 和 d7 兩組資料集的分布形態如圖 4-3(a)，我們首先依照 3.3 節提出的流程，對原始資料進行一次的相似度轉換，新產生的屬性分布形態如圖 4-3(b)；接著比較轉換前(圖 4-3(a))與轉換後(圖 4-3(b))的分群相似程度，將兩者進行分群後產生的兩組分群結果並不相似，Cohen’s Kappa 一致性數必定小於門檻值。我們的目標是找出一組恰當的相似度轉換次數，新的屬性關係必需不受相似度轉換而產生劇烈變化，因此，再進行第二次的相似度轉換，新的屬性分布形態如圖 4-3(c)。我們同樣比較轉換前(圖 4-3(b))與轉換後(圖 4-3(c))的分群相似程度，兩者的分群結果呈現高度相似且 Cohen’s Kappa 一致性數符合設定之門檻值，因此我們認為 d6 和 d7 所需的相似度轉換迭代次數皆為一次，選擇圖 4-3(b)之屬性關係做為相似度轉換的最佳解。

圖 4-3 說明，經過相似度轉換後，可以明顯的改變資料分布情形，使得新的資料分布關係不僅能夠符合正確分群，同時凸顯集群之間的邊界。

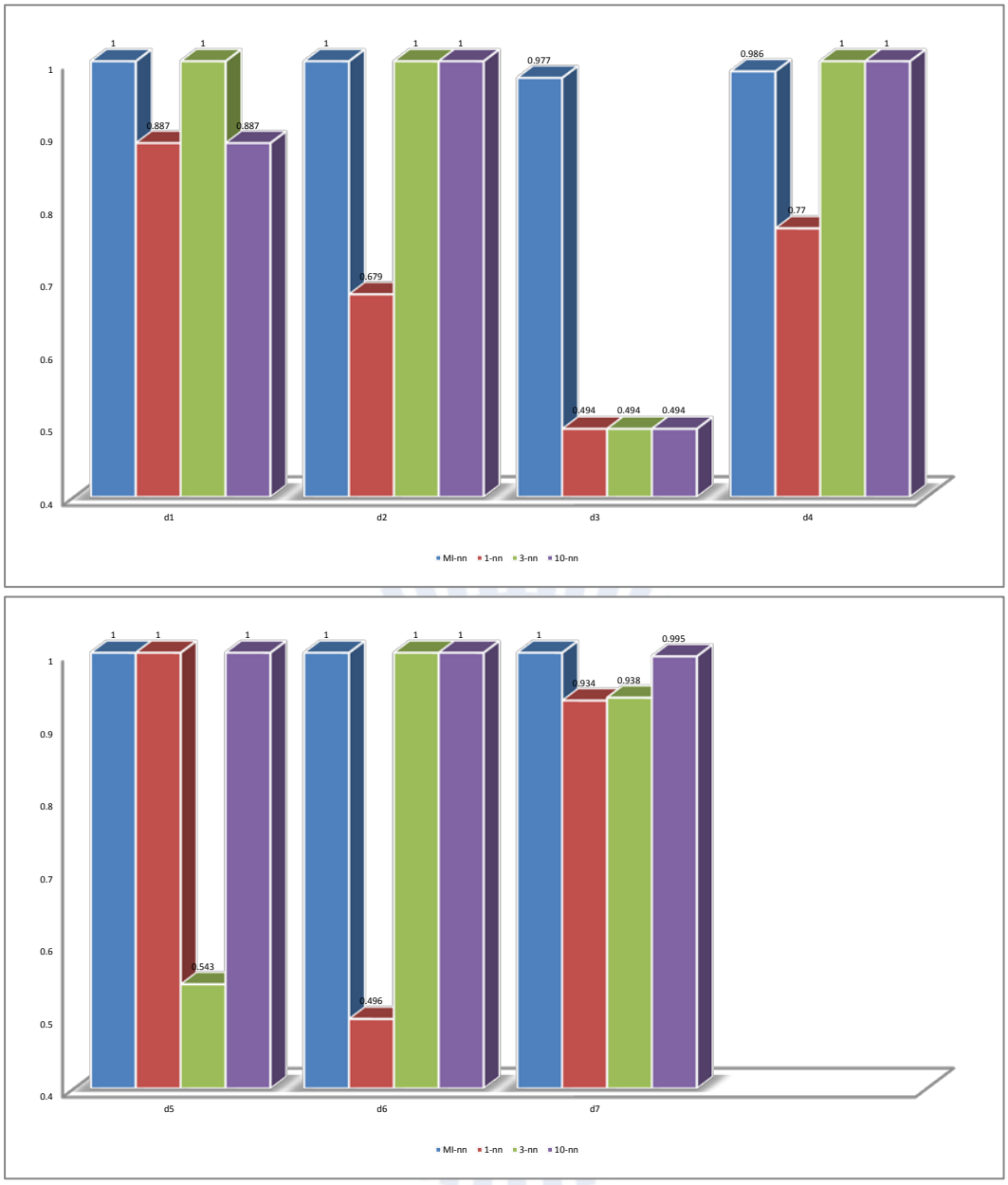
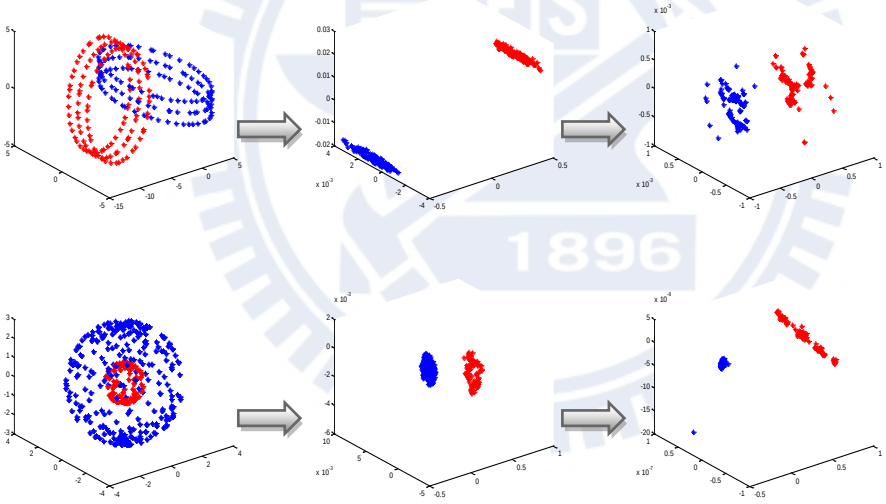


圖 4-2. 使用以 MI-nn、1-nn、3nn、10nn 為基礎之相似度轉換應用至 freq K-medoids 演算法，7 組人工資料集的 RI 比較

表 4-3. 使用以 MI-nn、1-nn、3nn、10nn 為基礎之相似度轉換應用至 freq K-medoids 演算法，7 組人工資料集之 RI 平均值與 Wilcoxon signed rank test 分析

Method	Average RI
MI-nn	0.995
1-nn	0.751
3-nn	0.854
10-nn	0.911

Method comparison	p-Value
MI-nn vs. 1-nn	0.0313
MI-nn vs. 3-nn	0.25
MI-nn vs. 10-nn	0.375



(a) (b) (c)

圖 4-3. 相似度轉換過程中資料分布之變化情形，選擇圖 4-2(b)做為相似度轉換的最佳解

說明：由上至下分別是 d6 和 d7，兩組人工設計資料集

4.3.1.2 其他非監督式分群演算法比較

我們將經過相似度轉換後的全新屬性帶入 freq K-medoids 演算法，接著再與另外數種常見的分群演算法做比較，它們分別是 K-means[1]、K-medoids[4]、CLUTO[5](一種實作 Chameleon[5]演算法的工具)以及 DBSCAN[6]。

K-means 使用由 R statistics[23]提供的函式，其所需的參數僅有分群個數 K；CLUTO[5]使用作者公布的原始碼，其所需的參數相當多種，包含：分群方法(clmethod)、相似度函式(sim)和 K 最近鄰個數(nnbrs)。此部分的實驗將分群方法(clmethod)設為 graph、相似度函式(sim)設為歐式距離，由於 CLUTO[5]採用 K 最近鄰圖動態調整相鄰半徑，最主要影響分群結果的參數即是 K 最近鄰個數(nnbrs)，我們分別測試了三種不同的參數值，分別是：10、25、40(CLUTO 之預設值)；DBSCAN[6]採用 2001 年由 Daszykowski 等[27]提出的改進方法。由於 DBSCAN 對輸入的兩項參數(半徑 ε 和 MinPts)非常敏感，尤其是半徑 ε，因此 Daszykowski 等[27]提出一種新的方法，經由設定的 MinPts 來尋找更為適當的半徑 ε，降低輸入的參數個數並同時減少演算法對參數的依賴性，在此我們分別將 MinPts 設定為 3、5、10。接下來的實驗我們將分別比較人工設計資料集與真實世界資料集。

人工設計資料集的實驗部分，我們同樣依照 3.3 節提出的流程尋找相似度轉換的最佳解，針對每一組資料集尋找適當的相似度轉換次數，如表 4-2。接著將經過相似度轉換後產生的新屬性應用至 freq K-medoids 演算法中，並使用 20 組隨機抽取的測試集同步評量其他 3 種非監督式分群演算法，實驗結果如圖 4-4 和表 4-4。上個小節已說明以可互相包函最近鄰(MI-nn)為區域最近鄰的相似度轉換方法在處理分群資料大小不均衡、密度不均衡的凸狀圖形或凹狀圖形時，都能夠較接近正確的分群，而在與其他分群的比較實驗中，其結果亦顯示我們所提出的方法優於其餘三種演算法。接著觀察 K-means 和 K-medoids 演算法產生的結果，由於兩者設計的目標都是解決凸性最佳化問題(convex optimization problem)，因此並不擅於處理上述的情況，特別是凹狀形狀的資料集 d5 和 d7，它僅能將圖形

水平一分為二，難以找出一內一外的兩個分群。繼續觀察 CLUTO 的實驗結果，雖然 CLUTO 已被實驗證明能夠解決任意形狀分布的資料集，然而 CLUTO 對於參數的設定是相當敏感的。實驗結果顯示 CLUTO 的確受到參數 nnbrs 的影響，雖然部分資料集在特定的參數 nnbrs 可以接近正確，我們仍然無法找出一個適當的參數 nnbrs 應用於所有資料集。由於分群的過程是一種非監督式學習法，對於參數的設定往往只能透過反覆測試，進而尋找出一個足以讓使用者信服的答案。與此篇論文提出的方法相比，可互相包含最近鄰(MI-nn)為基礎的相似度轉換方法對於參數的依賴程度遠低於 CLUTO。最後觀察 DBSCAN 的實驗結果，DBSCAN 是一種以密度為基礎的演算法，其弱點便是在處理分群密度不均衡的情況，實驗結果證實 DBSCAN 應用至四組密度不均衡的資料集(d1、d2、d3、d4)，分群準確率較低，而應用至凹狀資料集中(d5、d6、d7)，在特定的參數下仍然能獲得正確的分群結果。綜合以上結果，以可互相包含最近鄰(MI-nn)為基礎的相似度轉換方法能處理任意形狀的群集，且參數的依賴程度較低。

觀察 10 組來自於 UCI 資料庫的資料集，我們使用其提供的資料類別做為資料的正確分群。由於 UCI 資料集來自於真實世界，大多數擁有較高的資料維度，而高維度空間內的資料分布通常更為鬆散，難以預期其分布特性。我們希望藉由相似度轉換的幫助，突顯資料分布的特性和集群邊界。

在 UCI 資料集的實驗中，我們同樣依照 3.3 節提出的流程尋找相似度轉換的最佳解，針對每一組資料集，尋找適當的相似度轉換次數，如表 4-5。將經由相似度轉換後產生的新屬性代入至 freq K-medoids 演算法中，並使用 20 組隨機抽取的測試集同步評量其他 3 種不同的分群分法，實驗結果如圖 4-5 和表 4-6。實驗結果說明在多數的情況下，採用以可互相包函最近鄰(MI-nn)為區域最近鄰的相似度轉換方法都能獲得相對高的分群準確率，特別是 iris、glass、balance、WDBC、soybean，此五組資料集是顯著高於其他三種分群方法。

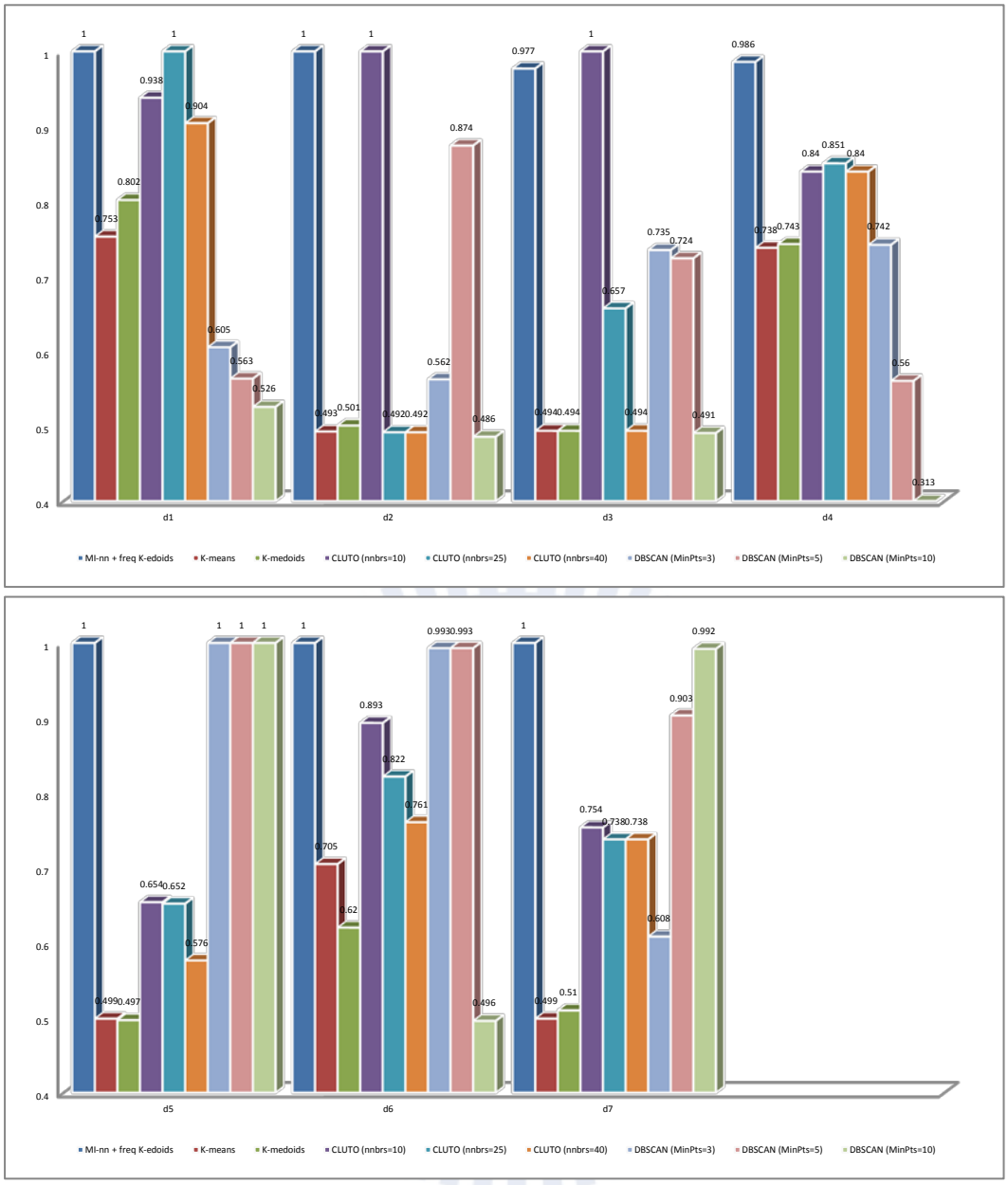


圖 4-4. 非監督式分群演算法，7 組人工資料集的 RI 比較

說明：由左至右：MI-nn based transform and freq K-medoids、K-means、K-medoids、CLUTO (nnbrs=10) 、CLUTO (nnbrs=25) 、CLUTO (nnbrs=40)、DBSCAN (MinPts=3)、DBSCAN (MinPts=5)、DBSCAN (MinPts=10)

表 4-4. 非監督式分群演算法，7 組人工資料集之 RI 平均值與 Wilcoxon signed rank test 分析

Method	Average RI
MI-nn + freq K-edoids	0.995
K-means	0.597
K-medoids	0.595
CLUTO (nnbrs=10)	0.868
CLUTO (nnbrs=25)	0.745
CLUTO (nnbrs=40)	0686
DBSCAN (MinPts=3)	0.749
DBSCAN (MinPts=5)	0.802
DBSCAN (MinPts=10)	0.615

Method comparison	p-Value
MI-nn + freq K-edoids vs. K-means	0.0156
MI-nn + freq K-edoids vs. K-medoids	0.0156
MI-nn + freq K-edoids vs. CLUTO (nnbrs=10)	0.0625
MI-nn + freq K-edoids vs. CLUTO (nnbrs=25)	0.0313
MI-nn + freq K-edoids vs. CLUTO (nnbrs=40)	0.0156
MI-nn + freq K-edoids vs. DBSCAN (MinPTS=3)	0.0313
MI-nn + freq K-edoids vs. DBSCAN (MinPTS=5)	0.0313
MI-nn + freq K-edoids vs. DBSCAN (MinPTS=10)	0.0313

表 4-5. 10 組 UCI 資料集之相似度轉換次數

Dataset	Iris	Wine	Glass	Balance	Ionosphere
	0	1	5	13	0
Dataset	Breast cancer	WDBC	Soybean	Segmentation	Pima diabetes
	1	0	0	2	5

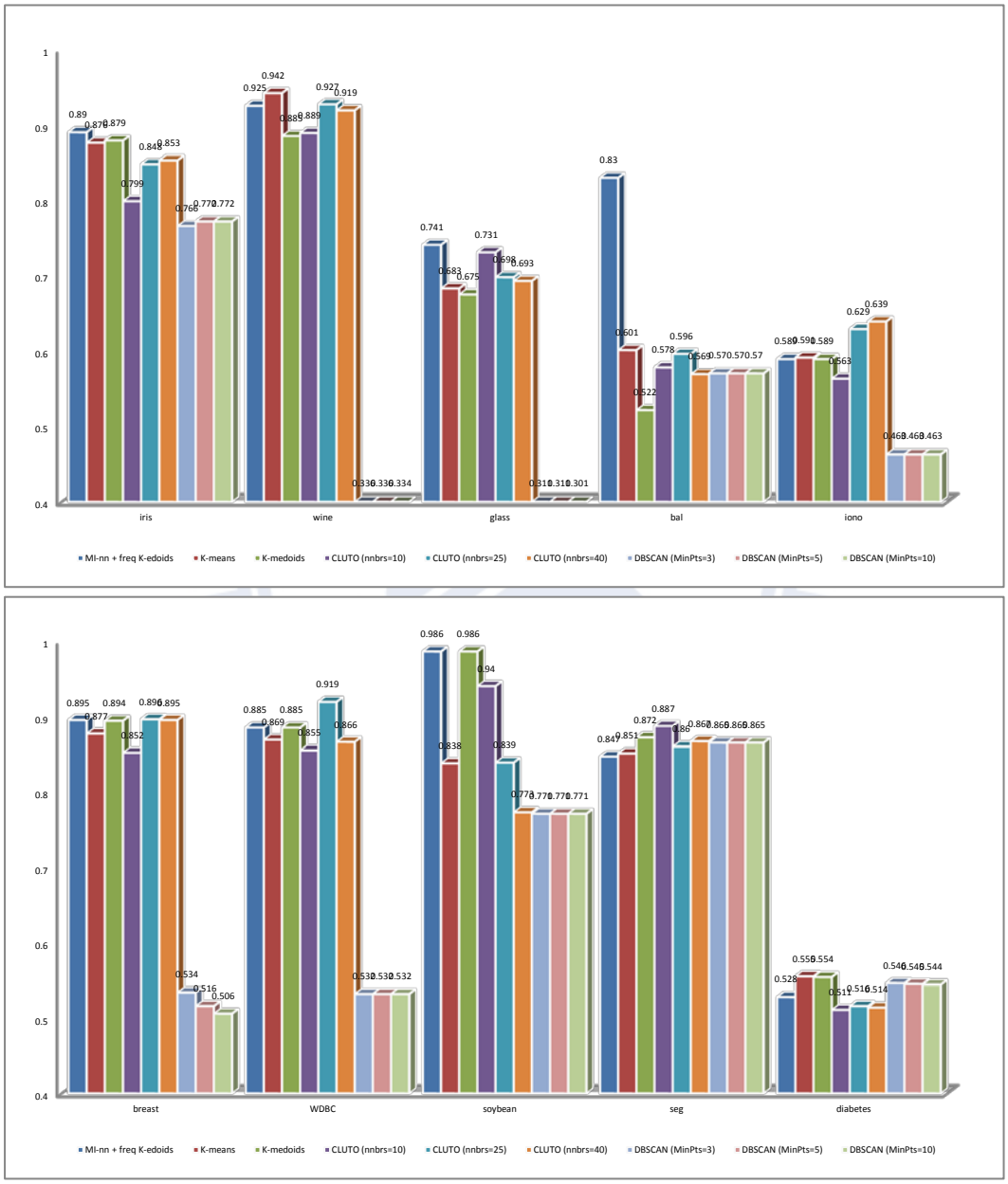


圖 4-5. 非監督式分群演算法，10 組 UCI 資料集的 RI 比較

說明：由左至右：MI-nn based transform and freq K-medoids、K-means、K-medoids、CLUTO (nnbrs=10) 、CLUTO (nnbrs=25) 、CLUTO (nnbrs=40)、DBSCAN (MinPts=3)、DBSCAN (MinPts=5)、DBSCAN (MinPts=10)

表 4-6. 非監督式分群演算法，10 組 UCI 資料集之 RI 平均值與 Wilcoxon signed rank test 分析

Method	Average RI
MI-nn + freq K-edoids	0.812
K-means	0.768
K-medoids	0.774
CLUTO (nnbrs=10)	0.761
CLUTO (nnbrs=25)	0.773
CLUTO (nnbrs=40)	0.759
DBSCAN (MinPts=3)	0.569
DBSCAN (MinPts=5)	0.568
DBSCAN (MinPts=10)	0.566

Method comparison	p-Value
MI-nn + freq K-edoids vs. K-means	0.2324
MI-nn + freq K-edoids vs. K-medoids	0.2969
MI-nn + freq K-edoids vs. CLUTO (nnbrs=10)	0.0273
MI-nn + freq K-edoids vs. CLUTO (nnbrs=25)	0.375
MI-nn + freq K-edoids vs. CLUTO (nnbrs=40)	0.2031
MI-nn + freq K-edoids vs. DBSCAN (MinPTS=3)	0.0098
MI-nn + freq K-edoids vs. DBSCAN (MinPTS=5)	0.0098
MI-nn + freq K-edoids vs. DBSCAN (MinPTS=10)	0.0098

4.3.1.3 相似度轉換方法與非監督式分群演算法合併之比較

相似度轉換方法可以被當作一種資料前處理的步驟。在此小節中，我們首先使用相似度轉換方法產生一組全新的屬性關係，接著將新的屬性關係代入至 K-means[1]、CLUTO[5]和 DBSCAN[6]，三種非監督式的分群演算法，比較原始資料屬性和經由相似度轉換方法求出的新屬性關係，兩者的分群準確率。

人工資料集的實驗結果如圖 4-6 和表 4-7。首先觀察 K-means 的實驗結果，我們已知 K-means 演算法的目標在於處理凸性最佳化問題(convex optimization problem)，在 4.3.1.1 小節中已經證實相似度轉換方法能夠凸顯集群之間的邊界(如圖 4-2)，有利於 K-means 演算法尋找正確分群，進而提高分群的準確率；實驗結果顯示，在七組人工設計資料集中，扣除 d4 資料集，我們都能找出正確的分群。接著觀察 CLUTO 和 DBSCAN 的實驗結果，由於兩者對於參數的設定都是相當敏感的，我們認為對採用相似度轉換方法改變資料的分布情形，能夠降低演算法對參數的依賴程度；實驗結果顯示，DBSCAN 方法在七組人工資料集中，對於實驗所採用的三組參數都能同時提高分群準確率，而少部分資料集例如：d1 和 d3 更能找出正確的分群；對於 CLUTO 的實驗部分，扣除 d1 和 d4 二組資料集，在剩餘的資料集中仍然可以得到接近甚至更佳的分群準確率。

UCI 資料集的實驗結果如圖 4-7 和表 4-8。由於 UCI 資料集式搜集自真實世界，資料的分布通常是更加散亂的，我們較難預測集群的分布情形。實驗結果顯示，相似度轉換並不能增加準確性，僅僅能維持相近的結果。

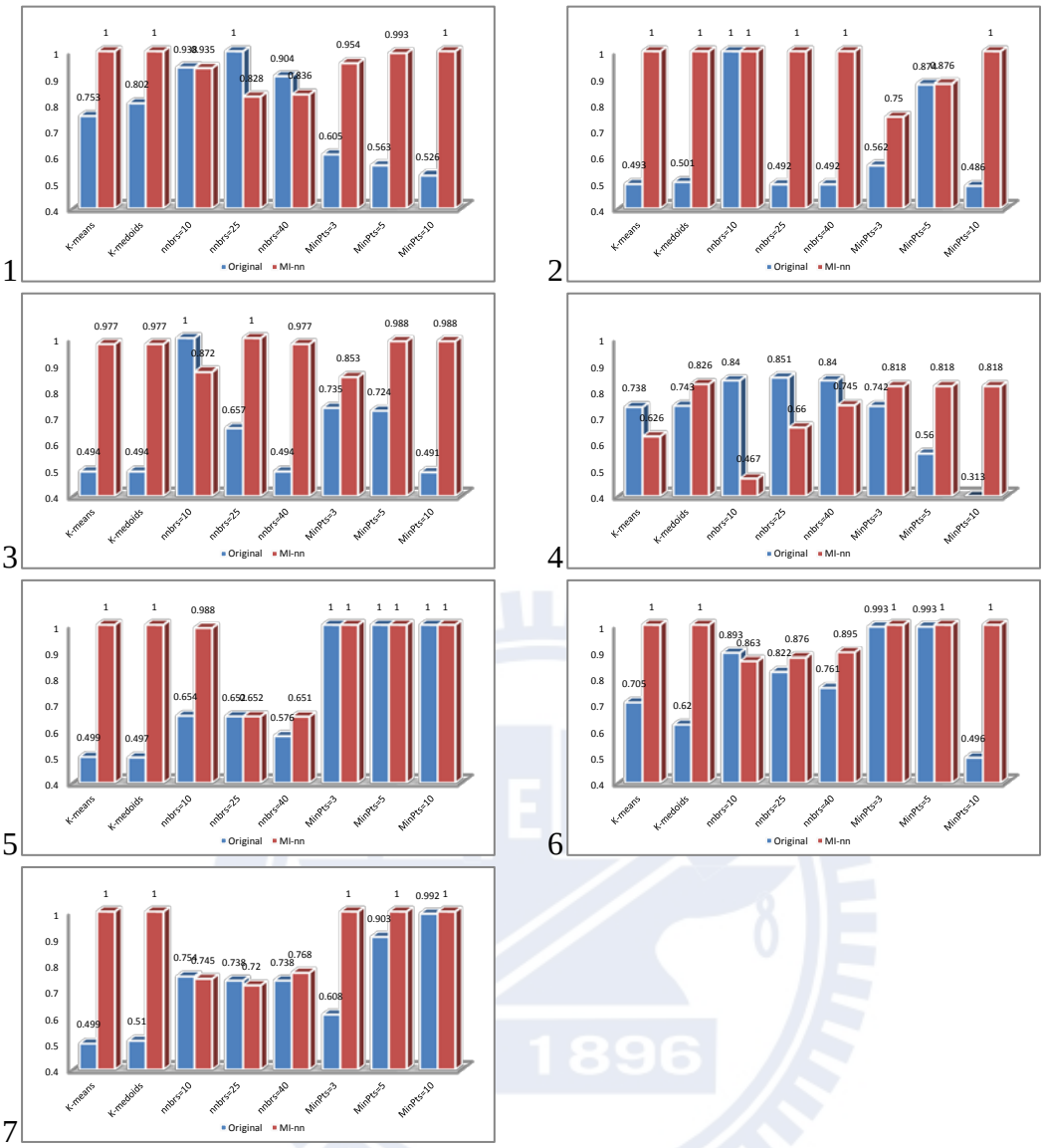


圖 4-6. 相似度轉換方法應用至 K-means、K-medoids、CLUTO 和 DBSCAN 方法，和未使用相似度轉換之 7 組人工資料集的 RI 比較

說明：由左至右，上至下分別是：1. d1, 2. d2, 3. d3, 4. d4, 5. d5, 6. d6, 7. d7

說明：藍色長條表示使用原始資料，紅色長條表示使用 MI-nn 轉換所產生之新屬性

說明：由左至右：K-means、K-medoids、CLUTO (nnbrs=10) 、CLUTO (nnbrs=25) 、CLUTO (nnbrs=40)、DBSCAN (MinPts=3)、DBSCAN (MinPts=5)、DBSCAN (MinPts=10)

表 4-7. 相似度轉換方法應用至 K-means、K-medoids、CLUTO 和 DBSCAN 方法，和未使用相似度轉換之 7 組人工資料集之 RI 平均值與 Wilcoxon signed rank test 分析

Method	Average RI	
	Original	MI-nn
K-means	0.597	0.943
K-medoids	0.595	0.972
CLUTO (nnbrs=10)	0.868	0.839
CLUTO (nnbrs=25)	0.745	0.819
CLUTO (nnbrs=40)	0.686	0.839
DBSCAN (MinPts=3)	0.749	0.911
DBSCAN (MinPts=5)	0.802	0.954
DBSCAN (MinPts=10)	0.615	0.972

Original vs. MI-nn	p-Value
K-means	0.0313
K-medoids	0.0156
CLUTO (nnbrs=10)	0.3125
CLUTO (nnbrs=25)	0.6875
CLUTO (nnbrs=40)	0.2188
DBSCAN (MinPts=3)	0.0313
DBSCAN (MinPts=5)	0.0313
DBSCAN (MinPts=10)	0.0313

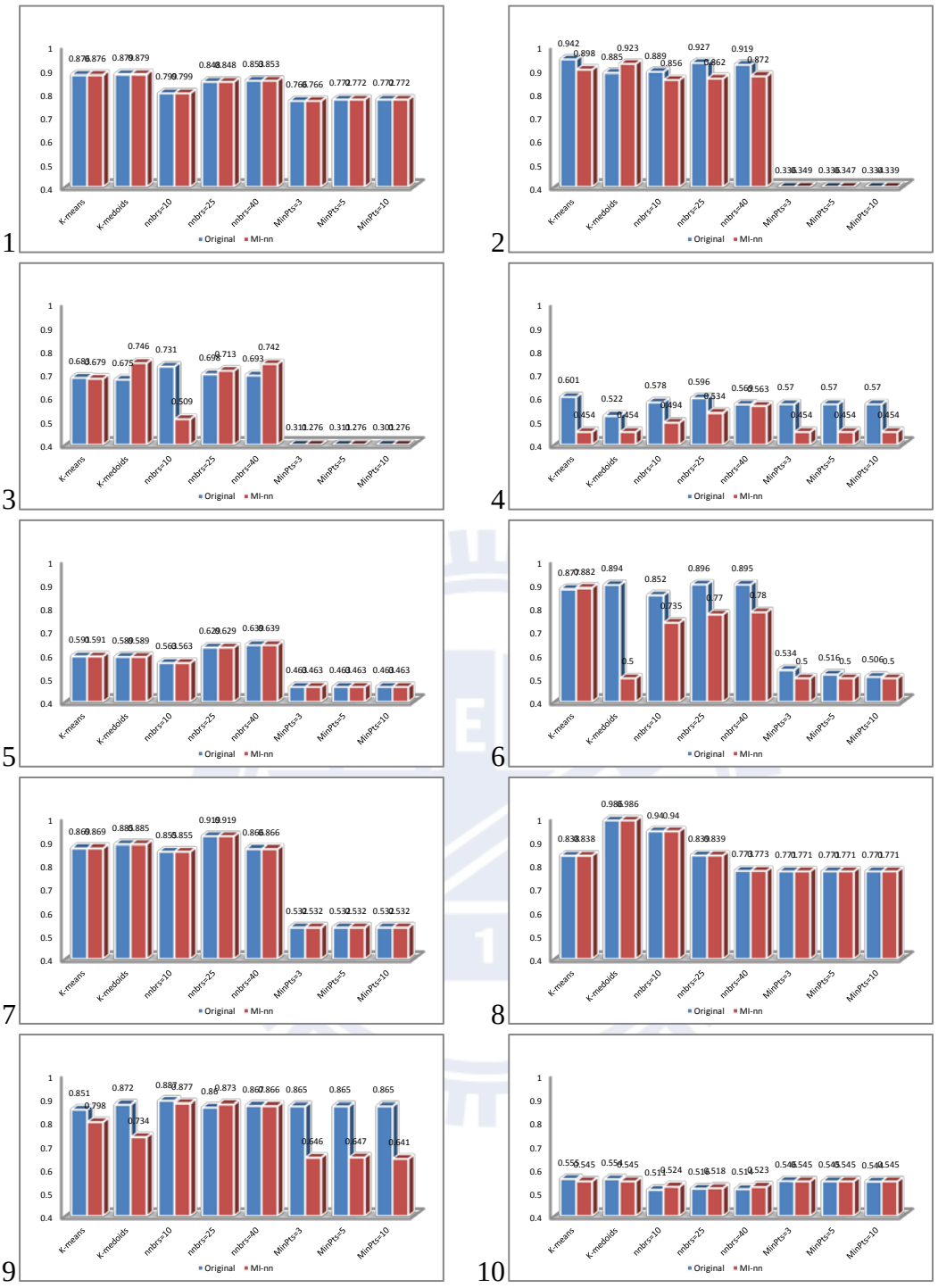


圖 4-7. 相似度轉換方法應用至 K-means、K-medoids、CLUTO 和 DBSCAN 方法，和未使用相似度轉換之 10 組 UCI 資料集 RI 比較

說明：由左至右，上至下分別是：1. Iris, 2. Wine, 3. Glass, 4. Balance, 5. Ionosphere, 6. Breast cancer, 7. WDBC, 8. Soybean, 9. Segmentation, 10. Diabetes

說明：藍色長條表示使用原始資料，紅色長條表示使用 MI-nn 轉換所產生之新屬性

說明：由左至右：K-means、K-medoids、CLUTO (nnbrs=10) 、CLUTO (nnbrs=25) 、CLUTO (nnbrs=40)、DBSCAN (MinPts=3)、DBSCAN (MinPts=5)、DBSCAN (MinPts=10)

表 4-8. 相似度轉換方法應用至 K-means、K-medoids、CLUTO 和 DBSCAN 方法，和未使用相似度轉換之 10 組 UCI 資料集之 RI 平均值與 Wilcoxon signed rank test 分析

Method	Average RI	
	Original	MI-nn
K-means	0.768	0.743
K-medoids	0.774	0.724
CLUTO (nnbrs=10)	0.761	0.715
CLUTO (nnbrs=25)	0.773	0.751
CLUTO (nnbrs=40)	0.759	0.748
DBSCAN (MinPts=3)	0.569	0.53
DBSCAN (MinPts=5)	0.568	0.531
DBSCAN (MinPts=10)	0.566	0.529

Original vs. MI-nn	p-Value
K-means	0.0938
K-medoids	0.4375
CLUTO (nnbrs=10)	0.0938
CLUTO (nnbrs=25)	0.4375
CLUTO (nnbrs=40)	0.6875
DBSCAN (MinPts=3)	0.0938
DBSCAN (MinPts=5)	0.125
DBSCAN (MinPts=10)	0.1563

4.3.2 半監督式分群演算法比較

將 3.4 節提出的代價函式與 freq K-medoids 演算法合併，使得 freq K-medoids 演算法從非監督式變更成半監督式。由於此篇論文提出的方法涉及相似度轉換，與 similarity-based 的半監督式分群法更為接近，因此在本章節的實驗中，我們將經過相似度轉換後的全新屬性帶入至 freq K-medoids with cost function 演算法，同時與另外三種一樣屬於 similarity-based 的半監督式分群法：Xing[11]、RCA[12]和 LMNN[16]做比較。Xing[11]和 RCA[12]提出的兩種半監督式分群法在學習一組新的相似度函式後，都會和屬於 search-based 的 Cop K-means[8]方法結合，在尋找分群的過程重複利用配對限制(pairwise constraints)，再次提升分群準確率。LMNN[16]在學習相似度函式的過程中使用 K 最近鄰(K-nn)建構最大化邊界(large margin)，與此篇論文提出的方法使用相似的概念，因此我們將其納入比較。然而，LMNN[16]的原始目標是應用於分類(classification)領域而非分群，除了給予配對限制的類型外，尚須給予配對限制所包含的資料之類別(class)。與 Xing[11]和 RCA[12]方法相同，在利用 LMNN[16]方法學習一組新的相似度函式後，我們比較經由 K-medoids 演算法產生之結果的準確率。由於此篇論文提出的方法是以 K-medoids 演算法為基礎，基於公平性，我們對 Cop K-means 稍做修改，將分群演算法由 K-means 更換成 K-medoids，而 Cop K-means 的主要目標仍然不變。

半監督式分群法的目標在於將使用者或專家提供的配對限制整合至演算法中。我們從訓練集中隨機挑選固定數量的鏈結用以模擬配對限制，並使用測試集評量分群的準確率。在配對限制的抽取過程中，我們盡可能平均的抽出 MUST-link 和 CANNOT-link 的數量，避免受到其中一種型態的配對限制影響。實驗設計上，分別自資料集中抽出三種固定數量的配對限制，數量分別是 50 組、100 組和 200 組，每當完成配對限制的抽取，將遵循眾多學者的作法，進一步對 MUST-link 和 CANNOT-link 建構遞移包(transitive closure)，藉此尋找更多的配對限制。

人工資料集的結果顯示，經過相似度轉換後的資料集，即便缺少專家提供的意見的協助(例如：配對限制)，我們的分群結果仍可達到近乎完全正確的結果。反觀其餘三組方法，如圖 4-8 和表 4-9 所示，即便將原始的配對限制個數提供至二百對，除了資料分布相對規律的 d2 之外，Xing [11]、RCA[12]、 LMNN[16]對於其他分布較為特殊的資料集仍然無法完整區分，僅能有限度的提升準確性。

反觀 UCI 資料集的實驗結果，如圖 4-9 和表 4-10 所示。Xing[11]、RCA[12]和 LMNN[16]三種方法憑藉著配對限制的幫助重新學習相似關係，因此可以預期新的相似度函式所描述的相似關係可以滿足多數的配對限制，進而提升整體的分群準確率。我們的實驗結果顯示，先利用可互相包含最近鄰(MI-nn)作為基礎之相似度轉換方法找出新的屬性關係，再採用 freq K-medoids with cost function 演算法找出的分群，其準確性不但接近其餘三種方法，甚至在部分資料集的分群結果上擁有更佳的表現。比較非監督式和半監督式分群法的實驗結果，將經過相似度轉換的新屬性分別應用至非監督式的 freq K-medoids 或半監督式的 freq K-medoids with cost function 演算法，兩者的差異並不是非常顯著。我們認為將相似度轉換方法應用於目前實驗所使用的資料集，都能完整描述資料間真實的相似關係，同時符合專家提供的配對限制，因此較難突顯配對限制在改進分群準確率的優點。若是應用於其他資料集的分群上，專家的意見仍然可以被用來彌補相似度轉換所無法涵蓋的範圍，以提升分群的效能。

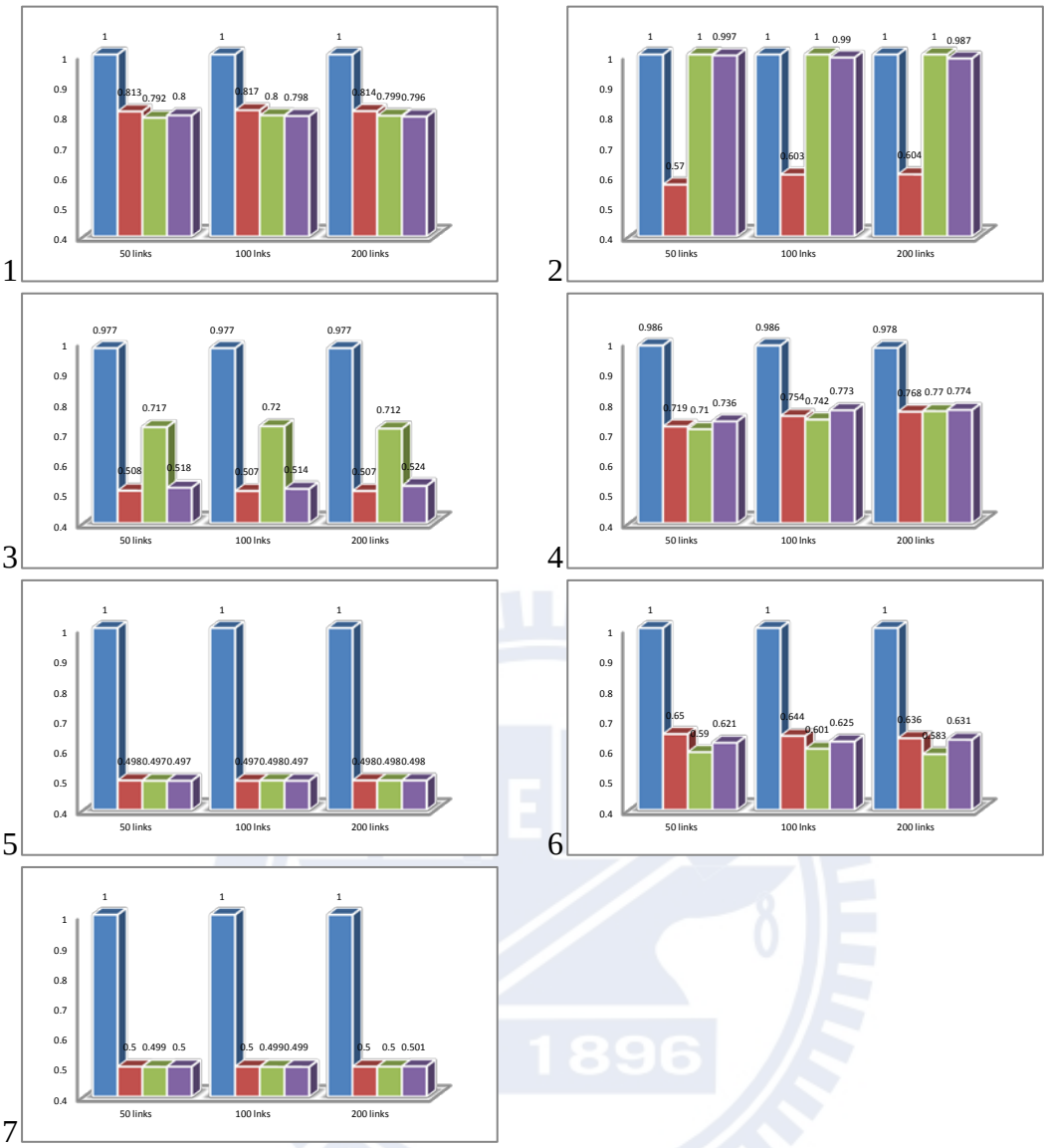


圖 4-8. 半監督式分群演算法，7 組人工資料集的 RI 比較

說明：由左至右，上至下分別是：1. d1, 2. d2, 3. d3, 4. d4, 5. d5, 6. d6, 7. d7

說明：圖表中由左至右： MI-nn based transformation and freq K-medoids with cost function、Cop K-medoids[2] over the feature space suggested by Xing[11]、Cop K-medoids[2] over the feature space suggested by RCA[12]、K-medoids over the feature space suggested by LMNN[16]

表 4-9. 半監督式分群演算法，7 組人工資料集之 RI 平均值與 Wilcoxon signed rank test 分析

Number of piarwise constraints	Method	Average RI
50	MI-nn	0.995
	Xing	0.608
	RCA	0.686
	LMNN	0.667
100	MI-nn	0.995
	Xing	0.617
	RCA	0.694
	LMNN	0.671
200	MI-nn	0.995
	Xing	0.618
	RCA	0.695
	LMNN	0.673

Number of pairwise constraints	Method comparison	p-Value
50	MI-nn vs. Xing	0.0156
	MI-nn vs. RCA	0.0313
	MI-nn vs. LMNN	0.0156
100	MI-nn vs. Xing	0.0156
	MI-nn vs. RCA	0.0313
	MI-nn vs. LMNN	0.0156
200	MI-nn vs. Xing	0.0156
	MI-nn vs. RCA	0.0313
	MI-nn vs. LMNN	0.0156

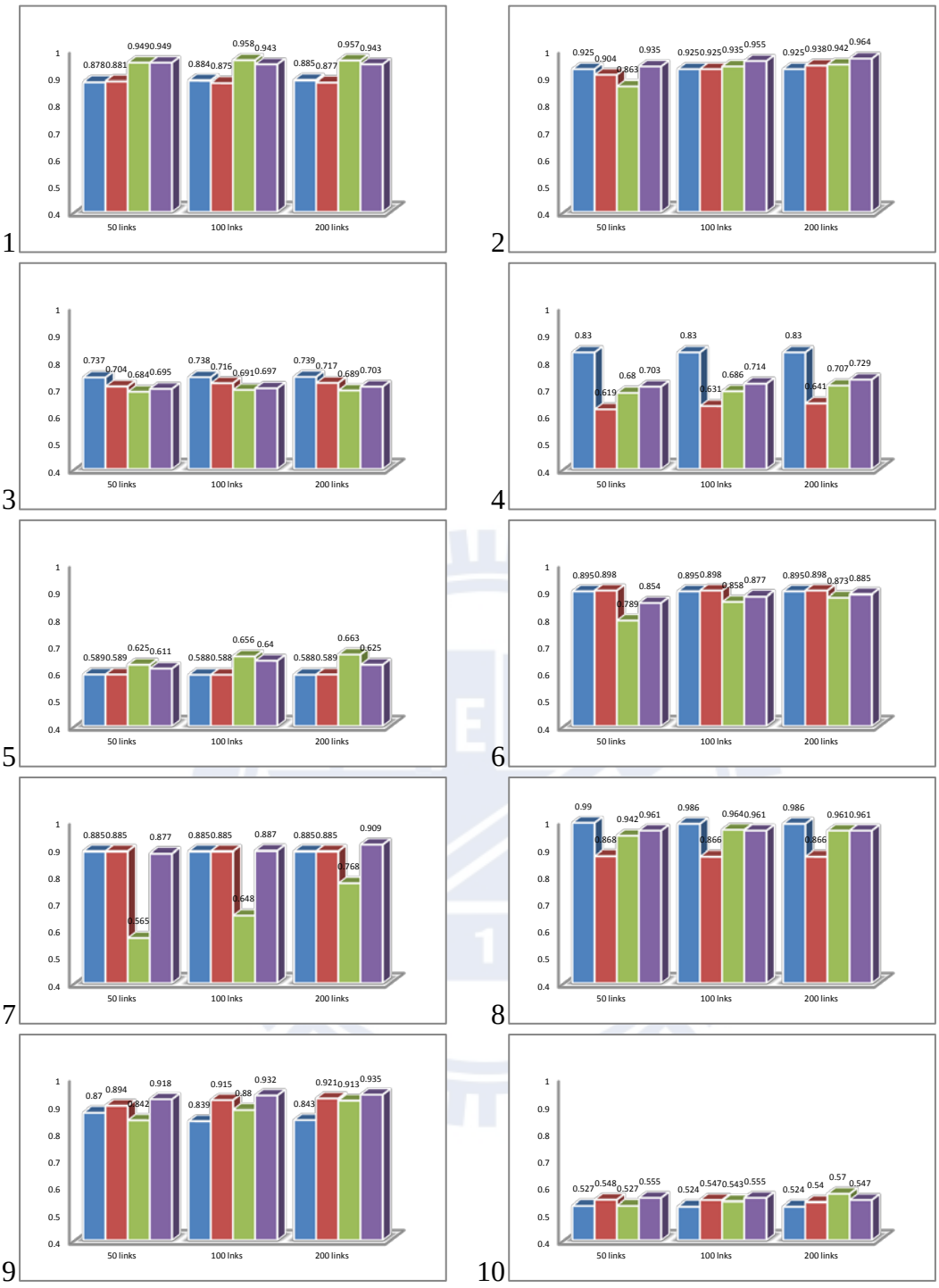


圖 4-9. 半監督式分群演算法，10 組 UCI 資料集的 RI 比較

說明：由左至右，上至下分別是：1. Iris, 2. Wine, 3. Glass, 4. Balance, 5. Ionosphere, 6. Breast cancer, 7. WDBC, 8. Soybean, 9. Segmentation, 10. Diabetes

說明：圖表中由左至右： MI-nn based transformation and freq K-medoids with cost function、Cop K-medoids[2] over the feature space suggested by Xing[11]、Cop K-medoids[2] over the feature space suggested by RCA[12]、K-medoids over the feature space suggested by LMNN[16]

表 4-10. 半監督式分群演算法，10 組 UCI 資料集之 RI 平均值與 Wilcoxon signed rank test 分析

Number of pairwise constraints	Method	Average RI
50	MI-nn	0.813
	Xing	0.779
	RCA	0.747
	LMNN	0.806
100	MI-nn	0.809
	Xing	0.785
	RCA	0.782
	LMNN	0.816
200	MI-nn	0.81
	Xing	0.787
	RCA	0.804
	LMNN	0.82

Number of pairwise constraints	Method comparison	p-Value
50	MI-nn vs. Xing	0.3984
	MI-nn vs. RCA	0.0977
	MI-nn vs. LMNN	0.9219
100	MI-nn vs. Xing	0.5781
	MI-nn vs. RCA	0.6953
	MI-nn vs. LMNN	0.5566
200	MI-nn vs. Xing	0.7344
	MI-nn vs. RCA	0.9219
	MI-nn vs. LMNN	0.4922

4.3.3 維度縮減應用於相似度轉換方法

我們將維度縮減應用至相似度轉換方法，以不影響分群準確率為前提下，預期可以減少資料計算量，進而提升相似度轉換方法的速度。在此小節，我們首先以非度量性多元尺度法做為維度縮減的方法，探討維度縮減對於分群準確率和相似度轉換方法執行時間的影響。主成分分析法(principle component analysis, PCA)是常用的維度縮減方法，在此我們另行比較使用非度量性多元尺度法或主成分分析法對資料集做維度縮減後，對於分群準確度之影響。

4.3.3.1 非度量性多元尺度法所選取之維度數對於相似度轉換的影響

非度量性多元尺度法的主要目標是根據資料的相似關係，於特定的維度空間內，使資料在此維度空間內的實際歐式距離分布盡可能與相似關係保持一致。如何選擇建構的空間維度個數是一項重要的問題，在 3.2 節中，我們首先將經過相似度轉換後的相似關係重新描述至與原始資料維度個數相同的空間內。由於真實世界的資料集其資料量通常是非常大量且多數都是高維度分布，對於效能是一項重大負擔，也是造成演算法的計算量增加的主要原因，透過維度縮減方法能夠改善上述情況。若是將非度量性多元尺度法預定建構的空間維度設定為低於原始維度，其過程相當於維度縮減。在此小節中，我們將分析以非度量性多元尺度法做為維度縮減方法，對於相似度轉換所造成的影響。

在 3.3 節中我們提出一種相似度轉換的迭代方法，藉此尋找一個合適的相似度轉換次數。當相似度轉換所需的迭代次數超過一次時，我們僅在第一次迭代時縮減資料維度，自第二次迭代開始便不再縮減資料維度，將資料維度設定在第一次迭代後縮減的維度個數。

由於人工設計的資料分布於二維空間，因此在本小節的實驗我們僅考慮 UCI 資料集。針對 10 組 UCI 真實世界資料集，我們分別嘗試在首次進行相似度轉換時將資料維度縮減至原始維度個數的 50%、60%、70%、80%、90%，接著依照 3.3 節提出的流程完成相似度轉換。由於維度個數不同，即便是相同的資料集所

找出的最佳相似度轉換次數亦不相同，如表 4-11 所示。在完成相似度轉換後，我們將產生的新屬性帶入至 freq K-medoids 中，比較在不同維度縮減的條件下，對於分群準確性所造成的影響，如圖 4-10 和表 4-12 所示。

表 4-11. 10 組 UCI 資料集，不同維度個數之相似度所需轉換次數

Reduction percentage	0.5	0.6	0.7	0.8	0.9	Reduction percentage	0.5	0.6	0.7	0.8	0.9
Iris	4	4	2	2	0	Breast cancer	1	1	1	1	4
Wine	1	1	1	1	1	WDBC	0	4	5	10	0
Glass	4	4	4	2	6	Soybean	0	0	0	0	0
Balance	2	2	5	5	13	Segmentation	6	2	5	2	2
Ionosphere	2	0	0	0	0	Pima diabetes	5	1	5	5	6

實驗結果顯示對於擁有較高維度的資料集而言，降低維度並不會顯著的影響分群結果，在將維度降低至原始維度的一半之後，我們仍可得到相近甚至更佳的結果，例如：扣除 balance 之外的九組資料集；對於擁有較低維度的資料集，由於本身用來描述資料的維度個數較低，若是再進行維度縮減則會影響非度量性多元尺度法方法產生的新屬性，使新產生的屬性與相似矩陣的一致性較低，進而影響分群的準確率，例如： balance 的維度為 4 維。

在 3.3 節中提出之尋找相似度轉換次數最佳解的流程，其效能受制於非度量性多元尺度法的運算速度，維度縮減能夠降低資料計算量以提升速度。我們接著針對維度個數超過十個以上的 UCI 資料集(wine、ionosphere、WDBC、soybean 和 segmentation)，比較原始維度個數和縮減至 50%所需的相似度轉換方法執行時間，如表 4-13 所示。

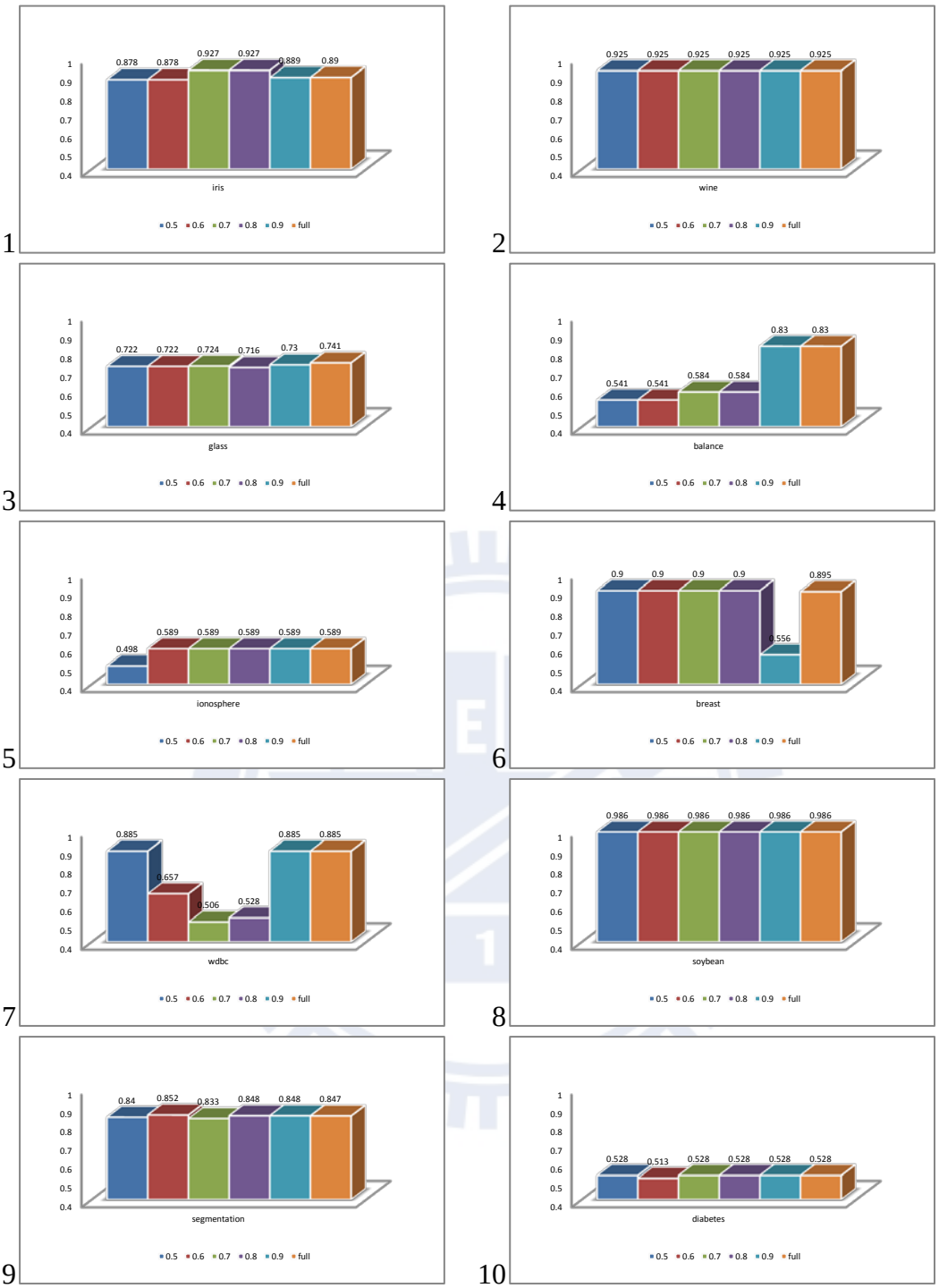


圖 4-10.不同維度縮減個數，10 組 UCI 資料集的 RI 比較

說明：10 組 UCI 資料集，由左至右，上至下分別是： 1. Iris, 2. Wine, 3. Glass, 4. Balance, 5. Ionosphere, 6. Breast cancer, 7. WDBC, 8. Soybean, 9. Segmentation, 10. Diabetes

說明：圖表中由左至右：50% of dimension、60%、70%、80%、90%、100%

表 4-12. 不同維度縮減個數，10 組 UCI 資料集之 RI 平均值與 Wilcoxon signed rank test 分析

Reduction percentage	Average RI
50%	0.77
60%	0.756
70%	0.75
80%	0.753
90%	0.777
Full	0.812

Reduction percentage comparison	p-Value
50% vs. Full	0.0625
60% vs. Full	0.0781
70% vs. Full	0.3125
80% vs. Full	0.5625
90% vs. Full	0.375

表 4-13. 原始維度個數和縮減至 50%所需的一次相似度轉換方法執行時間，以秒做為時間單位

	Wine	Ionosphere	WDBC	Soybean	Segmentation
Full	14	262	605	1	31
50%	10	96	216	1	16

實驗結果顯示，扣除 Soybean 之外的四組資料集，將維度縮減至原始維度個數的 50%後，都能減少相似度轉換方法所需的執行時間。由於 Soybean 的資料量較少(僅有 47 筆)，本身的資料計算量更低，我們難以看出維度縮減所帶來的好處。

相似度轉換方法是一種迭代演算法，若是在第一次重新描述資料相似度時能夠降低資料維度，則後續的迭代步驟都能受到維度縮減所帶來的好處。實驗結果證實，適度的縮減維度不僅不會顯著影響分群準確率，同時可以提升相似度轉換方法的執行速度。

4.3.3.2 主成分分析法比較

主成分分析法(principal component analysis)是一種多變量分析方法。它將資料集中高度相關性的屬性轉化成彼此互相獨立維度的線性組合，以較低維度的線性組合解釋原始資料屬性之變異性，可被應用於維度縮減。在此小節的實驗中，我們將分別比較在無配對限制的條件下，主成分分析法與非度量性多元尺度法對於分群準確率的影響。

在此小節的實驗中，對於主成分分析法選擇的成分(component)個數，我們遵從非度量性多元尺度法所建構的空間維度數，若是非度量性多元尺度法建構在 N 維空間內，那麼主成分分析法便選擇前 N 個成分。我們使用主成分分析法對原始資料做維度縮減，可以產生一組全新的屬性關係，實驗目標在於比較經由相似度轉換或主成分分析法縮減資料維度後，使用 freq K-medoids 方法進行分群之準確率，實驗結果如圖 4-11 和表 4-15 所示。

由於主成分分析法之目標在於使用較少的成分個數解釋資料之變異量，在多數的研究中，如何選取適當的成分個數通常會參考對應的解說總變異量。在此小節的實驗中，對於較高維度分布的資料集，即便我們將選取的成分個數對應至原始維度個數的 50%，其解說總變異量都能接近或超過 90%，如表 4-14，說明能夠在不損失過多的資訊下縮減資料維度。在主成分分析法能充分解釋 90%變異量

之情形下，實驗結果顯示，五組維度較高的資料集中(wine、ionosphere、WDBC、soybean 和 segmentation)，若是以非度量性多元尺度法做為維度縮減的方法，WDBC 和 soybean 二組資料集的分群準確率仍然是顯著高於主成分分析法，而在 wine 和 ionosphere 二組資料集則和主成分分析法接近，僅在 segmentation 資料集中，主成分分析法是優於非度量性多元尺度法。

表 4-14. 主成分分析法，選取之成分個數對應至原始維度個數的 50%，其累積解說變異量

	原始維度個數	選取之成分個數	解說總變異量
Iris	4	2	95.769%
Wine	13	7	89.337%
Glass	9	5	89.281%
Balance	4	2	50.000%
Ionosphere	34	17	88.694%
Breast cancer	9	5	87.190%
WDBC	30	15	98.649%
Soybean	21	11	96.886%
Segmentation	19	10	97.959%
Pima diabetes	8	4	71.634%

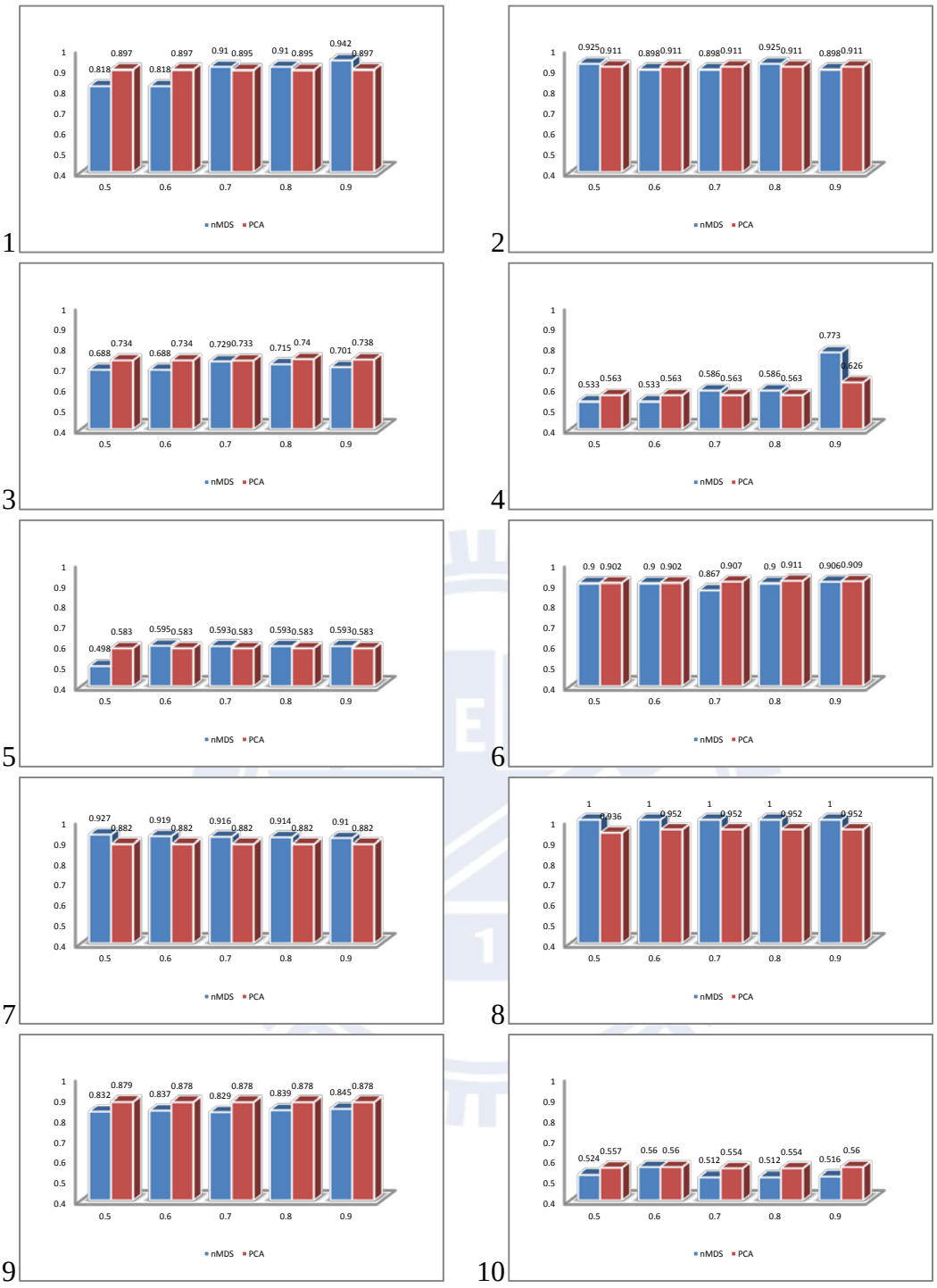


圖 4-11. 10 組 UCI 資料集，比較 nMDS 和 PCA 對於分群準確率的影響

說明：10 組 UCI 資料集，由左至右，上至下分別是： 1. Iris, 2. Wine, 3. Glass, 4. Balance, 5. Ionosphere, 6. Breast cancer, 7. WDBC, 8. Soybean, 9. Segmentation, 10. Diabetes

表 4-15. 10 組 UCI 資料集，nMDS 和 PCA 之 RI 平均值與 Wilcoxon signed rank test 分析

Reduction percentage	Average RI	
	nMDS	PCA
50%	0.77	0.784
60%	0.756	0.786
70%	0.75	0.786
80%	0.753	0.787
90%	0.777	0.794

nMDS vs. PCA	
Reduction percentage	p-Value
50%	0.2324
60%	0.1934
70%	0.8457
80%	0.8457
90%	0.8262

第五章，結論

此篇論文中，我們提出一種以區域性為基礎的相似度轉換方法。藉由觀察鏈結兩端點的區域最近鄰分布，重新調整鏈結權重，使得資料彼此的相似程度能依照提出的假設做修改，相當於對資料進行前處理。我們分別提出兩種尋找區域性鄰集的方法：K 最近鄰(K-nn)和可互相包含最近鄰(MI-nn)。實驗結果顯示可互相包含最近鄰(MI-nn)為基礎之相似度轉換方法能夠尋找任意形狀的群集，且對於參數的依賴度遠低於 K 最近鄰。我們認為以區域性鄰集為基礎的相似度轉換方法能夠凸顯資料之間的邊界，進而提升整體的準確率。

以區域性鄰集為基礎的相似度轉換方法有別於其他相似度轉換方法，不需任何配對限制的幫助即可達成預期目標。以非監督式分群的角度觀察，與其他非監督式分群演算法進行比較，對於分布較為特殊的資料集皆可獲得接近完全正確的結果，顯著優於 K-means、cluto[5]和 DBSCAN[27]；對於蒐集自真實世界的 UCI 資料集，多數的情況亦能獲得相對高的準確率。改以半監督式分群的角度觀察，由於提出的方法涉及相似度轉換，與 Xing[11]、RCA[12]、 LMNN[16]三種屬於 similarity-based 的半監督式分群演算法更為接近，比較之下準確性都可以獲得相近的結果，甚至在部分資料集有更高的準確率。

相似度轉換方法未必能完整描述資料的相似關係，因此我們認為將相似度轉換方法與半監督式分群法結合，可以透過專家的意見彌補相似度轉換所無法涵蓋的範圍，以提升分群的效能。

除了改善分群準確率外，我們亦能在相似度轉換的過程中縮減資料維度。縮減維度不僅能加快相似度轉換，同時能在不影響分群結果的前提下，提升分群演算法的速度。

總結以上實驗結果，此篇論文提出的方法有下列優點：

1. 透過相似度轉換改善分群演算法中所使用的相似度函式。
2. 不需過多的外部參數引導分群。

3. 在缺少配對限制的情況下，能夠有效的處理任意形狀分布的群集圖形。
4. 將相似度轉換方法與半監督式分群演算法結合，專家的意見能夠彌補相似度轉換所無法涵蓋的範圍，進一步提升整體的準確率。
5. 我們能透過相似度轉換方法適度的縮減資料維度。維度縮減並不顯著影響分群結果，亦能減少計算量以提升分群演算法的速度。

參考文獻

[1] Han, L., Kamber, M., Pei, J., “Data Mining: Concepts and Techniques”, Morgan Kaufmann, 2011

[2] Grira, N., Crucianu, M., Boujemaa, N., “Unsupervised and Semi-supervised Clustering: a Brief Survey”, A Review of Machine Learning Techniques for Processing Multimedia Content, Report of the MUSCLE European Network of Excellence, 2004

[3] MacQueen, J. B., “Some Methods for Classification and Analysis of Multivariate Observations”, Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, pp. 281-297, 1967

[4] Kaufman, L., Rousseeuw, P.J., “Finding Groups in Data: an Introduction to Cluster Analysis”, John Wiley & Sons, 2005

[5] Karypis, G., Han, E., Kumar, V., “CHAMELEON: A Hierarchical Clustering Algorithm Using Dynamic Modeling”, IEEE Computer: Special Issue on Data Analysis and Mining, pp. 68-75, 1999

[6] Ester, M., Kriegel, H., Sander, J., Xu ,X., “A Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise”, Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), pp. 226–231, 1996

[7] Basu, S., Banerjee, A., Mooney, R., “Semi-supervised Clustering by Seeding”, Proceedings of the 19th International Conference on Machine Learning, pp. 19-26, 2002

[8] Wagstaff, K., Cardie, C., Rogers, S., Schroedl ,S., “Constrained K-means Clustering with Background Knowledge”, 18th International Conference on Machine Learning, pp. 577-584, 2001

[9] Demiriz, A., Bennett, K., P., Embrechts, M., J., “Semi-supervised Clustering Using Genetic Algorithms”, Artificial Neural Networks in Engineering, pp. 809-814, 1999

[10] Basu, S., Banerjee, A., Mooney, R., “Active Semi-Supervision for Pairwise Constrained Clustering”, Proc. 4th SIAM Intl. Conf. on Data Mining (SDM-2004)

[11] Xing, P., Ng, Y., Jordan, M., Russell, S., “Distance Metric Learning, with Application to Clustering with Side-information”, Neural Information Processing Systems, pp. 521-528, 2002

[12] Bar-Hillel, A., Hertz, T., Shental, N., Weinshall, D., “Learning Distance Functions using Equivalence Relations”, Proceedings of the 20th International Conference on Machine Learning”, pp. 11-18, 2003

[13] Basu, S., Bilenko, M., Mooney, R., “Comparing and Unifying Search-Based and Similarity-Based Approaches to Semi-Supervised Clustering”, Proceedings of the ICML-2003 Workshop on the Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining Systems, pp.42-49, 2003

[14] Basu, S., Bilenko, M., Mooney, R., “A Probabilistic Framework for Semi-supervised Clustering”, Proceedings of Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 59-68, 2004

[15] Klein, D., Kamvar, S., Manning, C., “From Instance-level Constraints to Space-level Constraints: Making the Most of Prior Knowledge in Data Clustering”, Proceeding ICML '02 Proceedings of the Nineteenth International Conference on Machine Learning, pp. 307-314, 2002

[16] Weinberger, K., Blitzer, J., Saul, L., ”Distance Metric Learning for Large Margin Nearest Neighbor Classification”, Neural Information Processing Systems, pp. 1473-1480, 2006

[17] Cox, T., Cox, M., “Multidimensional Scaling, 2nd Edition”, Chapman & Hall, 2001

[18] Kruskal, J.B., “Nonmetric Multidimensional Scaling: a Numerical Method”, Psychometrika, pp. 115-129, 1964

[19] Rand, W.M., “Objective Criteria for the Evaluation of Clustering Methods”, Journal of the American Statistical Association, pp. 846-850, 1971

[20] Hubert, L., Arabie, P., “Comparing Partitions”, Journal of Classification, pp. 193-218, 1985

[21] Cohen, J., “A Coefficient of Agreement for Nominal Scales”, Educational and Psychological Measurement 196037-196046, 1960

[22] Landis, J.R., Koch, G.G., “The Measurement of Observer Agreement for Categorical Data”, Biometrics, pp. 159-174, 1977

[23] Reilly, C., Wang, C., Rutherford, M., “A Rapid Method for The Comparison of Cluster Analysis”, Statistica Sinica, pp. 19-33, 2005

[24] Ihaka, R., Gentleman, R., “R: A Language for Data Analysis and Graphics”, Journal of Computational and Graphical Statistics, pp. 299-314, 1996

[25] Kleinberg, J., Tardos, E., “Approximation algorithms for classification problems with pairwise relationships: Metric Labeling And Markov Random Field”, Journal of the ACM, pp. 616-639, 2002

[26] Blake, C., Merz, C., “UCI repository of machine learning databases”, 1998

[27] Daszykowski, M., Walczak, B., Massart, D., “Looking for Natural Patterns in Data. Part 1: Density Based Approach”, Chemometrics and Intelligent Laboratory Systems, pp. 83-92, 2001