國立交通大學

電控工程研究所

碩士論文

一近似的費雪線性鑑別分析於分群的應用

An Approximate Fisher Linear Discriminant Analysis for Clustering

研 究 生：楊承綱

指導教授：周志成　博士

中華民國一百年六月

一近似的費雪線性鑑別分析於分群的應用

An Approximate Fisher Linear Discriminant Analysis for Clustering

研 究 生：楊承綱　　Student：Cheng-Gang Yang

指導教授：周志成　　Advisor：Chi-Cheng Jou

國 立 交 通 大 學

電控工程研究所

碩 士 論 文

A Thesis

Submitted to Department of Electrical and Control Engineering

College of Electrical Engineering

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Master

in

Electrical and Control Engineering

June　2011

Hsinchu, Taiwan, Republic of China

中華民國一百年六月

一近似的費雪線性鑑別分析於分群的應用

學生：楊承綱　　指導教授：周志成

國立交通大學電控工程研究所

摘　　要

在大量資料取得越來越容易的時代，資料分群顯得更為重要。分群的困難處在於每一筆資料都有多種統計數據，稱為特徵，我們如何選擇特徵或其組合尤其影響分群結果。主成份分析是一種常見的特徵提取方法，然而提取最大變異成分未必對分類或分群有最好的效果。本論文針對特徵提取進行改善，我們結合在分類應用上具有優秀特徵提取功能的費雪線性鑑別分析，與傳統的 K-平均分群法 (K-means) 成一個近似費雪線性鑑別分析演算法 (approximate Fisher linear discriminant, AFD)。先令 K-平均分群後的結果作為已知類別，再利用費雪線性鑑別分析尋找最佳特徵，之後又使用此特徵重新分群再作費雪分析，又得到新分群結果的最佳特徵，如此反覆直到收斂。本論文選用兩種含有三個類別的資料 Iris 和 Wine 進行實驗，並根據真實類別比對分群結果的準確率。實驗結果發現，變異最大的成份雖保有原始資料最多的訊息，但並非都對分群有幫助，透過 AFD 演算法提取關鍵的特徵再進行分群，證實比主成份分析來的優秀，在相同的特徵數下能有較好的分群結果。

An Approximate Fisher Linear Discriminant Analysis for Clustering

Student：Cheng-Gang Yang　　Advisor：Dr. Chi-Cheng Jou

Department of Electrical and Control Engineering

National Chiao Tung University

ABSTRACT

In the era we get the large amounts of data more and more easily, the data clustering becomes more and more important. The difficulty of clustering is that every case has many statistics which call features, how we choose these features or their combination will effect the clustering result extremely. Principal component analysis (PCA) is one of the common feature extraction methods, but extracting the components of maximum variance is uncertain best for both classification and clustering. This thesis focuses on improving the feature extraction, we combine Fisher linear discriminant (FLD) which can extract the features excellently for classification and the traditional K-means clustering to an approximate Fisher linear discriminant (AFD) algorithm. Let the K-means clustering result is the known class, then use FLD to find the best features, after that, use these features to cluster and then do FLD again, we also get the best features for this new clustering result. Repeat above process until convergence. This thesis chooses two kinds of the data, Iris and Wine, that have three classes to do experiment, and compare the clustering accuracy by the real class. By experiment we

find that even though the components of maximum variance can contain the most information of the original data, but it is not useful for clustering. Extracting the key features by AFD algorithm to cluster is better than PCA, and in the same number of features AFD algorithm has better clustering result than PCA.

誌謝

碩士畢業了，首先要感謝我的父母，從小一路栽培我至今，也因為他們的支持與鼓勵，讓我在學業上能無後顧之憂的全力以赴。

研究所兩年生活中，最要特別感謝的便是我的指導教授周志成老師。周老師的平行思考往往讓我有仰之彌高的感覺，對我思考模式有大的啟發，在我研究陷入死胡同時，老師都能指引一條明路讓我走，除此之外，老師也很健談，讓每周固定的討論時間不會無聊，能順利的完成碩士論文，要謝謝老師。

此外，也十分感謝實驗室的夥伴們，能在研究之餘一同休閒，讓研究所生活增添不少樂趣，學業上也得之於夥伴們的互相幫忙，讓我課業問題能得以解決。

最後，謝謝口試委員對論文的建議以及提點，讓我可以知道改進的方向，也讓這份論文能更加完善。

目錄

圖目錄

表目錄

第一章　　序論

1.1　前言

在機器學習 (machine learning)領域上，大致可以分為兩類：

1. 監督式學習 (supervised learning)：現有資料含有明確的訊息，這些資料稱為訓練資料 (training data)，把每一筆資料當作輸入變數，皆對應一個輸出變數，這個輸出變數可以是連續變數或是離散變數，若為連續變數，則訓練資料可以用來建立迴歸模型，當某一筆測試資料輸入時，可以用此模型來預測輸出；若是離散變數，代表是類別訊息，可以用訓練資料建立分類模型，並預測測試資料所屬的類別。
2. 非監督式學習 (unsupervised learning)：與監督式學習相反，現有的資料皆不帶任何明確訊息，每一筆資料當作輸入，沒有對應的輸出，無法建立任何模型。

分類 (Classification)是根據資料已知的離散類別變數來建立分類模型，屬於監督式學習的應用。分群 (Clustering)則是在全部皆為未標記 (unlabeled)的資料上進行集群分析，屬於非監督式學習，如下圖：

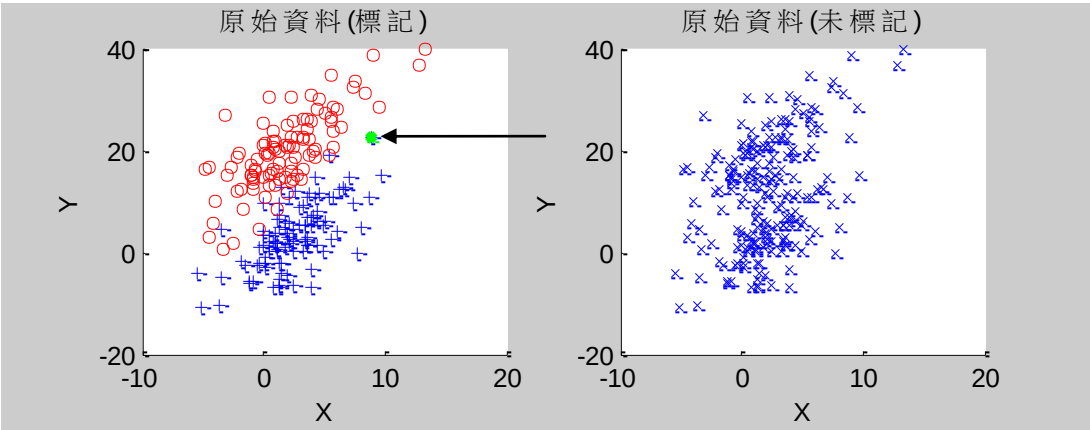

圖 1-1 分類(左)與分群(右)的差異

上圖左，當問綠色的點是屬於紅色還是藍色那類，這是屬於分類問題；相反的，

上圖右若問裡面應該有幾群資料，又該怎麼劃分，這就屬於分群問題。而為什麼要進行分群呢？透過分群，我們可以量化資料，還能找出圖形的結構，並且把資料宇集合分成數個子集合，子集合內的資料有較相似的屬性，因此在各群內可以用較少的資料代表此群全體資料，借此可以減少計算量。

分群過程可分為下圖幾個步驟：

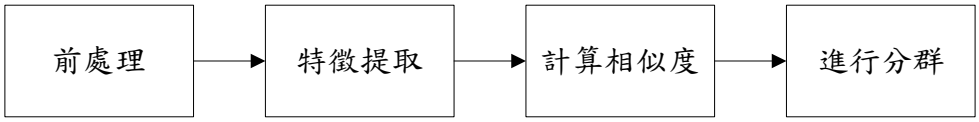

圖 1-2 分群流程

前處理包括過濾離群值 (outlier)以及刪除或補足遺漏值 (Missing)，目的是為了增加資料的有效性。進行分群時，是以相似度為依據，期望各組之內的相似度越大，而組與組之間的相似度越小，相似度的計算則可以依需求制定，如歐式距離(Euclidean distance)以及馬式距離 (Mahalanobis distance)等。歐式距離則是一種常用的相似度計算方式。分群可以針對變數或是資料，變數分群可以探討變數之間的相關性，本論文則是針對資料進行分群。分群被廣泛應用於資料探勘、圖形分類、文件檢索、生物資訊，以及影像分割等。分群演算法可粗略分為兩種：

1. 階層式 (hierarchical)
2. 分割式 (partitioning)

其中，階層式包含聚合法、分裂法兩種方式。常見的聚合法有單一連結聚合法、全部連結聚合法、平均連結聚合法，華德法等。分割式則有 K-means 演算法和 fuzzy C-means 演算法，其中尤以 K-means 演算法為最常見的方法。

1.2 研究動機與目的

從分群流程圖 1-2 中，可以發現能改變分群結果的步驟為：特徵提取、計算

相似度，以及進行分群的部分。要有意義的改變相似度的計算方式必須先觀察資料散佈的情形，如馬氏距離的計算是依據各變數間的共變矩陣 (covariance matrix)，然而這樣的改變未必對分群是有幫助的，因此改變相似度的方式多半應用在分類模型。而分群流程的最後一個步驟—進行分群，主要改善的方向在於討論合理的分群數目，這方面也已經有許多的分群指標可以用來判斷合理的分群數目，待後面章節我們會介紹。剩下特徵提取的部分，作特徵提取的原因在於原始變數中往往有許多對分群無用甚至有害的訊息，例如兩個相關性很強的變數 $\mathbf{x}_1$ 和 $\mathbf{x}_2$，當 $\mathbf{x}_1$ 的值很大時，$\mathbf{x}_2$ 值也會很大，如此一來在計算兩點距離時會因為 $\left|x_1^1 - x_1^2\right|$ 很大，$\left|x_2^1 - x_2^2\right|$ 也跟著很大(註：上標代表不同筆數，下標代表不同變數)，兩點距離就變很遠，那麼變數 $\mathbf{x}_1$ 和 $\mathbf{x}_2$ 就可能成為分群的重要依據，或者發生下圖情形時，X 軸顯然會對分群較無幫助。

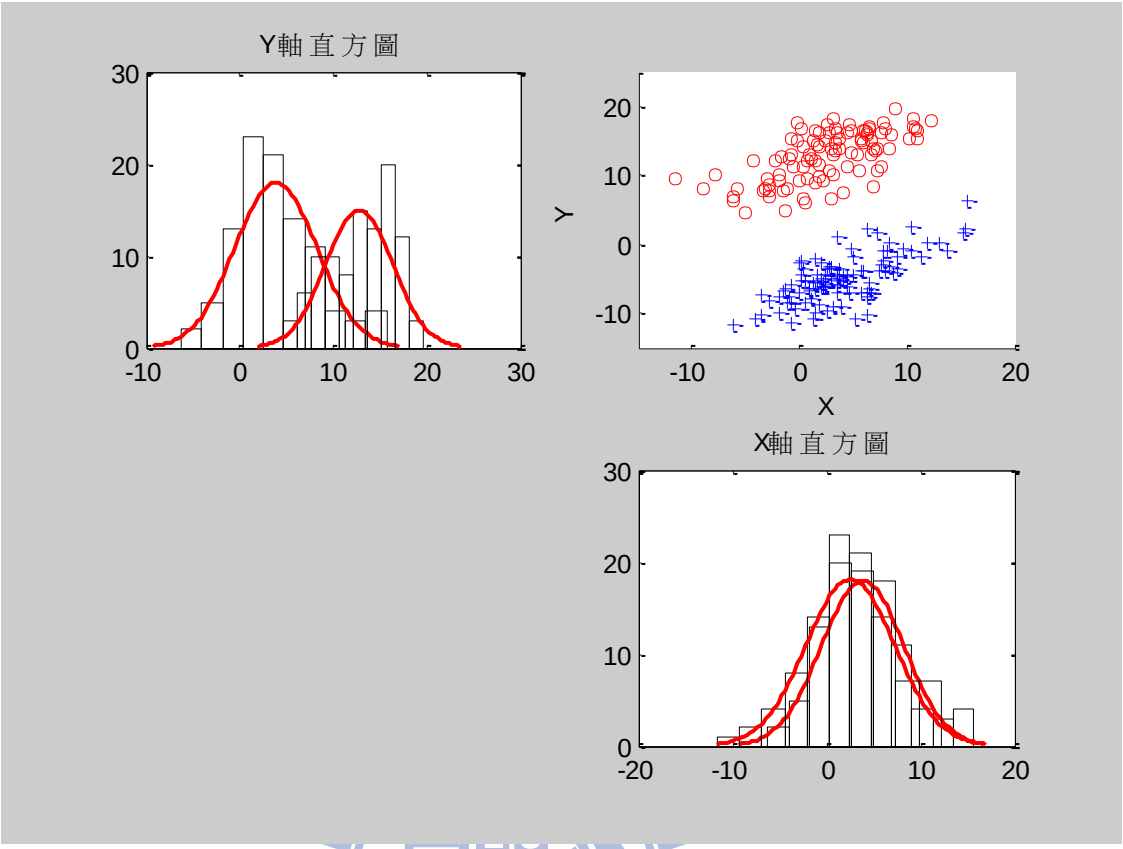

圖 1-3 示範變數無用的情形

對於變數存有相關性的問題，傳統的主成份分析 (Principal Component Analysis, PCA)可將原始變數透過線性轉換，轉成一組彼此無相關性的變數，即便如此，仍然未解決無用變數的問題，因此本論文便針對特徵提取作出改善。改善的目標即為挑選出對分群有幫助的特徵，方法如下：

1. 使用投影的方法。由於焦點放在統計資料的分群上，不同於人造資料或圖形式的資料，自然的統計資料其群聚邊界多半為線性，且使用線性的投影方法也較為容易。
2. 投影軸必須對於不同群集有良好的鑑別能力。如此一來投影上去的資料，才能明顯的觀察出各群聚邊界，也期望能藉此更容易判斷出該分成幾群。

基於以上兩點，若在監督式學習上，就是費雪線性鑑別分析 (Fisher Linear

Discriminant Analysis, FLD)，但是 FLD 無法應用在分群上，我們無法從未標記資料上找出這個投影軸。現在假設在未標記資料上可以找出這一個分群結果較佳的投影軸，反過來說，在這個軸上進行分群就會有較佳的結果，所以期望能應用這種監督式學習的方法在分群上，產生一個新的學習方法，可以用來提取特徵，並且預期會有較好的分群結果。

1.3 論文架構

第一章序論為問題描述，簡短說明作本研究的動機與目的以及研究的方向；第二章介紹特徵提取的方法以及分群的演算方法；第三章為本文所提出的方法並接著探討，之後在第四章使用含有真實類別的統計資料做實驗；第五章為結論。

第二章　　分群演算法及特徵提取

當資料的變數多且數據量大時，將不易運用這些資料，若能經由數學變換找出具有代表性的特徵，保留少數重要訊息，不但後續的應用能更為正確，也能減少系統運算量，這便是特徵提取的目的。

如同之前所提，分群方法已存在許多知名且常用的方法，本章節將介紹其中幾種，並在之後的章節用這些方法進行實驗。

2.1　分割式分群法

2.1.1　K-means 演算法

K-means 演算法是 J. B. MacQueen 於 1967 提出的演算法[1]。以 k 為輸入參數，欲把 n 筆資料分為 k 群，以使其各群內具有較高的相似度，而各群與各群之間的相似度較低。相似度的計算是根據群中資料點的平均值來進行。其目的在於最小化誤差平方總合：

$$E = \sum_{i=1}^{k} \sum_{X \in C_i} \|X - m_i\|^2 \tag{2.1}$$

X 為任一筆資料，m_i 為第 i 個群集的中心，k 為群集數目。

其演算法步驟如下：

輸入：全部的資料以及分群數目 k

1. 隨機選取 k 筆資料作為初始 k 個群集中心。
2. 計算每一筆資料到各個中心之間的距離，並指派此筆資料給距離最近的群集，此時會形成一個群集邊界，產生了群集的成員集合。
3. 根據邊界內的每一筆資料重新計算出該群集的中心，並取代上一次的中心。

4. 重複步驟 2 和步驟 3，直到群集成員不再變動為止。

2.1.2 K-medoids 演算法

由於離群值對中心點位置有極大的影響，中心點的位置又會影響群集的邊界，因此 K-means 演算法對離群值是敏感的，為了改善這種敏感性，誕生了 K-medoids 演算法[2]，其過程類似 K-means 演算法，差別在於後者直接以群集平均值當作中心，前者則是以最靠近此平均值的真實資料點作為中心。其演算過程如下：

輸入：全部的資料以及分群數目 k

1. 隨機選取 k 筆資料當作起始中心。
2. 計算每一筆資料到各個中心之間的距離，並指派此筆資料給距離最近的群集
3. 各群隨機選取任一不為中心點的資料計算其成本，即以此筆資料當作中心，計算其誤差平方和，若成本小於原先的中心，便以此筆資料當作新的中心點。
4. 重複步驟 2 和步驟 3，一直執行到群集成員不再變動為止。

以下圖作為範例說明 K-means 和 K-medoids 的差異。最左邊的點(-200,0)是離群值，右邊三個點(10,10),(5,0),(8,-10)是對分群有幫助的資料樣本。

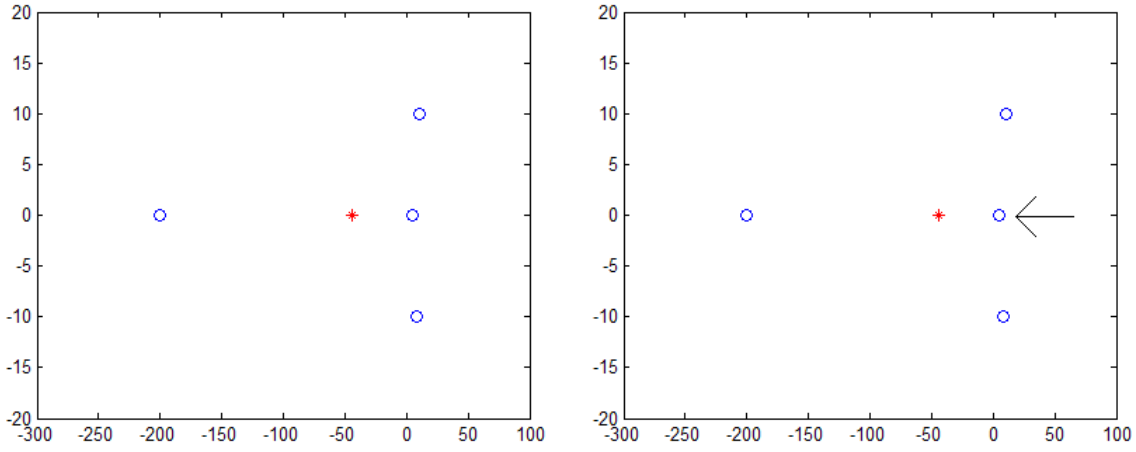

圖 2-1 (a)K-means 的中心　　(b)K-medoids 的中心，箭頭所指處

從圖 2-1 可看出 K-means 的中心受到左邊的離群值影響，而 K-medoids 的中心是從真正存在的資料上選取，因此與真正有用的右邊三個點的中心更為相近。這就是 K-medoids 較能抵抗離群值的原因，但其缺點是計算量比 K-means 來的多。若總共有 n 筆資料，K-means 的複雜度為 $O(n)$，而 K-medoids 為 $O(n^2)$，因此較適合在資料筆數少的情況。

2.1.3 模糊 C-means 演算法

最早由 Dunn 於 1973 年首先提出[3]，並在經 Bezdek 改善[4]。其目的是透過模糊邏輯的概念，希望能進一步提升分群的效果。與 K-means 相似，差別在於任何一筆資料，可以用 0 到 1 的數字表示屬於某一群集的程度，而不像 K-means 只有屬於和不屬於兩種選擇。假設 $\mathbf{U}$ 為一個 $c \times n$ 的矩陣，其中 n 為資料樣本數目，c 為欲分的群數，則 u_{ij} 代表第 j 筆資料屬於第 i 群的程度，而每一行總和為 1，即

$$\sum_{i=1}^{c} u_{ij} = 1 \tag{2.2}$$

在此條件下欲最小化目標函數

$$J = \sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}^{m} \left\| X_j - C_i \right\|^2 \tag{2.3}$$

m 為大於 1 的權重指數，X_j 為第 j 筆資料，C_i 為第 i 群中心，代入拉格朗日 (Lagrangian)條件方程可得

$$c_i = \frac{\sum_{j=1}^{n} u_{ij}^{m} X_j}{\sum_{j=1}^{n} u_{ij}^{m}} \tag{2.4}$$

所以其演算過程如下

1. 初始化矩陣 $\mathbf{U}$。

2. 透過式(2.4)計算各群集中心。

3. 更新 $\mathbf{U}$，$u_{ij} = \dfrac{1}{\sum_{k=1}^{c}\left(\dfrac{\|X_j - C_i\|}{\|X_j - C_k\|}\right)^{\frac{2}{m-1}}}$

4. 透過式(2.3)計算 J，若與前次差距 $\left|J^{(t+1)} - J^{(t)}\right| < \varepsilon$，則停止，否則重複步驟 2 到步驟 4。

最後，分群結果依據矩陣 $\mathbf{U}$，將第 j 筆資料指派給 $\arg\max_i u_{ij}$。

2.2 階層式分群法

階層式分群法會產生一樹狀結構，由樹狀結構可看出其分群結果，分為兩種：

1. 聚合式：由樹狀底部開始，一開始每個資料點都自成一個群集，並逐一將相似度較大的兩個合併，慢慢往上生成頂部。
2. 分裂式：由樹狀頂部開始，一開始全部資料都是同一個群集，逐一將相似度小的分離，慢慢往下生成底部。

聚合法為較常用的方法，以下介紹聚合法中較常使用的距離度量方式。

單一連結法 (single linkage)：

群集 X 與群集 Y 之間的距離定義為兩群之中最接近兩點的距離：

$$D(X,Y) = \min_{x\in X, y\in Y} d(x,y) \tag{2.5}$$

全部連結法 (complete linkage)：

群集 X 與群集 Y 之間的距離定義為兩群之中最遠兩點的距離：

$$D(X,Y) = \max_{x\in X, y\in Y} d(x,y) \tag{2.6}$$

平均連結法 (average linkage)：

群集 X 與群集 Y 之間的距離定義為兩群之間各點到各點的距離平均：

$$D(X,Y)=\frac{\sum_{x\in X,y\in Y} d(x,y)}{|X||Y|} \tag{2.7}$$

華德法 (Ward's method)：

群集 X 與群集 Y 間的距離定義為在將兩群合併後，各點到合併後中心的距離平方和

$$D(X,Y)=\sum_{v\in X\cup Y}\|v-m\|^2 \tag{2.8}$$

m 為合併後的中心。

下圖為針對圖 1-3 的兩類人造資料分別使用四種連結方法的樹狀圖。

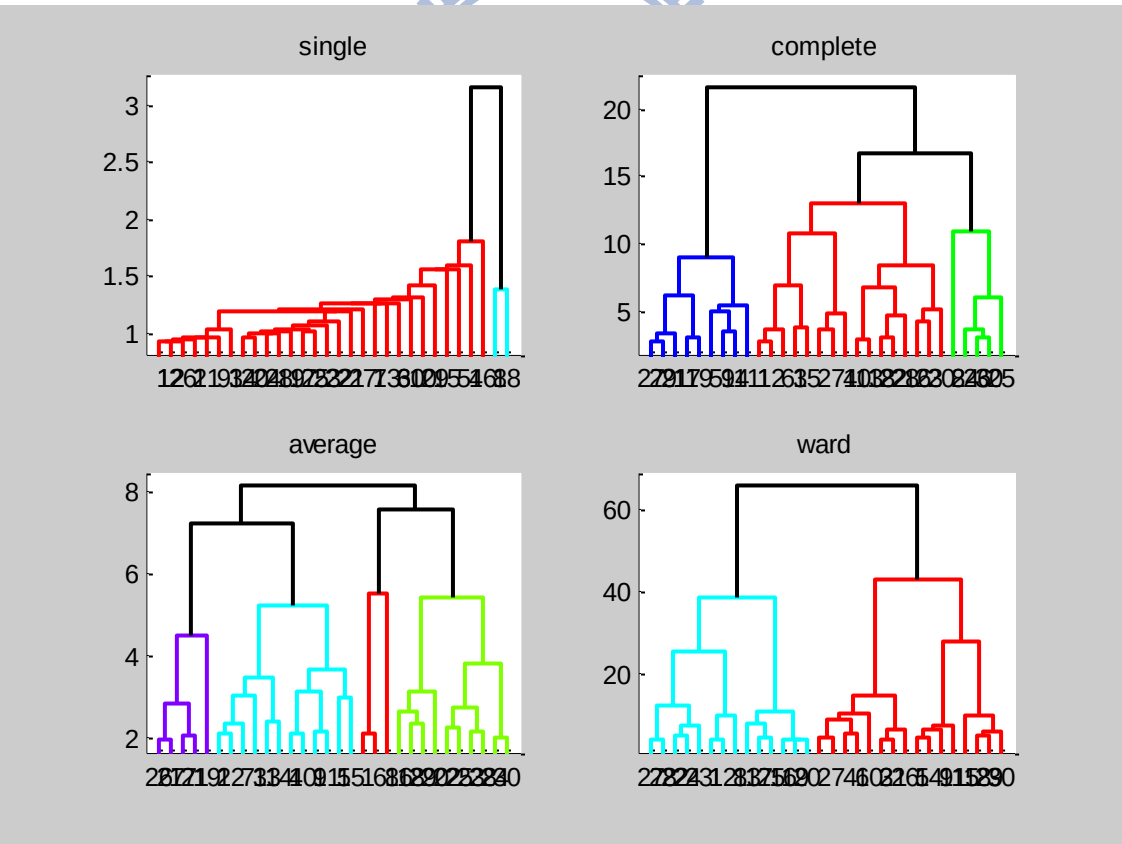

圖 2-2 四種分群樹狀圖

我們可以發現，單一連結法出現極不對稱的結果，因為每次合併都只考慮最近的

距離。對某個點來說，距離它最近的點有越大的機會在越多點的群內，因此大的群會越來越大，所以這四種連結方法以此為最少人使用。平均連結以及完整連結則是比較容易出現群數較多的情形，因為群內的點一旦變多時，外部的點與此群的距離就會越來越大，要併入的機會也越來越小。華德法則是四種方法裡較佳的方法。

2.3 主成份分析

主成份分析由 Pearson 於 1901 年提出，並在 1933 年經由 Hotelling 加以發展。在統計學上，主成份分析是一種維度簡化的技術。它是將原始變數經過線性變換後得到一組新的變數，而這組新的變數其變數與變數之間並沒有相關性，彼此是正交的。並且，原始數據投影上去後，第一個坐標軸擁有最大的變異量，稱為第一主成份，第二個坐標軸擁有第二大變異量，稱為第二主成份，以此類推。相較於其他基底，主成份分析可以提取對資料全體變異量有最大貢獻的特徵。

原理敘述如下：

為方便起見，假設 N 筆資料資料已先經過中心化 (centered)，為 $\mathbf{x}^1,\mathbf{x}^2,\mathbf{x}^3,\cdots,\mathbf{x}^N$ ，維度為 p ，現有一組正交且單位長的基底 $V=\{\mathbf{v}_1,\mathbf{v}_2,\mathbf{v}_3,\cdots,\mathbf{v}_p\}$ ，即

$$\mathbf{v}_i^T\mathbf{v}_j=\begin{cases}1 & i=j\\ 0 & i\neq j\end{cases} \tag{2.9}$$

原始空間中的資料皆可以表示為此基底的線性組合

$$\mathbf{x}^n=\sum_{i=1}^{p} z_i^n\mathbf{v}_i \tag{2.10}$$

$$z_i^n=(\mathbf{x}^n)^T\mathbf{v}_i \tag{2.11}$$

$$\mathbf{z}_i=\mathbf{X}\mathbf{v}_i\ ,\ \mathbf{X}=[\mathbf{x}^1,\mathbf{x}^2,\mathbf{x}^3,\cdots,\mathbf{x}^N]^T\quad i=1,2,\cdots,p \tag{2.12}$$

主成份分析的目的是要在 $\mathbf{z}_i$, $i=1,2,\cdots,p$ 之中找出擁有最大變異量的變數 $\mathbf{z}$ 。

樣本變異數 $\text{var}(\mathbf{z})=\frac{1}{N-1}\sum_{n=1}^{N}(z^n-\bar{z})^2$ ，因為中心化，所以 $\sum_{n=1}^{N}\mathbf{x}^n=0$ ，將式(2.10)代入， $\sum_{n=1}^{N}\mathbf{x}^n=0\Rightarrow\sum_{n=1}^{N}\sum_{i=1}^{p}z_i^n\mathbf{v}_i=\sum_{i=1}^{p}\sum_{n=1}^{N}z_i^n\mathbf{v}_i=0$ ，又因為 $\mathbf{v}_i$ 彼此獨立，故可得

$\sum_{n=1}^{N}z_i^n=0\Rightarrow\bar{z}_i=0$, for $i=1,2,\cdots p$ ，因此 $\text{var}(\mathbf{z})=\frac{1}{N-1}\sum_{n=1}^{N}(z^n)^2=\frac{1}{N-1}\mathbf{z}^T\mathbf{z}$ ，式(2.12)代入可得

$$\text{var}(\mathbf{z})=\frac{1}{N-1}\mathbf{v}^T\mathbf{X}^T\mathbf{X}\mathbf{v}=\mathbf{v}^T\mathbf{C}\mathbf{v} \tag{2.13}$$

$\mathbf{C}=\frac{1}{N-1}\mathbf{X}^T\mathbf{X}$，為樣本共變異數矩陣(sample covariance matrix)。欲在條件 $\mathbf{v}^T\mathbf{v}=1$ 下最大化式(2.13)，可代入拉格朗日條件方程：

$$L(\mathbf{v},\lambda)=\mathbf{v}^T\mathbf{C}\mathbf{v}-\lambda(\mathbf{v}^T\mathbf{v}-1) \tag{2.14}$$

將式(2.14)偏微分

$$\frac{\partial L}{\partial\mathbf{v}}=2\mathbf{C}\mathbf{v}-2\lambda\mathbf{v} \tag{2.15}$$

$$\frac{\partial L}{\partial\lambda}=1-\mathbf{v}^T\mathbf{v} \tag{2.16}$$

令式(2.15)及式(2.16)等於 0 可得

$$\mathbf{C}\mathbf{v}=\lambda\mathbf{v} \tag{2.17}$$

$\mathbf{v}$ 即為 $\mathbf{C}$ 的最大特徵值所對應的單位長特徵向量，稱為第一主成份向量，而擁有最大變異量的 $\mathbf{z}$ ，即 $\mathbf{X}$ 投影在 $\mathbf{v}$ 的座標，稱為第一主成份得點 (score)。第二主成份向量即為 $\mathbf{C}$ 的第二大特徵值所對應的單位長特徵向量，以此類推。

主成份分析性質如下：

1. 將式(2.17)代入式(2.13)重新整理可得

   $\text{var}(\mathbf{z}) = \mathbf{v}^T\mathbf{C}\mathbf{v} = \mathbf{v}^T\lambda\mathbf{v} = \lambda\mathbf{v}^T\mathbf{v} = \lambda$，換句話說，變異量第 i 大的變數 $\mathbf{z}_i$，其變異量就是樣本共變異數矩陣第 i 大的特徵值 λ_i。

2. $\mathbf{z}_i^T\mathbf{z}_j = (\mathbf{X}\mathbf{v}_i)^T(\mathbf{X}\mathbf{v}_j) = \mathbf{v}_i^T\mathbf{X}^T\mathbf{X}\mathbf{v}_j = \mathbf{v}_i^T(\mathbf{X}^T\mathbf{X}\mathbf{v}_j) = (N-1)\lambda_2\mathbf{v}_i^T\mathbf{v}_j = 0$, for $i \neq j$，可看出變數與變數之間並沒有相關性，彼此正交。

3. 若變數 $\mathbf{x}$ 的度量尺度不同，可以先將其標準化，此時共變異數矩陣 $\mathbf{C}$ 等於相關係數矩陣 $\mathbf{R}$，所有變數 $\mathbf{z}_i$ 的變異量總和等於維度數 p

$$\text{var}\left(\sum_{i=1}^{p}\mathbf{z}_i\right) = \sum_{i=1}^{p}\text{var}(\mathbf{z}_i) = tr\begin{bmatrix}\lambda_1 & & \\ & \ddots & \\ & & \lambda_p\end{bmatrix} = tr(\mathbf{D}) = tr(\mathbf{V}^T\mathbf{R}\mathbf{V}) = tr(\mathbf{R}\mathbf{V}\mathbf{V}^T) = tr(\mathbf{R}) = p$$

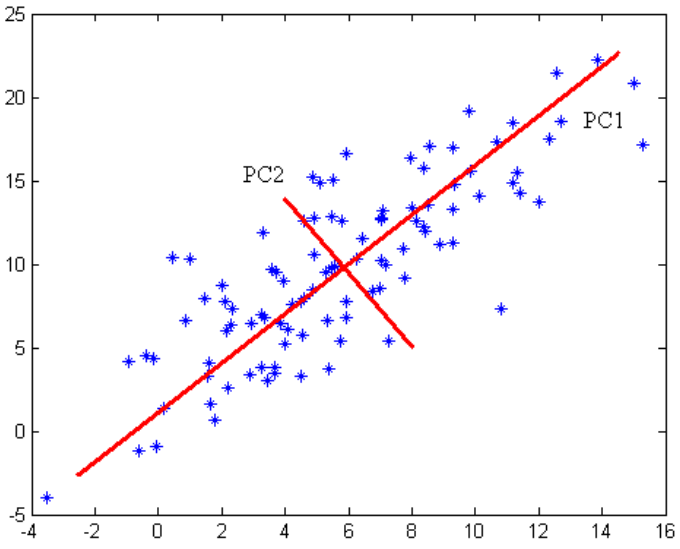

圖 2-3 兩個主成份方向 PC1 和 PC2

變異量 $\text{var}(\mathbf{z}) = \frac{1}{N-1}\sum_{n=1}^{N}(z^n)^2 \propto \sum_{n=1}^{N}(z^n)^2$ 與能量成正比，因此比較大的主成份有較大的能量，當我們捨棄較小的主成份時，形同捨棄了較小的能量，而較小的能量通常是由雜訊所造成，所以主成份分析不但能有效降低維度，同時能保留真正的訊

號，移除雜訊的影響，可謂一舉兩得。

2.4 費雪線性鑑別

在給定一主成份分析的基底之後，資料投影在此空間前後的差距會最小，然而費雪線性鑑別並不是如此，資料在投影之後群集能有明顯的區別，以利於分類。如下圖：L 線便是 FLD 找出的方向。

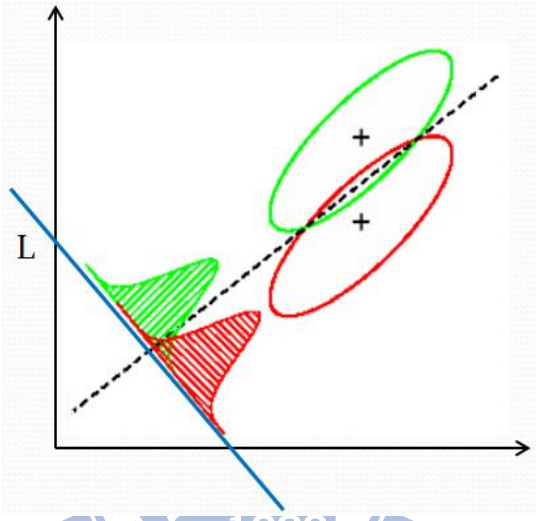

圖 2-4 FLD 範例

原理敘述如下：

FLD 是希望能找出一個軸，能將兩群標記資料投影上去之後，同一群內資料能越近越好，而兩群彼此之間能越遠越好。令這個軸為 $\mathbf{a}$，兩群原始資料的平均分別為 $\mathbf{m}_1 = \frac{1}{n_1}\sum_{\mathbf{x}\in\mathbf{C}_1}\mathbf{x}$, $\mathbf{m}_2 = \frac{1}{n_2}\sum_{\mathbf{x}\in\mathbf{C}_2}\mathbf{x}$，投影上去之後的平均值為

$$\tilde{m}_1 = \frac{1}{n_1}\sum_{\mathbf{x}\in\mathbf{C}_1}\mathbf{a}^T\mathbf{x} = \mathbf{a}^T\mathbf{m}_1 \text{ , } \tilde{m}_2 = \frac{1}{n_2}\sum_{\mathbf{x}\in\mathbf{C}_2}\mathbf{a}^T\mathbf{x} = \mathbf{a}^T\mathbf{m}_2 \tag{2.18}$$

定義度量群與群之間分散程度的方式為

$$(\tilde{m}_1 - \tilde{m}_2)^2 \tag{2.19}$$

各群內分散的程度為

$$S_i^2 = \sum_{\mathbf{x}\in\mathbf{C}_i} (\mathbf{a}^T\mathbf{x} - \tilde{m}_i)^2 \tag{2.20}$$

根據 FLD 所期望的結果，可以最大化下式：

$$J(\mathbf{a}) = \frac{(\tilde{m}_1 - \tilde{m}_2)^2}{S_1^2 + S_2^2} \tag{2.21}$$

稱為費雪準則 (Fisher criterion)。將式(2.18)代入式(2.19)重新整理

$$\begin{aligned}(\tilde{m}_1 - \tilde{m}_2)^2 &= (\mathbf{a}^T\mathbf{m}_1 - \mathbf{a}^T\mathbf{m}_2)^2 \\ &= (\mathbf{a}^T\mathbf{m}_1 - \mathbf{a}^T\mathbf{m}_2)(\mathbf{m}_1^T\mathbf{a} - \mathbf{m}_2^T\mathbf{a}) \\ &= \mathbf{a}^T(\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T\mathbf{a} \\ &= \mathbf{a}^T\mathbf{S}_B\mathbf{a}\end{aligned} \tag{2.22}$$

其中 $\mathbf{S}_B$ 稱為組間共變異矩陣 (between-class covariance matrix)。式(2.18)和式(2.20)代入式(2.21)的分母

$$\begin{aligned}S_1^2 + S_2^2 &= \sum_{\mathbf{x}\in\mathbf{C}_1} (\mathbf{a}^T\mathbf{x} - \tilde{m}_1)^2 + \sum_{\mathbf{x}\in\mathbf{C}_2} (\mathbf{a}^T\mathbf{x} - \tilde{m}_2)^2 \\ &= \sum_{\mathbf{x}\in\mathbf{C}_1} (\mathbf{a}^T\mathbf{x} - \mathbf{a}^T\mathbf{m}_1)(\mathbf{x}^T\mathbf{a} - \mathbf{m}_1^T\mathbf{a}) + \sum_{\mathbf{x}\in\mathbf{C}_2} (\mathbf{a}^T\mathbf{x} - \mathbf{a}^T\mathbf{m}_2)(\mathbf{x}^T\mathbf{a} - \mathbf{m}_2^T\mathbf{a}) \\ &= \mathbf{a}^T\left(\sum_{\mathbf{x}\in\mathbf{C}_1} (\mathbf{x} - \mathbf{m}_1)(\mathbf{x} - \mathbf{m}_1)^T + \sum_{\mathbf{x}\in\mathbf{C}_2} (\mathbf{x} - \mathbf{m}_2)(\mathbf{x} - \mathbf{m}_2)^T\right)\mathbf{a} \\ &= \mathbf{a}^T\mathbf{S}_W\mathbf{a}\end{aligned} \tag{2.23}$$

其中 $\mathbf{S}_W$ 稱為組內共變異矩陣 (within-class covariance matrix)。

將式(2.22)和式(2.23)代入式(2.21)可得

$$J(\mathbf{a}) = \frac{\mathbf{a}^T\mathbf{S}_B\mathbf{a}}{\mathbf{a}^T\mathbf{S}_W\mathbf{a}} \tag{2.24}$$

$$\begin{aligned}\frac{\partial J(\mathbf{a})}{\partial \mathbf{a}} &= \frac{(\mathbf{a}^T\mathbf{S}_W\mathbf{a})\mathbf{S}_B\mathbf{a} - (\mathbf{a}^T\mathbf{S}_B\mathbf{a})\mathbf{S}_W\mathbf{a}}{(\mathbf{a}^T\mathbf{S}_W\mathbf{a})^2} = 0 \\ &\Rightarrow (\mathbf{a}^T\mathbf{S}_W\mathbf{a})\mathbf{S}_B\mathbf{a} = (\mathbf{a}^T\mathbf{S}_B\mathbf{a})\mathbf{S}_W\mathbf{a} \\ &\Rightarrow \mathbf{S}_B\mathbf{a} = c\cdot\mathbf{S}_W\mathbf{a} \\ &\Rightarrow \mathbf{S}_W^{-1}\mathbf{S}_B\mathbf{a} = c\cdot\mathbf{a}\end{aligned} \tag{2.25}$$

最後可以得到 $\mathbf{a}$ 就是 $\mathbf{S}_W^{-1}\mathbf{S}_B$ 的最大特徵值對應到的特徵向量，注意到式(2.25)整理一下可得

$$\mathbf{a} \propto \mathbf{S}_W^{-1}\mathbf{S}_B\mathbf{a} = \mathbf{S}_W^{-1}(\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T\mathbf{a} \propto \mathbf{S}_W^{-1}(\mathbf{m}_1 - \mathbf{m}_2) \tag{2.26}$$

所以我們無須解特徵值問題。

以上是兩類別一個軸的問題，現在將問題延伸到 c 類別 p 個軸：$\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \cdots, \mathbf{a}_p]$。修改

$$\mathbf{S}_W = \sum_{i=1}^{c}\sum_{\mathbf{x}\in\mathbf{C}_i}(\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T \tag{2.27}$$

$$\mathbf{S}_B = \sum_{i=1}^{c} n_i(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T \tag{2.28}$$

$\mathbf{m}$ 為全部資料平均，$\mathbf{m}_i$ 為第 i 類資料平均，n_i 為第 i 類資料數目，c 為類別數目。

欲讓同一類別內散佈情形越靠近，不同類別間散佈越分開，可最大化費雪準則：

$$J(\mathbf{A}) = \frac{\det(\mathbf{A}^T\mathbf{S}_B\mathbf{A})}{\det(\mathbf{A}^T\mathbf{S}_W\mathbf{A})} \tag{2.29}$$

或者

$$J(\mathbf{A}) = \frac{tr(\mathbf{A}^T\mathbf{S}_B\mathbf{A})}{tr(\mathbf{A}^T\mathbf{S}_W\mathbf{A})} \tag{2.30}$$

取行列式值 (determinant)或是跡 (trace)都是量化散步矩陣的方式。

解之可得

$$\mathbf{S}_W^{-1}\mathbf{S}_B\mathbf{A} = c\cdot\mathbf{A} \tag{2.31}$$

$\mathbf{A}$ 即為 $\mathbf{S}_W^{-1}\mathbf{S}_B$ 的特徵向量。與主成份向量不同的是，解出來並非是正交基底。

第三章　　研究方法

有鑑於費雪線性鑑別對於分類應用有良好的特徵提取能力，唯其無法用在未標記資料上。因此若是能找出最佳分群結果的關鍵特徵，並用此特徵進行分群，預期能有良好的分群結果。以下稱本論文所提出的方法為近似的費雪線性鑑別分析 (approximate Fisher linear discriminant, AFD)，簡稱 AFD。

先令前次分群的結果當作已知的類別來作 FLD 找出最佳特徵，再用此特徵進行分群，此次新分群的結果再作一次 FLD 又可得到新的最佳特徵，如此反覆的用分群結果更新特徵，便是 AFD 演算法的架構。

3.1　AFD 演算法

先介紹步驟，之後再解釋各個步驟的含意。

其步驟如下：

1. 資料先做主成份分析，找出其第一主成份。

   很直覺的，這麼作的目的是為了讓收斂的速度變快。

2. 使用各筆資料的第一主成份做 K-means 演算法分群。

   注意到 K-means 演算法會受到起始中心的影響，因此我們根據 Barakbah 和 Kiyoki [5]提出的方法改善初始值造成的局部最佳化。

3. 根據 K-means 分群的結果，依式(2.31)算出 FLD 第一鑑別向量

4. 將資料投影到步驟 3 找出的向量之後，其投影得點即為最佳特徵，用此特徵重新做 K-means 分群

5. 重複步驟 3 到步驟 4，直到收斂為止。

此時，我們已經找出第一個有最佳特徵的向量，但由於太少的特徵往往會失去許

多重要的資訊，因此欲找齊全部 p 個向量。根據 Duchene 及 Leclercq 所提出的方法能改善傳統費雪方法[6]，其法是找出一組正交的向量，有別於傳統不一定正交，其實驗結果也較為出色，故之後也選擇找出正交的鑑別向量，方法如下：

假設第一次找出的單位長鑑別向量為 $\mathbf{a}_1$，欲找出與其正交的向量 $\mathbf{a}_2$，在與 $\mathbf{a}_1$ 垂直的空間中找即可，因此對 $\mathbf{a}_1$ 做奇異值分解可以得到由左奇異向量組成的矩陣，即 $\mathbf{a}_1\mathbf{a}_1^T$ 的特徵向量組成的矩陣 $\mathbf{U}$

$$\mathbf{a}_1\mathbf{a}_1^{\mathbf{T}}\mathbf{U} = \mathbf{U}\mathbf{V}\ ,\ \mathbf{V} = \mathrm{diag}(\lambda_1, \lambda_2, \cdots, \lambda_p) \tag{3.1}$$

由於 $\mathbf{a}_1\mathbf{a}_1^{\mathbf{T}}$ 的秩為 1，根據線性代數定理，將會有 $p-1$ 個特徵值是 0，即

$$\begin{aligned} &\mathbf{a}_1\mathbf{a}_1^{\mathbf{T}}\mathbf{u}_i = 0\cdot\mathbf{u}_i = 0 \\ \Rightarrow\ &\mathbf{a}_1^{\mathbf{T}}\mathbf{u}_i = 0\ ,\ i = 2,3,\cdots,p \quad \text{since}\ \ \mathbf{a}_1 \neq \vec{0} \end{aligned} \tag{3.2}$$

又，$\sum_{i=1}^{p}\lambda_i = \lambda_1 = \mathrm{trace}(\mathbf{a}_1\mathbf{a}_1^{\mathbf{T}}) = \mathrm{trace}(\mathbf{a}_1^{\mathbf{T}}\mathbf{a}_1) = \mathrm{trace}(1) = 1 \Rightarrow \lambda_1 = 1$，即

$$\begin{aligned} &\mathbf{a}_1\mathbf{a}_1^{\mathbf{T}}\mathbf{u}_1 = 1\cdot\mathbf{u}_1 \\ \Rightarrow\ &c\mathbf{a}_1 = \mathbf{u}_1 \\ \Rightarrow\ &\mathbf{a}_1 = \mathbf{u}_1\ ,\ \text{if}\ |\mathbf{a}_1| = |\mathbf{u}_1| = 1 \end{aligned} \tag{3.3}$$

由式(3.2)和式(3.3)可知 $\mathbf{U}$ 即為自己和與自己垂直的一組正交基底所組成。當算出第一個鑑別向量 $\mathbf{a}_1$ 時，可以找出這組 $p-1$ 個的基底，再將原始空間的資料全部投影上去，並在這空間中找出第二個鑑別向量，重複此方法直到找出所有正交的鑑別向量為止。

全部演算流程如下頁圖：

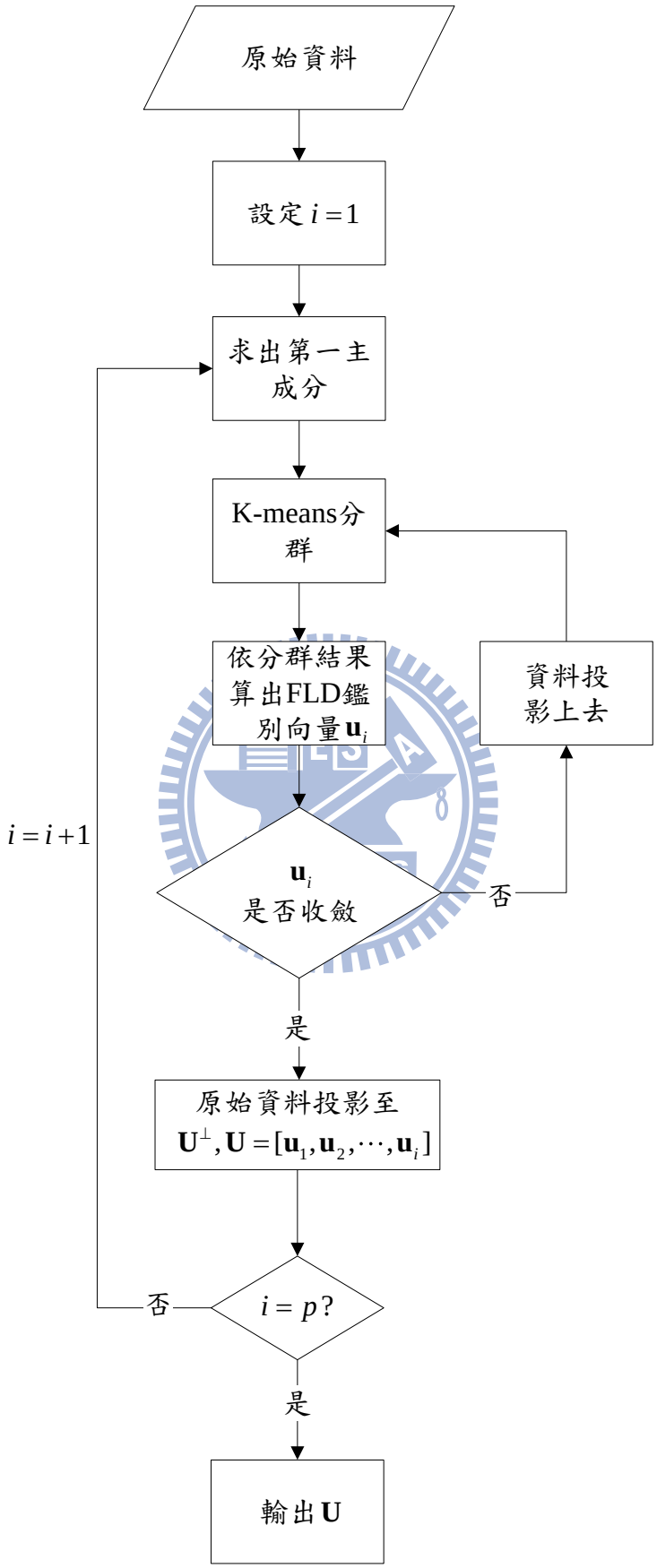
原始資料
設定 $i=1$
求出第一主成分
K-means分群
依分群結果算出FLD鑑別向量$\mathbf{u}_i$
資料投影上去
$\mathbf{u}_i$ 是否收斂
否
是
$i=i+1$
原始資料投影至 $\mathbf{U}^{\perp}, \mathbf{U}=[\mathbf{u}_1,\mathbf{u}_2,\cdots,\mathbf{u}_i]$
$i=p$?
否
是
輸出 $\mathbf{U}$

圖 3-1 演算流程

找出正交基底在已標記資料的分類應用上不但具有較好的實驗結果，在未標記資料上，這種做法還可以避免第二個以後的鑑別向量被第一個鑑別向量影響，因為之後找出來的鑑別向量可能全部都以很小的夾角圍繞在第一鑑別向量上，如此一來便失去找多個特徵軸的意義。

3.2 方法探討

由式(2.1)的目標函數 E 和式(2.27)，

$$\begin{aligned} trace(\mathbf{S}_W) &= trace\left(\sum_{i=1}^{c} \sum_{\mathbf{x}\in\mathbf{C}_i} (\mathbf{x}-\mathbf{m}_i)(\mathbf{x}-\mathbf{m}_i)^T \right) \\ &= \sum_{i=1}^{c} \sum_{\mathbf{x}\in\mathbf{C}_i} trace\left((\mathbf{x}-\mathbf{m}_i)(\mathbf{x}-\mathbf{m}_i)^T \right) \\ &= \sum_{i=1}^{c} \sum_{\mathbf{x}\in\mathbf{C}_i} trace\left((\mathbf{x}-\mathbf{m}_i)^T(\mathbf{x}-\mathbf{m}_i) \right) \\ &= \sum_{i=1}^{c} \sum_{\mathbf{x}\in\mathbf{C}_i} trace\left(\|\mathbf{x}-\mathbf{m}_i\|^2 \right) \\ &= \sum_{i=1}^{c} \sum_{\mathbf{x}\in\mathbf{C}_i} \|\mathbf{x}-\mathbf{m}_i\|^2 \\ &= E \end{aligned} \tag{3.4}$$

可以得到 K-means 演算法的目標就是 min $trace(\mathbf{S}_W)$。再由主成份分析的原理可得知，其目的是要找出單位向量 $\mathbf{a}$

$$\arg\max_{\mathbf{a}} \mathbf{a}^T\mathbf{C}\mathbf{a} = \arg\max_{\mathbf{a}} trace(\mathbf{a}^T\mathbf{S}_T\mathbf{a}) \tag{3.5}$$

其中總共變異矩陣 (total covariance matrix) $S_T = (N-1)\cdot\mathbf{C} = \mathbf{S}_B + \mathbf{S}_W$

而根據式(2.30)費雪線性鑑別目標為

$$\begin{aligned}\arg\max_{\mathbf{b}} \frac{trace(\mathbf{b}^T\mathbf{S}_B\mathbf{b})}{trace(\mathbf{b}^T\mathbf{S}_W\mathbf{b})} &= \arg\max_{\mathbf{b}} \left(\frac{trace(\mathbf{b}^T\mathbf{S}_T\mathbf{b})}{trace(\mathbf{b}^T\mathbf{S}_W\mathbf{b})} - \frac{trace(\mathbf{b}^T\mathbf{S}_W\mathbf{b})}{trace(\mathbf{b}^T\mathbf{S}_W\mathbf{b})} \right) \\ &= \arg\max_{\mathbf{b}} \left(\frac{trace(\mathbf{b}^T\mathbf{S}_T\mathbf{b})}{trace(\mathbf{b}^T\mathbf{S}_W\mathbf{b})} - 1 \right) \\ &= \arg\max_{\mathbf{b}} \frac{trace(\mathbf{b}^T\mathbf{S}_T\mathbf{b})}{trace(\mathbf{b}^T\mathbf{S}_W\mathbf{b})}\end{aligned} \tag{3.6}$$

結合式(3.4)、式(3.5)，和式(3.6)，如下

PCA

$$\text{FLD} = \frac{trace(\mathbf{b}^T\mathbf{S}_T\mathbf{b})}{trace(\mathbf{b}^T\mathbf{S}_W\mathbf{b})}$$

K-means

當向量$\mathbf{b}$讓分母最小，分子最大時，就是最大的鑑別向量，但由於$\mathbf{S}_W$未知，所以初始時就先讓分子最大，也就是式(3.5)的主成份分析找出$\mathbf{a}$，之後再作 K-means 分群讓分母最小，即式(3.4)，如此一來，收斂的速度會最快。

找出近似的鑑別向量後，我們可以由向量元素的大小看出原始變數$\mathbf{X}=[\mathbf{x}_1\ \mathbf{x}_2 \ldots \mathbf{x}_p]$中哪些是對分群有幫助的，若找出的向量為$\mathbf{u}=\left[u_1\ u_2 \ldots u_p\right]^T$，那麼新的變數即為$\mathbf{z}=\mathbf{X}\mathbf{u}=u_1\mathbf{x}_1+u_2\mathbf{x}_2+\ldots+u_p\mathbf{x}_p$，兩點距離為：

$(z^m-z^k)^2=\left[u_1(x_1^m-x_1^k)+u_2(x_2^m-x_2^k)+\ldots+u_p(x_p^m-x_p^k)\right]^2$，假設$u_i$很大，表示原始變數$\mathbf{x}_i$對區別群集有很大的貢獻，在判斷距離上給予$\mathbf{x}_i$很高的權重。

最後一個問題：在非監督式學習的前提下，該如何證明 AFD 找出來的軸是比較好的呢？找出來的軸既然是用來分群，就依分群的結果來決定軸的優劣。甚麼樣的分群才是好的分群結果，這向來沒有一定的對與錯，也因此有許多指標可以用來評分，只要指標定義的方式符合分群的想法，也就是組間相似度小，組內相似度大即可，因此使用費雪準則當做我們的指標，以下簡稱為 FCI。常用的指標還

有側影值 (Silhouette value)和 DBI 指數 (Davies-Bouldin index，DBI)。側影值定義如下：令 $d(\mathbf{x}^i, A)$ 為 $\mathbf{x}^i$ 到 A 群所有點的平均距離，則 $a(i) = d(\mathbf{x}^i, A)$, $\mathbf{x}^i \in A$ ，$b(i) = \min_B d(\mathbf{x}^i, B)$, $\mathbf{x}^i \notin B$ ，

$$S(i) = \begin{cases} 1 - \dfrac{a(i)}{b(i)} & a(i) < b(i) \\ 0 & a(i) = b(i) \\ \dfrac{b(i)}{a(i)} - 1 & a(i) > b(i) \end{cases} \tag{3.7}$$

即 $S(i) = \dfrac{b(i) - a(i)}{\max\{a(i)\ ,\ b(i)\}}$ ， $-1 \le S(i) \le 1$

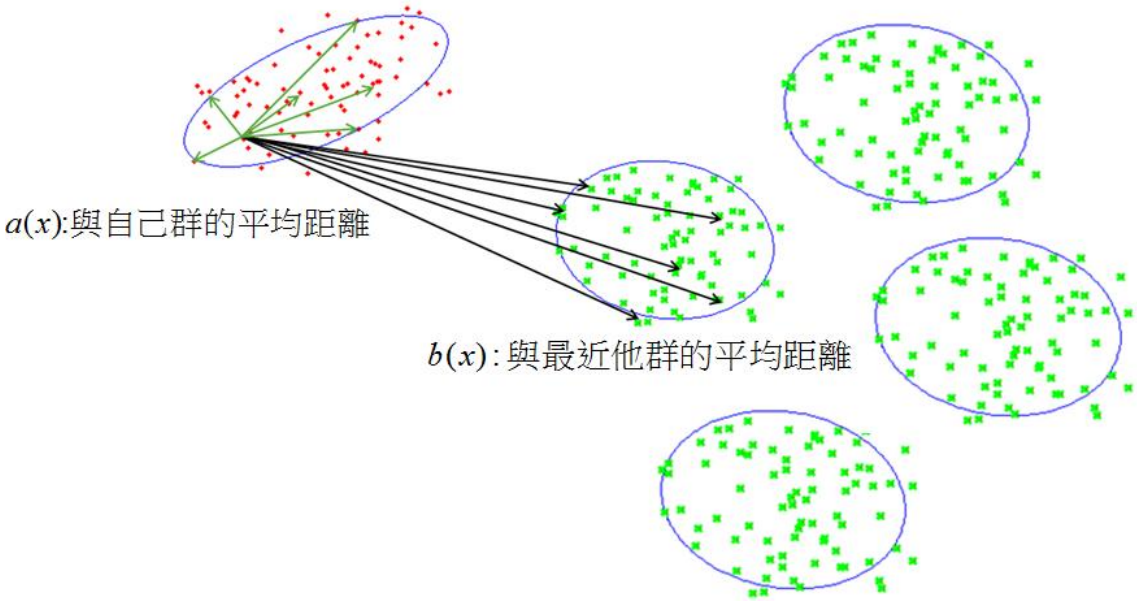

圖 3-2 側影值示意圖

最後分群結果的側影值計算為：先分別算出每一群所有點的側影值平均，有幾群便有幾個數值，再算出這些數值的平均，即

$$SC(k) = \frac{1}{k}\sum_{i=1}^{k}\frac{1}{|C_i|}\sum_{x \in C_i} S(x) \tag{3.8}$$

以下用範例來說明 AFD 演算法收斂的過程：先用高斯分布隨機產生兩包資料，分別為紅色(o)和藍色(+)，再假裝我們都不知道類別訊息來進行分群。

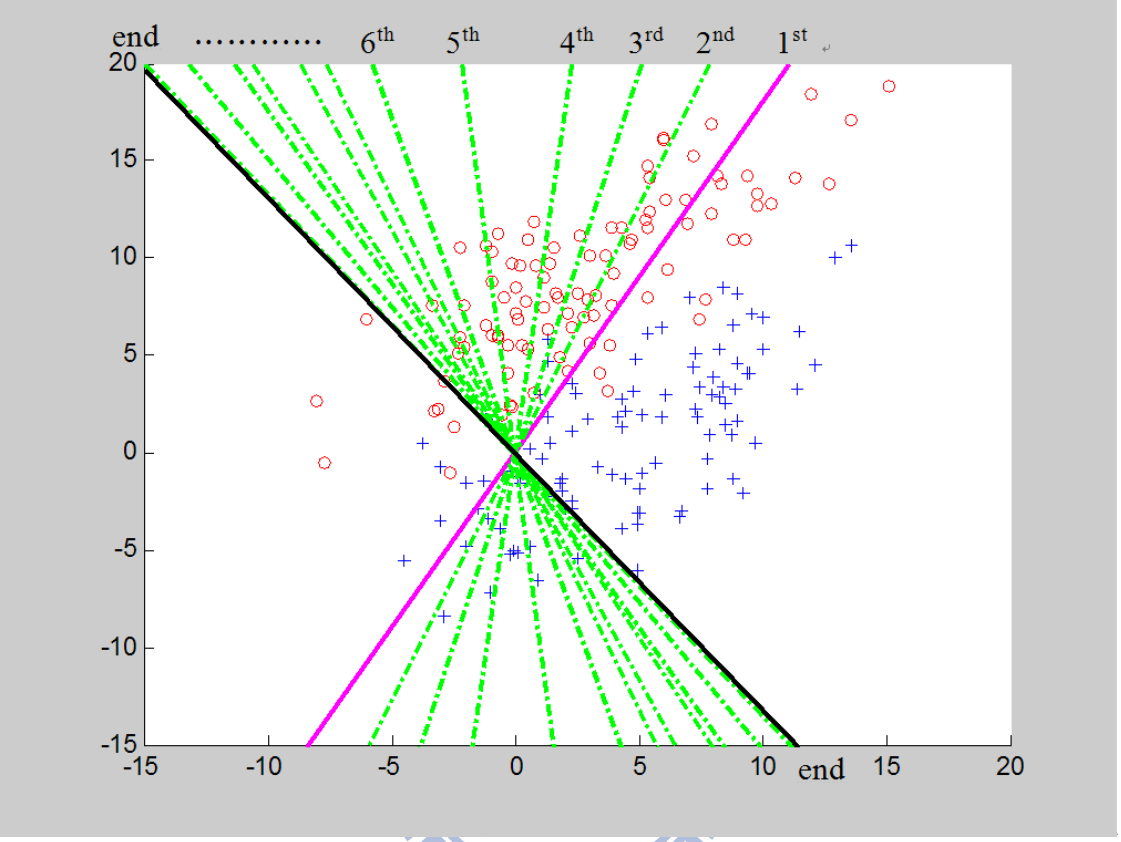

圖 3-3 AFD 向量收斂過程

第一次找出來的方向其實就是 PCA 的第一主軸，跟最後收斂出來的方向幾乎要垂直，代表在此例子中 AFD 跟 PCA 幾乎選擇了完全相反的特徵。由圖上原本的兩類別來看，AFD 所找出的方向的確比 PCA 更適合拿來分群。

會產生這樣正確的收斂過程，是因為每次找出方向後，資料投影上去作分群都會有更接近原本真實類別的分群結果，如此不斷的先分群再修正方向直到收斂。

現在就再以人造資料來觀察選出的軸其 FCI 值與分群結果之間的關係：

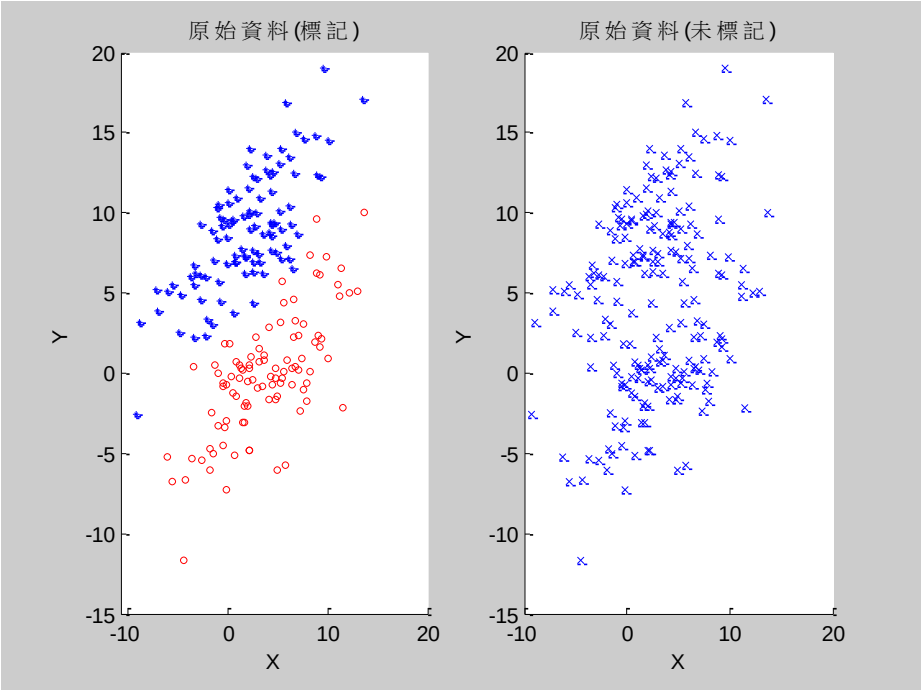

圖 3-4 兩類人造資料

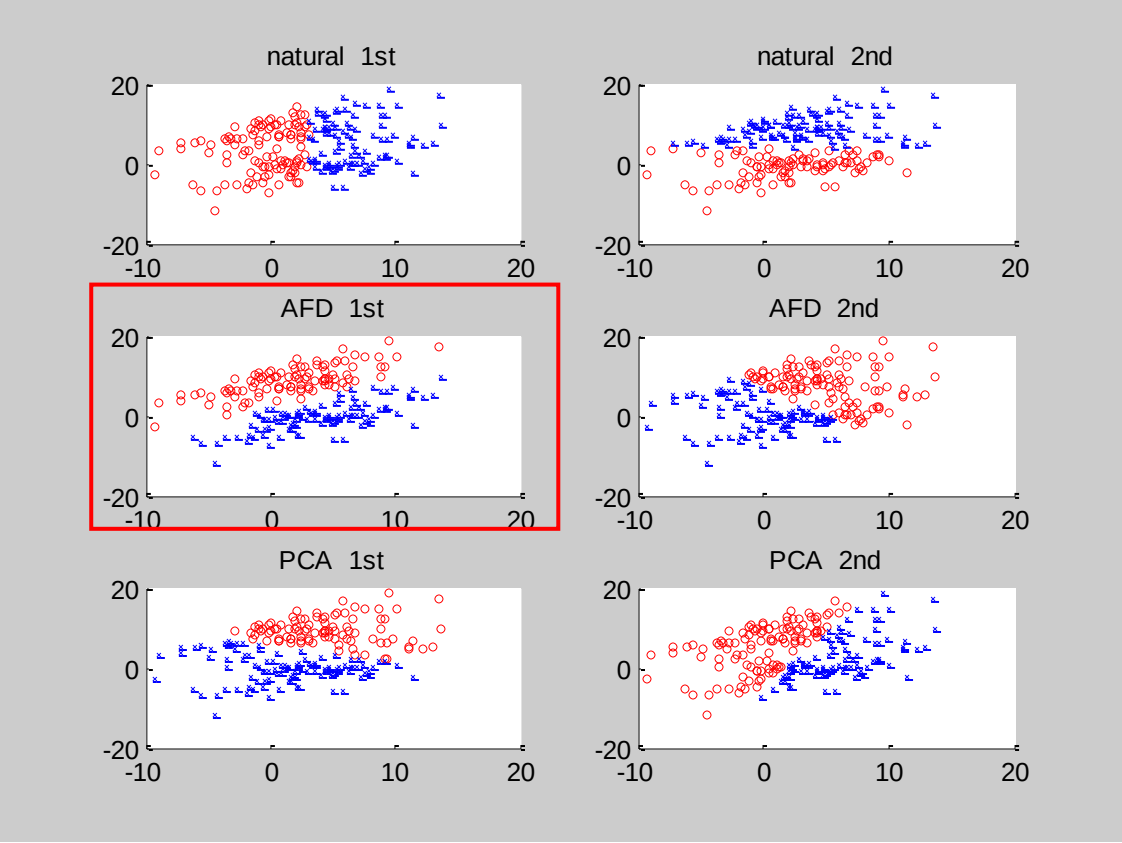

圖 3-5 各軸分群結果

FCI	1st	2nd
自然基底	1.791	2.635
AFD	4.176	1.775
PCA	2.143	1.842

表 3-1 各軸 FCI

由表可斷定 AFD 的第一個軸，以及自然基底的 Y 軸，有較好的分群結果，我們從圖上來看可以發現，確實在這兩個軸上的分群較能接近資料原本的分類情形。

第四章　　實驗結果

本論文使用著名的 Iris 和 Wine 資料來做實驗。Iris 最初是由安德森從加拿大加斯帕半島上的鳶尾屬花朵中提取的數據，後來由費雪作為判別分析的一個例子，運用到統計學中。Iris 是由四個變數，150 筆資料所形成的三種類別，每類各 50 筆，變數分別為花萼和花瓣的長度以及寬度；Wine 是由十三個變數，178 筆資料，所形成的三種類別葡萄酒，每類分別有 59、71，和 48 筆資料，其變數都是化學成分如酒精，蘋果酸，...等。這兩種資料都有已知的三種類別，所以可以拿他來比對分群的結果。

4.1　Iris 實驗結果

下圖為 Iris 資料的四個變數交互散佈圖矩陣。

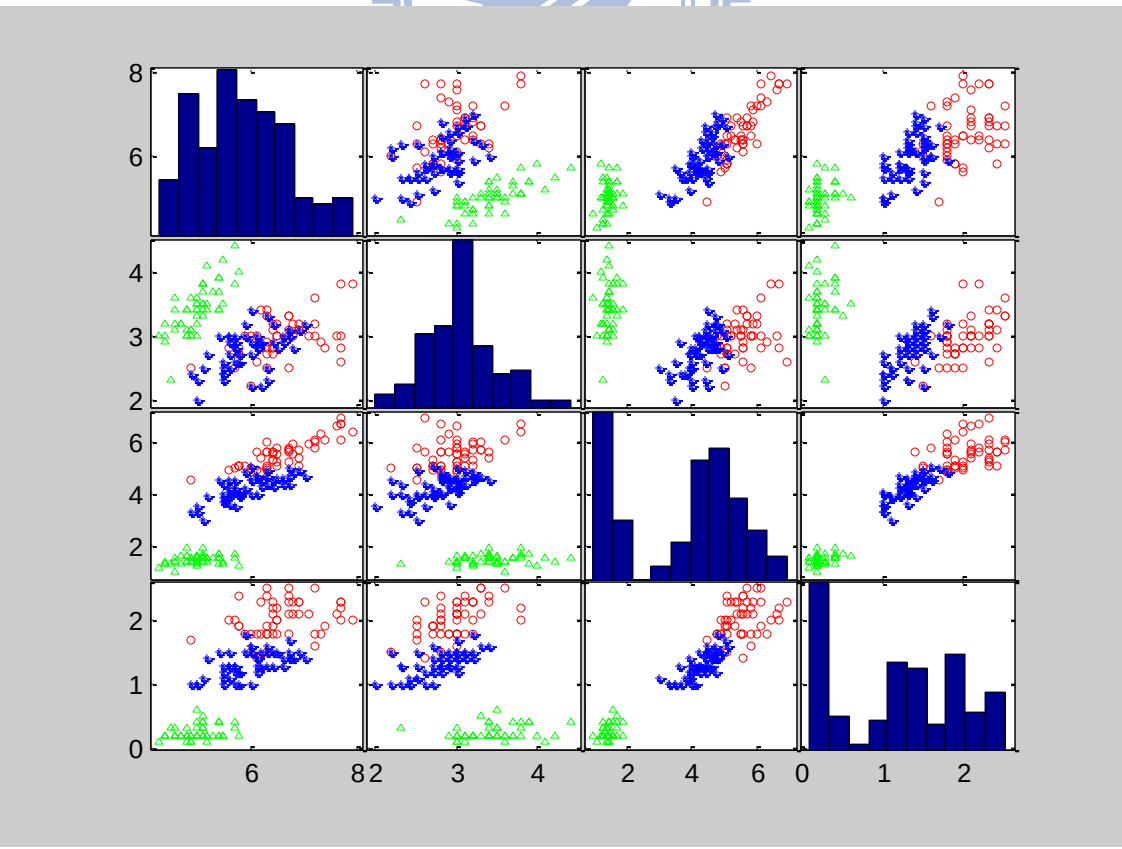

圖 4-1 Iris 資料散佈圖矩陣

首先，可以由變數與類別之間的互信息 (mutual information)了解彼此之間的相關性，互信息的含意為兩個變數之間能互相解釋的程度，其定義如下：

$$I(X;Y) = H(Y) - H(Y|X) = H(X) - H(X|Y) \tag{4.1}$$

其中

$$\begin{aligned} H(X) &= -\sum_{x} p(x) \log p(x,y) \\ H(X|Y) &= -\sum_{x}\sum_{y} p(x,y) \log p(x|y) \end{aligned} \tag{4.2}$$

$H(X)$ 為 X 的熵 (entropy)，代表了 X 的混亂程度，$H(X|Y)$ 就代表了當已知 Y 以後，X 剩下的混亂程度。在算出互信息之後，將其正規化，使其較具參考價值，即

$$\frac{I(X;Y)}{\min\left(H(X),H(Y)\right)} \tag{4.3}$$

由上式可以算出下表：

變數	X1	X2	X3	X4
正規化的互信息	0.475	0.278	0.852	0.940

表 4-1 各變數與類別變數的互信息

從表上可看出變數 $\mathbf{x}_3$ 和 $\mathbf{x}_4$ 比較能解釋三種類別，直接從圖 4-1 上也可看出這兩個變數對於分辨三種花，有明顯的區別。而 $\mathbf{x}_2$ 則完全看不出與三種類別之間的關聯性，對分群顯然是較無用的變數。單獨使用各變數來觀察分群的準確率。下圖為三類別之於各變數的直方圖

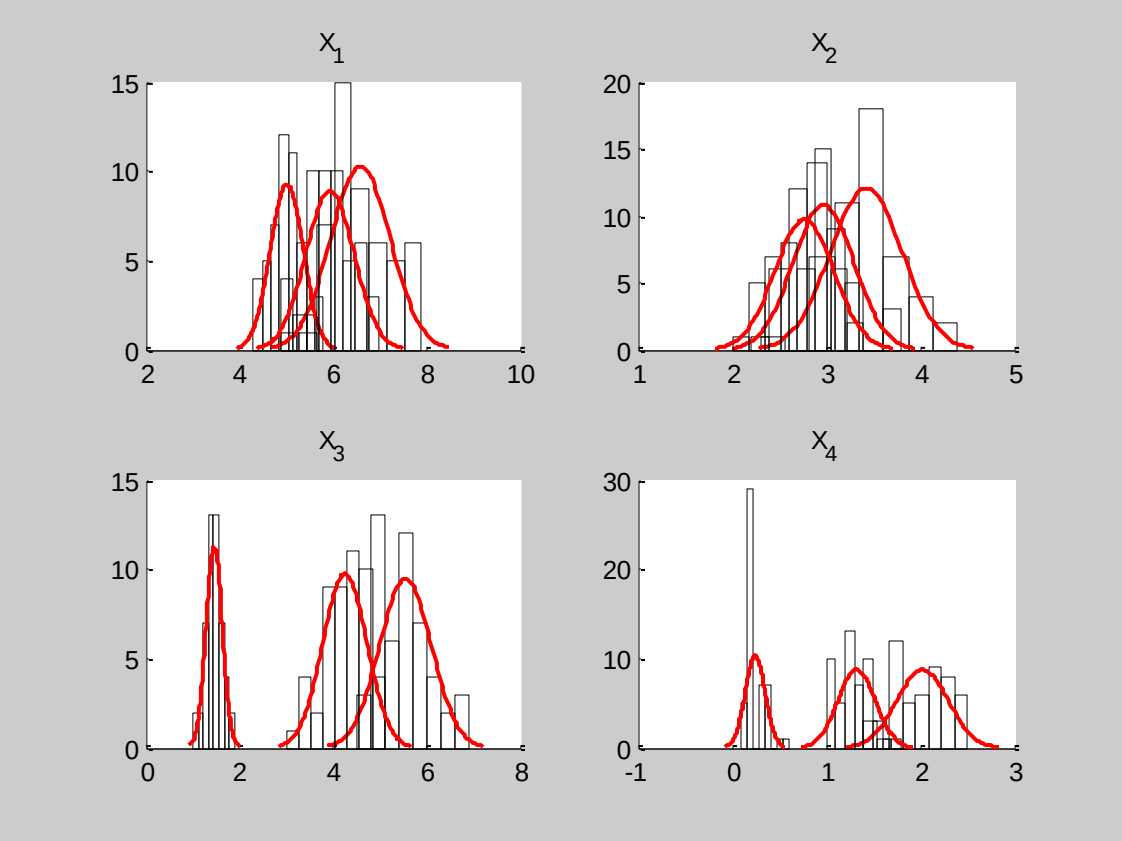

圖 4-2 各原始變數直方圖

	X1	X2	X3	X4
準確率(%)	72.00	51.33	89.33	96.00

表 4-2 原始變數的分群準確率

可以發現，互信息越大，其準確率也越高。這是理所當然的，因為互信息越大代表就是造成三類別差異的主要原因。

現在用 AFD 演算法嘗試找出近似的費雪鑑別向量，其向量收斂過程如下圖：每張圖代表每個向量的各個元素收斂的過程。

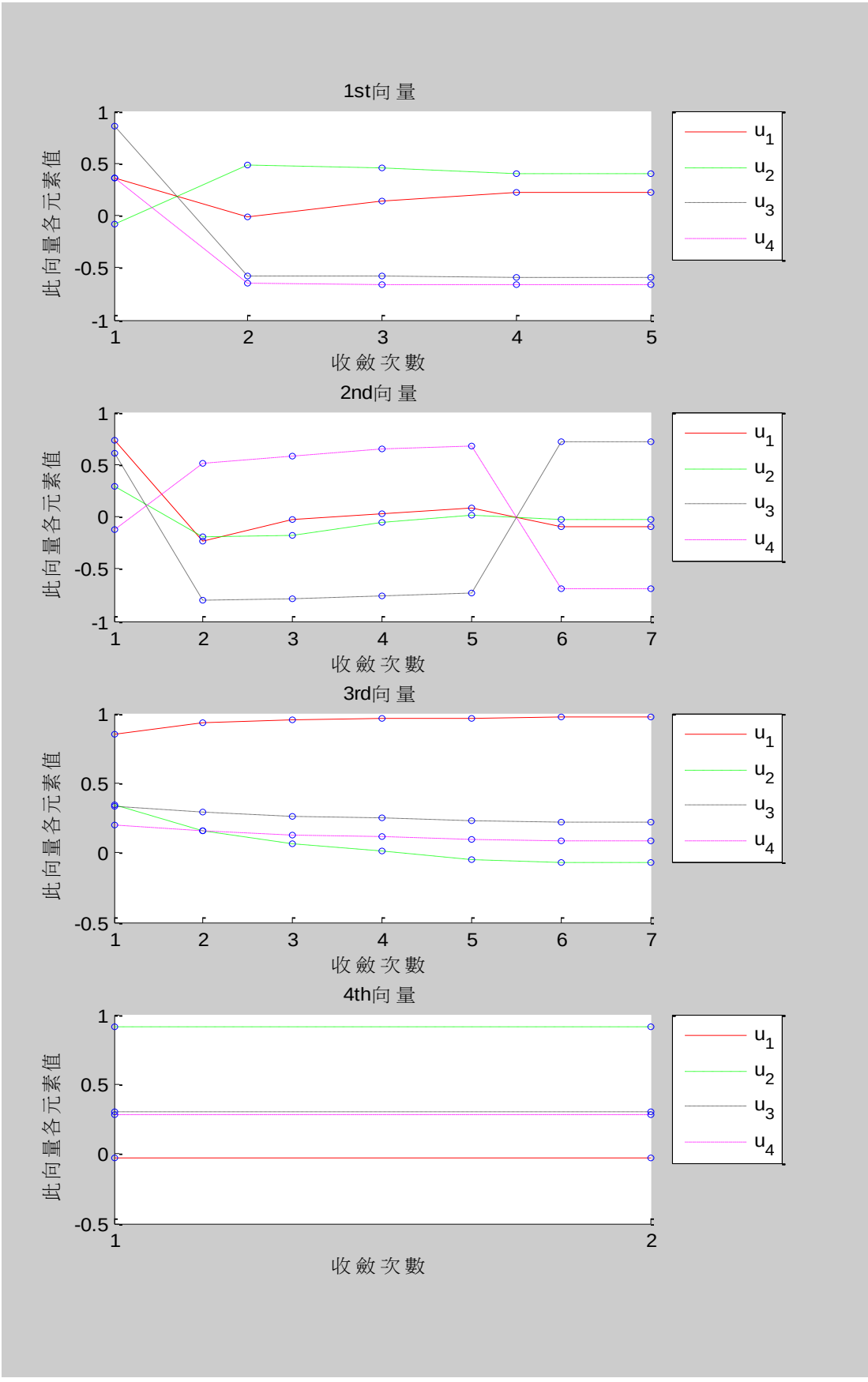
1st向量
此向量各元素值
收斂次數
u1
u2
u3
u4
2nd向量
此向量各元素值
收斂次數
u1
u2
u3
u4
3rd向量
此向量各元素值
收斂次數
u1
u2
u3
u4
4th向量
此向量各元素值
收斂次數
u1
u2
u3
u4

圖 4-3 向量收斂過程圖

可以看出四個向量幾乎在第二次收斂時便很接近最後結果，也證實使用主成份分析當作起始軸收斂會較快的事實。第二個鑑別向量在第六次收斂時發生變號，但其實不影響結果。第四個鑑別向量，理所當然一次就會被決定，因為它必須與前三個鑑別向量正交。找出的鑑別向量是否真的對分群有比較好的效果，來看看跟主成份分析的比較，以及各軸的 FCI 值。

AFD	1st	2nd	3rd	4th
	0.222	-0.103	0.969	-0.032
	0.407	-0.029	-0.067	0.911
	-0.589	0.716	0.221	0.302
	-0.662	-0.690	0.088	0.280
PCA	1st	2nd	3rd	4th
	0.361	0.657	0.582	-0.315
	-0.085	0.730	-0.598	0.320
	0.857	-0.173	-0.076	0.480
	0.358	-0.075	-0.546	-0.754

表 4-3 AFD 和 PCA 各軸向量

FCI	1st	2nd	3rd	4th
AFD	33.804	15.434	7.591	5.788
PCA	15.628	3.666	4.306	4.212

表 4-4 各軸 FCI 值

準確率(%)	1st	2nd	3rd	4th
AFD	98.67	80.67	80.67	71.33
PCA	91.33	42.67	52.00	43.33

表 4-5 各軸分群準確率

AFD 各軸鑑別能力皆比 PCA 高出許多。比對可以發現，較大的 FCI 值對應著較高準確率，再次證明根據 FCI 值來斷定軸的優劣是可行的。接著，我們畫散佈圖觀察群聚分佈的情形：

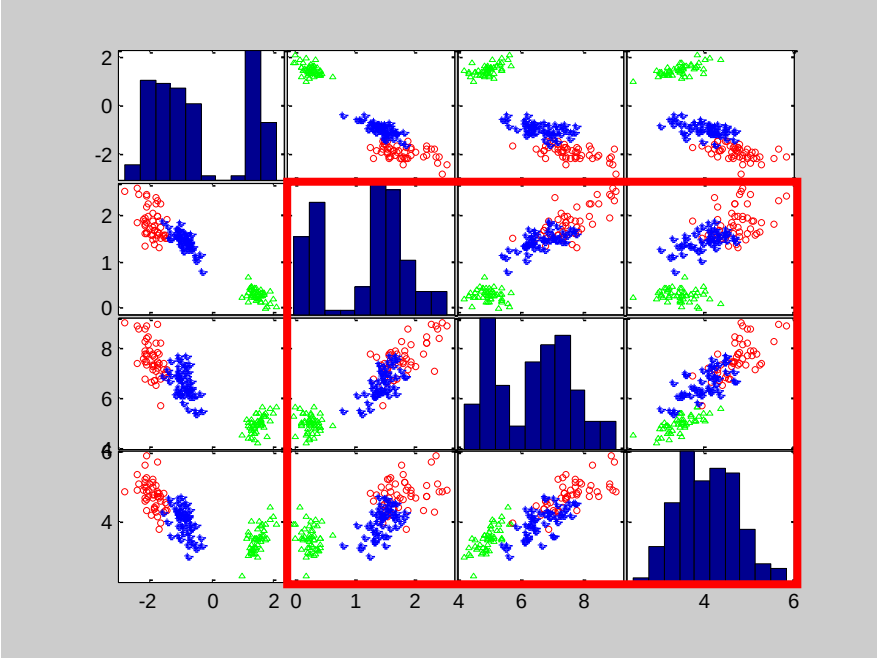

圖 4-4 Iris 資料 AFD 散佈圖矩陣

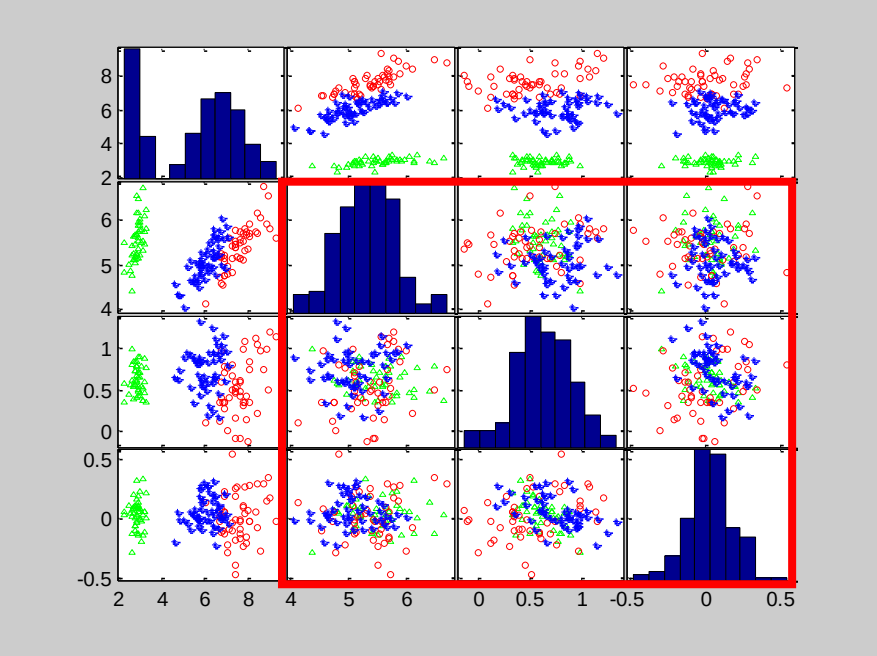

圖 4-5 Iris 資料 PCA 散佈圖矩陣

從上面兩張圖可以看到在第一軸上 AFD 的群聚都比 PCA 來的緊密，並且在第二軸以後(框起來的部分)，PCA 完全分不出群聚，AFD 還能清楚的分出來。主成份分析在各軸上的分群準確率都不及我們所找出的鑑別向量，可由因素負荷(factor loading)來解釋。因素負荷定義為兩變數之間的相關性，即 $f_{XY} = corr(X,Y)$。因素負荷的絕對值大小代表了因素解釋變數的能力。

PCA	factor1	factor2	factor3	factor4
X1	0.897	0.391	0.197	-0.059
X2	-0.399	0.825	-0.384	0.113
X3	0.998	-0.048	-0.012	0.042
X4	0.967	-0.049	-0.200	-0.153

表 4-6 PCA 因素負荷矩陣

AFD	factor1	factor2	factor3	factor4
X1	-0.790	0.835	0.980	0.861
X2	0.537	-0.499	-0.257	0.142
X3	-0.985	0.978	0.952	0.830
X4	-0.970	0.889	0.905	0.851

表 4-7 AFD 因素負荷矩陣

由於主成份分析在第二主成份以後，解釋變數 $\mathbf{x}_3$ 和 $\mathbf{x}_4$ 的能力很少，然而由表 4-1 知道變數 $\mathbf{x}_3$ 和 $\mathbf{x}_4$ 對分群才有明顯的幫助，因此主成份分析在這些軸上的分群效果並不理想。然而 AFD 在四個軸上對 $\mathbf{x}_3$ 和 $\mathbf{x}_4$ 都有很強的解釋能力，也因此每一個軸都有不錯的分群準確率。觀察到在四個鑑別向量上，$\mathbf{x}_2$ 的因素負荷都是最小，然而 PCA 第二軸對 $\mathbf{x}_2$ 的因素負荷很大，準確率又非常的低，因此可以推測變數 $\mathbf{x}_2$ 不

但對分群沒有幫助，甚至會讓結果變得更糟。

接著看收斂過程是否如預期般慢慢將群與群之間分隔開來，以 AFD 第一鑑別向量為例。由於主成份分析在第一個軸上的鑑別能力不差，因此我們改由隨機產生的方向當做起始方向，也可看看收斂的穩定性如何。

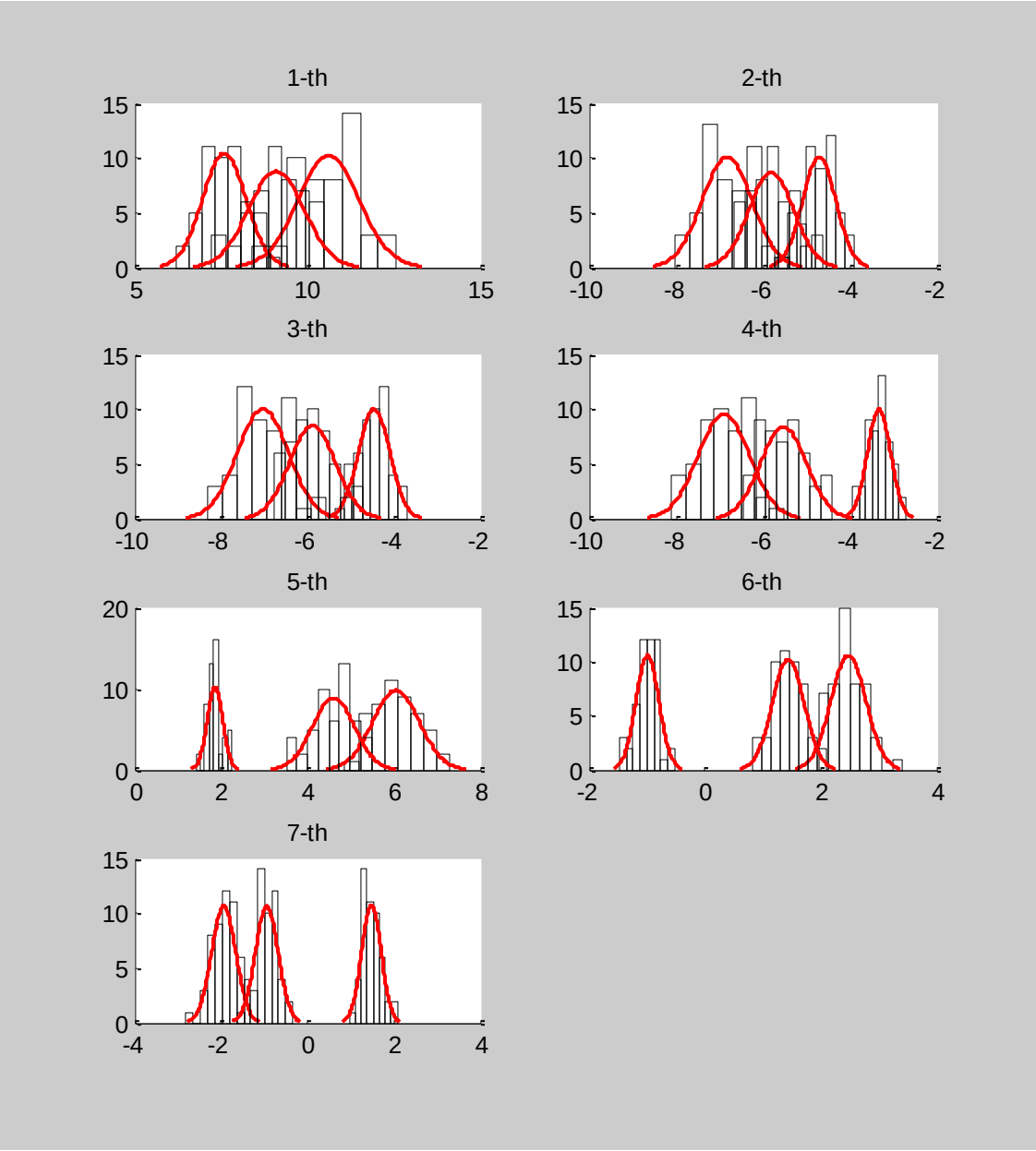

圖 4-6 Iris AFD 第一鑑別向量收斂過程

由圖可以看出在收斂過程，群聚漸漸明顯。不但群與群之間越分越開，群內散佈也越來越緊密。如下表：

with-class variance	class1	class2	class3	SUM	Between-class
1-th	0.670	0.992	0.866	**2.529**	**2.075**
2-th	0.437	0.677	0.593	**1.707**	**2.765**
3-th	0.393	0.662	0.577	**1.631**	**3.811**
4-th	0.285	0.675	0.594	**1.554**	**9.704**
5-th	0.181	0.547	0.487	**1.215**	**16.204**
6-th	0.198	0.303	0.275	**0.775**	**31.386**
7-th	0.224	0.278	0.264	**0.765**	**32.104**

表 4-8 組內變異和組間變異

下圖為 Iris 資料在 AFD 四軸上的直方圖

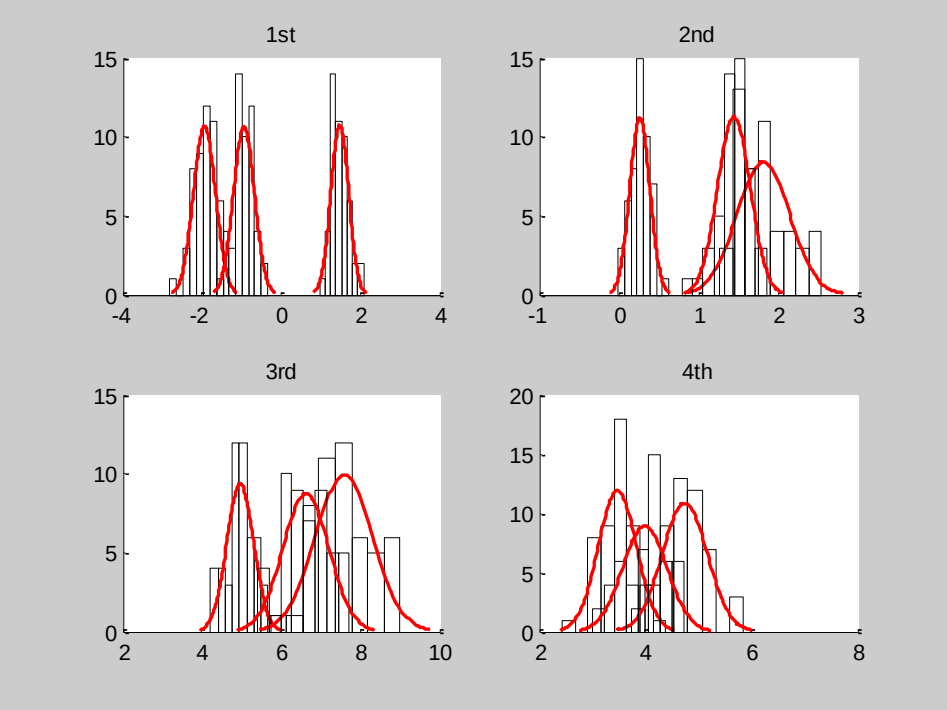

圖 4-7 三類別在 AFD 四軸上的直方圖

AFD 第一向量	FLD 第一向量	PCA 第一向量
0.222	0.209	0.361
0.407	0.386	-0.085
-0.589	-0.554	0.857
-0.662	-0.707	0.358

表 4-9

AFD 演算法所求出的第一鑑別向量與真實 FLD 的第一鑑別向量，其夾角為 $\cos^{-1}(\frac{\mathbf{a}\cdot\mathbf{b}}{|\mathbf{a}||\mathbf{b}|}) \approx 3.6°$，而 PCA 第一向量與 FLD 第一向量夾角為 46.8°。雖然在高維空間已經看不出向量夾角的幾何意義，但投影到兩個夾角很小向量，其座標可以確定是很接近的。

4.2 Wine 實驗結果

Wine 資料有 13 維，取前四個 AFD 和 PCA 向量來畫散佈圖矩陣

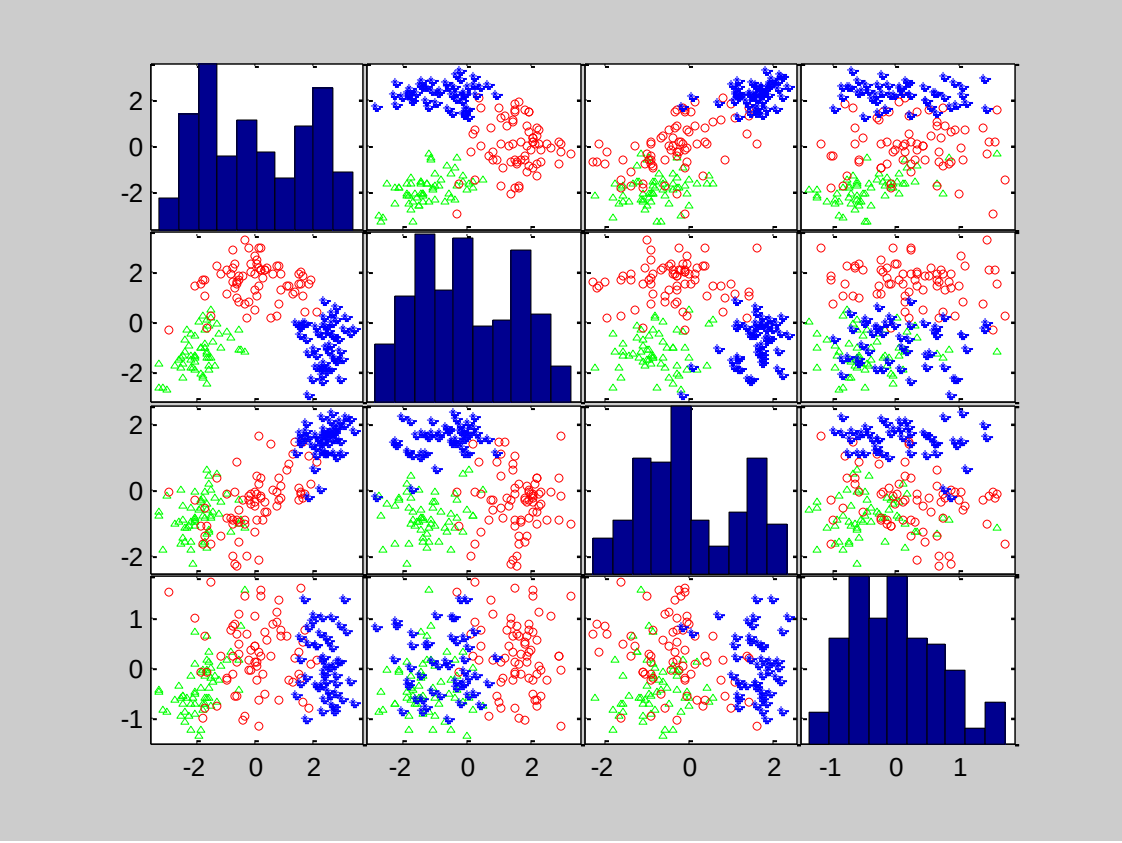

圖 4-8 Wine 資料 AFD 散佈圖矩陣

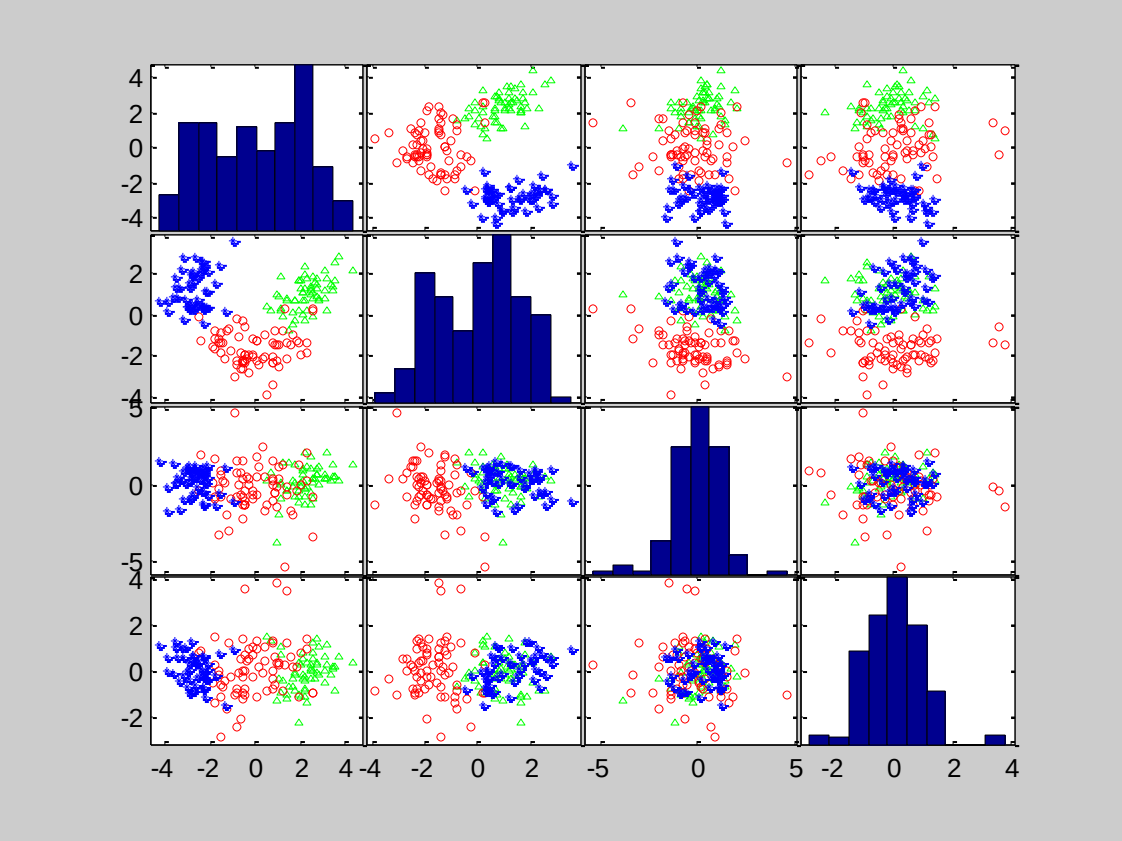

圖 4-9 Wine 資料 PCA 散佈圖矩陣

上面兩圖搭配各軸比較如下表：

準確率(%)	1st	2nd	3rd	4th
AFD	83.10	70.80	66.90	46.60
PCA	80.30	66.90	39.30	37.10
FCI	1st	2nd	3rd	4th
AFD	10.927	8.85	8.875	6.082
PCA	8.774	7.228	2.722	2.821

表 4-10 Wine 各軸的 FCI 值和準確率

AFD 找出來的各軸準確率都比對應的 PCA 各軸來的好。可以發現到 AFD 第三鑑別向量的 FCI 值雖大，卻沒有相應的準確率。先來看看鑑別向量大小(註：鑑別向量大小便是指鑑別向量找出的先後順序)與 FCI 和準確率之間的關係：

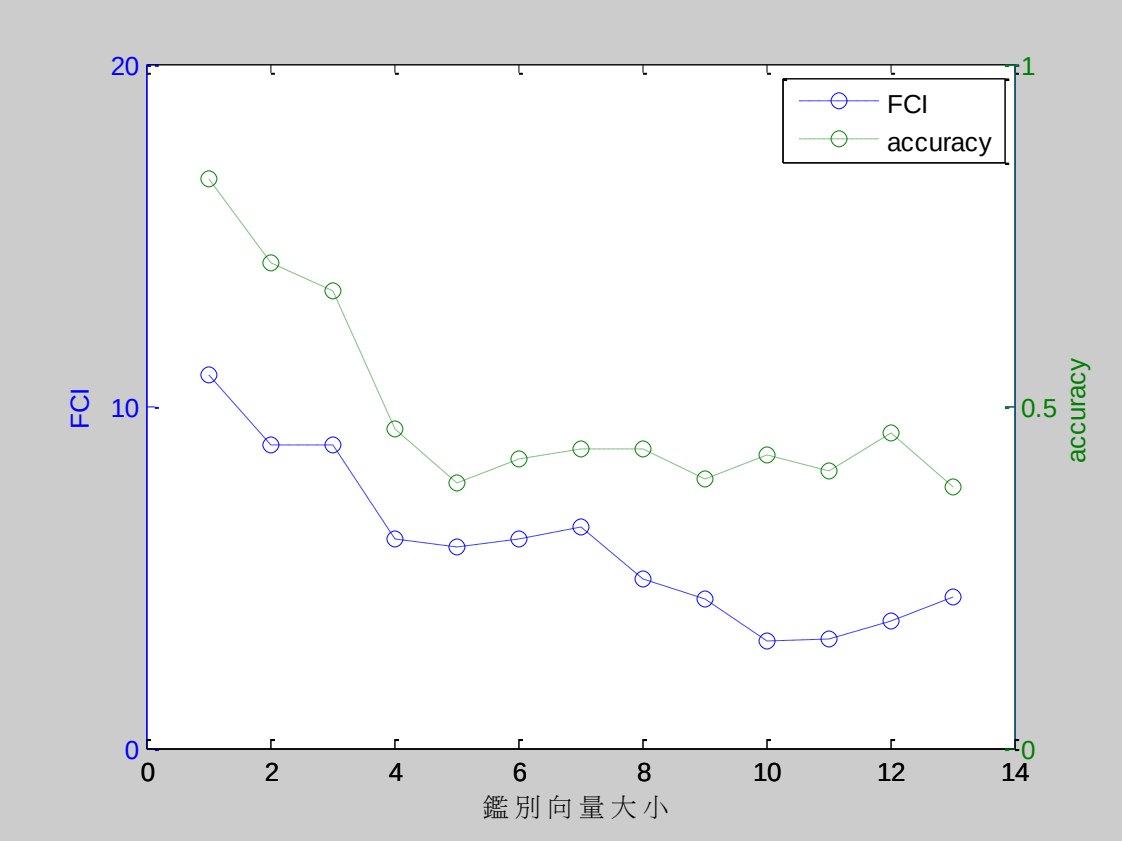

圖 4-10 Wine 資料各鑑別向量的 FCI 值與準確率

我們發現越大(越前面找出的)的鑑別向量，並不一定有較大的 FCI 值，也不一定有較高的準確率。顯然，指標並不是完美的，然其大致上的趨勢還是足以採信。

4.3 軸數與分群數

上述實驗都是使用準確率做最後判斷的依據，那麼在未標記資料上該如何斷定分群的優劣呢？這有幾個必須討論的方向：1. 資料該分成幾群 2. 要用多少個軸來分群 3. 分群數與軸數都決定了以後，該怎麼挑選軸。

先看第三個問題，決定了軸數以後我們便選擇前幾大的鑑別向量，在之前的實驗已經看過 FCI 值，越大(越先找出)的鑑別向量通常有越大的 FCI 值，越大的 FCI 值通常有越高的準確率，Iris 的實驗結果也證實具有可信度，下圖為 FCI 值與準確率的關係圖，並且輔以側影值當做對照。隨機抓 1000 個軸與 AFD 第一鑑別

向量來比較。

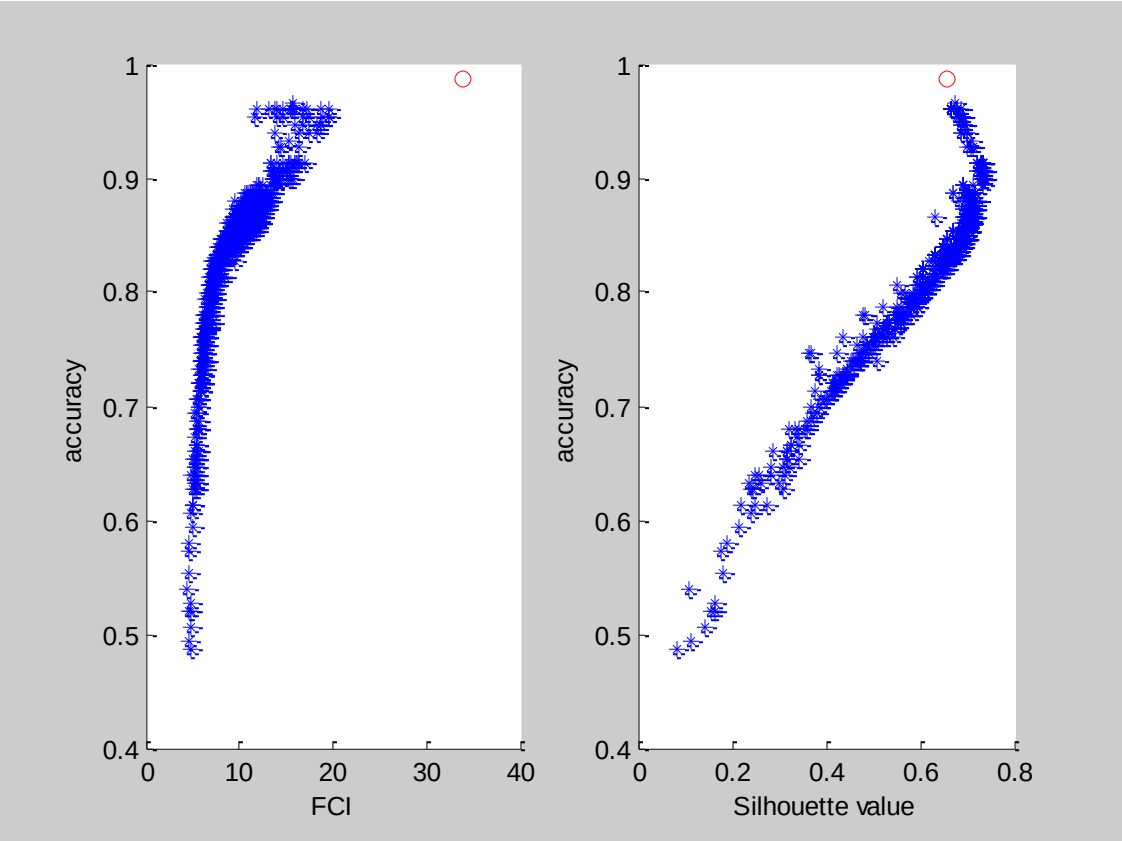

圖 4-11 Iris 資料 FCI 值和側影值 VS 準確率

隨機挑選的軸在側影值的大小上可以逼近甚至超過第一鑑別向量，然而在 FCI 值的大小上卻很難逼近，這代表如果 AFD 演算法找出來的軸其 FCI 值較其他軸來的大許多，那麼就是找到了最佳分群結果的鑑別向量。可以發現，Iris 資料的 FCI 值或側影值對於預測準確率是相當不錯的，然而我們在 Wine 的實驗發現越大的 FCI 值，卻不見得有越高的準確率，這是因為在 Wine 資料上，指標的趨勢不像 Iris 資料那麼明顯。隨機產生 200 個軸的指標對準確率作圖

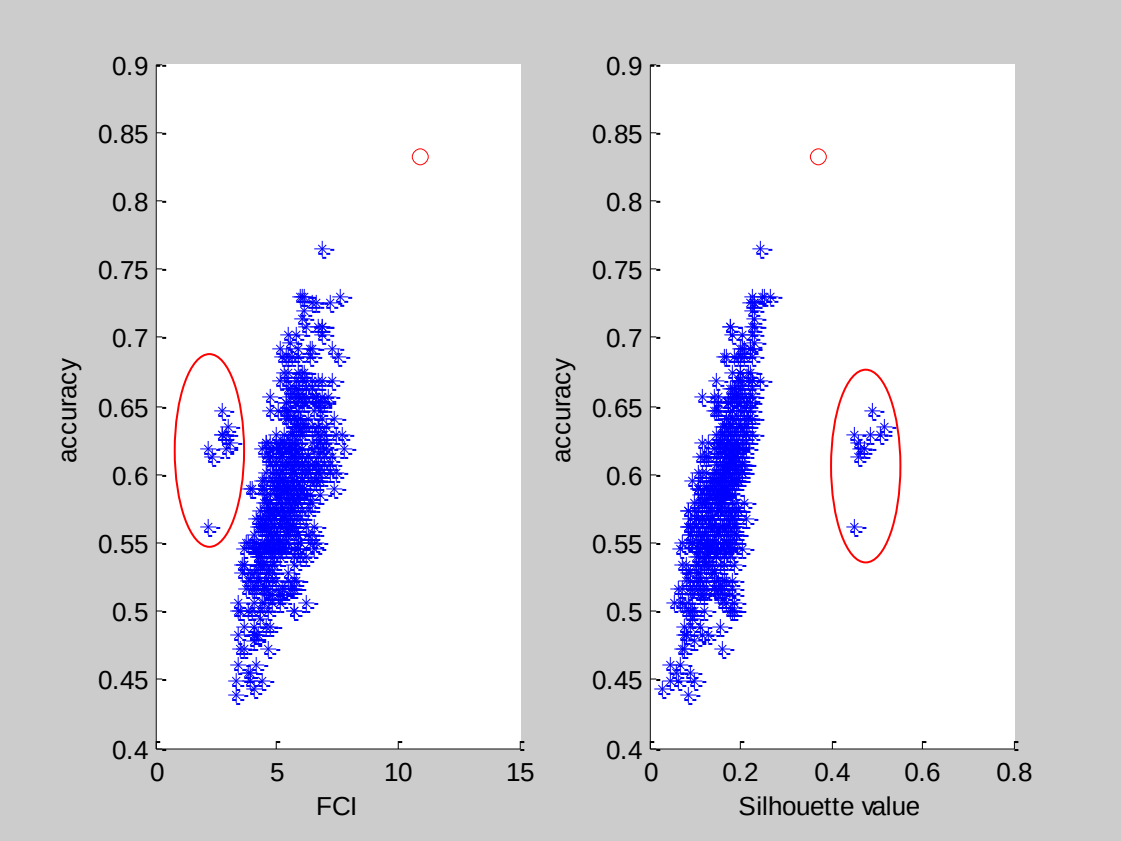

圖 4-12 Wine 資料 FCI 值和側影值 VS 準確率

如上圖，指標的大小無法精確的反應出 Wine 資料分群的準確率，猜測這是由於 Wine 資料原始的分類模型較為複雜，不易在線性轉換的空間用分群去近似。可以看到圈起來的地方有不尋常的情形，這是由於 K-means 分群產生極不平衡的群集大小所造成。

第二個問題，要使用幾個軸分群呢？來看看使用軸數與準確率的關係

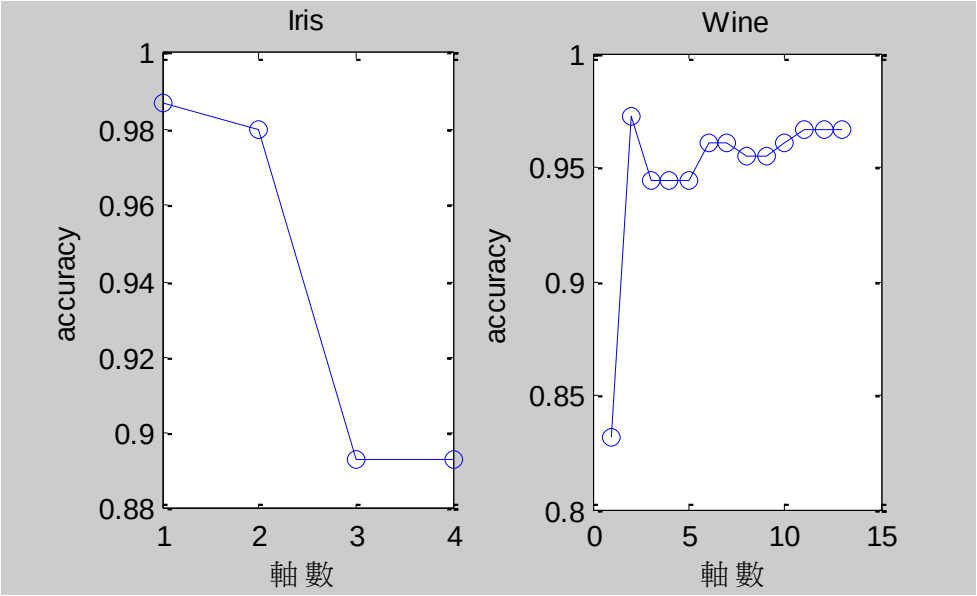

圖 4-13 AFD 軸數 VS 準確率

圖上顯示越多的軸不見得有越高的準確率，因此選擇適當的軸數便顯得重要，我們從單一各軸的 FCI 值觀察：

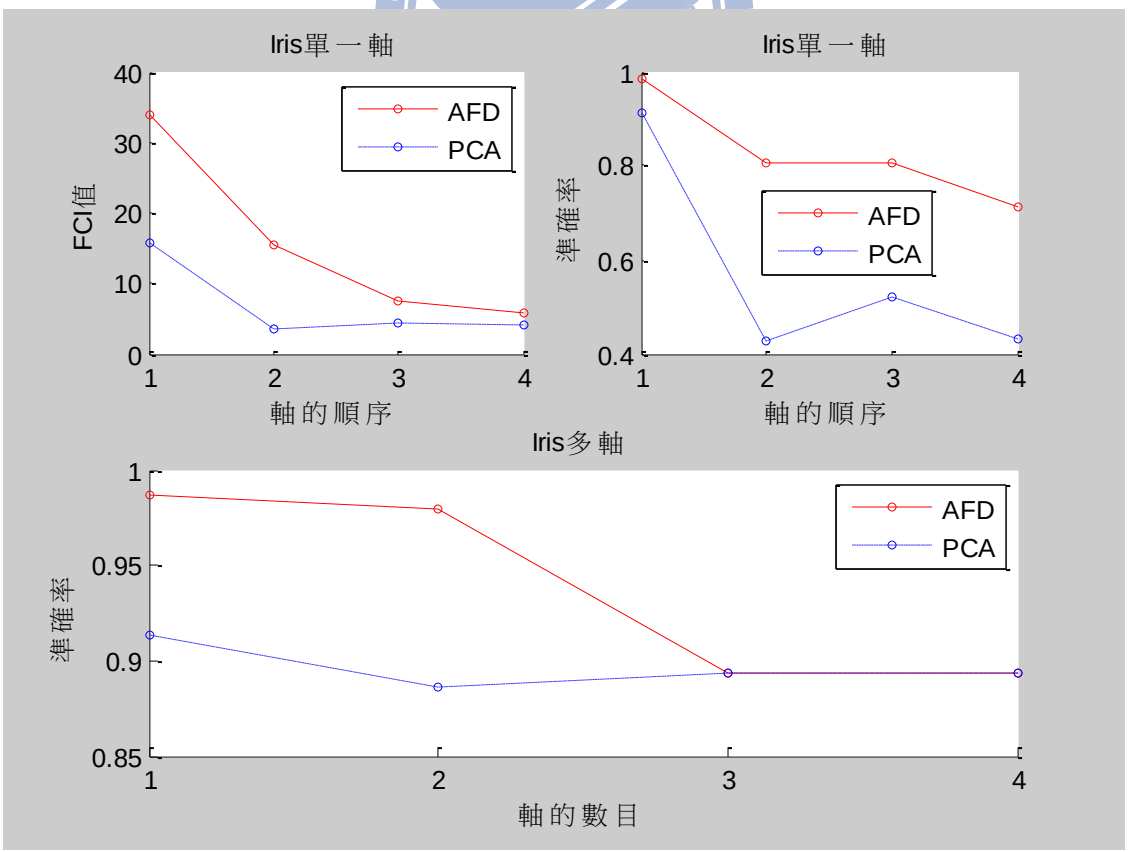

圖 4-14 Iris 單一軸與多軸的準確率

從左上的圖可以看到不論是 AFD 或是 PCA 在第二軸以後，FCI 值有很大的落差，因此可以研判，在 Irsi 資料中，第二軸以後便已經對分群無用了，從右上的圖可以證實第二軸以後的單一軸分群準確率的確大幅下降，因此 Irsi 資料就使用一個軸分群即可。最下邊的圖顯示只使用一軸的準確率的確是最高的。

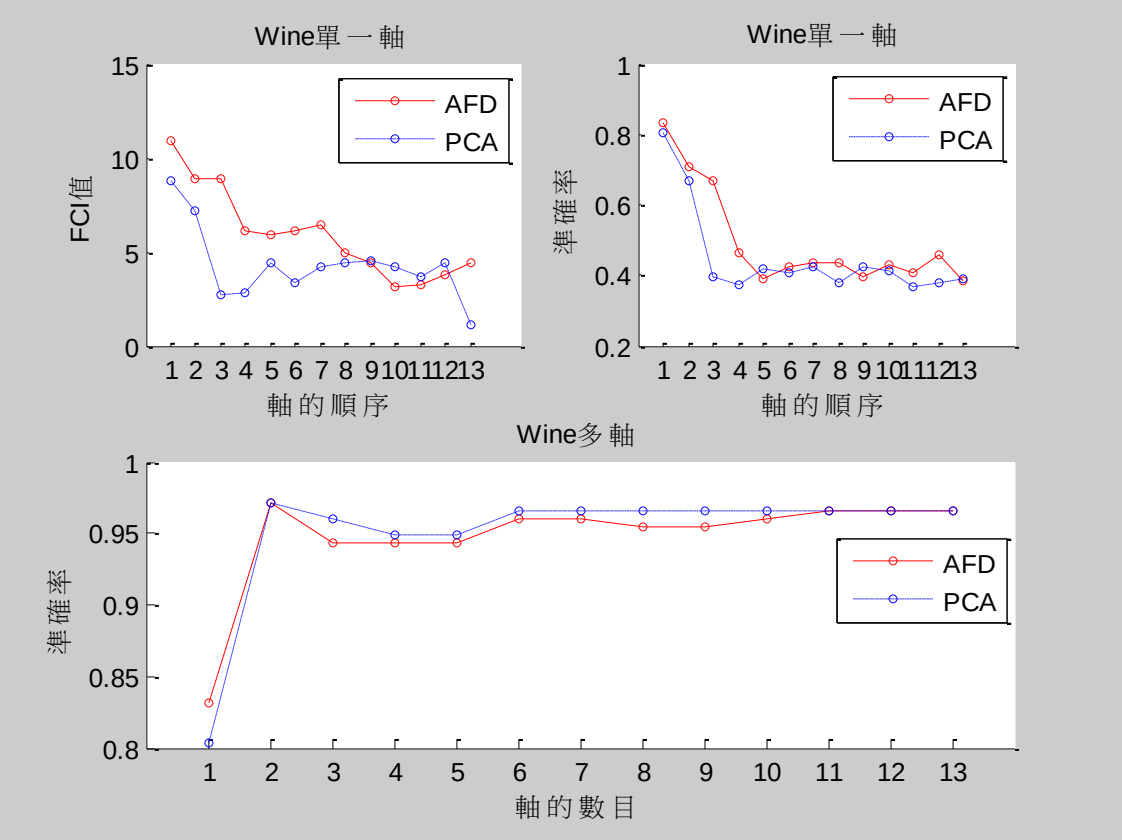

圖 4-15 Wine 單一軸與多軸的準確率

在 Wine 資料中，可以看到 PCA 的 FCI 值在第二軸和第三軸之間有極大落差，代表對 PCA 來說，取前兩軸就夠了，而 AFD 的特徵提取能力較 PCA 好，AFD 理所當然不會比 PCA 還需要更多的軸，因此 PCA 取兩軸 AFD 也是取兩軸就夠了。

最後，分群的數目就使用側影值來決定，透過式(3.8)，可以算出每次分群的側影值，取發生最大側影值時的分群數，$\arg\max_k SC(k)$

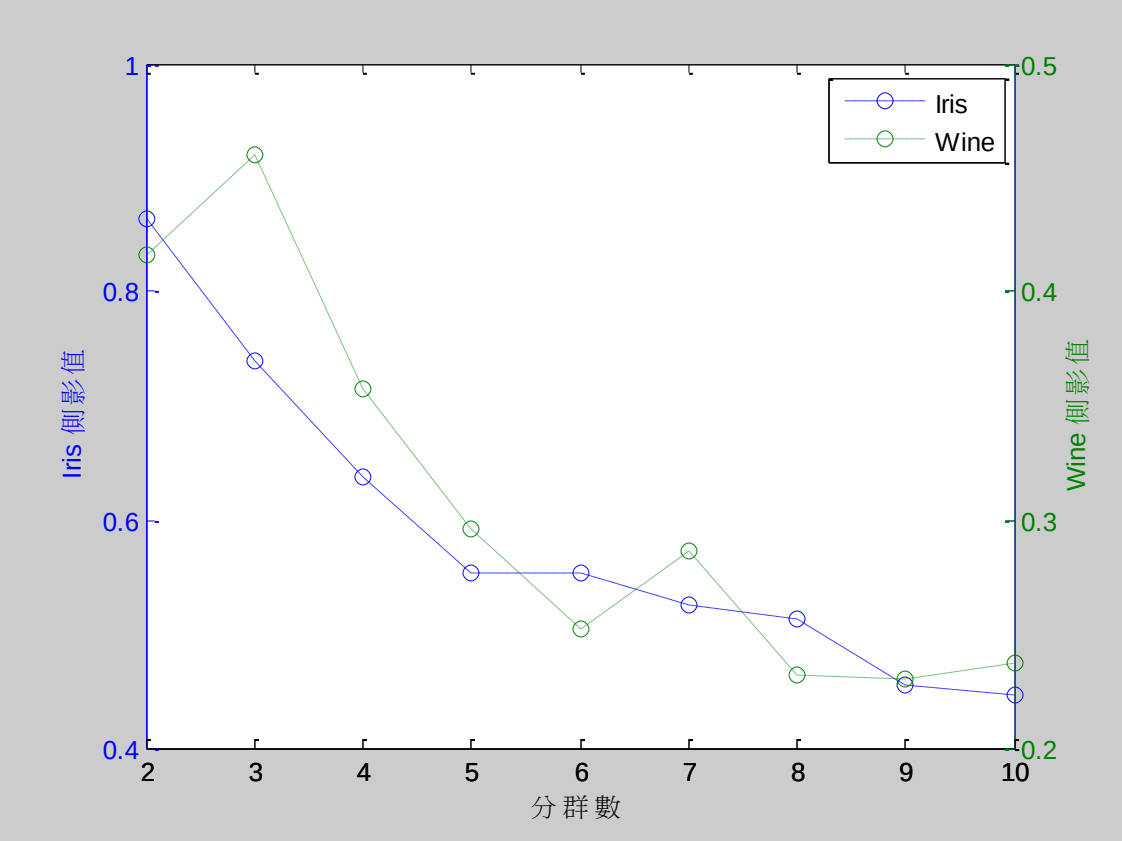

圖 4-16 分群數 VS 側影值

Wine 資料透過側影值的判斷與真實類別數目一致，然而 Iris 側影值判斷為兩群，事實上根據下圖，誤判為兩群確實有其道理。

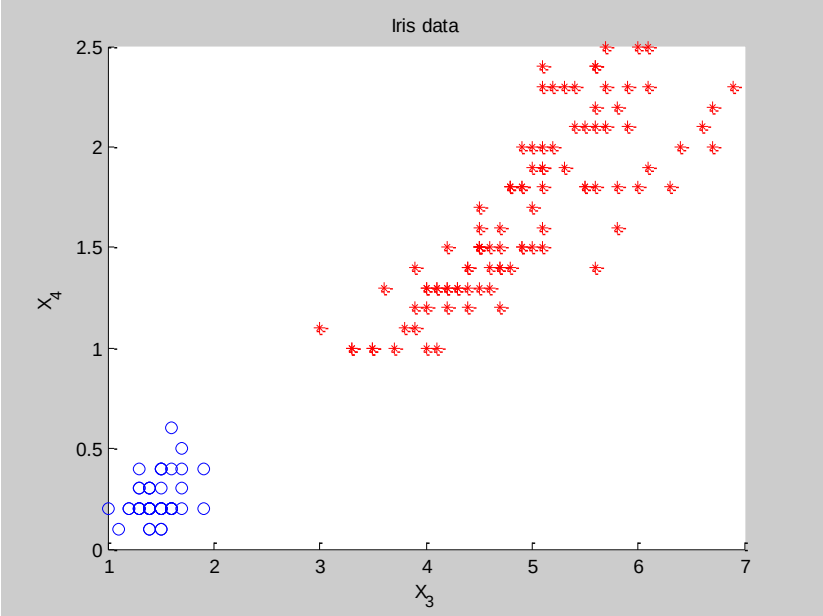

圖 4-17 Iris 分兩群

4.4 結果比較

根據上一節決定了軸的數目之後，Iris 資料使用一個軸，Wine 資料使用兩個軸。最後用第二章所提到的分群演算法及特徵提取法，進行實驗比較：

準確率(%)	Kmeans	Kmedois	fuzzy C-means	Ward
自然基底	89.33	92.67	89.33	89.33
AFD	98.67	98.67	98.67	96.67
PCA	91.33	91.33	91.33	90.00
FCI	Kmeans	Kmedois	fuzzy C-means	Ward
自然基底	7.641	7.191	7.622	7.593
AFD	33.804	33.804	33.804	33.245
PCA	15.628	15.628	15.628	15.394

表 4-11 Iris 資料總結果

準確率(%)	Kmeans	Kmedois	fuzzy C-means	Ward
自然基底	96.63	89.33	96.63	92.70
AFD	97.19	97.19	96.63	95.51
PCA	97.19	94.94	97.19	96.63
FCI	Kmeans	Kmedois	fuzzy C-means	Ward
自然基底	0.811	0.762	0.811	0.773
AFD	3.903	3.903	3.895	3.793
PCA	3.935	3.928	3.937	3.898

表 4-12 Wine 資料總結果

綜合比較 Iris 和 Wine 的實驗結果可以看出用 AFD 提取特徵於各種分群演算法大

致上比 PCA 來的好，分群效果皆能有效的提升，並且 FCI 值在判斷分群準確率也有一定的可信度。雖然在 Wine 資料我們由各種實驗圖可以看出，不論是 FCI 指標還是側影值指標，都無法非常有效的反應準確率，對此也只能猜測 Wine 資料的分散情形不是那麼的理想。

透過 AFD，也較能分辨出該分成三群：

側影值(SC)	分兩群	分三群	差距	百分比(%)
自然基底	0.865	0.739	0.126	14.57
AFD 第一軸	0.951	0.887	0.064	6.73

表 4-13 側影值改善情形

可以看出，側影值的差距縮小，代表在 AFD 第一軸上，三群群聚的情形比自然基底中來的明顯，對於分成三群變得更有說服力了。

第五章　　結論

主成份分析是一個很好的維度化簡方法，在同樣的維度上，能保有全部資料最多的訊息，在迴歸應用上是個很好的特徵提取方法，然而對分群而言，資料全部的訊息不見得是有幫助的，往往有許多對分群無用的變數，真正有幫助的是每一群之間差異最大的訊息，因此本論文所提出的是一個訓練特徵的想法，我們應用費雪線性鑑別分析提取並訓練特徵，透過不斷修正向量的方向，漸漸的給予原始空間中重要變數較高的權重係數，這意味著不同方向的距離不再是一視同仁，改變了距離量測的方式。實驗結果發現 AFD 演算法確實對分群準確率有幫助，能找出對分群效果最佳的特徵。

透過特徵提取保留有用的少數特徵，也能減少分群系統的負荷，由於進行分群最為耗時的步驟就是在於計算相似度，若維度很大，計算量將會很重，維度簡化後的資料，將能減少大量的相似度計算時間。

本論文訴求的重點以及貢獻便在於針對分群給出有別於 PCA 的另一組正交基底，並且能用最少的特徵保留大部分對分群有用的訊息，而且此正交基底各軸分群的能力都較 PCA 來的好。用 AFD 找出基底後，並非用越多的軸就會有越好的分群結果，這是因為少數的軸往往已經包含有用的全部訊息，剩下的軸含有過多的雜訊，使用越多，越會干擾分群的結果。實際應用在 Wine 資料實驗中，我們發現 FCI 指標無法很正確的反應出準確率，研判造成 Wine 資料發生不如預期情況的原因如下：Wine 資料散佈情形不如 Iris 來的理想，或者說群距的邊界不是理想的直線，這個原因導致了指標在這包資料上面的可信度下降。針對這點，我們期望在未來改善分群的目標便在於：嘗試找出非線性的邊界。關於這個問題，近年來

已有非常熱門的核化法 (kernel method)，可以將資料轉換到另一個高維空間中，而在此空間中，群聚邊界是線性的，至於要怎麼應用改善，便是未來的目標。若未來能實現，相信可以對分群問題有莫大的改善。

參考文獻

[1] J. B. MacQueen, "Some Methods for classification and Analysis of Multivariate Observations," *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*, Berkeley, University of California, pp. 1:281-297, 1967.

[2] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: an Introduction to Cluster Analysis*: John Wiley & Sons, 1990.

[3] J. C. Dunn, "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters," *Journal of Cybernetics*, pp. 3:32-57, 1973.

[4] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algoritms*. New York: Plenum Press, 1981.

[5] A. R. Barakbah and Y. Kiyoki, "A pillar algorithm for K-means optimization by distance maximization for initial centroid designation," presented at the IEEE, 2009.

[6] J. Duchene and S. Leclercq, "An Optimal Transformation for Discriminant and Principal Component Analysis," *transactions on pattern analysis and machine intelligence*, vol. 10, pp. 978-983, 1988.

[7] J. Han and M. Kamber, *Data mining*, first ed.: Morgan Kaufmann, 2003.

[8] 張智星. *資料群聚與樣式辨認*.

[9] T. Hastie, *et al.*, *The elements of statistical learning: data mining, inference, and prediction*: Springer, 2001.