



支援向量機於信用評等之應用

*黃承龍 **陳穆臻 ***王界人

*高雄第一科技大學 資訊管理系

e-mail: cl.huang@msa.hinet.net

**國立台北科技大學 經營管理系

e-mail: bmcchen@ntut.edu.tw

***華梵大學 資訊管理系

e-mail: m9225002@cat.hfu.edu.tw

摘 要

由於信用卡被廣泛的使用，各銀行累積相當多信用卡有關的資料，所以如何利用這些資料來對未來的發卡用戶進行信用判斷就相當重要。支援向量機(Support Vector Machine)分類法，近幾年來被廣泛地運用在解決各領域的分類問題。本研究嘗試以支援向量機來解決信用卡分類問題，然而，使用不同的核心函數來建立支援向量機分類系統，會有不同的參數需要設定，而使用不同的參數將會影響到支援向量機分類的正確率。本研究使用格子點演算法來調整支援向量機的參數，使支援向量機有最佳的分類能力。實驗資料取自 UCI 資料庫之信用卡領域兩個資料集，以支援向量機建立分類器，實驗結果顯示依據支援向量機演算法建立的分類系統，在分類的正確率有不錯的表現。

關鍵字：支援向量機，格子點演算法，資料探勘，分類。

一、緒 論

由於國內發卡銀行積極推廣市場，信用卡使用率持續快速成長，信用卡變成金融銀行很重要的服務項目與收入來源之一。然而，隨著信用卡的普及使用，相對的信用卡之犯罪率也持續擴大成長。依據財政部金融局之資料顯示，至 92 年 5 月止國內發卡數已超過 6 千萬張，流通卡量也達 3 千 4 百萬張以上，而在市場快速成長同時，信用卡犯罪也隨之升高，銀行偽卡風險損失提高，92 年超過新台幣三十億元以上[1]。



信用卡詐欺問題是金融發卡界最急欲解決的問題。如果可以利用資訊系統幫助信用卡發卡者進行詐欺偵測，當有新的用戶申請發卡時，先對其進行信用評估來考慮是否發卡，這樣應該可以降低信用卡詐欺的風險，減少銀行的損失。目前常使用詐欺偵測的方法就是資料探勘(Data Mining)，資料探勘是由大量資料中探勘出不明確的、未知的以及潛在的有用資訊之過程[2]，利用資料探勘技術之預測能力，建構出一套信用卡授信決策模式，判斷是否核卡及授權以減少呆帳，可以幫助銀行發卡決策者做出正確的決策。

資料探勘有許多不同的功能，如分類(Classification)、預測(Prediction)、關聯法則(Association)、分群(Clustering)等。其中「分類」是從已定義類別的屬性集合中，根據目標屬性建立出在資料間的法則，來描述屬性與類別之間關係的模型，此模型可以對其它新的未知資料進行分類，以得到決策的依據。分類問題是資料探勘或者機器學習領域最常被探討的問題，而在資料探勘演算法中可以做分類的方法有很多，例如：統計方法、類神經網路、決策樹等[3]，各適用在不同的情況與資料性質。資料探勘應用於信用卡詐欺偵測已有不錯的成果，如：最近鄰居法(k-Nearest Neighbor)[4]、邏輯斯迴歸法(Logistic Regression)[5]、類神經網路[6~10]和決策樹[11~13]等方法，皆能有效分類申請者之客戶類型。

除了上述的方法外，近幾年中又有許多新的方法被提出，支援向量機(Support Vector Machine；SVM)是其中之一。支援向量機是由統計學習理論(Statistical Learning Theory)衍生而成的學習演算法，從統計學習理論中的簡易向量分類器(Simple Vector Classifiers)，逐漸發展成為超平面分類器(Hyperplane Classifiers)，到目前大家所熟悉的支援向量分類器(Support Vector Classifiers)[14]。支援向量機在處理分類與預測方面的問題都有不錯的表現，且廣泛的運用於不同的領域，例如：影像辨識、手寫辨識、文字分類與生物科技等相關分類問題都獲得相當好的成果[15~17]。

支援向量機的參數設定對於分類器的正確率有相當大的影響，所以如何調整支援向量機參數來建立分類器就相當重要。現今調整參數常使用的演算法為格子點演算法(Grid Algorithm) [18,19]。因此，本研究利用格子點演算法尋找支援向量機最佳化參數，來建立支援向量機的分類系統，提高分類的正確率。以信用卡資料集為實驗對象，嘗試使用支援向量機建立分類系統，以評估此法應用在信用卡資料之分類效果。

後續章節將詳細介紹支援向量機、實驗資料、格子點演算法調整支援向量機參數之過程以及支援向量機分類法之實驗結果。

二、支援向量機簡介

支援向量機(Support Vector Machines ; SVM)是由 Vapnik 在 1995 年和 AT & T 實驗室團隊所提出的一個新方法，其主要的理論是來自統計學習理論中結構化風險最小誤差法 (Structural Risk Minimization, SRM)[19]。支援向量機最主要是利用區分超平面 (Separating Hyperplane) 來分隔兩個或多個不同類別 (Class) 的資料，處理資料探勘中分類 (Classification) 的問題。在典型的分類問題中，我們通常會定義以下基本的表示方法：

x_i ：是一個向量，用來描述某筆 N 維資料的樣式 (Pattern) 或是屬性 (Attribute)，

$$x_i \in R^N, i = 1, 2, 3, \dots, m$$

y_i ：稱為標註 (Label) 或是目標 (Target)，通常用 $\{\pm 1\}$ 表示 (假設我們的分類目標為兩類)，

$$+1 \text{ 和 } -1 \text{ 皆表示兩不同的類別 (Class)。 } y_i \in \{\pm 1\}, i = 1, 2, 3, \dots, m$$

SVM 處理不同的問題又可分為線性支援向量機和非線性支援向量機兩種不同的類型 [21, 22]，以下將分別介紹。

2.1 線性支援向量機

線性支援向量機分類主要在輸入的訓練資料 (Training Data) 中，找出一個可以將資料分隔開最大邊界 (Margin) 的區分超平面 (Separating Hyperplane)。如圖 1，先定義區分平面之邊界為 $d_+ + d_-$ ， d_+ 和 d_- 分別表示標記為 +1 和 -1 不同類別的訓練資料分別距離區分超平面之最短距離，此類型的資料必須符合下面的條件：

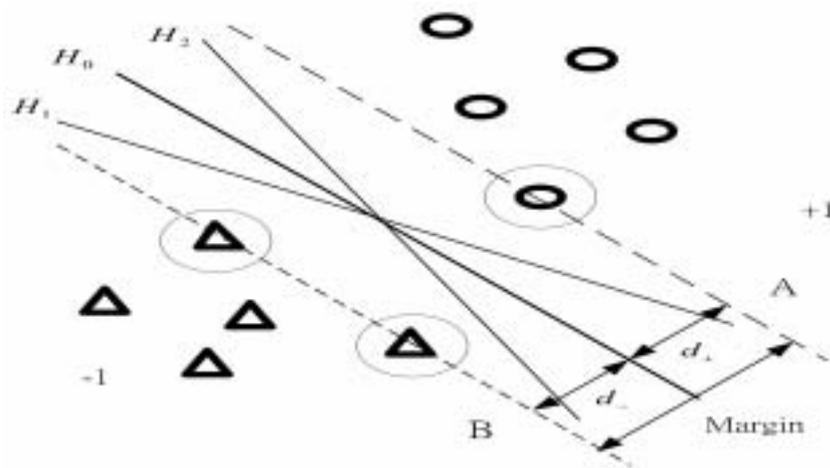
$$(x_1, y_1), \dots, (x_i, y_i), x_i \in R^d, y_i \in \{-1, 1\}$$

現在如果有一個超平面，可以將這兩類資料區分開來，則此超平面即可以稱為區分超平面，落在區分超平面的所有的 x 必須滿足， $w \cdot x + b = 0$ ， w 為超平面之法向量 (Normal Vector)， b 則是偏移量。所以我們可以把 $f(x) = w \cdot x + b$ 稱為決定函數 (Decision Function)，當輸入一筆測試資料 (Test Data) 時，就可以依據決定函數的值來分類。若 $f(x) > 0$ ，則將該筆資料歸類為 +1，若 $f(x) < 0$ 時就將該筆資料歸類於 -1。

支援向量機就是希望可以在不同類別的資料中，找出最大邊界 (Margin) 的區分超平面。我們可以從圖一中發現超平面 H_0 、 H_1 、 H_2 都可以達到區分類別的效果，而 H_0 是



最好的，因為 H_0 與邊界 A 和邊界 B 的距離最大。這種型態的學習過程稱為結構化風險最小誤差法(Structural Risk Minimization ; SRM)其目的就是希望分類器能在期望的誤差(Expected Risk)中找到最小值。



圖一 最佳化區分超平面

以下是支援向量機的演算法，先定義區分平面之邊界為 $d_+ + d_-$ ， d_+ 和 d_- 分別表示標記為+1 類別和-1 類別距離區分超平面之最短距離，此類型的資料必須符合下面的條件：

$$x_i \cdot w + b \geq +1 \quad \text{for } y_i = +1 \quad (1)$$

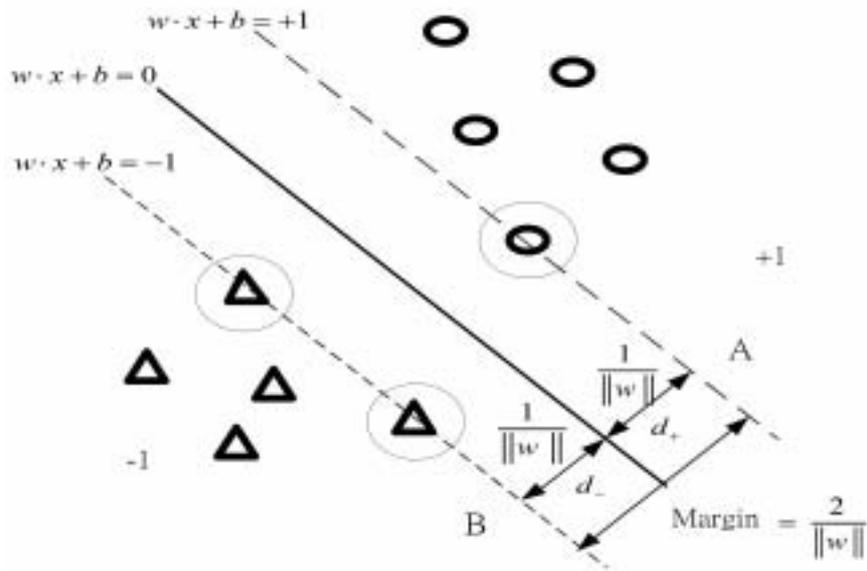
$$x_i \cdot w + b \leq -1 \quad \text{for } y_i = -1 \quad (2)$$

可將方程式(1)和方程式(2)兩式結合成以下不等式：

$$y_i(x_i \cdot w + b) - 1 \geq 0 \quad \forall i \quad (3)$$

由(1)(2)分別到 $w \cdot x + b = 0$ 的距離為 $\frac{1}{\|w\|}$ ，也就是說 $d_+ = d_- = \frac{1}{\|w\|}$ ，所以邊界為 $\frac{2}{\|w\|}$ 。

如果我們要尋找區分平面的最大邊界，就如同在符合限制式(3)的條件之下，求的 $\|w\|^2$ 最小值。在限制式(3)中，如果有任何一個 x_i 可使得等號成立，我們即可稱 x_i 為支援向量(Support Vector)。如圖二，使用二維度的訓練資料為例，用圓形圈起來的資料點即稱為支援向量。



圖二 最大邊界圖

在限制式(3)的情況下，求 $\|w\|^2$ 的最小值，可以利用拉格藍吉(Lagrange)最佳化問題來處理，使用拉格藍吉最佳化方法有兩個理由：第一，限制式(3)可以被拉格藍吉係數(Lagrange Multipliers)取代，更容易處理；第二，經過整理後的公式中訓練資料將可以用向量內積的形式來表示，方便計算。

Lagrange function 表示如下：

$$L_p \equiv L(w, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^m \alpha_i [y_i (w \cdot x_i) + b] - \sum_{i=1}^m \alpha_i \quad (4)$$

其中拉格藍吉係數 $\alpha_i, i = 1, 2, 3, \dots, m$ 對應到方程式(3)中的每一個不等式，且 $\alpha_i > 0$ 。這樣我們就可以把原本的問題轉變成為求 L_p 最小值且限制式為 $\alpha_i > 0$ 。此時，使用拉格藍吉最佳化對偶問題(Lagrange Dual Optimization Problem)來解決，先對方程式(4)的 w 和 b 偏微分：

$$\frac{\partial}{\partial w} L_p = 0 \quad , \quad \text{求得 } w = \sum_{i=1}^m \alpha_i y_i x_i \quad (5)$$

$$\frac{\partial}{\partial b} L_p = 0 \quad , \quad \text{求得 } \sum_{i=1}^m \alpha_i y_i = 0 \quad (6)$$

將方程式(5)，方程式(6)代入方程式(4)後，整理後可以得到方程式(7)，我們給予這個函式一個新的符號 L_D 以避免混淆，原本求 L_p 的最小值問題，其對偶問題變成求 L_D 的最大值。



$$L_D = \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=0}^m \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \quad (7)$$

$$\text{Subject to: } \alpha_i \geq 0 \quad i=1, \dots, m \quad \text{and} \quad \sum_{i=1}^m \alpha_i y_i = 0$$

根據 Karush Kuhn-Tucker(KKT)理論[23]，代入最佳化解 (w^*, b^*) 將可以得到方程式(8)，且 α_i^* 之值必定大於或等於零，所以說如果 α_i^* 之值大於零則 $y_i(<w_i^* \cdot x_i > + b^* - 1)$ 必定是等於零。

$$\sum_{i=1}^m \alpha_i^* [y_i^* (<w_i^* \cdot x_i > + b^* - 1)] = 0 \quad (8)$$

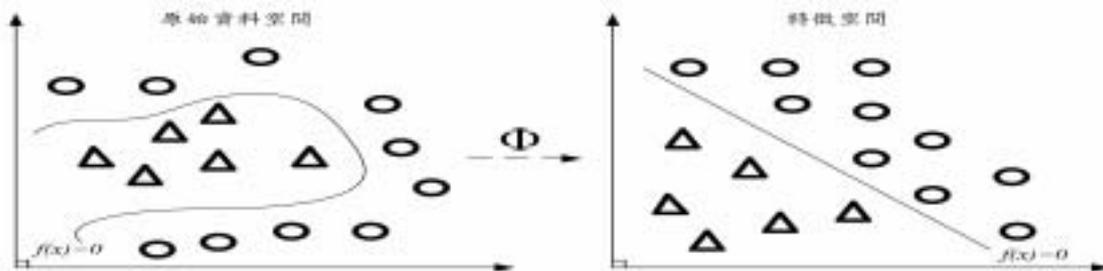
這意思就是說如果有 x_i 可以滿足上式，這些 x_i 就是最接近最佳化區分超平面的向量，也就是說如果有一個 x_i 的 $\alpha_i^* \geq 0$ ，則該 x_i 即可稱為支援向量。所以我們找出支援向量後，即可找到最大邊界。最後我們可以得到一個分類處理問題的函數：

$$f(x) = \text{sgn}(\sum_{i=1}^m y_i \alpha_i \cdot (x_i \cdot x_j) + b) \quad (9)$$

當 $f(x) > 0$ 時，表示該筆資料與標註為“+1”的資料屬於同一類；反之則是屬於另外一類。

2.2 非線性支援向量機

在現實世界中並不是所有的資料都可以找到線性區分超平面，所以如果遇到非線性方面的問題，在原始資料空間中無法找到線性區分超平面。Boser 和 Vapnik[24]等人針對非線性函數的問題，發現如果將原始資料透過非線性的映射函數 Φ 轉換到另外一個較高維度的特徵空間(Feature Space)中 $(\Phi: R^d \rightarrow F)$ ，然後在特徵空間上執行線性分類，可以獲得更好的正確率，請參考圖三。



圖三 原始資料透過 Φ 轉換到特徵空間中

在最佳化對偶問題(7)中, $(x_i \cdot x_j)$ 會影響到最後結果, 如果將資料轉換到特徵空間中最後會影響結果的就是 $\Phi(x_i) \cdot \Phi(x_j)$, 而 $\Phi(x_i)$ 和 $\Phi(x_j)$ 的內積則可以用核心函數(Kernel Function)來取代, 因此只要利用核心函數計算出資料在特徵空間的內積值, 就不需要直接把資料映射到特徵空間:

$$k(x_i, x_j) := (\Phi(x_i) \cdot \Phi(x_j)) \quad (10)$$

所以非線性支援向量機所處理的最佳化問題函式可改寫為:

$$L_D = \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=0}^m \alpha_i \alpha_j y_i y_j k(x_i \cdot x_j) \quad (11)$$

$$\text{Subject to: } 0 \leq \alpha_i \leq C \quad i=1,2,3,\dots,m \quad \text{and} \quad \sum_{i=1}^m \alpha_i y_i = 0$$

常用的核心函數有線性(Linear)、多項式(Polynomial)、放射(Radial Basis Function; RBF) 和 S 型(Sigmoid)等 4 種[24]。每個核心函數都有不同的參數, 使用者必須依據不同的核心函數調整。

Linear kernel : $k(x_i, x_j) = x_i \cdot x_j^T$

Polynomial kernel : $k(x_i, x_j) = (1 + x_i \cdot x_j)^d$

Radial Basis Function kernel : $k(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$

Sigmoid kernel : $k(x_i, x_j) = \tanh(kx_i \cdot x_j - \delta)$

本研究採用放射型(RBF)核心函數來建立分類系統, 因為放射型函數能分類非線性且高維度的資料, 而需調整的參數只有 C 和 γ 兩個, 是選擇核心函數的優先選擇[25]。

三、以支援向量機建立信用卡資料分類系統

3.1 實驗資料

本研究實驗採用 UCI 資料庫[26]之兩個信用卡資料集, 資料集摘要如下:



1. Australian credit database

澳洲信用卡公司的 690 個案例，根據申請人特徵決定是否發卡，總共有 15 個屬性，其中包含 6 個連續型屬性、8 個名目型屬性和 1 個目標屬性(接受或拒絕)。目標屬性中共有 307 個接受案例，383 個拒絕案例，這個資料集的特色是在連續屬性的部分，最大值跟最小值的差距相當大。

2. German credit database

有 1000 位德國客戶的信用卡審核資料，共有 24 個連續型屬性和一個目標屬性(接受或拒絕)。這 24 個屬性描述客戶的情況，包含了 credit history、account balance、loan purpose、loan amount、employment status、personal information、age、housing 與 job，目標屬性共有 700 個接受案例，有 300 個拒絕案例。

3.2 分類實驗方式

一般來說，正確率的評估有許多種，較常用的正確率評估方式為 Holdout、Bootstrap 和 Cross-Validation 這三種正確率的評估方法 [27,28]，本研究使用的方法是 k-fold-cross-validation [29]，此法是將資料分成大小相等 k 個部分，而且這 k 群資料彼此之間互斥，例如將資料集 A 分成 k 個部分 $\{A_1, A_2, \dots, A_k\}$ 。使用此方法，必須進行 k 次訓練和測試，輪流以 $A_t, t \in \{1, 2, \dots, k\}$ 為測試資料，剩下的資料為訓練資料，以測試資料驗證其正確率。進行 k 次後，再將 k 次的正確率加以平均求得此模型最後的平均正確率，這種方式能確保每筆資料皆能當作測試資料，而且全部的測試樣本都是獨立的。本研究以 10 組交叉驗證(k=10)的方式，將資料集隨機分成 10 組，每一組輪流當作測試樣本，其它則當作訓練樣本，也就是共有 10 組不同的訓練樣本和測試樣本。如此執行 10 次後，得到 10 組分類正確率，再平均 10 組分類正確率則是該資料集的平均正確率。

3.3 屬性篩選

當進行資料探勘時，探勘的資料集可能會有相當多的屬性，但並非每個屬性都重要，也就是說，某些屬性可能對目標值的影響相當小，甚至沒有。我們可以去掉這些不重要的屬性以提高資料探勘的正確率和電腦的運算效率。屬性篩選的方法相當多，如：逐步迴歸法(Stepwise Regression)、熵法(Entropy)、特徵權重法(Feature Weighting Methods)

[30]，在傳統統計和資料探勘屬性篩選中，逐步迴歸屬性篩選法是經常使用的方法之一 [31]，本研究嘗試採用逐步迴歸進行屬性篩選，刪除對目標屬性影響較小的輸入屬性，減少建立分類器時電腦運算所需要的時間，並把篩選後的正確率與未經屬性篩選的原始資料所計算出的正確率進行比較，比較是否有顯著的差異。逐步迴歸所刪除與保留的屬性如表一，澳洲資料集總共篩選掉 7 個屬性，德國資料集篩選掉 12 個，皆占全部屬性的 50%，「逐步迴歸保留屬性」資料欄中的每個數字則是代表資料集中第幾項的屬性被保留下來。

表一 屬性篩選後保留的屬性

資料集	原始屬性數	篩選後屬性數	逐步迴歸保留屬性
澳洲	14	7	4,5,8,9,10,13,14
德國	24	12	1,2,3,5,6,7,9,11,15,16,17,21

3.4 以支援向量機建立分類系統

本研究以 LIBSVM [32]在 Python 的環境下進行支援向量機的處理，處理的流程如圖 4，分為四個步驟，詳細步驟說明如下。

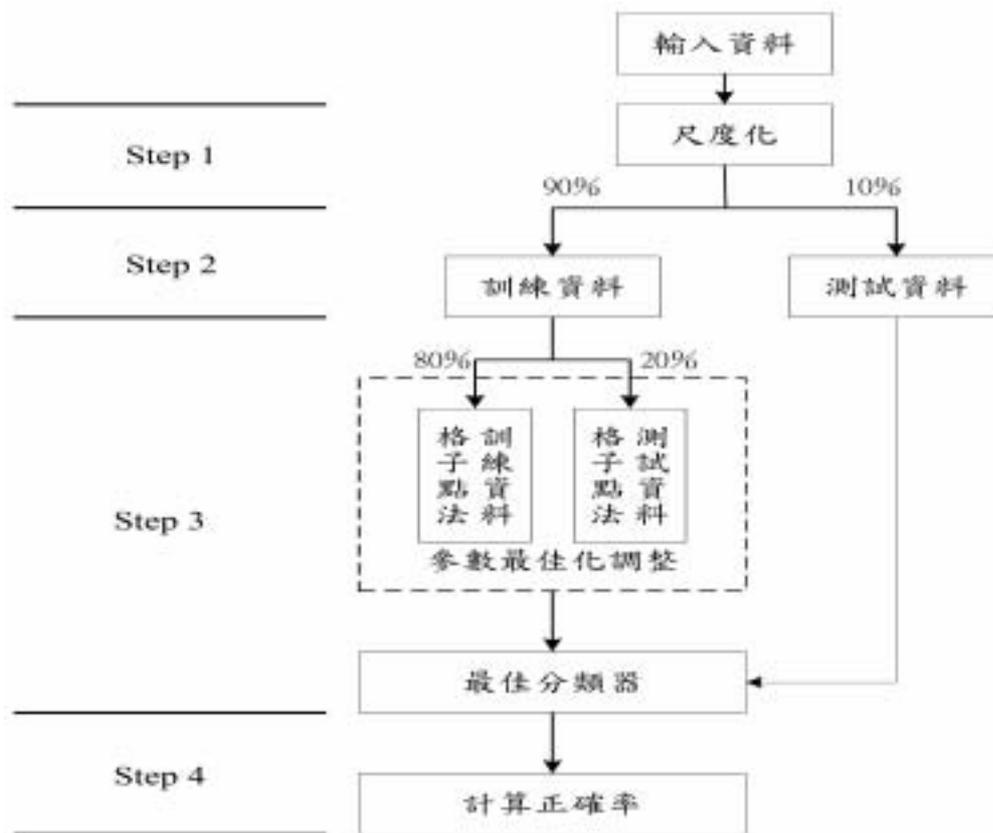
步驟一：尺度化

把原始資料進行尺度化(Scaling)，尺度化就是把屬性的數值轉換到特定範圍內，執行這個步驟最主要的目的是不要讓數字的範圍過大，控制在一定區間的範圍內，以增加支援向量機正確性。一般來說大多把所有屬性值轉換到 1 到-1 或是 1 到 0 之間，使用的公式如(12)， v 是原始資料，而 v' 是經過轉換的資料， \max_a 、 \min_a 分別代表尺度化之上界(Upper bound)與下界(Low bound)。

$$v' = \frac{v - \min_a}{\max_a - \min_a} \quad (12)$$

步驟二：10 組交叉驗證

將資料集隨機分成 10 組，每一組輪流當作測試樣本，其餘 9 組則當作訓練樣本。

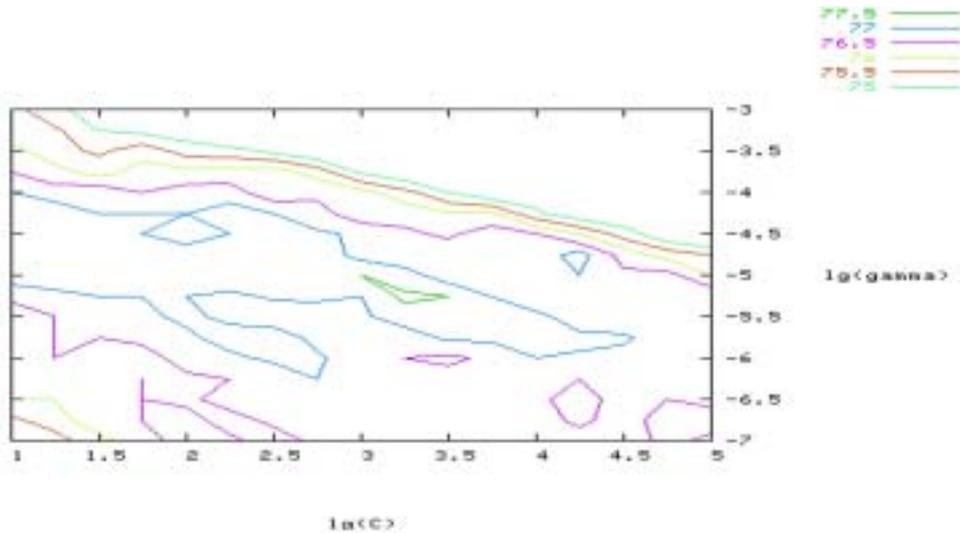


圖四 SVM 建立分類系統之流程圖

步驟三：建立分類系統

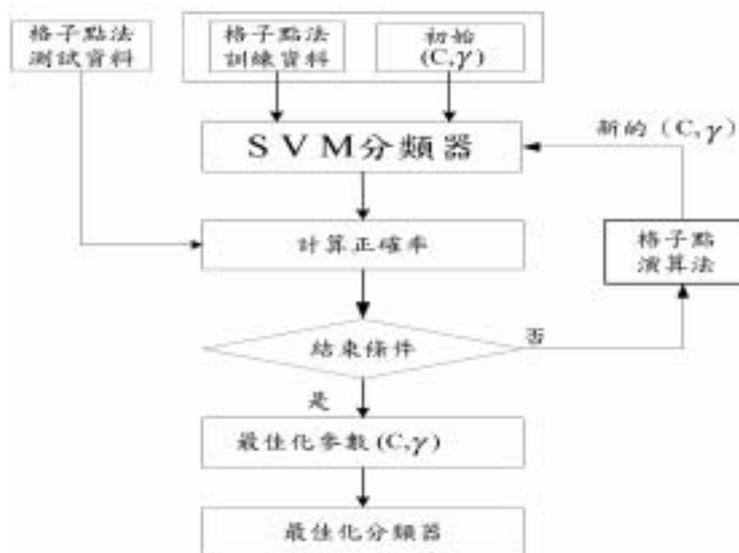
使用支援向量機法的 RBF 核心函數來建立分類系統，使用不同的 C 和 γ 參數會產生不同的正確率，所以如何找到最佳化的 C 和 γ 來建立分類系統是重要的關鍵。為了尋找最佳化 C 和 γ ，我們把占全部資料 90% 的訓練資料集，進行第二次的交叉驗證，也就是把這 90% 的訓練資料再做 5 組交叉驗證 ($k=5$)，使用格子點演算 (Grid algorithm) 來尋找最佳化參數 C 和 γ 。

所謂格子點演算法，就是把不同的 C 與 γ 值的可能解組合 (C, γ)，以支援向量機算出其正確率。在這個演算過程中，當正確率更高的解被找到的時候，其它正確率比目前最佳解低的點就會被去除。如果最佳解並不只一組，再對每一組解的周圍做細部尋找，例如：有一組解為 (2,2)，則我們會對 (2.1,2.1) 或是 (1.9,1.9) 等周圍的解做嘗試，直到找到最佳解為止。格子點法會計算各種可能的解組合 (C, γ) 之正確率，以德國資料集為例，圖 5 為根據不同正確率的解，所畫出的等高線圖，不同的等高線代表不同 (C, γ) 的正確率。



圖五 正確率等高線圖

使用格子點演算法來進行支援向量機參數調整過程如下：隨機產生一組 C 和 γ 代入 SVM 分類器，把 SVM 分類器所產生的分類結果與測試資料的目標屬性比對計算正確率，並判斷是否滿足結束條件。結束條件通常由使用者定義，一般都是設定正確率達到某個程度，最佳預期的正確率通常為 100%，但並不是每個資料集都有可能達到這樣的目標，所以使用者可以針對不同的資料集做適當的調整。如果滿足結束條件則會得到最佳化的參數 (C, γ) ，如果不滿足，則使用格子點演算法來尋找，直到滿足結束條件為止其流程圖如圖六，最後我們就可以用最佳化的 C 和 γ 參數來建立分類器。



圖六 格子點演算法調整參數流程圖



步驟四：計算分類正確率

最後把測試資料代入最佳化的分類系統，並且計算出正確率。

四、實驗結果

所得到的正確率及最佳化的(C, γ)參數值與平均正確率如表二和表三，常用來衡量分類成功與否的標準為計算分類正確率(Accuracy Rate)，即正確分類的筆數佔全部資料集筆數的比例，在德國資料集中，使用原始資料所得到的平均正確率為 76%，而經過屬性篩選資料所得到的平均正確率為 76.8%。在澳洲資料集中，使用原始資料所得到的平均正確率為 84.7%，而經過屬性篩選資料所得到的平均正確率為 85.4%。

為了比較屬性篩選對正確率是否有影響，以無母數檢定中兩個相關樣本的 Wilcoxon 符號等級檢定(Signed Rank Test)對德國資料集、澳洲資料集之正確率進行檢定，得到 p -值在德國與澳洲資料集中分別為 0.234 和 0.129，皆大於 0.05，表示原始資料是否經過屬性篩選其正確率並沒有顯著差異。

表二 SVM 對德國資料所計算出的正確率(%)、C 和 γ 值

德國	原始資料			經過屬性篩選		
	正確率%	最佳化 C	最佳化 γ	正確率%	最佳化 C	最佳化 γ
第 1 組	77	2048	0.000488281	77	2048	77
第 2 組	74	8	0.0078125	72	512	74
第 3 組	77	512	0.001953125	80	512	77
第 4 組	81	8192	0.00012207	81	512	81
第 5 組	74	512	0.00012207	74	8	74
第 6 組	78	128	0.000488281	78	8	78
第 7 組	81	32768	0.000035	83	2048	81
第 8 組	72	8	0.0078125	74	2048	72
第 9 組	68	2048	0.00012207	71	512	68
第 10 組	78	512	0.00012207	78	2048	78
平均	76			76.8		

表三 SVM 對澳洲資料所計算出的正確率(%)、C 和 γ 值

澳洲	原始資料			經過屬性篩選		
	正確率%	最佳化 C	最佳化 γ	正確率%	最佳化 C	最佳化 γ
第 1 組	85.5	0.03125	0.125	85.5	0.125	0.0078125
第 2 組	76.8	0.03125	0.125	78.2	32768	0.03125
第 3 組	76.8	128	0.03125	76.8	0.5	2
第 4 組	86.9	32768	0.001953125	88.4	8192	0.03125
第 5 組	91.3	0.5	0.001953125	86.9	8192	0.03125
第 6 組	86.9	8192	0.001953125	89.8	32768	0.03125
第 7 組	85.5	128	0.03125	86.9	2048	0.03125
第 8 組	88.4	0.125	0.0078125	88.4	8192	0.125
第 9 組	82.6	8192	0.001953125	84	2048	0.125
第 10 組	86.9	0.125	0.0078125	89.8	32768	0.03125
平均	84.7			85.4		

經過平均計算後是否經過屬性篩選的分類正確率百分比如表四：

表四 信用卡樣本資料集的分類平均正確率(%)與屬性數目

方法 資料集	未經屬性篩選		經過屬性篩選		檢定 Wilcoxon 之 p-值
	輸入 屬性數	平均正 確率	輸入 屬性數	平均正 確率	
澳洲	14	84.7%	7	85.4%	0.129
德國	24	76%	12	76.8%	0.234

從實驗的結果可以看到，在所有資料集中，支援向量機經過屬性篩選後與原始資料所得到的正確率類似。經過屬性篩選後，去除許多沒有影響或是影響很小的屬性，在建立分類器時減少許多計算時間，提高執行的效率，而且可以發現正確率並沒有顯著的下降，所以利用迴歸來進行屬性篩選是一種可以被接受的方法。



五、結 論

隨著信用卡發卡量的快速增加，信用卡的犯罪也隨之升高，利用資料探勘的工具對發卡用戶進行詐欺偵測是逐漸被廣於使用的方式，預測發卡用戶的信用評等，以減少銀行的損失。

本研究以支援向量機演算法建立分類系統，並利用格子點演算法調整支援向量機之參數，建立最佳化的支援向量機分類系統，並且利用逐步迴歸進行屬性篩選，最後以信用卡資料集實驗，發現以支援向量機建立的分類系統有良好的分類的效果。實驗的結果顯示，支援向量機對於信用卡詐欺偵測的準確度皆能夠達到一般實務上的要求，再對有屬性篩選、無屬性篩選的兩個支援向量機分類結果進行無母數檢定後，發現兩者的正確率沒有顯著差異，但經過屬性篩選後所建立支援向量機分類器速度較快。

以支援向量機分類器的預測能力，能夠有效的協助管理者進行信用卡詐欺偵測與監控，本研究之結果可以提供金融界進行預防信用卡詐欺問題決策之參考。未來研究可以嘗試採用不同的屬性篩選方法，並搭配支援向量機分類器，以及使用本土資料進行信用卡評等之資料探勘。

參考文獻

1. Bureau of Monetary Affairs, Financial Supervisory Commission, ROC. Available: <http://www.boma.gov.tw/modules/ensection/english.php>
2. J. Han, M., Kamber, 2001, *Data Mining: Concepts and Techniques*, CA: Morgan Kaufmann Publishers, San Francisco, USA.
3. D. Michie, D. J. Spiegelhalter & C. C. Taylor, 1994, *Machine learning, neural and statistical classification*, Ellis Horwood, New York.
4. W. E. Henley, D. J. Hand, 1996, "A k-nearest neighbor classifier for assessing consumer credit risk," *Statistician*, vol. 44, pp. 77-95.
5. W. E. Henley, 1995, *Statistical aspects of credit scoring*, Dissertation, The Open University, Milton Keynes, UK.
6. R. Malhotra, D. K. Malhotra, 2002, "Differentiating between good credits and bad credits using neuro-fuzzy systems," *European Journal of Operational Research*, vol. 136, pp.



- 190-211.
- 7.D. West, 2000, "Neural network credit scoring models," *Computers and Operations Research*, vol. 27, pp.1131-1152.
- 8.R. Brause, T. Langsdorf & M. Happ, 1999, "Neural data mining for credit card fraud detection," *Proceedings 11th IEEE International Conference on Tools with Artificial intelligence*, pp.103-106.
- 9.P. K. Chan, W. Fan, A. L. Prodromidis & S. J. Stolfo, 1999, "Distributed data mining in credit card fraud Detection," *IEEE Intelligent Systems*, vol. 14, no. 6, pp. 67-74.
- 10.S. Ghosh, D.L. Reilly, 1994, "Credit card fraud detection with a natural-network System Sciences," *Information Systems Proceedings of the 7th Hawaii International Conference*, vol. 3, pp.621-630.
- 11.I. K. Sethi, M. Otten, 1990, "Comparison between entropy net and decision tree classifiers," *IJCNN International Joint Conference*, vol. 3, no. 1, pp.63-68.
- 12.M. L. Hambaba, 1996, "Intelligent hybrid system for data mining," *Computational Intelligence for Financial Engineering of the IEEE/IAFE*.
- 13.R. H. Davis, D. B. Edelman & A. J. Gammernan, 1992, "Machine learning algorithms for credit-card applications," *IMA Journal of Mathematics Applied in Business and Industry*, vol. 4, pp.43-51.
- 14.B. Schölkopf, A. J. Smola, 2000, *Statistical learning and kernel methods*, Cambridge, USA.
- 15.M. Pontil, A. Verri, 1998, "Support vector machines for 3D object recognition", *IEEE Transaction On PAMI*, vol. 20, pp.637-646.
- 16.T. Joachims, 1998, "Text categorization with support vector machines", *In Proceedings of European Conference on Machine Learning (ECML)*.
- 17.M. Brown, W. Grundy, D. Lin, N. Cristianini, C. Sugnet, T. Furey, M. Ares & D. Haussler, 1999, "Knowledge-base analysis of microarray gene expression data using support vector machines", *Technical report*, University of California in Santa Cruz.
- 18.C. W. Hsu, C. J. Lin, 2002, "A simple decomposition method for support vector Machine," *Machine Learning*, vol. 46, pp.219-314.
- 19.H.T. Lin, C.J. Lin, 2003, *A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods*, Technical report, Department of Computer Science & Information Engineering, National Taiwan University.



- 20.V. N. Vapnik, 1995, *The Nature of Statistical Learning Theory*, Springer-Verlag, NY, USA.
- 21.C. J. C. Burges, 1998, “A tutorial on support vector machines for pattern recognition,” *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp.955-974.
- 22.B. Schölkopf, C. J. C. Burges & A. J. Smola, 1999, “Introduction to support vector learning, advances in kernel methods-support vector learning,” *Cambridge, MA*, pp. 1-15.
- 23.R. Fletcher, 1987, *Practical methods of optimization*, John Wiley and Sons, Inc., 2nd Edition.
- 24.S. R. Gunn, 1998, “Support Vector machines for classification and regression,” *Technical Report*, University of Southampton.
- 25.C. W. Hsu, C. C. Chang & C. J. Lin, 2003, A practical guide to support vector classification. Available: <http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf>.
- 26.S. Murthy, D. Aha, 1996, UCI repository of machine learning data tables. Available: <http://www.ics.uci.edu/~mlearn/>
- 27.B. Elton, R. Tibshirani, 1995, “Cross-validation and the bootstrap: estimating the error rate of a prediction rule,” *Technical Report*, Department of Statistics Stanford University.
- 28.R. Kohavi, 1995, “a study of cross-validation and bootstrap for accuracy estimation and model selection,” *IJCAI*, pp.1137-1145.
- 29.S. L. Salzberg, 1997, *On comparing classifiers: pitfalls to avoid and a recommended approach*, Data Mining and Knowledge Discovery, Kluwer Academic Publishers, Boston. Manufactured in The Netherlands, vol. 1, pp.317–327.
- 30.H. Liu, H. Motoda & L. Huan, 1998, *Feature selection for knowledge discovery and data mining*, Boston Kluwer, USA.
- 31.P. Pudil, K. Fuka, K. Beránek & P. Dvorač, 1999, “Potential of artificial intelligence based feature selection methods in regression models,” *IEEE 3rd International Conference on Computational Intelligence and Multimedia Applications*, pp.159-163.
- 32.C. C. Chang, C. J. Lin, 2003, “LIBSVM: a library for support vector machine,” Available: <http://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html>.



Credit Card Scoring by Support Vector Machine

*Cheng-Lung Huang **Mu-Chen Chen ***Chieh-Jen Wang

*Kaohsiung First University Department of Science & Technology

E-mail: cl.huang@msa.hinet.net

***Taipei University of Technology Department of Business Management

E-mail: bmcchen@ntut.edu.tw

**Huafan University Department of Information Management

E-mail: m9225002@cat.hfu.edu.tw

Abstract

Credit card industry grows rapidly recently, and thus huge of consumers' credit data are collected by the credit card department of the bank. The credit card scoring manager often evaluates the consumer's credit with intuitive experience. However, with the supporting of credit classification model, the manager can accurately evaluate the applicant's credit score. Support Vector Machine (SVM) classification is currently an active research area, and successfully solves classification problems in many domains. This study uses the Grid Algorithm to optimize the SVM classifier parameter, and the SVM classifier obtained to evaluate the applicant's credit score from the applicant's input features. Two credit card datasets from UCI database are selected as the experimental data to demonstrate the accuracy of the SVM classifier. This study concludes that SVM is a promising addition to the existing methods, and it can offer the manager an informative insight into credit scoring decisions.

Keywords: Support vector machine, Grid algorithm, Data mining, Classification

