

次序性支持向量機與多類別支持向量機於協同推薦上之比較

蕭文峰

林文星

莊可任

國立屏東商業技術學院

國立屏東商業技術學院

國立屏東商業技術學院

wfsiao@gmail.com

tnwxds@hotmail.com

muki770824@yahoo.com.tw

摘要

支持向量機(SVM)在過去已被證明其穩健性和良好的性能，但仍無法適切處理次序類別之預測。因此許多學者嘗試提出能處理次序類別的支持向量機。然而這些方法在真實世界資料集的表現如何是本研究所想探討的。本研究比較了 oSVM(發表在 JMLR)，SVOR(發表於 ICML 共兩個版本)以及 LIBSVM 的多類別 SVM 等方法在學習次序性資料的表現。實驗資料集為 GroupLens 的電影資料集。由實驗結果顯示，SVOR-IM 不論在 zero-one error 或 absolute error 的評估上都優於其他兩個方法。而三個次序尺度方法在協同推薦的表現上都優於多類別 LIBSVM 法，顯示次序尺度方法的有效性。

關鍵字：次序類別、支持向量機、oSVM、SVOR

1. 緒論

在日常生活中，我們常需將某些人事物加以區分，使它們之間有著程度上的差異。例如，在 Amazon 線上購物網站，消費者可以對商品給予 1~5 顆星的評等，並提供文字評論。通常消費者會依喜好程度給予不同的星等，5 顆星代表最喜歡/最推薦、接著是 4 顆星、3 顆星、2 顆星、一直到 1 顆星代表最不喜歡/最不推薦的商品。這些星等間只代表著次序關係，不代表程度差異的量，因此 4 顆星的喜好程度未必是 2 顆星的程度的 2 倍。因此在處理次序類別的資料並不像區間尺度資料可以進行加減運算。因此，以迴歸方法或多類別分類器來處理次序尺度的分類問題通常並無法獲得讓人滿意的結果。

支持向量機(SVM)在過去已被證明其

穩健性和良好的性能，但在處理次序類別之預測上一直沒有良好的解法，因此許多學者嘗試提出能處理次序類別的支持向量機(Cardoso et al., 2007; Chu and Keerthi, 2005)。然而這些方法在真實世界的資料集(e.g. 協同推薦資料集)的表現如何是本研究所想探討的。本研究比較了 oSVM(Cardoso et al., 2007)(發表在 JMLR)，SVOR(Chu and Keerthi 2005)(發表於 ICML，共有兩個版本，SVOR-IM 與 SVOR-EX)以及多類別 LIBSVM(Chang and Lin, 2001)的 SVM。我們想瞭解，次序類別的支持向量機是否會優於多類別支持向量機在次序尺度資料集上的表現。

2. 文獻探討

2.1. Support Vector Machine for Ordinal data

Cardoso and da Costa (2007)提出了一種新的分類機器學習方法，用於處理次序類別的分類問題，此方法稱之為"資料複製法"。利用加入虛擬變數(dummy variables)將資料複製並映射到更高的維度，如此原本在低維度空間無法線性求解的問題可在高維度上順利處理。以二維上的三類資料學習為例(圖 1)，原本不可能找到一個超平面將三個類別完美切割，但若將原資料擴充成三維(加入一個虛擬變數)，並將資料複製成兩份，其中一份在第三維是 0，另一份在第三維是 1(如圖 2 所示)，並根據資料點所在位置，將其所屬類別重新分配輸出為 +1 或 -1(如圖 3 所示，在底層的資料原本正方形資料點會被重新分配成 -1，圓形與三角形則是 +1；上層的資料則是正方形和圓形被分成 -1，三角形是 +1)。

如此我們可以用一線性超平面來將資料點完美切割。當新資料點(test point)欲進

行預測時，新資料點也會在被擴充並複製。接著透過擴充後的資料點在切割超平面上的分類結果可以得出其預測值。以圖 4 的測試資料點為例，它在底層的預測值是-1，在上層的預測值為+1，故其所屬類別會是 $2(+1$ 的個數加 1)。最後再將其資料降低維度成原始的維度，如圖 5 所示。從降低維度後來看，似乎不可以分割的資料可以利用映射到更高的維度空間來進行線性分割，而且 SVM 還可以同時學習到兩個最接近類別的邊際。

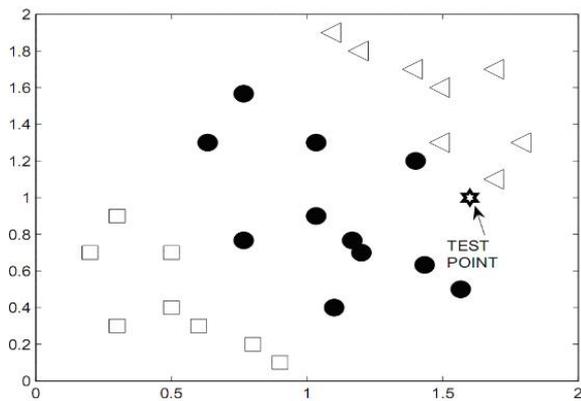


圖 1. Data replication in oSVM.

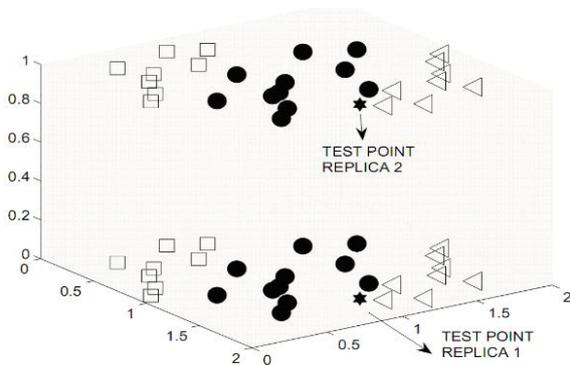


圖 2. Data replication in oSVM.

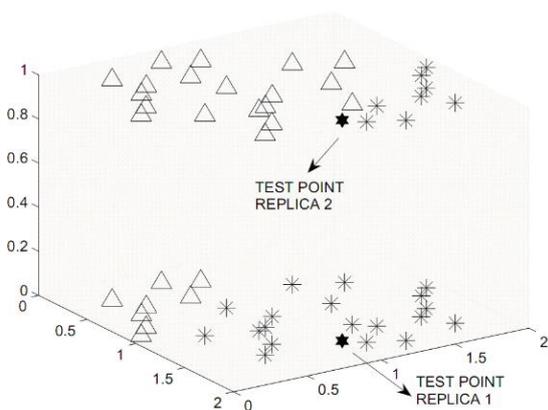


圖 3. Data classes are reassigned in oSVM.

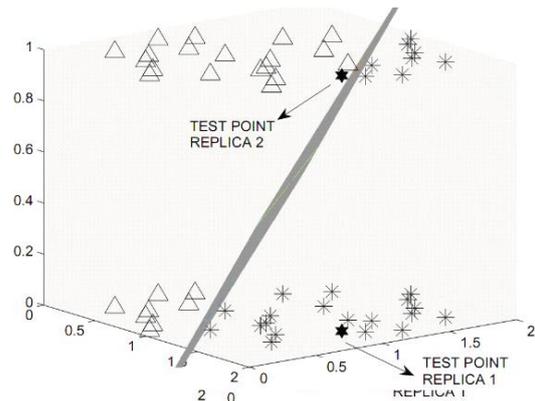


圖 4. Data are reassigned in oSVM.

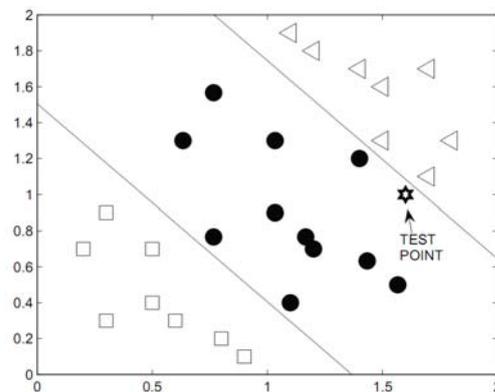


圖 5. The data remapped to lower dimension
(Cardoso and da Costa, 2007)

2.2. Support Vector Ordinal Regression(SVOR)

Chu and Keerthi (2005)提出兩個支持向量方法在次序迴歸上，在次序尺度上最佳化多重閾值來定義平行的判別超平面。這兩種方法保證閾值在最佳化上是適當的規則。而這些最佳化問題的數量大小和訓練範例是呈現線型的增加。他們使用了SMO演算法來解決這些最佳化問題，因為SMO是非常容易且有效地解決二次規劃問題。

Chu and Keerthi (2005)採用一個新的方法在支持向量次序迴歸，替代了只有從相鄰類別的範例中來決定閾值的錯誤，它們允許所有類別的範例來在所有閾值上計算。例如限制的閾值不是明確的在新的公式上，閾值的次序不等式會自動滿足在最佳解。

圖 6 解釋新的鬆馳變數 ξ 和 ξ^* ，對於一個閾值 b_j 從所有較低層類別的所有範例的函數值，應該小於較低邊際 b_{j-1} ；如果不成立，將 $\xi_{ki}^j = \langle w, \phi(x_i^k) \rangle - (b_j - 1)$ 視為有關在 $k \leq j$ 的範例 x_i^k 在 b_j 有誤。相同地，從所有較上類別的所有範例的函數值應該大於較高邊際 $b_j + 1$ ；否則 $\xi_{ki}^{*j} = (b_j + 1) - \langle w, \phi(x_i^k) \rangle$ 視為有關範例 x_i^k 在 b_j 有誤，這裡的 $k > j$ 。這此下標 k_j 表示鬆馳變數有關於在 k 類的第 i 個輸入範例；上標 j 表示鬆馳變數有關於 b_j 的較低類別；上標 $*j$ 表示鬆馳變數有關於 b_j 的較高類別。

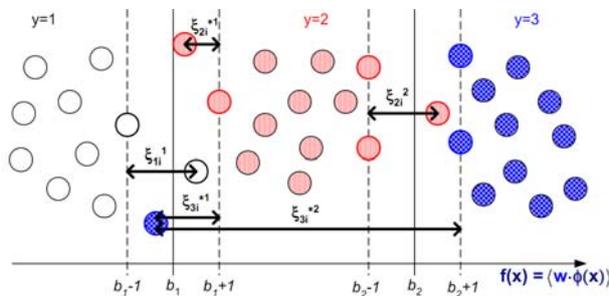


圖 6. The diagram of Implicit Constraints on Thresholds(Chu and Keerthi, 2005)

2.3. 協同過濾

協同過濾(Collaborative Filtering)是藉由計算自己的偏好與他人的偏好之間的相似度，來進行個人化的推薦。其主要想法是要自動化以前人們口耳相傳的產品推薦流程。相較於內容過濾法(content-based filtering)，協同過濾已被證明更適合於個人化推薦。根據 Adomavicius and Tuzhilin (2005)，協同過濾可以區分為記憶為基(memory-based)的協同過濾及模式為基(model-based)的協同過濾。

2.3.1. 記憶為基的協同過濾

記憶式演算法指的是那些基於使用者項目評分矩陣(維度為 $m * n$)來做預測的經驗法則(heuristics)。亦即，要預測使用者 u 對項目 i 的評分，可以使用其它使用者在項目 i 的綜合結果來代表。先計算已評價項目和待預測項目的相似度，並以相似度作為權重，加權各已評價項目的分數，得到待預測項目的預測值。常用的演算法有相關

式相似度(correlation-based similarity)(1)、餘弦式相似度(cosine-based similarity) (2)、修正餘弦相似度(adjusted cosine similarity) (3)。其中， $R_{u,i}$ 代表使用者 u 對項目 i 的評分， \bar{R}_i 表項目 i 的平均得分。在餘弦式相似度中，項目 i 與項目 j 可視為是 m 維使用者空間中的兩個向量，由其夾角可以定出兩項目之間的相似度。相關式相似度及餘弦式相似度都未考慮使用者評估尺度的差異性，因此修正餘弦相似度被提出以修正此問題，此處 \bar{R}_u 表使用者 u 的平均評分。

$$\sin(i, j) = \text{corr}_{i,j} = \frac{\sum_{u \in U} (R_{u,i} - \bar{R}_u)(R_{u,j} - \bar{R}_u)}{\sqrt{\sum_{u \in U} (R_{u,i} - \bar{R}_u)^2} \sqrt{\sum_{u \in U} (R_{u,j} - \bar{R}_u)^2}} \dots \dots \dots (1)$$

$$\sin(i, j) = \cos(i, j) = \frac{\bar{i} \cdot \bar{j}}{\|\bar{i}\|_2 * \|\bar{j}\|_2} \dots \dots \dots (2)$$

$$\sin(i, j) = \frac{\sum_{u \in U} (R_{u,i} - \bar{R}_u)(R_{u,j} - \bar{R}_u)}{\sqrt{\sum_{u \in U} (R_{u,i} - \bar{R}_u)^2} \sqrt{\sum_{u \in U} (R_{u,j} - \bar{R}_u)^2}} \dots \dots (3)$$

2.3.2. 模式為基的協同過濾

以記憶為基的協同過濾技術的缺點是當資料稀疏時難以處理大資料量，因此便有學者提出以模式為基的協同過濾技術。與記憶式為基的方法相反，模式為基的演算法會由評分矩陣去學習一個模式，並以此來對新項目進行預測。例如 Breese, et al. (1998)提出一個機率式方法(4)。其中 $\text{Pr}(\cdot)$ 項代表的意義是，給定使用者先前所評比過的項目的評分，求得使用者 u 對項目 i 評為 k 分的機率。當然，使用者 u 對項目 i 評分的預測值則可由計算期望值而得。

其它模式為基的演算法尚包括：貝氏模式(Bayesian model)(Chien and George, 1999)、機率關聯模式(probabilistic relational model)(Getoor and Sahami, 1999)、線性迴歸(linear regression) (Sarwar, et al., 2001)、最大熵模式(maximum entropy model)(Pavlov and Pennock, 2002)、及支持向量機(support vector machine)(Xia, et al., 2006)等。惟，這些方法不是未考慮到資料的特性(次序尺度，非名目尺度或區間尺度)，就是純粹從使用者的評分矩陣(未考量評論意見)做處

理。

$$r_{u,i} = E(r_{u,i}) = \sum_{k=0}^n i \times \Pr(r_{u,i} = k | r_{u,i'}, i' \in I_u) \dots \dots (4)$$

3. 研究方法

3.1. 研究架構

我們透過 Cardoso and da Costa (2007) 所提出的 oSVM 和 Chu and Keerthi (2005) 的 SVOR-IM 和 SVOR-EX，以及與 Chang and Lin (2001) 所提出的多類別支持向量機 LIBSVM 方法的比較，利用這四種方法，進行實驗與比較。

而我們評估比較的方法是透過 Mean zero-one 以及平均絕對誤差 (Mean absolute error, MAE)，設 f_i 為程式預測的類別，而 y_i 為資料正確的類別，而 e_i 即為 $|f_i - y_i|$ ，透過平均絕對誤差的公式(5)，將每一項的預測值減正確值取絕對值後的加總，再除以個數，即為平均絕對誤差。

$$MAE = \frac{1}{n} \sum_{i=1}^n |f_i - y_i| = \frac{1}{n} \sum_{i=1}^n |e_i| \dots \dots (5)$$

3.2. 實驗步驟

3.2.1. 獲取資料集

本實驗所採用的原始資料集自於 GroupLens 網站所提供，而我們採取 943 個使用者對 1682 部電影評筆資料集，其原始資料為用戶編號、電影編號、評分、評分時間標記所組成的 10 萬筆資料集，作為我們實驗的最初資料集。

首先我們選出十個評比數最多的評分者來做為十個資料點，然後計算這十個資料點之間的距離，算出那一個類別與大多資料點最接近(也就是重疊率最高)的做目標點(U_T)。

第二步驟將這個目標點的評分者所有評分過的電影當作資料集的輸出屬性，並且以目標點評分者有著一定的相同評比數的評分者的評分作為輸入屬性，如表 1 所示， u_i 代表評分者 i ， m_j 代表電影項目 j ，假設包含目標點和與目標點有一定的相同評比數的評分者有 n 個人，共對 k 部電影

評分，而那些以 "-" 表示評分對此部電影沒有評分，對於這些沒評分的空缺，我們以使用者所有的評分的平均值來填補(mean imputation)，例如 u_1 使用者評分 5 部電影有 2 部沒有評分，所以空缺以 $(4+4+5+3+5)/5=4.2$ ，故 m_2 和 m_5 都以 4.2 來計算。

這些與目標點有 $\alpha\%$ 以上的相同評比率使用者，這個 $\alpha\%$ 我們採用 20%、25%、30%、35% 來做為比較，產生的資料格式如表 2 所示。

表 1. 電影資料集特徵描述

	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	...	U_T
m_1	4	5	4	5	3	5	-	-	...	4
m_2	-	4	-	-	-	-	-	3	...	1
m_3	4	4	5	5	4	5	5	2	...	5
m_4	5	4	5	5	-	-	5	4	...	5
m_5	-	-	-	4	3	-	5	3	...	3
m_6	3	-	3	-	4	4	-	-	...	2
...
m_k	5	4	4	3	4	5	4	3	...	4

表 2. 資料集特徵描述

Movie Lens	Dataset	Instances	Features	Classes
	20%	737	103	5
	25%	737	54	5
	30%	737	29	5
	35%	737	14	5

3.2.2. 資料正規劃至 0~1 區間

資料集的屬性資料值需加以正規化至 0~1 的區間內，採用如(6)的公式，對每一個欄位的資料進行正規化，其 x 為該欄屬性任一點資料， \min 為該欄資料最小值，而 \max 為該欄資料最大值，而 x' 為其正規劃結果，而其結果會在 0~1 之間。

$$x' = \frac{x - \min}{\max - \min} \dots \dots (6)$$

3.2.3. 主要流程

流程說明如下：為使所有方法都用相同的設置，在此我們使用 RBF 的 kernel function，且最佳的 C 和 K 值都統一。

3.3. 實驗結果

由 0-1 誤差¹的比較(表 3)顯示，IM 方

¹ 0-1 誤差即是傳統的多類別預測錯誤率，當預測值與實際值相等，其誤差為 0，否則為 1，並不考慮

法在5個資料集中的表現有3個是最佳的，而LIBSVM則在5個資料集的表現都不好。由平均絕對誤差的比較(表 4)，則可看出IM在五個資料集的表現都是最好的，顯是此方法，即使分錯類別，其分錯類別都是在鄰近，並不會有將第1類分成第5類，這種離譜的錯誤。

表 3. Mean zero-one comparison among oSVM, IM, EX, LIBSVM

	oSVM	IM	EX	LIBSVM
15%	0.5414	0.3604	0.3649	0.6306*
20%	0.5287	0.3511	0.3629	0.6250*
25%	0.4835	0.3574	0.3744	0.6166*
30%	0.4713	0.3912	0.3744	0.6072*
35%	0.4044	0.4150	0.3709	0.5394*

(表中最好的數值用粗體顯示，最差則加註*號)

表 4. Mean absolute error comparison among oSVM, IM, EX, LIBSVM

	oSVM	IM	EX	LIBSVM
15%	0.9089*	0.8041	0.8881	0.9057
20%	0.9495*	0.7758	0.8645	0.9216
25%	0.8623	0.7914	0.9200	0.944*
30%	0.9047	0.8264	0.8899	0.9648*
35%	0.9674	0.9206	0.9487	1.1047*

(表中最好的數值用粗體顯示，最差則加註*號)

4. 結論與未來研究方向

綜合而言，次序類別是介於名目尺度(nominal scale)與區間尺度(interval)之間，其類別間具有次序關係，但卻沒有等距的性質，因此比較難以處理。由比較的三個次序性支持向量機與多類別支持向量機在電影推薦的資料集的表現可知，次序性支持向量機的確較多類別支持向量機能處理次序性資料的預測。而其中SVOR-IM的表現是最佳的，此結論的另一層意義是發表在研討會的方法(尤其是頂尖研討會)可能會優於期刊中所提之方法。

由於本實驗只有依靠一份資料集，其可靠性仍需加強，未來應多增加實驗的資料集，來預測並且做比較。

預測值與實際值距離的遠近。

致謝

本研究為國科會計畫(NSC 99-2410-H-251-014)補助之成果，特此致謝。

參考文獻

- [1] Adomavicius, G., Tuzhilin, A. (2005). "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 17(6), pp. 734-749.
- [2] Breese, J.S., Heckerman, D., and Kadie, C. (1998). "Empirical Analysis of Predictive Algorithms for Collaborative Filtering," *Proc. 14th Conf. Uncertainty in Artificial Intelligence*.
- [3] Cardoso, J. S., and da Costa, J. F. P. (2007), "Learning to classify ordinal data: The data replication method," *Journal of Machine Learning Research*, Vol. 8, pp. 1393-1429.
- [4] Chang, C.C. and Lin, C.J. (2001). "LIBSVM: a Library for Support Vector Machines," <http://www.csie.ntu.edu.tw/~cjlin/libsvm>
- [5] Chien, Y.-H. and George, E.I. (1999). "A Bayesian Model for Collaborative Filtering," *Proc. Seventh Int'l Workshop Artificial Intelligence and Statistics*.
- [6] Chu, W. and Keerthi, S. S. (2005), "New approaches to support vector ordinal regression", in *Proc. of International Conference on Machine Learning (ICML-22)*:145-152.
- [7] Getoor, L. and Sahami, M., (1999). "Using Probabilistic Relational Models for Collaborative Filtering," *Proc. Workshop Web Usage Analysis and User Profiling (WEBKDD '99)*.
- [8] Keerthi, S.S., Shevade, S.K., Bhattacharyya, C., Murthy, K.R.K. (2001). "Improvements to Platt's SMO Algorithm for SVM Classifier Design," *Neural Computation*. 13(3):637-649
- [9] Pavlov, D. and Pennock, D. (2002). "A Maximum Entropy Approach to Collaborative Filtering in Dynamic,

- Sparse, High-Dimensional Domains," Proc. 16th Ann. Conf. Neural Information Processing Systems (NIPS '02).
- [10] Sarwar, B., Karypis, G., Konstan, J., and Riedl, J., (2001). "Item-Based Collaborative Filtering Recommendation Algorithms," Proc. 10th Int'l WWW Conf.
- [11] Tang, H. and Qu. L.S (2008), "Fuzzy support vector machine with a new fuzzy membership function for pattern classification", *Proceedings of the Seventh International Conference on Machine Learning and Cybernetics*, Kunming, 12-15 July 2008.
- [12] Xia, Z., Dong, Y., and Xing, G. (2006). "Support vector machines for collaborative filtering," ACM-SE 44: Proceedings of the 44th annual southeast regional conference, pp. 169-174.