以密度為基礎的微聚合方法

溫宗賢　　林志麟　　汪嘉翔
元智大學資管所碩士生　元智大學資管所助理教授　元智大學資管所碩士生
s966211@mail.yzu.edu.tw　jun@saturn.edu.tw　terrorist0111@gmail.com

摘要

微聚合(microaggregation)是廣泛被採用的統計洩漏控制技術，防止統計資料庫中公開發行的微資料(microdata)被用來找出個別資料的真實身分。微聚合將牽涉到隱私及機密的微資料集分成大小不少於 k 的若干群組，之後在釋出這些資料前，每一群的資料會被所屬群組的中心點所取代，以達到保護資料隱私的效果。然而，為確保處理後資料集的可用性，有效的微聚合技術應降低此一處理所造成的資訊損失。本文提出兩個根據資料密度為基礎來進行微聚合的演算法，並透過實驗評估這兩個演算法的效能。

關鍵詞：微聚合，統計洩漏控制，微資料保護

Abstract

Microaggregation is a statistical disclosure control (SDC) technique that has been widely used to protect publicly released microdata from individual identification. It works by dividing the privacy-related microdata set into groups of size no less than k, and prior to releasing the dataset, each record is replaced by the centroid of the group to which this record belongs. However, to ensure the quality of the anonymized dataset, an effective microaggregation method must try to minimize the information loss caused the anonymization process. This work proposes two density-based methods for microaggregation, and studies the performances of both methods via experiments.

Keywords: Microaggregation, statistical disclosure control, microdata protection

1. 前言

隨著資訊科技的急速演進，大量的資料得以迅速地搜集、儲存及傳遞。透過大型資料庫的建構與資料探勘(data mining)技術的結合，吾人可以探究隱藏在資料背後的知識及資訊。然而，此類應用卻也很可能洩漏了個人隱私資料[1]。Statistical disclosure control(以下簡稱 SDC)[10][12]是目前常用於保護個人資料隱私的一種作法。統計資料庫在將資料發行供分析之用以前，會先進行 SDC 處理 ，以確保隱私資料的安全性。

SDC 技術可應用在表格資料(tabular data)、動態式查詢資料庫(dynamically queryable database)及微資料(microdata)等三種形式的資料[4]，而本文所關注的是微資料。微資料集是紀錄(record)或是資料向量(data vector)的集合，它涵蓋的個體資料範疇包含了個人或公司等[5]。微資料集中的個體 h 被視為一個資料向量 x_h ，由鑑別屬性(identifiers)、機密屬性(confidential outcome attributes)及關鍵屬性(key attributes)三類屬性所構成[14]。鑑別屬性可以直接辨識出一個個體，例如：學號、公司統一編號等，因此，在前處裡時就必須將這些欄位移除。機密屬性包含了個體敏感的資訊，例如：薪資。剩下的其他屬性視為關鍵屬性，透過關鍵屬性的組合並連結外部資訊，例如：醫療資料連結選舉名冊，可間接辨識出個體[13]。注意，任一屬性並無強制歸類於某一特定屬性，而是依照不同的問題有不同之定義。

沒有運用 SDC 技術，原始資料等同被直接使用，相關的隱私及機密未加以保護，毫無安全可言。近年來，有鑒於人們對個人隱私意識高漲，SDC 保護技術的使用似乎有勢在必行的趨勢。依據文獻[6][7][18]，SDC 保護技術大都須對原始資料做某種程度的更改或遮蔽，因此在更改原始資料的情況下，如何在防止隱私及機密洩露的同時，將資訊損失(information loss)降到可接受的程度，是 SDC 技術所面臨的重要課題，如圖 1 所示。

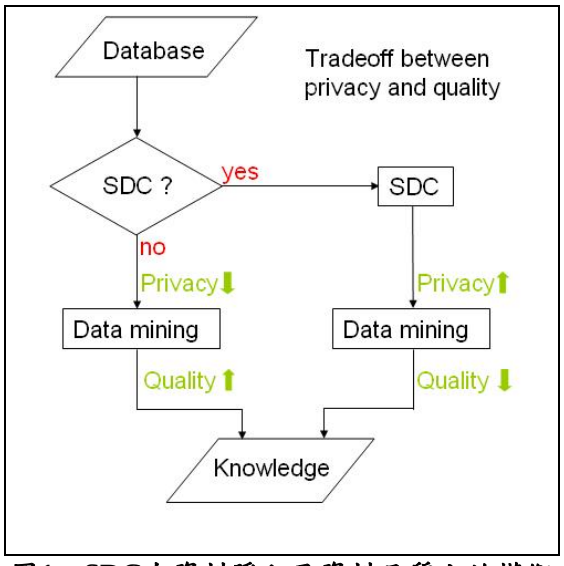

圖1　SDC在資料隱私及資料品質上的權衡

微聚合(microaggregation)是一種常用之SDC保護技術。它的作法是先將資料分群，使每一群都至少有k筆資料，然後計算出每一群記錄的中心點，最後以各群的中心點來取代該群內的記錄。由於處理後的資料中，每一筆紀錄至少會和其他$k-1$個筆紀錄相同，滿足了所謂k匿名[13][17]的要求，因此較難透過與外部資料庫的連結來辨識個體。但是透過中心點來遮蔽並取代原始資料以防止隱私及機密洩露，卻也付出了資訊損失的代價。因此，一個好的微聚合技術應增加群體內的紀錄同質性以降低資訊損失。此一目的和傳統的分群(clustering)技術不謀而合，然而傳統的分群技術並不會限制每一群的資料筆數下限k。文獻[5]證明，一個最佳的微聚合技術所產生的分群結果中，每一群所包含的資料筆數一定不會超過$2k-1$。

本研究提出根據資料密度的順序來進行微聚合的作法，並評估其效能。本文其餘的內容架構如下：第二節回顧微聚合的基本概念及相關文獻；第三節介紹本文所提出的以密度為基礎微聚合方法(density-base microaggregation)；第四節討論實驗結果；最後，第五節總結本文。

2.微聚合

在過去幾年，由於人們對隱私相關議題越來越關注，因此有許多方法及研究投入在微聚合及其相關領域。本節首先回顧不同的微聚合方法，然後說明微聚合概念及評估其效能的資訊損失(information loss)計算方式。

2.1 微聚合文獻回顧

微聚合本身是針對數值型欄位(numerical attribute)的SDC技術，而Domingo-Ferrer及Torra[4]已將其延伸應用在順序型欄位(ordinal attribute)及類別型欄位(categorical attribute)。微聚合問題依據其資料向量維度可分為：

- 單變數微聚合(univariate microaggregation)。單變數微聚合代表所處理的資料向量集合$X=R^p=\{x_1,x_2,...,x_n\}$、$p=1$。此類問題的最佳解可用文獻[8]所提出的方法找出。
- 多變數微聚合(multivariate microaggregation)。多變數微聚合代表所處理的資料向量集合$X=R^p=\{x_1,x_2,...,x_n\}$、$p>1$，此問題已被証明是個NP-hard問題[12]。因此目前多數的方法是為了解決多變數微聚合問題。

鑒於每個群體要有不小於k個的資料向量，因此微聚合方法可依據每個群體內資料向量個數是否固定分為兩大類：

- 固定大小微聚合(fixed-size microaggregation)。X為資料向量的集合，若$|X|=n$能被k整除，則會產生n/k個資料向量個數為k的群體；若n無法被k整除，則固定大小的微聚合會產生$\lfloor n/k \rfloor$個資料向量個數為k的群體，而剩下$n\%k$個資料向量則分配給先前的$\lfloor n/k \rfloor$個群體，使最多$k-1$個群體其資料向量個數會介於k到$2k-1$間。
- 變動大小微聚合(variable-size microaggregation)。透過變動大小微聚合方法，每個群體的資料向量個數會介於k到$2k-1$間，因此對於資料的聚合有較高的彈性，也較符合資料導向(data oriented)，但相對的，變動大小微聚合方法通常也相對複雜。

典型固定大小微聚合不外乎是利用「最遠距離」(maximum distance)的概念找出某一起始資料向量，及其最近的$k-1$個資料向量，以構成大小為k之群體。以CBFS(centroid-based fixed-size)微聚合演算法[6][9]為例：CBFS不斷從未被分配之資料向量中更新中心點$\tilde{x}$，並從未被分配之資料向量中，找出距中心最遠的點x_r及$k-1$個距x_r最近點，以構成大小為k之群體；而MDAV(maximum distance to average vector)微聚合演算法[4]類似CBFS，不僅找出x_r，還另找一個距x_r最遠的點x_s，及距x_s最近的$k-1$個點；文獻[5][9]使用了

DBFS(diameter-based fixed-size) 微聚合，DBFS 不斷的從未被分配之資料向量中找兩個相距最遠的點 x_r 和 x_s，及該兩點最近之 $k-1$ 點，以構成大小為 k 之兩個群體；在文獻[3]中，Chang 等人為了降低時間複雜度，根據資料向量的分佈，決定出兩個固定虛擬極端點 x'_r 及 x''_r，並經由不斷的交替使用這兩個虛擬極端點，從未被分配之資料向量中，找出距極端點最遠的點 x_r 及距 x_r 最接近的 $k-1$ 點，以構成大小為 k 之群體。

有別於固定大小微聚合技術常使用最遠距離的概念，變動大小微聚合技術使用許多不同的想法。 Solanas 及 Martínez-Ballesté 提出改善自 MDAV 方法的 V-MDAV[15]。以 d_{in} 代表已分配資料向量到未分配資料向量中最近的距離，d_{out} 代表前述未分配資料向量 x_r 到其他未被分配資料向量中最近的距離。V-MDAV 的作法是，當 $d_{in} < \lambda d_{out}$ 時，便將 x_r 分配至前述已分配資料向量所屬群體，其中 λ 為使用者自定的權重。Solanas 等[16]使用了基因演算法(genetic algorithm)處理較小資料集的微聚合。但為了改善基因演算法只能用在較小資料集的限制，Martínez-Ballesté 等[11]利用「分割與擊破」(divide and conquer)概念，先以 MDAV 演算法把資料集微聚合成適合基因演算法的大小，再使用基因演算法做細部的調整。Laszlo 及 Mukherjee[9]把每一個資料向量以最小擴張樹(minimum spanning tree)連結，並對邊線(edge)的長度依遞減排序，最後依順序嘗試移除邊線，以切出符合微聚合的森林。Domingo-Ferrer 及 Mateo-Sanz[5]改變 Ward's Method[19]的合併條件，限制群體與群體的合併，以滿足微聚合。Domingo-Ferrer 等[6]採 NPN(nearest point next)的概念在群體層次(group-level)及資料層次(data-level)做排序，並將排序後結果用在單變量微聚合方法[8]上。

2.2 微聚合概念

微聚合本質為將資料集 X 轉換成匿名化版本 X' 的過程。如圖 2 所示，假定 X 是大小為 n，維度為 p 的數值型資料集；$G=\{G_1,G_2,...,G_g\}$ 是各群體之集合(g 代表群體的個數)，$G_{i\in[1,g]}$ 分割自資料集 X，$\bigcup_{i\in[1,p]} G_i = X$，且在任意 $\hat{i} \neq \check{i}$ 的情況下 $G_{\hat{i}} \cap G_{\check{i}} = \{\ \}$，並要求 $n_{i\in[1,g]} \geq k$，意即每一個資料向量必須且只能屬於一個群體，且各群體所含資料向量個數 $n_{i\in[1,g]}$ 必須有 k 個以上；最後每個群體所含的資料向量由該組中心點 $\bar{x}_i$ 取代(群體內所有資料向量之算術平均數)，以達成 X 的匿名化版本 X'。

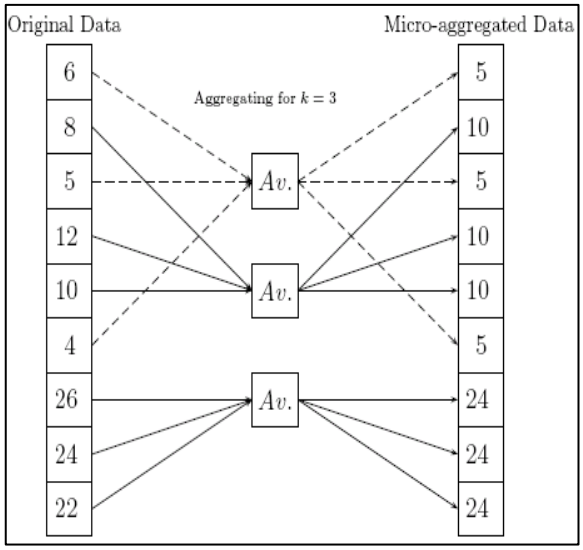

圖2　*k*=3之單變量微聚合[14]

雖然資料集 X 易於達成微聚合要求，但為增加資料集 X' 的品質，群體內資料向量的同質性(homogeneity)必須高，而同質性可經由誤差平方和(Sum of Squares Error, SSE)做定量描述，其定義如下：

$$SSE = \sum_{i=1}^{g} SE(G_i) = \sum_{i=1}^{g}\sum_{j=1}^{n_i} \left(x_{ij} - \bar{x}_i\right)'\left(x_{ij} - \bar{x}_i\right) \quad (1)$$

x_{ij} 代表在第 i 個群體中第 j 個資料相量，$SE(G_i)$ 為第 i 個群體內各資料向量對其中心點的平方差，SSE 為各群體平方差加總，SSE 較大，代表匿名後資料集 X' 各群體的同質性低，反之則同質性高，而最佳化的微聚合代表擁有最小的 SSE 及最小的資訊損失(information loss, IL)；資訊損失為 SSE 更一般性描述，並透過 SST 將數值標準化在 0 到 1 之間，其定義如下：

$$IL = \frac{SSE}{SST} \quad (2)$$

$$SST = \sum_{i=1}^{g}\sum_{j=1}^{n_i} \left(x_{ij} - \bar{x}\right)\left(x_{ij} - \bar{x}\right) \quad (3)$$

$\bar{x}$ 為所有資料向量的中心點(所有資料向量之算術平均數)，SST(Sum of Square total)為總誤

差平方和。

3. Density Base Algorithms

本文所提出的「以密度為基礎的微聚合演算法」屬於固定大小的微聚合方法，意即每一群體在產生過程中所含資料向量個數為 k；而「密度」的概念等同任一點資料向量 x_c (起始點,initial point)及其最近 $k-1$ 點在群體內的平方差，以 $Density(G_c)$ 表示。當 $Density(G_c)$ 較大時稱「低密度」，反之則稱「高密度」。$Density(G_c)$ 公式如下：

$$Density(G_c)=\sum_{j=1}^{k}\left(x_{c_j}-\bar{x}_c\right)'\left(x_{c_j}-\bar{x}_c\right) \qquad (4)$$

x_{c_j} 為距 x_c 第 j 近的資料向量(x_{c_1} 即為資料向量 x_c)，$\bar{x}_c$ 為 x_c 及其最近 $k-1$ 點的算術平均數。

依據不同的「候選群體建立順序」，本研究之演算法可分為「低密度優先」及「高密度優先」,分述如下之 Algorithm 1 與 Algorithm 2。

Algorithm 1: Low-Density First Algorithm

Input: Data set X, Integer k, Integer n.
Result: *k*-partition.

1. $RR = n$; // Remaining Records in X
2. $i = 0$;
3. **while** $RR \geq k$ **do**
4. x_c =InitialPointWithLowestDensity(X , k);
5. G_i =Group(x_c ,Around(x_c , $k-1$));
6. $X = X$ - x_c -Around(x_c , $k-1$);
7. $i = i+1$;
8. $RR = RR-k$;
9. **end**
10. $G_1...G_{i-1}$ = AssignRest($X, G_1...G_{i-1}$);
11. **return** $G_1...G_{i-1}$;

透過 Algorithm 1(低密度優先)及 Algorithm 2(高密度優先)，在每次的迭代都會建立一個大小為 k 的群體。Algorithm 1 會從候選群體中找尋密度最低的，而 Algorithm 2 會從候選群體中找尋密度最高的。

在迭代過程中，首先將資料集 X 內所有資料向量都各別視為一「起始點」，並分別找其最近的 $k-1$ 點，以建立候選群體，隨後將所有候選群體中密度最小(大)群體之起始點 x_c 傳回(4 行)，下一步，再將 x_c 及其最鄰近 $k-1$ 個點加入 G_i (5 行)，最後更新資料集 X 、群體編號 i 、以及未被分派的資料個數 RR (6-8 行)，而一直到未被分配的資料向量小於 k 個時才跳離迴圈；在演算法的最後，若 $X \neq \{\}$ 則必須將剩餘的資料向量分配給最接近的群體(10 行)。

Algorithm 2: High-Density First Algorithm

Input: Data set X, Integer k, Integer n.
Result: *k*-partition.

1. $RR = n$; // Remaining Records in X
2. $i = 0$;
3. **while** $RR \geq k$ **do**
4. x_c =InitialPointWithLargestDensity(X , k);
5. G_i =Group(x_c ,Around(x_c , $k-1$));
6. $X = X$ - x_c -Around(x_c , $k-1$);
7. $i = i+1$;
8. $RR = RR-k$;
9. **end**
10. $G_1...G_{i-1}$ = AssignRest($X, G_1...G_{i-1}$);
11. **return** $G_1...G_{i-1}$;

4. 實驗結果

本節測試「低密度優先」及「高密度優先」微聚合方法，所有的實驗都在 Intel Core2 Duo 2.2GHz CPU 及 2GB RAM 的 MS Windows XP 作業系統底下運作。

本實驗使用 Tarragona、Census 及 EIA 等三個常用於評估微聚合技術效能的資料集[2]，Tarragona 及 Census 資料集有 13 個欄位(維度)，並分別有 843 及 1080 筆紀錄(資料向量)，而 EIA 有 4092 筆紀錄及 15 個欄位(*UTILITYID*、*UTILNAME*、*STATE*、*YEAR*、*MONTH*、*RESREVENUE*、*RESSALES*、*COMREVENUE*、*COMSALES*、*INDREVENUE*、*INDSALES*、*OTHREVENUE*、*OTHRSALES*、*TOTREVENUE*、*TOTSALES*)，其中 *UTILNAME*、*STATE*、*YEAR* 及 *MONTH* 等 4 個欄位則會事先移除。

本實驗測試上述資料集，在不同 k ($k = 3,4,5,10$)狀況下的資訊損失情況。實驗

結果如表 1 所示，可以看出「低密度優先」較「高密度優先」通常有較佳的表現，但在某些情況下卻是「高密度優先」較佳。因此，兩種方法都值得做進一步的探索。

表1　LDF及HDF微聚合演算法的資訊損失量

單位：IL*100%		*k*=3	*k* =4	*k* =5	*k*=10
EIA	HDF	1.09	0.84	1.9	4.27
	LDF	0.76	1.10	2.17	4.17
Census	HDF	6.14	9.13	10.84	15.79
	LDF	6.46	8.49	10.12	15.93
Tarragona	HDF	20.7	23.83	26	35.39
	LDF	17.15	19.44	23.25	33.49

5. 結論

本研究利用「密度」的概念，提出新的固定大小微聚合方法。這兩個演算法雖然在某些狀況下有較好的表現，但在部分情況下仍表現不夠理想。後續研究將朝更符合資料導向(data-oriented)來修改這兩個演算法，以進一步降低資訊損失。

參考文獻

[1] Atallah, M., Bertino, E., Elmagarmid, A., Ibrahim, M. and Verykios, V., "Disclosure Limitation of Sensitive Rules," ***In: Proceedings of 1999 Workshop on Knowledge and Data Engineering Exchange***, Chicago, IL, pp. 45-52, 1999.

[2] Brand, R., Domingo-Ferrer, J. and Mateo-Sanz, J.M., "Reference Data Sets to Test and Compare SDC Methods for Protection of Numerical Microdata," ***European Project IST-2000-25069 CASC***, http://neon.vb.cbs.nl/casc, 2002.

[3] Chang, C.-C., Li, Y.-C. and Huang, W.-H., "TFRP: An Efficient Microaggregation Algorithm for Statistical Disclosure Control," ***Journal of Systems and Software***, Vol. 80, No. 11, pp. 1866-1878, 2007.

[4] Domingo-Ferrer, J. and Torra, V., "Ordinal, Continuous and Heterogenerous K-anonymity through Microaggregation," ***Data Mining and Knowledge Discovery***, Vol. 11, No. 2, pp. 195-212, 2005.

[5] Domingo-Ferrer, J. and Mateo-Sanz, J.M., "Practical Data-Oriented Microaggregation for Statistical Disclosure Control," ***IEEE Transactions on Knowledge and Data Engineering***, Vol. 14, No. 1, pp. 189-201, 2002.

[6] Domingo-Ferrer, J., Martínez-Ballesté, A., Mateo-Sanz, J.M. and Sebé, F., "Efficient Multivariate Data-Oriented Microaggregation," ***VLDB Journal***, Vol. 15, No. 4, pp. 355-369, 2006.

[7] Doyle, P., Lane, J.I., Theeuwes, J.J. and Zayatz, L.V., ***Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies***, North-Holland, Amsterdam, 2001.

[8] Hansen, S.L. and Mukherjee, S., "A Polynomial Algorithm for Optimal Univariate Microaggregation," ***IEEE Transactions on Knowledge and Data Engineering***, Vol. 15, No. 4, pp. 1043-1044, 2003.

[9] Laszlo, M. and Mukherjee, S., "Minimum Spanning Tree Partitioning Algorithm for Microaggregation," ***IEEE Transactions on Knowledge and Data Engineering***, Vol. 17, No. 7, pp. 902-911, 2005.

[10] Liew, C.K., Choi, U.J. and Liew, C.J., "Data Distortion by Probability Distribution," ***ACM Transactions on Database Systems***, Vol. 10, No. 3, pp. 395-411, 1995.

[11] Martínez-Ballesté, A., Solanas, A., Domingo-Ferrer, J. and Mateo-Sanz, J.M., "A Genetic Approach to Multivariate Microaggregation for Database Privacy," ***2007 23rd International Conference on Data Engineering Workshop***, pp. 180-185, 2007.

[12] Oganian, A. and Domingo-Ferrer, J., "On the Complexity of Optimal Microaggregation for Statistical Disclosure Control," ***Statistical Journal of United Nations Economic Commission for Europe***, Vol. 18, No. 4, pp. 345-354, 2001.

[13] Samarati, P., "Protecting Respondents' Identities in Microdata Release," ***IEEE Transactions on Knowledge and Data Engineering***, Vol. 13, No. 6, pp. 1010-1027, 2001.

[14] Solanas, A., Sebé, F. and Domingo-Ferrer, J., "Micro-aggregation-Based Heuristics for p-sensitive k-anonymity: One Step Beyond," ***IEEE Transactions on Knowledge and Data Engineering***, Nantes, France, pp. 61-69, 2008.

[15] Solanas, A. and Martínez-Ballesté, A., "V-MDAV: A Multivariate Microaggregation with Variable Group Size," ***COMPSTAT 2006***, pp. 917-925, 2006.

[16] Solanas, A., Martínez-Ballesté, A., Mateo-Sanz, J. M. and Domingo-Ferrer, J., "Multivariate Microaggregation Based

Genetic Algorithms," ***IEEE Conference on Intelligent Systems***, pp. 65-70, 2006 .

[17] Sweeney, L., "K-anonymity: A Model for Protecting Privacy," ***International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems***, Vol. 10, No. 5, pp. 557-570, 2002.

[18] Willenborg, L. and DeWaal, T., ***Elements of Statistical Disclosure Control***, Springer. Berlin Heidelberg New York, 2001.

[19] Ward, J., "Hierarchical Grouping to Optimize an Objective Function," ***J. Am. Statistical Assoc.***, Vol. 58, pp. 236-244, 1963.