# Métodos de microagregación para $k$-anonimato: privacidad en bases de datos

Agusti Solanas, Antoni Martínez-Ballesté, Josep Domingo-Ferrer,
Susana Bujalance y Josep M. Mateo-Sanz

Dept. Enginyeria Informàtica i Matemàtiques,
Universitat Rovira i Virgili,
Av. Països Catalans 26,
E-43007 Tarragona, Catalonia, Spain
e-mail {agusti.solanas,antoni.martinez,josep.domingo,
josepmaria.mateo}@urv.cat, susana.bujalance@estudiants.urv.cat

**Resumen** El Control de Revelación Estadística es una disciplina de protección de datos cuyo objetivo es impedir que la identidad de los respondedores quede revelada cuando terceras partes tienen acceso a grandes ficheros de estadísticas. En este artículo presentamos un conjunto de métodos para $k$-anonimato en bases de datos estadísticas usando microagregación. Los métodos son analizados y comparados y se desarrolla el análisis computacional de cada uno.

## 1. Introducción

El uso masivo de los ordenadores para tratar y distribuir información electrónica ha propiciado la aparición de métodos para gestionar esta información, sobre todo cuando ésta debe ser tratada por terceras partes. En este sentido, la privacidad de los individuos debe ser preservada cuando sus datos son usados en campos como la medicina o la economía.

Con este fin, se han desarrollado varias técnicas para la preservación de la privacidad en bases de datos, conocidas como Control de Revelación Estadística (*Statistical Disclosure Control*, SDC). En este artículo estudiamos diversos métodos de SDC, prestando especial atención a dos conceptos relacionados con la privacidad en las bases de datos: el $k$-anonimato y la microagregación. Proteger la información individual de los individuos cuyos datos contiene la base de datos no es sencillo puesto que los datos deben ser publicados o transferidos telemáticamente de forma que el receptor no sea capaz de revelar su identidad.

Dentro de los distintos métodos de SDC se encuentran los métodos *perturbadores*, los cuales modifican los datos con el fin de dificultar el uso malintencionado de los datos. El $k$-anonimato es una propiedad

relacionada con la protección de la privacidad en bases de datos y la *microagregación* es un método perturbador con el cual se puede obtener el $k$-anonimato.

### 1.1. El $k$-anonimato y la microagregación

Definimos $X$ como un conjunto de datos protegido, donde los identificadores han sido borrados. Dado un subconjunto de atributos (conocidos como cuasi-identificadores) de $X$, el cual es conocido por un intruso gracias a fuentes externas de identificadores $X$' (e.g. guías telefónicas, censos electorales, etc.), se dice que el conjunto de datos protegido $X$ es $k$-anónimo si, como mínimo, existen registros en $X$ que comparten cualquier combinación de valores cuasi-identificadores. De este modo, lo mejor que el intruso puede conseguir es asociar un registro de $X'$ *con un conjunto de* $k$ *registros de* $X$. Un ejemplo ilustrativo de la situación anterior, es el caso hipotético en que un intruso quisiera conocer la identidad de una persona cuyo teléfono conoce gracias a la guía telefónica e intenta encontrar en una base de datos $X$ información sobre dicha persona. Sin embargo se encuentra con la dificultad de que $X$ es 3-anónima puesto que 3 personas en $X$ comparten ese número de teléfono.

El procedimiento computacional que inicialmente se propuso para implementar los métodos de $k$-anonimato estaba basado en la supresión/generación de valores de atributos cuasi-identificadores. Recientemente, se ha visto que microagregando los cuasi-identificadores se puede conseguir el mismo resultado sin necesidad de suprimir parte de los datos ni añadir datos de más [12]. La microagregación se ha usado durante varios años en diversos países: empezó en la agencia Eurostat [13] sobre los años noventa y desde entonces se usa en Alemania [14] y otros países [15]. La microagregación no sólo es relevante para SDC, sino que también lo es en inteligencia artificial [16]. En este último campo, su aplicación consiste en incrementar el grado de *conocimiento* de un sistema de toma de decisiones y representación de dominios. Las técnicas de microagregación pueden usarse también en minería de datos con el propósito de reducir o comprimir un conjunto de datos minimizando la pérdida de información.

La microagregación satisface la condición de $k$-anonimato mediante la agrupación ($k$-partición) de registros de un conjunto de datos en grupos de cómo mínimo $k$ registros. En el fichero protegido, cada registro es reemplazado por el centroide de su grupo. Con objetivo de minimizar la pérdida de información causada por la microagregación, los grupos deben

construirse de forma que la homogeneidad del grupo sea máxima. Existen diversas medidas de homogenización en la literatura, basadas en diferentes definiciones de distancia (e.g. distancia Euclidiana, distancia Minkowski, distancia de Chebyshev, etc.). La medida de homogeneidad más común para hacer agrupaciones es la SEC, o *suma de errores cuadráticos* [17][18][19][20][21][22][23]. SEC consiste en la suma del cuadrado de las distancias entre el centroide de un grupo y cada uno de los registros que pertenecen al grupo. Dada una $k$-partición,

$$SEC = \sum_{i=1}^{s} \sum_{j=1}^{n_i} (\mathbf{x}_{ij} - \bar{\mathbf{x}}_i)'(\mathbf{x}_{ij} - \bar{\mathbf{x}}_i) \tag{1}$$

donde es el número de grupos en la $k$-partición y ni es el número de registros en el i-ésimo grupo. La expresión (1) se calcula sobre los datos después de estandarizarlos, es decir, después de restar a los valores de cada atributo al promedio del atributo y dividir el resultado por su desviación típica.

Una $k$-partición es óptima cuando su SEC es mínimo. En [24] se demuestra que encontrarla es NP-hard. En consecuencia, los únicos métodos prácticos de microagregación son heurísticos. En [22] se muestra que los grupos con una $k$-partición óptima contienen entre $k$ y $2k - 1$ registros.

### 1.2. Objetivo del artículo

En este artículo hacemos un análisis de distintos métodos perturbadores de microagregación. El análisis se centra en la complejidad computacional y en la homogeneidad de los grupos en los datos microagregados. En las secciones 2, 3 y 4 se explican distintos métodos y se muestra su análisis computacional. En la sección 5 comparamos los métodos así como la homogeneidad en los datos microagregados. En la sección 6, concluimos el artículo.

## 2. El método de Distancia Máxima

El método de Distancia Máxima (*Maximum Distance*, MD) se propuso en [22] como un método de microagregación multivariante. A continuación se describe el algoritmo y se determina su complejidad computacional.

### 2.1. El algoritmo

El algoritmo MD construye una $k$-partición de la siguiente manera:

1. Hallar $r$ y $b$, los dos registros más distantes en el conjunto de datos, usando la distancia Euclidiana; formar un grupo con $r$ y los $k$-1 registros más cercanos a $r$; formar un grupo con $s$ y los $k$-1 registros más cercanos a $s$.
2. Si hay al menos $2k$ registros que no pertenecen a ninguno de los grupos formados en el paso 1, volver al paso 1 considerando como nuevo conjunto de datos al conjunto de datos original menos los grupos formados en el paso 1.
3. Si hay entre $k$ y $2k - 1$ registros que no pertenezcan a ningún grupo formado en el paso 1, formar un nuevo grupo con dichos registros y finalizar.
4. Si hay menos de $k$ registros que no pertenezcan a ninguno de los grupos formados en el paso 1, añadir cada uno al grupo más cercano.

En este momento, se tiene una $k$-partición del conjunto de datos con una serie de grupos. Tal como se ha explicado anteriormente, dada una $k$-partición, los datos microagregados se consiguen reemplazando cada registro por el centroide del grupo al que pertenece.

### 2.2. Coste computacional

La dimensión del conjunto de datos (cantidad de atributos) no se considerará en el análisis a causa de que éste es comúnmente mucho menor que $n$ y sólo supone un leve coste computacional sobre el cálculo de la distancia. Se supone que $2k$ es divisor de $n$, puesto que el hecho de realizar los pasos 3 y 4 supone un coste despreciable. Para un conjunto de datos con $n$ puntos, la complejidad de computar una $k$-partición usando MD es el coste de formar $\lfloor n/k \rfloor$ grupos de $k$ puntos, dos por iteración.

La complejidad de una iteración puede dividirse en:

- Encontrar los dos puntos más distantes entre los puntos restantes que no pertenecen a ningún grupo. Si tenemos un promedio de $n/2$ puntos aún sin agrupar, se requiere computar una matriz triangular superior, de media $n/2$ filas, esto es, $(n/2)\frac{n/2-1}{2} = \frac{n(n-2)}{8}$ cálculos de distancia.
- Formar 2 grupos con $k$-1 puntos cercanos a cada uno de los puntos formados en el punto anterior. Para formar un grupo, debemos calcular una fila de la matriz de la distancia con un promedio de $n/2$ columnas e identificar las columnas con las distancias mínimas, lo

que supone $((k-1)n)/2$ cálculos. En consecuencia, formar dos grupos cuesta $(k-1)n$ operaciones.

Por lo tanto, como son $\lfloor n/2k \rfloor$ iteraciones, la computación de $k$-particiones necesita $O(n^3/k)$ operaciones.

## 3. El método de Distancia Máxima al Vector Promedio

El principal problema de MD es su complejidad computacional debida al coste de encontrar los registros más lejanos en cada iteración. El método de la distancia máxima al vector promedio (*Maximum Distance to Average Vector*, MDAV) mejora a MD en términos de complejidad computacional, manteniendo la calidad en cuanto al SEC resultante. MDAV se propuso en [12], [25] como parte del método de microagregación multivariante implementado en el paquete $\mu$-Argus para el control de revelación estadística. También aparece ligeramente modificado en [10].

### 3.1. El algoritmo

El algoritmo del método MDAV para construir una $k$-partición es el siguiente:

1. Calcular el centroide (registro promedio) de todos los registros del conjunto de datos. Buscar el registro más lejano $r$ al centroide. Encontrar el registro $s$ más lejano a $r$.
2. Formar dos grupos alrededor de $r$ y $s$: el primero contiene a $r$ y al grupo de $k-1$ elementos más cercanos a $r$; el otro contiene a $s$ y a los $k-1$ registros más cercanos a $s$.
3. Si hay como mínimo $2k$ registros que no pertenezcan a ninguno de los grupos formados en el paso anterior, volver al paso 1 tomando como nuevo conjunto de puntos al conjunto resultante de restar al conjunto original todos aquellos registros que ya han sido agrupados en el paso 2.
4. Si el número de registros sin agrupar en el paso 2 es entre $k$ y $2k-1$, formar un nuevo grupo con estos registros y finalizar el algoritmo.
5. Si el número de registros sin agrupar en el paso 2 es menos de $k$, añadir cada uno al grupo más cercano.

En este momento, se tiene una $k$-partición del conjunto de datos en una serie de grupos. Se procede de la misma manera que con MD para obtener los datos microagregados.

### 3.2. Coste computacional

La complejidad computacional de MDAV es muy inferior al coste que supone ejecutar MD sobre un conjunto de $n$ registros, concretamente, su coste es $O(n^2)$. De este modo, MDAV puede microagregar conjuntos que contengan 10000 registros en pocos segundos. Al igual que en MD, se supone que $2k$ divide $n$.

MDAV consiste en formar $\lfloor n/k \rfloor$ grupos de $k$ puntos, dos en cada iteración. Asumiendo que hay una media de $n/2$ registros no agrupados, la complejidad de formar un grupo se puede dividir en:

- Calcular el centroide de los registros no agrupados.
- Encontrar el registro más lejano $r$ al centroide, lo cual requiere $n/2$ cálculos de distancia.
- Encontrar el registro más lejano $s$ a $r$, lo cual requiere $n/2-1$ cálculos de la distancia.
- Se forman dos grupos, uno alrededor de $r$ y otro alrededor de $s$, para los cuales, igual que en MD, se necesitan $O((k-1)n)$ operaciones.

En consecuencia, como se realizarán $\lfloor n/2k \rfloor$ iteraciones, el cálculo de la $k$-partición necesita $O(n^2)$ operaciones.

## 4. MDAV de tamaño variable

MDAV genera grupos de tamaño fijo $k$, por tanto no tiene la flexibilidad para adaptar el tamaño del grupo a la distribución de los registros del conjunto de datos. Esto resulta en una $k$-partición cuya homogeneidad aún mejorable. MDAV de tamaño variable (*Variable*-MDAV, V-MDAV)[26] es un nuevo algoritmo que intenta solventar estas limitaciones calculando $k$-particiones de tamaño variable con un coste similar al de MDAV.

### 4.1. El algoritmo

El algoritmo utilizado para construir una $k$-partición usando V-MDAV es el siguiente:

1. Calcular las distancias entre registros y almacenarlas en una matriz de distancias.
2. Encontrar el centroide $c$ del conjunto de datos.
3. Mientras haya más de $k-1$ registros que no hayan sido aún asignados a un grupo:

- Buscar el registro más lejano $e$ al centroide $c$
- Formar un grupo alrededor de $e$ que contenga los $k$-1 registros más cercanos a $e$.
- Extender el grupo, lo cual consiste en añadir registros al grupo, siguiendo los siguientes pasos: i) encontrar $e_{min}$, el registro aún no asignado más cercano a cualquiera de los puntos del conjunto y definir $d_{in}$ como la distancia entre $e_{min}$ y el grupo; ii) definir $d_{out}$ como la mínima distancia entre $e_{min}$ y los demás registros sin asignar; iii) si $d_{in} < d_{out}$ entonces se asigna emin al grupo. En [26] se discute el valor de $\gamma$.

4. Si quedan menos de $k$ registros sin asignar a ningún grupo, asignar dichos registros a sus grupos más cercanos.

### 4.2. Coste computacional

De forma similar a MDAV, el coste computacional es $O(n^2)$. En [26] se puede encontrar un análisis computacional exhaustivo del V-MDAV. Las diferencias entre MDAV i V-MDAV se limitan a:

- V-MDAV calcula el centroide del conjunto de datos sólo una vez. Por el contrario, MDAV calcula el centroide en varias iteraciones.
- Sin embargo, V-MDAV cuenta con un paso adicional para extender el grupo, con coste computacional $O(n)$.

## 5. Comparación de los métodos

En esta sección se comparan las heurísticas presentadas a lo largo del artículo. A modo de resumen de lo presentado:

- MD y MDAV generan una $k$-partición en la que todos los grupos, excepto quizá el último, son de tamaño $k$. V-MDAV genera grupos de tamaño variable.
- El coste computacional de MD es $O(n^3/k)$, mientras que MDAV y V-MDAV tienen coste $O(n^2)$. En consecuencia, MD es mucho más costoso incluso para un número moderado de registros (e.g. 10000); usar MD en conjuntos grandes o moderadamente grandes necesita bloquear atributos para dividir el conjunto en diversos bloques manejables.

La Tabla 1 muestra varios resultados experimentales donde se puede comprobar la pérdida de información que causa cada uno de los métodos de microagregación anteriores. Los tests se han ejecutado usando diferentes valores de y diversos conjuntos de datos:

- "Disperso": es un conjunto de datos sintéticos con $n = 1000$ registros y $d = 2$ atributos. Este conjunto de datos es disperso en el sentido de que no aparecen conjuntos agrupados de forma natural. Los valores que toman los atributos están comprendidos en un rango [-10000, 10000].
- "Agrupado": es un conjunto de datos con $n = 1000$ registros y $d = 2$ atributos. Este conjunto decimos que es agrupado ya que sus registros forman grupos de forma natural. Los valores de los atributos son iguales al caso anterior, pero forman grupos de entre 3 y 5 registros cada uno.
- "Census": es un conjunto real de datos que contiene 1080 registros con 13 atributos numéricos.
- "EIA": también es un conjunto real de datos que contiene 4092 registros con 11 atributos numéricos.

**Cuadro1.** SEC de los conjuntos microagregados por las heurísticas descritas en este artículo.

| Conjunto de Datos | Método | k=3 | k=4 | k=5 | k=10 |
|---|---|---|---|---|---|
| Disperso | MD | 4.71 | 7.21 | 9.67 | 22.38 |
| | MDAV | 4.72 | 7.37 | 9.95 | 21.74 |
| | V-MDAV | 4.57 | 6.98 | 9.82 | 22.56 |
| Agrupado | MD | 3.53 | 5.03 | 7.03 | 18.56 |
| | MDAV | 3.57 | 4.79 | 6.86 | 18.73 |
| | V-MDAV | 1.52 | 3.85 | 7.51 | 19.99 |
| Census | MD | 803.09 | 1072.70 | 1264.51 | 2021.27 |
| | MDAV | 799.18 | 1053.78 | 1276.02 | 1997.03 |
| | V-MDAV | 798.49 | 1055.51 | 1260.56 | 1974.75 |
| EIA | MD | 212.60 | 347.45 | 751.44 | 1671.78 |
| | MDAV | 217.38 | 302.18 | 750.20 | 1728.31 |
| | V-MDAV | 240.70 | 337.87 | 511.20 | 1270.90 |

Los dos últimos conjuntos de datos se propusieron como conjuntos de referencia en el proyecto "CASC" [27] y se usaron en los artículos [10], [12], [22], [28]-[30]. Estudiando los resultados del conjunto "Disperso", se puede observar que el comportamiento de las heurísticas es muy similar. El valor de SEC se incrementa cuanto mayor es el valor de $k$. En el conjunto "Agrupado", se aprecia una importante mejora al usar V-MDAV con $k$=3 y $k$=4. Cabe destacar que V-MDAV, debido a su flexibilidad, mejora MDAV y MD en conjuntos agrupados. Sin embargo, como el conjunto

"Agrupado" tiene grupos sintéticos con 3, 4 o 5 registros cada uno, V-MDAV no resulta ser mejor que MD o MDAV para $k$=5 y $k$=10.

En lo que respecta a los conjuntos "Census" y "EIA", la pérdida de información que causa V-MDAV es, en general, similar a la causada por MD y MDAV. Sin embargo, en el conjunto "EIA" se puede observar que V-MDAV obtiene $k$-particiones con un menor SEC para $k$=5 y $k$=10. Esto se debe a que "EIA" presenta varios grupos naturales que se han tenido en cuenta al construir una $k$-partición con V-MDAV para los mencionados valores de $k$.

## 6. Conclusiones

En este artículo se han presentado un análisis comparativo entre un conjunto de métodos para $k$-anonimato en bases de datos estadísticas usando microagregación. Los métodos han sido analizados y se han comentado tanto sus ventajas como sus puntos débiles.

Podemos concluir que V-MDAV supera a las heurísticas que forman grupos de tamaño fijo, con un coste computacional similar. Esto hace que la flexibilidad que ofrece V-MDAV sea realmente atractiva. Los resultados experimentales muestran una notable mejoría al usar V-MDAV cuando se aplica a conjuntos con subconjuntos naturalmente agrupables.

## Agradecimientos

Los autores están parcialmente subvencionados por el proyecto SEG2004-04352-C04-01 "PROPRIETAS" y por la Generalitat de Catalunya bajo la concesión 2005 SGR 00446.

## Referencias

[1] M. A. Palley, "Security of statistical databases - compromise through attribute correlational modeling", en Proceedings of the Second International Conference on Data Engineering. Washington, DC, USA: IEEE Computer Society, 1986, pp. 67-74.

[2] D. E. Denning, P. J. Denning, y M. D. Schwartz, "The tracker: a threat to statistical database security", ACM Transactions on Database Systems, vol. 4, no. 1, pp. 76-96, 1979.

[3] N. R. Adam y J. C. Wortmann, "Security-control methods for statistical databases: a comparative study", ACM Comput. Surv., vol. 21, no. 4, pp. 515-556, 1989.

[4] J. Pokorny, "Conceptual modeling of statistical data", en Proceedings., Seventh International Workshop on Database and Expert Systems Application. IEEE Computer Society, 1996, pp. 377-382.

[5] G. DeGiacomo y P. Naggar, "Conceptual data model with structured objects for statistical databases", en Proceedings., Eighth International Conference on Scientific and Statistical Database Systems. IEEE Computer Society, 1996, pp. 168-175.
[6] R. Agrawal, J. Kiernan, R. Srikant, y Y. Xu, "Hippocratic databases." en VLDB, 2002, pp. 143-154.
[7] R. Agrawal, R. J. B. Jr., C. Faloutsos, J. Kiernan, R. Rantzau, y R. Srikant, "Auditing compliance with a hippocratic database", en VLDB, 2004, pp. 516-527.
[8] J. Domingo-Ferrer, A. Martínez-Ballesté, y J. M. Mateo-Sanz, "Efficient multivariate data-oriented microaggregation", Manuscript, 2005.
[9] J. Domingo-Ferrer y V. Torra, "Ordinal, continuous and heterogenerous $k$-anonymity through microaggregation", Data Mining and Knowledge Discovery, vol. 11, no. 2, 2005.
[10] M. Laszlo y S. Mukherjee, "Minimum spanning tree partitioning algorithm for microaggregation", IEEE Transactions on Knowledge and Data Engineering, vol. 17, no. 7, pp. 902-911, 2005.
[11] A. Solanas, A. Martínez-Ballesté, J. M. Mateo-Sanz, y J. Domingo-Ferrer, "Multivariate microaggregation based on a genetic algorithm", Manuscript, 2006.
[12] J. Domingo-Ferrer y V. Torra, "Ordinal, continuous and heterogenerous $k$-anonymity through microaggregation", Data Mining and Knowledge Discovery, vol. 11, no. 2, pp. 195-212, 2005.
[13] D. Defays y P. Nanopoulos, "Panels of enterprises and confidentiality: the small aggregates method", en Proc. of 92 Symposium on Design and Analysis of Longitudinal Surveys. Ottawa: Statistics Canada, 1993, pp. 195-204.
[14] M. Rosemann, "Erste ergebnisse von vergleichenden untersuchungen mit anonymisierten und nicht anonymisierten einzeldaten amb beispiel der kostenstrukturerhebung und der umsatzsteuerstatistik", en G. Ronning y R. Gnoss (editores) Anonymisierung wirtschaftsstatistischer Einzeldaten, Wiesbaden: Statistisches Bundesamt, 2003, pp. 154-183.
[15] E. C. for Europe, "Statistical data confidentiality in the transition countries: 2000/2001 winter survey", en Joint ECE/Eurostat Work Session on Statistical Data Confidentiality, 2001, invited paper n.43.
[16] J. Domingo-Ferrer y V. Torra, "On the connections between statistical disclosure control for microdata and some artificial intelligence tools",Information Sciences, vol. 151, pp. 153-170, Mayo 2003.
[17] A. W. F. Edwards y L. L. Cavalli-Sforza, "A method for cluster analysis", Biometrics, vol. 21, pp. 362-375, 1965.
[18] J. H. Ward, "Hierarchical grouping to optimize an objective function",Journal of the American Statistical Association, vol. 58, pp. 236-244, 1963.
[19] A. D. Gordon y J. T. Henderson, "An algorithm for euclidean sum of squares classification", Biometrics, vol. 33, pp. 355-362, 1977.
[20] P. Hansen, B. Jaumard, y N. Mladenovic, "Minimum sum of squares clustering in a low dimensional space", Journal of Classification, vol. 15, pp. 37-55, 1998.
[21] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations", en Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, 1967, pp. 281-297.
[22] J. Domingo-Ferrer y J. M. Mateo-Sanz, "Practical data-oriented microaggregation for statistical disclosure control", IEEE Transactions on Knowledge and Data Engineering, vol. 14, no. 1, pp. 189-201, 2002.

[23] Y. Jung, H. Park, D.-Z. Du, y B. L. Drake, "A decision criterion for the optimal number of clusters in hierarchical clustering", Journal of Global Optimization, vol. 25, pp. 91-111, Ene. 2005.

[24] A. Oganian y J. Domingo-Ferrer, "On the complexity of optimal microaggregation for statistical disclosure control", Statistical Journal of the United Nations Economic Comission for Europe, vol. 18, no. 4, pp. 345-354, 2001.

[25] A. Hundepool, A. V. deWetering, R. Ramaswamy, L. Franconi, A. Capobianchi, P.-P. DeWolf, J. Domingo-Ferrer, V. Torra, R. Brand, y S. Giessing, $\mu$-ARGUS version 4.0 Software y User's Manual. Voorburg NL: Statistics Netherlands, mayo 2005, http://neon.vb.cbs.nl/casc.

[26] A. Solanas y A. Martínez-Ballesté, "V-MDAV: Variable group size multivariate microaggregation", 2006, manuscript.

[27] R. Brand, J. Domingo-Ferrer, y J. M. Mateo-Sanz, "Reference data sets to test and compare sdc methods for protection of numerical microdata", 2002, european Project IST-2000-25069 CASC, http://neon.vb.cbs.nl/casc.

[28] J. Domingo-Ferrer, F. Sebé, y A. Solanas, "A polynomial-time approximation to optimal multivariate microaggregation", Manuscript, 2005.

[29] W. E. Yancey, W. E. Winkler, y R. H. Creecy, "Disclosure risk assessment in perturbative microdata protection", en Inference Control in Statistical Databases, ser. LNCS, J. Domingo-Ferrer, Ed., vol. 2316. Berlin Heidelberg: Primavera, 2002, pp. 135-152.

[30] R. Dandekar, J. Domingo-Ferrer, y F. Sebé, "Lhs-based hybrid microdata vs rank swapping and microaggregation for numeric microdata protection", en Inference Control in Statistical Databases, ser. LNCS, J. Domingo-Ferrer, Ed., vol. 2316. Berlin Heidelberg: Primavera, 2002, pp. 153-162.