Microaggregation for Database and Location Privacy

Josep Domingo-Ferrer

Rovira i Virgili University of Tarragona
Dept. of Computer Engineering and Maths
Av. Països Catalans 26
E-43007 Tarragona, Catalonia
josep.domingo@urv.cat

Abstract. Data aggregation is a central principle underlying many applications in computer science, from artificial intelligence to data security and privacy. Microaggregation is a special clustering problem where the goal is to cluster a set of points into groups of at least k points in such a way that groups are as homogeneous as possible. A usual homogeneity criterion is the minimization of the within-groups sum of squares. Microaggregation appeared in connection with anonymization of statistical databases. When discussing microaggregation for information systems, points are database records. This paper extends the use of microaggregation for k-anonymity to implement the recent property of p-sensitive k-anonymity in a more unified and less disruptive way. Then location privacy is investigated: two enhanced protocols based on a trusted-third party (TTP) are proposed and thereafter microaggregation is used to design a new TTP-free protocol for location privacy.

Keywords: Microaggregation, k-Anonymity, Statistical database privacy, Location privacy, Wireless systems.

1 Introduction

Microaggregation [3, 4] is a special clustering problem which became relevant in connection with privacy in statistical databases, a discipline also known as statistical disclosure control (SDC) whose purpose is to prevent released individual data records (microdata) from being linkable with the respondents they correspond to. The microaggregation problem is to cluster a set of d-dimensional points (points represent records in the database application) into groups of at least k points in such a way that groups are as homogeneous as possible.

Let $\mathbf{X}$ be a dataset formed by n points in a d-dimensional numerical space. Microaggregation is operationally defined in terms of two steps. Given a parameter k, the first step partitions points in $\mathbf{X}$ into groups of at least k points each. The second step replaces each point by the centroid of its group to obtain the masked dataset $\mathbf{X}'$. In a microaggregated dataset where points correspond to individuals (*e.g.* each point is a record with individual data, and the attribute values in the record are the point co-ordinates), no re-identification within a

O. Etzion, T. Kuflik, and A. Motro (Eds.): NGITS 2006, LNCS 4032, pp. 106–116, 2006.

group is possible, because all k points in a group are identical: the best that an intruder can hope is to track the group where a target individual has been masked into.

Microaggregating with minimum information loss has been known to be an important —and difficult— issue ever since microaggregation was invented as an SDC masking method for microdata. However, it was often argued that optimality in SDC is not just about minimum information loss but about the best tradeoff between low information loss and low disclosure risk. The recent application [8] of microaggregation to achieve k-anonymity [24, 23, 26, 27] for numerical microdata leaves no excuse to circumvent the problem of trying to reduce information loss as much as possible: once a value k is selected that keeps the re-identification risk low enough, the only job left is to k-anonymize (that is, to microaggregate) with as little information loss as possible.

A partition P such that all of its groups have size at least k is called a k-partition ([4]) and microaggregation with parameter k is sometimes denoted as k-microaggregation.

In [4], optimal microaggregation is defined as the one yielding a k-partition maximizing the within-groups homogeneity. The rationale is that, the more homogeneous the points in a group, the less variability reduction when replacing those points by their centroid and thus the less information loss. The within-groups sum of squares SSE is a usual measure of within-groups homogeneity in clustering [34, 9, 11, 12], so a reasonable optimality criterion for a k-partition $P = \{G_1, \ldots, G_g\}$ is to minimize SSE, *i.e.* to minimize

$$SSE(P) = \sum_{i=1}^{g} \sum_{j=1}^{|G_i|} (x_{ij} - c(G_i))'(x_{ij} - c(G_i))$$

where $|G_i|$ is the number of points in the i-th group, $c(G_i)$ is the centroid of the i-th group and x_{ij} is the j-th point in the i-th group. It was shown in [4] that groups in the optimal k-partition have sizes between k and $2k - 1$.

For the univariate case (one-dimensional points), the optimal k-partition can be computed in polynomial time using the algorithm in [13]. For the multivariate case ($d > 1$), the optimal microaggregation problem has been shown to be NP-hard ([20]). Therefore, algorithms for multivariate microaggregation are heuristic [2, 4, 25, 14, 15]. Quite recently, the first approximation heuristic for multivariate microaggregation has been presented ([6]); it yields a k-partition whose SSE is no more than an certain multiple of the minimum SSE.

Microaggregation can be extended for categorical spaces (that is, for records with categorical attributes) if the Euclidean distance is replaced by an appropriate categorical distance and the mean is replaced by a suitable categorical average when computing centroids [30].

1.1 Contribution and Plan of This Paper

In this work we will review the use of microaggregation to provide disclosure control and k-anonymity in statistical databases. In this context we will give

new results on how to adapt microaggregation to implement the recently defined property of p-sensitive k-anonymity in a more unified and less disruptive way (without generalizations nor suppressions). We will then investigate how microaggregation can be adapted for application to location privacy.

Section 2 deals with the application to statistical databases. Section 3 is about the new application to location privacy. Conclusions and lines for future research are sketched in Section 4.

2 Microaggregation for Statistical Databases

Microaggregation became known as a statistical disclosure control technique proposed by Eurostat researchers in the early nineties [3]. The heuristics developed at that time split the records in a dataset into groups of exactly k records, except for the last group, which contained between k and $2k-1$ records. The operation of such heuristics was basically *univariate*: either the records were projected onto a single dimension —*e.g.* using the first principal component, the sum of z-scores or a particular attribute— or microaggregation was conducted independently for each attribute in the dataset —what is known as individual ranking microaggregation [21]—.

Projecting multivariate data onto a single dimension caused a very high loss of information (data utility) [7], while using individual ranking provides very little protection against disclosure risk [7, 5]. The authors of [4] were the first to define *multivariate* microaggregation as a cardinality-constrained multivariate clustering problem (in the way described in Section 1) and give a heuristic for it. Going multivariate allowed a tradeoff between information loss and disclosure risk to be struck and turned microaggregation into one of the best SDC methods for microdata [7].

Since it was proposed at Eurostat, microaggregation has been used in Germany [22, 16], Italy [21] and several other countries, detailed in the surveys [32, 33] by the United Nations Economic Commission for Europe.

2.1 k-Anonymity Using Microaggregation

In [8], microaggregation was shown to be useful to implement the property of k-anonymity [24, 23, 26, 27]. To recall the definition of k-anonymity, we need to enumerate the various (non-disjoint) types of attributes that can appear in a microdata set $\mathbf{X}$:

- *Identifiers.* These are attributes that *unambiguously* identify the respondent. Examples are passport number, social security number, full name, etc. Since our objective is to prevent confidential information from being linked to specific respondents, we will assume in what follows that, in a pre-processing step, identifiers in $\mathbf{X}$ have been removed/encrypted.
- *Key attributes.* Borrowing the definition from [1, 23], key attributes are those in $\mathbf{X}$ that, in combination, can be linked with external information to re-identify (some of) the respondents to whom (some of) the records in $\mathbf{X}$

refer. Unlike identifiers, key attributes cannot be removed from **X**, because any attribute is potentially a key attribute.
- *Confidential outcome attributes.* These are attributes which contain sensitive information on the respondent. Examples are salary, religion, political affiliation, health condition, etc.

Now, the k-anonymity property can be stated as:

Definition 1 (k-Anonymity). *A dataset is said to satisfy k-anonymity for $k > 1$ if, for each combination of values of key attributes (*e.g. *name, address, age, gender, etc.), at least k records exist in the dataset sharing that combination.*

In the seminal k-anonymity papers [24, 23, 26, 27], the computational procedure suggested to k-anonymize a dataset relies on suppressions and generalizations. The drawbacks of partially suppressed and coarsened data for analysis were highlighted in [8]. Joint multivariate microaggregation of all key attributes with minimum group size k was proposed in [8] as an alternative to achieve k-anonymity; besides being simpler, this alternative has the advantage of yielding complete data without any coarsening (nor categorization in the case of numerical data).

2.2 A New Property: p-Sensitive k-Anonymity Using Microaggregation

We will show here how microaggregation can be used to implement a recently defined privacy property called p-sensitive k-anonymity [31]. This is an evolution of k-anonymity whose motivation is to avoid that records sharing a combination of key attributes in a k-anonymous data set also share the values for one or more confidential attributes. In this case, k-anonymity does not offer enough protection.

Example 1. Imagine that an individual's health record is k-anonymized into a group of k patients with k-anonymized key attributes values *Age = "30"*, *Height = "180 cm"* and *Weight = "80 kg"*. Now, if all k patients share the confidential attribute value *Disease = "AIDS"*, k-anonymization is useless, because an intruder who uses the key attributes *(Age, Height, Weight)* can link an external identified record

(Name="John Smith", Age="31", Height="179", Weight="81")

with the above group of k patients and infer that John Smith suffers from AIDS. □

Based on the above, the p-sensitive k-anonymity property is defined as:

Definition 2 (p-Sensitive k-anonymity). *A dataset is said to satisfy p-sensitive k-anonymity for $k > 1$ and $p \leq k$ if it satisfies k-anonymity and, for each group of tuples with the same combination of key attribute values that exists in the dataset, the number of distinct values for each confidential attribute is at least p within the same group.*

In the above example, p-sensitive k-anonymity would require in particular that there be at least p different diseases in each group of people sharing the same k-anonymized age, height and weight.

The computational approach proposed in [31] to achieve p-sensitive k-anonymity is an extension of the generalization/suppression procedure proposed in papers [24, 23, 26, 27]. Therefore, it shares the same shortcomings pointed out in [8]: p-sensitive k-anonymized data are a coarsened and partially suppressed version of original data. Generalization is undesirable for a number of reasons: i) it transforms numerical (continuous) attributes into categorical ones; ii) if applied to all records, it causes a great loss of information and, if applied only locally, it introduces new categories that co-exist with the old ones and complicate subsequent analyses. On the other hand, partially suppressed data cannot be analyzed using standard statistical tools (techniques for censored data are needed).

Like we did for k-anonymity in [8], we propose here to achieve p-sensitive k-anonymity via microaggregation. The proposed algorithm is as follows.

Algorithm 1 (p-sensitive k-anonymization). *($\mathbf{X}$: dataset, k, p: integers)*

1. *If $p > k$, signal an error ("p-sensitive k-anonymity infeasible") and exit the Algorithm.*
2. *If the number of distinct values for any confidential attribute in $\mathbf{X}$ is less than p over the entire dataset, signal an error ("p-sensitive k-anonymity infeasible") and exit the Algorithm.*
3. *k-Anonymize $\mathbf{X}$ using microaggregation as described in Section 2.1. Let $\mathbf{X}'$ be the microaggregated, k-anonymized dataset.*
4. *Let $\hat{k} := k$.*
5. *While p-sensitive k-anonymity does not hold for $\mathbf{X}'$ do:*
   (a) *Let $\hat{k} := \hat{k} + 1$.*
   (b) *$\hat{k}$-Anonymize $\mathbf{X}$ using microaggregation. Let $\mathbf{X}'$ be the $\hat{k}$-anonymized dataset.*

The above algorithm in based on the following facts:

- A $k+1$-anonymous dataset is also k-anonymous.
- By increasing the minimum group size, the number of distinct values for confidential attributes within a group also increases.

The resulting p-sensitive k-anonymous dataset does not contain coarsened or partially suppressed data. This makes its analysis and explotation easier, with the additional advantage that numerical continuous attributes are not categorized.

3 Microaggregation for Location Privacy

The growth of location-detection devices (*e.g.* cellular phones, GPS-like devices, RFIDs and handheld devices) along with wireless communications has fostered

the development of location-based commercial services [17, 18, 28, 29]. These are applications that deliver specific information to their users *based* on their current location (*e.g.* list of hotels or pharmacies near the place where you are, etc.).

As noted in [35], although location-based services may be very convenient, they threaten the privacy and security of their customers. Several approaches to location privacy have been proposed, an overview of which is given in [19]. The latter paper also proposes a model in which a *location anonymizer* acts as a third party between the mobile users and the location-based database servers. The interaction between user, anonymizer and database server is completely mediated by the anonymizer according to the following protocol:

Protocol 1 (Mediator with location anonymization)

1. User U sends her query to the anonymizer A.
2. A anonymizes the location of U and forwards the query to the database server DS.
3. DS returns the query answer to A.
4. A forwards the query answer to U.

The location anonymizer's aim is to protect the privacy of queries issued by roaming users who sent their location as an input parameter for the query. The approach in [19] is for the location anonymizer to provide a special form of location k-anonymity. Location privacy differs from the database privacy problem dealt with in Section 2 in some respects:

- Locations are updated more frequently than typical databases. Therefore, k-anonymity algorithms for location privacy must be faster than those for databases.
- k-Anonymity alone may not be enough. If k roaming users happen to be in nearly the same place (*e.g.* a department store) and one of them issues a query, no privacy is gained if the location anonymizer forwards to the database server the minimum bounding rectangle containing the locations of the k users instead of the querying user's locations.

In view of the above peculiarities, the following requirements are stated in [19] on the algorithm used by the location anonymizer to *cloak* a user's location into a spatial region containing it:

1. The *cloaking* region should contain at least k users to provide k-anonymity. In addition to this, the cloaking region should cover an area A such that $A_{min} < A < A_{max}$. The minimum bound A_{min} is a privacy parameter and the maximum bound A_{max} is a utility parameter (too large a cloaking region is useless to obtain location-based services from the database server).
2. An intruder should not be able to know the exact location of the user within the cloaking region.
3. The cloaking algorithm should be computationally efficient to cope in real time with the continuous movement of mobile users.

Note 1. As a side remark, it is interesting to note that the k-anonymity used for location privacy in the literature [19, 10] is somewhat different from the standard k-anonymity discussed in Section 2. *What is perturbed (cloaked) in location privacy is the user's confidential attributes (latitude and longitude)*, whereas in database k-anonymity perturbation affects the key attributes.

3.1 Mediator Without Location Anonymization

In the architecture proposed in [19], a trusted third party is mediating all user queries to the database server. We claim that, in this context, there is no need for anonymizing the user location, since the mediator can withhold all the user's identifiers and key attributes. This would result in the following protocol:

Protocol 2 (Mediator without location anonymization)

1. User U sends her query to the mediator M.
2. M suppresses any identifiers or key attributes in the query and forwards only the question in the query plus the exact location of U (latitude, longitude) to the database server DS.
3. DS returns the query answer to M, based on the location provided.
4. M forwards the query answer to U.

Note that the DS only receives questions and (latitude, longitude) pairs without any identifying information. Therefore DS is unable to link those locations with the users they correspond to. Additionally, since exact locations are sent to the database server, query answers can be more accurate and useful. *E.g.* if the query is about hotels or pharmacies near the user's location, only those really close hotels or pharmacies will be included in the answer.

3.2 Location Anonymization Without Mediation

Only when there is direct interaction between the user and the database server is location anonymization really needed. This is the situation described by the following protocol:

Protocol 3 (Location anonymization without mediation)

1. User U requests the anonymizer A to anonymize her location.
2. A anonymizes the location of U. Like in Protocol 1, this is done by cloaking it with the locations of at least $k-1$ additional users, while taking spatial constraints into account (see above). A returns the anonymized location to U.
3. U sends her query to DS, using her anonymized location instead of her real location.
4. DS returns the query answer to U, based on the location provided.

Protocol 3 is a clear improvement regarding user privacy, because the anonymizer does not receive either user queries or their answers (unlike the anonymizer in Protocol 1 or the mediator in Protocol 2).

3.3 TTP-Free Location Anonymization Through Microaggregation

One might argue against Protocols 1, 2 and 3 that their use of a trusted third party (TTP) is a privacy weakness. Indeed, the TTP (anonymizer or mediator) learns the whereabouts and the identities of all users. Additionally, in Protocols 1 and 2, the TTP learns the content of the queries and the answers.

We present next a TTP-free protocol, whereby users do not need to disclose their exact location or their queries. The operating principle is to compute the centroid of at least k perturbed user locations and feed it directly to the database server.

Protocol 4 (TTP-free location anonymization)

1. User U does:
   (a) Generate a perturbed version of her location (x, y) by adding Gaussian noise to her latitude x and longitude y, that is
   $$(x', y') = (x + \epsilon_x, y + \epsilon_y)$$
   where ϵ_x is drawn from a $N(0, \sigma_x^2)$ random variable and ϵ_y is drawn from a $N(0, \sigma_y^2)$ random variable. All users use the same σ_x, σ_y.
   (b) Broadcast a message containing (x', y') to request neighbors to return perturbed versions of their locations.
   (c) Among the replies received, select $k-1$ neighbors such that the group G formed by them plus (x', y') spans an area A satisfying $A_{min} < A < A_{max}$, where A_{min} and A_{max} are input parameters with the meaning defined above. If there are several feasible groups G, choose the one minimizing the within-group sum of squares SSE. If there is no feasible G (not enough neighbors have replied or the above spatial constraint cannot be met) go back to Step 1b.
   (d) Compute the centroid (c'_x, c'_y) of G.
   (e) Send her query to DS, using (c'_x, c'_y) as location instead of U's real location.
2. DS returns the query answer to U, based on the location provided.

Lemma 1. *If each user perturbs her location using Gaussian noise $N(0, \sigma_x^2)$ for latitude and $N(0, \sigma_y^2)$ for longitude, the standard errors of the centroid (c'_x, c'_y) of k perturbed locations with respect to the centroid (c_x, c_y) of the real locations are $\sigma_x/\sqrt{k}$ and $\sigma_y/\sqrt{k}$, respectively.*

Proof. By elementary sampling theory, the sample mean of k observations of a $N(0, \sigma^2)$ random variable follows a $N(0, \sigma^2/k)$ distribution. Thus,

$$(c'_x, c'_y) = (c_x + \epsilon_{c,x}, c_y + \epsilon_{c,y})$$

where $\epsilon_{c,x}$ is drawn from a $N(0, \sigma_k^2/n)$ random variable and $\epsilon_{c,y}$ is drawn from a $N(0, \sigma_y^2/k)$ random variable. □

So, the perturbation added by each user to her own location it mitigated by the centroid computation. Even for small k and moderate σ_x, σ_y, the centroid (c'_x, c'_y) sent to the database server can be expected to be close to the real centroid (c_x, c_y). Therefore, the utility the query answer provided by the database server is largely unaffected by perturbation.

The reason for each user to perturb her own location is to defend her own privacy from collusions. Without perturbation, $k-1$ users could collude and pool their real locations to derive the remaining user's location. With perturbation, the colluders will get no more than the perturbed location of the remaining user. If a non-roaming user has already sent one or more perturbed versions of her constant real location in previous instances of Step 1b of Protocol 4, she should evaluate the privacy risk incurred if she sends a new perturbed version: by Lemma 1, anyone can average those perturbed locations to get closer to the user's real location. A roaming user does not face that problem, since her real location is continuously changing.

4 Conclusions and Future Research

We have presented several applications of the principle of microaggregation to providing privacy in databases and location-based services. In the case of databases, we have extended the use of microaggregation for k-anonymity to implement the recent property of p-sensitive k-anonymity. In the case of location privacy, we have proposed two TTP-based protocols which improve on the state of the art. Despite the improvements, TTP-based privacy remains weak, so we have also proposed a TTP-free protocol for location privacy based on microaggregation.

Future research will include refining and tuning the parameters for using microaggregation to implement p-sensitive k-anonymity and TTP-free location privacy.

Acknowledgments

The author is partly supported by the Catalan government under grant 2005 SGR 00446, and by the Spanish Ministry of Science and Education through project SEG2004-04352-C04-01 "PROPRIETAS".

References

1. T. Dalenius. Finding a needle in a haystack - or identifying anonymous census records. *Journal of Official Statistics*, 2(3):329–336, 1986.
2. D. Defays and N. Anwar. Micro-aggregation: a generic method. In *Proceedings of the 2nd International Symposium on Statistical Confidentiality*, pages 69–78, Luxemburg, 1995. Eurostat.
3. D. Defays and P. Nanopoulos. Panels of enterprises and confidentiality: the small aggregates method. In *Proc. of 92 Symposium on Design and Analysis of Longitudinal Surveys*, pages 195–204, Ottawa, 1993. Statistics Canada.

4. J. Domingo-Ferrer and J. M. Mateo-Sanz. Practical data-oriented microaggregation for statistical disclosure control. *IEEE Transactions on Knowledge and Data Engineering*, 14(1):189–201, 2002.
5. J. Domingo-Ferrer, Josep M. Mateo-Sanz, A. Oganian, and À. Torres. On the security of microaggregation with individual ranking: analytical attacks. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5):477–492, 2002.
6. J. Domingo-Ferrer, F. Sebé, and A. Solanas. A polynomial-time approximation to optimal multivariate microaggregation. *Manuscript*, 2005.
7. J. Domingo-Ferrer and V. Torra. A quantitative comparison of disclosure control methods for microdata. In P. Doyle, J. I. Lane, J. J. M. Theeuwes, and L. Zayatz, editors, *Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies*, pages 111–134, Amsterdam, 2001. North-Holland. http://vneumann.etse.urv.es/publications/bcpi.
8. J. Domingo-Ferrer and V. Torra. Ordinal, continuous and heterogenerous k-anonymity through microaggregation. *Data Mining and Knowledge Discovery*, 11(2):195–212, 2005.
9. A. W. F. Edwards and L. L. Cavalli-Sforza. A method for cluster analysis. *Biometrics*, 21:362–375, 1965.
10. B. Gedik and L. Liu. A customizable k-anonymity model for protecting location privacy. In *Proceedings of the International Conference on Distributed Computing Systems-ICDCS*, 2005.
11. A. D. Gordon and J. T. Henderson. An algorithm for euclidean sum of squares classification. *Biometrics*, 33:355–362, 1977.
12. P. Hansen, B. Jaumard, and N. Mladenovic. Minimum sum of squares clustering in a low dimensional space. *Journal of Classification*, 15:37–55, 1998.
13. S. L. Hansen and S. Mukherjee. A polynomial algorithm for optimal univariate microaggregation. *IEEE Transactions on Knowledge and Data Engineering*, 15(4):1043–1044, 2003.
14. A. Hundepool, A. Van de Wetering, R. Ramaswamy, L. Franconi, A. Capobianchi, P.-P. DeWolf, J. Domingo-Ferrer, V. Torra, R. Brand, and S. Giessing. *μ-ARGUS version 4.0 Software and User's Manual*. Statistics Netherlands, Voorburg NL, may 2005. http://neon.vb.cbs.nl/casc.
15. M. Laszlo and S. Mukherjee. Minimum spanning tree partitioning algorithm for microaggregation. *IEEE Transactions on Knowledge and Data Engineering*, 17(7):902–911, 2005.
16. R. Lenz and D. Vorgrimler. Matching german turnover tax statistics. Technical Report FDZ-Arbeitspapier Nr. 4, Statistische Ämter des Bundes und der Länder-Forschungsdatenzentren, 2005.
17. http://www.locatrix.com.
18. http://www.mapinfo.com.
19. M. F. Mokbel. Towards privacy-aware location-based database servers. In *Proceedings of the Second International Workshop on Privacy Data Management-PDM'2006*, Atlanta, GA, 2006. IEEE Computer Society.
20. A. Oganian and J. Domingo-Ferrer. On the complexity of optimal microaggregation for statistical disclosure control. *Statistical Journal of the United Nations Economic Comission for Europe*, 18(4):345–354, 2001.
21. D. Pagliuca and G. Seri. Some results of individual ranking method on the system of enterprise accounts annual survey, 1999. Esprit SDC Project, Deliverable MI-3/D2.

22. M. Rosemann. Erste Ergebnisse von vergleichenden Untersuchungen mit anonymisierten und nicht anonymisierten Einzeldaten am Beispiel der Kostenstrukturerhebung und der Umsatzsteuerstatistik. In G. Ronning and R. Gnoss, editors, *Anonymisierung wirtschaftsstatistischer Einzeldaten*, pages 154–183, Wiesbaden, Germany, 2003. Statistisches Bundesamt.
23. P. Samarati. Protecting respondents' identities in microdata release. *IEEE Transactions on Knowledge and Data Engineering*, 13(6):1010–1027, 2001.
24. P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical report, SRI International, 1998.
25. G. Sande. Exact and approximate methods for data directed microaggregation in one or more dimensions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5):459–476, 2002.
26. L. Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty, Fuzziness and Knowledge Based Systems*, 10(5):571–588, 2002.
27. L. Sweeney. k-anonimity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge Based Systems*, 10(5):557–570, 2002.
28. http://www.targusinfo.com.
29. http://www.telostar.com.
30. V. Torra. Microaggregation for categorical variables: a median based approach. In *Lecture Notes in Computer Science*, vol. 3050, pp. 162–174, Berlin Heidelberg, 2004. Springer.
31. T. M. Truta and B. Vinay. Privacy protection: p-sensitive k-anonymity property. *Manuscript*, 2005.
32. UNECE. United nations economic commission for europe. questionnaire on disclosure and confidentiality - summary of replies. In *1st Joint UNECE/Eurostat Work Session on Statistical Data Confidentiality*, Thessaloniki, Greece, 1999.
33. UNECE. United nations economic commission for europe. questionnaire on disclosure and confidentiality - summary of replies. In *2nd Joint UNECE/Eurostat Work Session on Statistical Data Confidentiality*, Skopje, Macedonia, 2001.
34. J. H. Ward. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58:236–244, 1963.
35. J. Warrior, E. McHenry, and K. McGee. They know where you are. *IEEE Spectrum*, 40(7):20–25, 2003.