RESEARCH REPORT SERIES
(*Statistics #2002-07)*

Using Simulated Annealing for k-anonymity

William E. Winkler

Statistical Research Division
U.S. Bureau of the Census
Washington D.C. 20233

Report Issued: November 19, 2002

Using Simulated Annealing for k-anonymity

William E Winkler[1], william.e.winkler@census.gov 2002Oct24

Abstract

In this note, we describe a method of simulated annealing for producing k-anonymity. For analytic purposes, there is no reason to expect that the method will be superior or worse than the method of applying genetic algorithms used by Iyengar (2002). The main appeal of simulated annealing is the amount of control it allows of the micro-aggregation process. The k-anonymity problem is known to be NP complete.

Introduction

Samarati (2001, also Samarati and Sweeney 1998, Sweeney 1999) have defined *k-anonymity* in a microdata release as the property that each combination of values of quasi-identifiers can be indistinctly matched to at least k individuals.

For instance, assume we have a microdata release of 2000 records having sex (two categories or value-states for the sex variable) and age (100 categories) in which the values states are uniformly distributed within each variable. Then using sex alone, we have 1000-anonymity; using age, alone we have 20-anonymity. If sex and age are uncorrelated, then using the two variables yields 10-anonymity. In other words, for each value of sex and each value of age there are 10 records. If we do not consider 10-anonymity to be sufficient, then we could collapse age into ten-year intervals. This would yield 100-anonymity. In many situations, k-values between 3 and 10 are considered sufficient.

Determining k-anonymity by collapsing value-states of a variable into aggregated value states is known to be a difficult combinatorial optimization problem. Domingo (2001) shows that micro-aggregation is NP complete. Using similar arguments, it is straightforward to deduce that k-anonymity is NP complete. Iyengar (2002) uses genetic algorithms to produce k-anonymity in a simulation that uses public-use Census data. He also shows that the k-anonymized data can still be used for classification (determining whether someone has an income above or below $50,000). In his application, he obtains k values as high as 250 yield anonymized data that can still be used for classification with accuracy comparable to the original un-anonymized data. In general, however, with k-values above 10, we can destroy the analytic usefulness of data for loglinear analyses or subdomain analyses.

In the remainder of this note, we give background and define terminology. After these preliminaries, we describe the method for simulated annealing for producing k-anonymity. We finish with discussion and an example that is a special case of the main application of Iyengar (2002).

Background and Definitions of Terms

We use a basic simulated annealing approach (e.g., Kirkpatrick et al. 1983, Geman and Geman 1984, Aarts and Lenstra 1997). For now, we do not include generalization restraints. Iyengar (2002) shows how a tree-based hierarchy (defined by users or subject matter experts) can be used naturally with genetic algorithms. Although the procedure described below can be generalized and computed easily with user-defined hierarchies, we do not include them. It is not clear how genetic algorithms could be developed that do not use user-defined hierarchies for collapsing value-states of individual variables. We do note that the user-defined hierarchies can significantly reduce the number of subsets of the value states of each variable that are considered. The problem, however, is still NP complete.

Let X_i, i=1,…,n, be variables (fields) with value-states x_{ij}, j = 1, …, n_i. The annealing procedure will go through the set of fields sequentially. In the following, $G(y_i)$ represents the set of values associated with value-state y_i of field i. The states of the system are is described in terms of the set of disjoint groups associated with the value-states of individual variables. In the application of Iyengar (2002), the objective function f() is a measure CM of the average penalty associated with suppressing values in a record (row of a table). As Iyengar emphasizes, the objective function f() needs to be chosen according to the analytic uses of the k-anonymized data. Let $r = (y_1, …, y_n)$ be a

record. For accurate classification, for each i = 1, …, n, $G(y_i)$ should be associated with one class. Following Iyengar (but with very slightly different notation), we define CM as the average penalty given by

$$f() = CM = (\sum_{\text{all records}} \text{Penalty (record r)})/N \quad (1)$$

where N is the number of records in the file. A record r is penalized if the value in some field is suppressed (i.e., set to a neutral value) or if the value y_i in some field i is different from class label associated with the majority or records associated with $G(y_i)$. The penalty for record r is given by

$$\text{Penalty (row r)} = \begin{cases} 1 \text{ if record r is suppressed (i.e., has a value in a field set to a neutral value)} \\ 1 \text{ if class r} \neq \text{majority } (G(y_i)) \text{ for some i.} \\ 0 \text{ otherwise} \end{cases} \quad (2)$$

For Iyengar (2002), the objective is relatively straightforward because it is associated with a yes/no classification task. For general contingency tables, our objective function might be associated with certain sufficient statistics needed for the loglinear modeling.

The Algorithm

Algorithm Sim1

Repeat until M cycles or the restriction on the objective function f().

Repeat thru for i=1, …,n fields

For i, sample an existing value-state, say y_i (each value state must have positive probability of being sampled.) All movement will be related to individual value states y_i. This is because all movement to different system states (in either forward or backward directions) can be described in terms of of individual y_i's. First, draw a value-state y_i at random. Then, draw a sample value associated with state y_i from the neighborhood of i and state y_i to get z_i. The neighborhood $N(y_i)$ $= \{ y_i \} \cup (\{ x_{ij}, j = 1, \ldots, n_i \} \setminus G(y_i))$. The neighborhood consists of y_i and all of the value states outside of the group $G(y_i)$ containing y_i. We have two operations. The first is associated with collapsing. Assume that we sample a value state outside of the group containing y_i to get a value state z_i. The potential move is to move z_i from its current group (could be a singleton) to the group $G(y_i)$. With the potential new configuration of the states of y_i, we compute $f(z_i)$ and $f(y_i)$. If $f(y_i)$ (the system state with the new configurations improves the objective, we accept the move; otherwise, we accept with probability $\exp((f(z_i) – f(y_i))/ c_i)$ where c_i is associated with the annealing schedule of field i. In this situation z_i represents collapsing. If y_i is sampled, we can move y_i from its group $G(y_i)$ and create a singleton group $G(y_i) = \{ y_i \}$. We compute the objectives $f(z_i)$ and $f(y_i)$. In this second situation, $f(z_i)$ represents uncollapsing. If $f(z_i)$ improves the objective f(), we accept. If it does not, we accept the move with probability $\exp((f(z_i) – f(y_i))/ c_i)$. We can vary (if necessary) the c_i as M increases in a manner so that c_i goes to zero.

Discussion

There are three subtle issues: (1) what is meant by the local neighborhood, (2) how does one define transition probabilities between the various states of the system and (3) what does the terminology $f(y_i)$ mean.

(1) Each state of the system consists of a set of fields and the set of subsets (groups) of value-states on the individual fields at a given time-point. A local neighborhood consists the set of fields and the subsets of the value-states of each field that can be reached in one move from the current system state.

(2) For each field i, movement is by changing the group in which single state x_{ik} is associated. We can move a single value-state from the current group in which it resides to a new group. Generally, the movement mechanism has a tendency toward collapsing into larger and larger groups. We also need to assure that we can uncollapse. Uncollapsing assures that we can move to any state of the system (i.e., irreducibility) and that the resultant Markov process is reversible (we can get back to earlier states of the system). We can also control the sampling mechanism more precisely. At early stages, we allow somewhat more collapsing. At later stages, we can slow down collapsing

by making by making the uncollapsing (i.e., separating a value-state x_{ik} from the group in which it resides) slightly more likely than at early stages.

(3) $f(z_i)$ is the CM measure (see Iyengar 2002) with the proposed change. $f(y_i)$ is the CM measure currently.

Example of Iyengar

An interesting observation. Classification accuracy can be good with high k-values. Iyengar uses 30162 records from a public-use Census file. The eight variables are *age, work class, education, marital status, occupation, race, gender, and native country*. He uses the additional binary variable *salary class* (above or below $50,000). The k-values are 10, 25, 50, 75, 100, 150, 200, 250, and 500. For k=250, the variables *work class, occupation, race, gender* and *native country* are generalized away (put in a single value-state). *Age* is collapsed into {([0,39], (0,∞)}, *education* into {four or more years college, some college, other}, and *marital status* into {married, was married, never married}. Classification accuracy with the k-anonymized data was comparable to the classification with original data (approximately an 18% 10-fold cross-validated error rate).

1/ This paper reports the results of research and analysis undertaken by Census Bureau staff. It has undergone a Census Bureau review more limited in scope than that given to official Census Bureau publications. This report is released to inform interested parties of research and to encourage discussion.

REFERENCES

Aarts, E. and Lenstra, J.K. (1997), *Local Search in Combinatorial Optimization*, Wiley-Interscience: Chichester, UK.

Domingo-Ferrer, J. (2001), "On the Complexity of Microaggregation," presented at the UNECE Workshop On Statistical Data Confidentiality, Skopje, Macedonia, May 2001.

Geman, S. and Geman, D. (1984), "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**, 721-741.

Iyengar, V. (2002), "Transforming Data to Satisfy Privacy Constraints," Association of Computing Machinery, Special Interest Group on Knowledge Discovery and Datamining '02.

Kirkpatrick, S, Gelatt, Jr. C.D., and Vecchi, M.P. (1983), "Optimization by Simulated Annealing," *Science*, **220**, 671-680.

Samarati, P. (2001), "Protecting Respondents' Identity in Microdata Release," *IEEE Transactions on Knowledge and Data Engineering*, **13** (6), 1010-1027.

Samarati, P., and Sweeney, L. (1998), "Protecting Privacy when Disclosing Information: k-anonymity and Its Enforcement through Generalization and Cell Suppression," Technical Report, SRI International.

Sweeney, L. (1999), "Computational Disclosure Control for Medical Microdata: The Datafly System" in *Record Linkage Techniques 1997*, Washington, DC: National Academy Press, 442-453.

Willenborg, L. and De Waal, T. (2000), *Elements of Statistical Disclosure Control*, Vol. 155, Lecture Notes in Statistics, Springer-Verlag, New York.