k-Anonymity Techniques for Privacy Preserving Notes

Module TA: Thanassis Giannetsos
Panos Papadimitratos

December 15, 2014

1 k-Anonymity

The collection of digital information by various entities (i.e., corporations, individuals, etc.) has created tremendous opportunities for knowledge- and information-based decision making. However, data in its original form, typically contains sensitive information about individuals and publishing such data will violate their privacy. The current practice in data publishing relies mainly on policies and guidelines as to what types of data can be published and on agreements on the use of published data. This approach alone may lead to excessive data distortion or insufficient protection. Therefore, publishing data about individuals without revealing sensitive information about them still remains an important problem.

In recent years, a new definition of privacy called k-anonymity [1] has gained popularity. The k-anonymity model was developed because of the possibility of indirect identification of records from public databases. The key idea behind k-anonymity is to build groups of k participants such that they share a common attribute (e.g., k participants located in the same district), rendering them indistinguishable form each other. In this way, in a k-anonymized dataset, each record is indistinguishable from at least $k - 1$ other records with respect to certain "identifying" attributes. Different methods can be used to find an appropriate and common attribute in order to construct groups of k users. These methods can be classified into the three main categories of *generalization*, *supression* and *perturbation* [2].

In the former, the original value of the attribute is generalized by a value with less degree of detail. For example, the exact coordinates of the k participants are replaced by the name of the district of their current location. In supresssion, individual attributes can be deleted; null or replaced with "$*$". In contrast, perturbation is based on replacing the original data by a new value resulting from a function applied to the k attribute values of the participants. For example, the location of each participant can be replaced by the average location of all participants.

Usually a data publisher has a table of the form [3]:

D(Explicit-Identifier, Quasi-Identifier, Sensitive Attributes, Non-Sensitive Attributes),

where *Explicit Identifier* is a set of attributes, such as name and social security number (SSN), containing information that explicitly identifies record owners; *Quasi Identifier* (QID) is a set of attributes that could potentially identify record owners; *Sensitive Attributes* consists of sensitive person-specific information such as disease, salary, and disability status; and *Non-Sensitive Attributes*

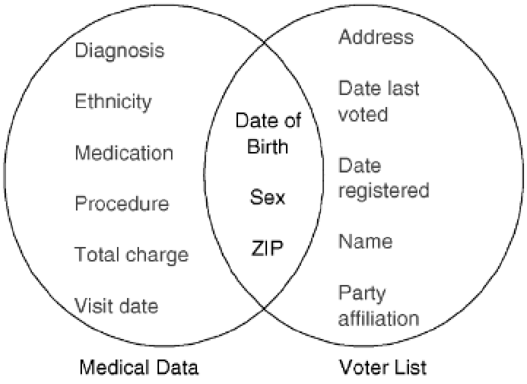


Figure 1: Linkgin to re-identify record owner [1].

contains all attributes that do not fall into the previous three categories. The four sets of attributes are disjoint.

In order to hide the identity and/or the sensitive data of record owners, clearly explicit identifiers must be removed. However, this is not enough; it has been shown, for example, how an individual's name in a public voter list was linked with his record in a published medical database through the combination of zip code, date of birth, and sex. Each of these attributes does not uniquely identify a record owner, but their combination, called the quasi identifier, often singles out a unique or small number of record owners. To perform such linking attacks, the attacker needs two pieces of prior knowledge: the victim's record in the released data and the quasi-identifier of the victim. Such knowledge can be obtained by observation. For example, the attacker noticed that his boss was hospitalized (and wants to know his vote in the voter list), and therefore knew that his boss's medical record would appear in the released patient database (see Fig. 1). Also, it was not difficult for the attacker to obtain his boss's zip code, date of birth, and sex, which could serve as the quasi-identifier in linking attacks.

To prevent linking attacks, the data publisher should provide an anonymous table:

$$T(QID', \text{Sensitive Attributes}, \text{Non-Sensitive Attributes}),$$

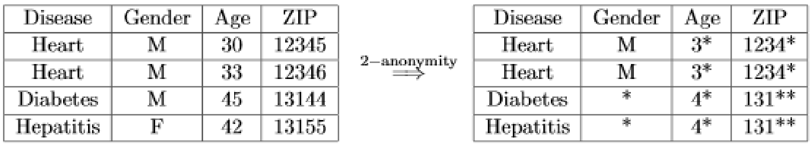
QID′ is an *anonymous* version of the original QID obtained by applying k-anonymity operations to the attributes in QID in the original table. If a person is linked to a record through QID′, that person is also linked to all other records that have the same value for QID′, making the linking ambiguous. In general, a data table T is said to satisfy k-anonymity if and only if each combination of quasi-identifier attributes in T occurs at least k times. The anonymization problem is to produce an anonymous table that satisfies a given privacy requirement determined by the chosen privacy model and to retain as much data utility as possible. Note that the Non-Sensitive Attributes are published if they are important to the data mining task.

For example, let the set {Age, Country, Zip Code} be a QID. Table 1 is one 2-anonymous view of such a dataset since there are five QID-groups and the size

Table 1: A 2-anonymous table view

Age	Country	Zip Code	Health Condition
<30	India	124***	Cancer
<30	India	124***	Cancer
<30	India	1242**	HIV
<30	India	1242**	HIV
>40	America	1110**	Phthisis
>40	America	1110**	Hepatitis
>40	America	1110**	Heart Disease
>40	America	1110**	Asthma
3*	India	1242*	Fever
3*	India	124**	Fever
3*	India	124**	Fever
3*	India	1242*	Indigestion

of each QID-group is at least 2. So k-anonymity can ensure that even though an intruder knows a particular individual in the k-anonymous dataset, she cannot infer which record in s corresponds to the individual with a probability greater than 1/k [4].



Disease	Gender	Age	ZIP
Heart	M	30	12345
Heart	M	33	12346
Diabetes	M	45	13144
Hepatitis	F	42	13155

2−anonymity

Disease	Gender	Age	ZIP
Heart	M	3*	1234*
Heart	M	3*	1234*
Diabetes	*	4*	131**
Hepatitis	*	4*	131**

Figure 2: Another example of 2-anonymity supression.

A risk to k-anonymity is the possibility of homogeneity attacks. Such attacks exploit the monotony of certain attributes to identify individuals from the set of k participants. Therefore, an extension to k-anonymity (termed l-diversity [5]) has been presented which additionally requires the participants to provide at least l different values for an attribute of interest. As a result, at least l distinct values for the sensitive attributes are present within each group, which represents an effective countermeasure to homogeneity attacks.

References

[1] L. Sweeney. 'K-anonymity: A Model for Protecting Privacy'. In: *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.* 10.5 (Oct. 2002), pp. 557–570. ISSN:

0218-4885. DOI: 10.1142/S0218488502001648. URL: http://dx.doi.org/10.1142/S0218488502001648.

[2] K. L. Huang, S. S. Kanhere, and W. Hu. 'Preserving privacy in participatory sensing systems.' In: *Computer Communications* 33.11 (2010), pp. 1266–1280. URL: http://dblp.uni-trier.de/db/journals/comcom/comcom33.html#HuangKH10.

[3] B. C. M. Fung, K. Wang, R. Chen, and P. S. Yu. 'Privacy-preserving Data Publishing: A Survey of Recent Developments'. In: *ACM Comput. Surv.* 42.4 (June 2010), 14:1–14:53. ISSN: 0360-0300. DOI: 10.1145/1749603.1749605. URL: http://doi.acm.org/10.1145/1749603.1749605.

[4] S. C. N and D. G. N. Srinivasan. 'Article: Survey on Recent Developments in Privacy Preserving Models'. In: *International Journal of Computer Applications* 38.9 (2012). Published by Foundation of Computer Science, New York, USA, pp. 18–22.

[5] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. 'L-diversity: Privacy Beyond K-anonymity'. In: *ACM Trans. Knowl. Discov. Data* 1.1 (Mar. 2007). ISSN: 1556-4681. DOI: 10.1145/1217299.1217302. URL: http://doi.acm.org/10.1145/1217299.1217302.