# Draft: A brief Introduction to Database Privacy

April 28, 2017

## 1 Introduction and Motivation

Jojo Hedaya, the CEO of `unroll.me`, a service that offers a ”free” service to scan your mail inbox for unwanted email and automatically ”unsubscribe”, wrote two days ago (23 Apr 2017) after it's users found out that the company is analyzing their entire email and selling aggregate results to other companies: [1]

> [...] it was heartbreaking to see that some of our users were upset to learn about how we monetize our free service. [...] Sure we have a Terms of Service Agreement and a plain-English Privacy Policy that our users agree they have read and understand before they even sign up, but the reality is most of us - myself included - don't take the time to thoroughly review them.

From a report titled “Personal Data: The Emergence of a New Asset Class” [13], published by the World Economic Forum in 2011:

> Too much transparency too soon presents as much a risk to destabilising the personal data ecosystem as too little transparency;

And from the same report:

> Most end users still remain unaware of just how much they are tagged, tracked and followed on the Internet. Few individuals realise how much data they implicitly give away, how that data might be used or even what is known about them. [...] When customers suddenly find out how their trusted brand of product or service was gathering and using their personal data, they tend to react with outrage, rather than reward the business for its transparency. Similarly, citizens fear Big Brother control and manipulation in the way government uses their personal information. As long as the risk of transparency outweighs the rewards, the personal data ecosystem will remain vulnerable to periodic seismic shocks.

[1] `http://blog.unroll.me/we-can-do-better/`

## 1.1 Database Privacy

When we talk about confidentiality in computer security we usually attempt to prevent someone from learning confidential information. We do this by making sure users are who they say they are (authentication), restricting access to authorized users (access control) and make information undecipherable when we can not guarantee the former (encryption).

But sometimes we want (on purpose) to make information available to ”unauthorized” users and thus have to de-classify this information. For example, consider the medical database of a hospital and a set of researchers wanting to do research on this data. Or a marketing company that collects detailed browsing profiles of web users and stores the result in an “anonymized” form. Our goal is then to provide a *sanitized* version of this database to those researchers and the confidential information that we seek to protect is the individual diagnosis data of patients. Towards this end we consider the following two threats: disclosure of identity (finding out the patient behind a record) and disclosure of attribute (finding out the diagnosis of a particular patient). It turns out that both are surprisingly difficult problems.

## 1.2 Positive Example: Randomized Response to hide Attributes

To begin with a positive example we consider something that has been applied by social scientist for decades: randomized response (taken from [3]). Consider the problem of asking a set of participants a Yes/No question who might not want to reveal the true answer, for example out of fear for embarrassment or legal repercussions. The goal is to provide a statistic about the fraction of Yes answers. Instead of giving a direct response, each participant is asked to follow the following protocol:

1. flip a (fair) coin,
2. if the result is tails, then respond truthfully
3. if the result is heads, flip a second coin and respond with *Yes* (heads) or *No* (tails).

This gives each participant not only some kind of “plausible deniability”, but also guarantees that the interviewer can not deduce the true result of one particular participant. Table 1 shows an example of a dataset (left side) and the true values and hidden coin flips (right side). This protocol does not directly give the true result of Yes answers, but given a “large enough” number of participants it is easy to estimate the true result with high confidence (see tutorials).

Notice that Table 1 also hides user identifiers by giving a generic number, and not an identifier that is related to an individual.

| Participant identifier | response | (true value) | coin flip 1 | coin flip 2 |
|---|---|---|---|---|
| 1 | Yes | Yes | tails | - |
| 2 | No | No | heads | tails |
| 3 | Yes | Yes | heads | heads |
| 4 | Yes | Yes | tails | - |
| 5 | No | No | tails | - |
| 6 | Yes | Yes | heads | - heads |
| 7 | No | No | tails | - |
| 8 | Yes | No | heads | - heads |

Table 1: Shows a table of reported results to a Yes/No question (columns on the left), where the true values of participants are "hidden" using the randomize response technique.

### 1.3 Negative examples: Naïve "anonymization" of Identifiers

In our previous example we mainly concentrated on hiding *attributes*. We now want to take a brief look on identifiers and give some negative examples that are still applied in practice. One problem of particular relevance is the anonymization of IP-Addresses in log files.

**Hashing Identifiers** Lets say we have a log file that looks like the following table and we want to anonymize the left side, the IP-Address.

| id | value |
|---|---|
| `23.66.121.155` | `/submit.cgi` |
| `77.87.229.22` | `/submit.cgi` |

One easy approach is to just apply a "strong" cryptographic hash function $H$ (here: SHA-128) to the IP-Address and store the result $H(\text{id})$ instead.

| id | value |
|---|---|
| `6aa9c8d5f4fa4fb0cc560c0f96a24ec53d5ec1ae` | `/submit.cgi` |
| `d3c348d35dead7a5d9e00a9596a47d7f2a7d8b36` | `/submit.cgi` |

However, IPv4-Addresses are only 32 bit long (less than $2^{32}$ values), and a brute-force attack (trying every IP-Address until the hash matches) is far from infeasible.

Truncating the result to, say 20 bit, will help since this reduces information and will cause *collisions* (two different IP-Addresses will map to the same hash value). However, given some background knowledge (the country of the IP-Address, for example), it might still be possible to rule out candidates. A similar idea applies to truncation of the IP-Address, for example by replacing the last 8-bit with 0.

While not giving sufficient privacy guarantees, the collisions furthermore reduces the utility of the identifier, which might hurt the purpose this log file is kept at all.

| id | query |
|---|---|
| 4417749 | big cuddly dog |
| 4417749 | jarrett t. arnold eugene oregon |
| 4417749 | loneliness |
| 4417749 | numb fingers |
| 4417749 | 60 single men |
| 4417749 | dog that urinates on everything |
| 4417749 | landscapers in Lilburn, Ga |
| 4417749 | homes sold in shadow lake subdivision gwinnett county georgia. |

Figure 1: Search queries of AOL user number 4417749 aka Thelma Arnold (excerpts).

**Random identifiers** A better approach is to replace each IP-Address with a *randomly* chosen pseudonym and "forget" the mapping between identifier and pseudonym afterwards. This can easily be implemented by adding a long (length depending on the hash function) secret prefix to the IP-Address and storing a hash value $H(\text{prefix}||id)$ similar to before.

However, practice has shown that even if the prefix is kept hidden and is securely erased after use (which is difficult enough), concentrating on the *immediately apparent* identifier like the IP-Address is not enough.

In 2006, AOL released a dataset that contained detailed records of search queries that has been "anonymized" using random identifiers. Table 1 shows an excerpt from this dataset. It did not take long for the New York Times to find out who the real person behind user #4417749 is, who exposed herself due to the contents of her search queries. You can read more about this story (and have a look in the database) at `http://search-id.com/aol/about`.

## 1.4 More de-anonymization

There has been considerable amount of examples where persons have been identified in a supposedly "anonymized" database, and we now want to briefly review a few of them. There is still a debate going on [2], where some claim that de-identification/anonymization of data is practical and possible, while others argue that this is not the case.

**Voter Records Deanonymization** Latyana Sweeny found in 2000 [11] that in the United States, the ZIP code, birth date and sex uniquely identify 87% of the population. Given the information about american voters (see Table 2) it was then easy to deanonymize

[2] See `https://fpf.org/2017/02/03/fpf-brussels-law-science-de-identification/` for a recent discussion

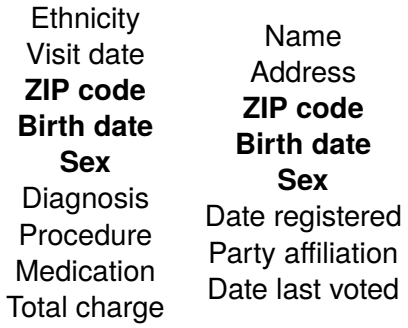


Figure 2: Excerpts of attributes of two public databases: one containing "Anonymous" Medical Data (left) and one containing voter data (right).

a previously published "anonymous" database containing medical data. Philippe Golle revisited this study in 2006 [5] and was able to identify 63% of the population of the US.

**Netflix Deanonymization** In part of a challenge to improve it's services, Netflix published 2006 a database of about 100,000,000 movie ratings of about 500,000 Netflix users (about $1/8$ of the userbase). The task of the challenge was to improve Netflix's movie recommendation service. Netflix assured it's customers on it's FAQ page that:

> No, all customer identifying information has been removed; all that remains are ratings and dates. [...] Even if, for example, you knew all your own ratings and their dates you probably couldn't identify them reliably in the data because only a small sample was included (less than one-tenth of our complete dataset) and that data was subject to perturbation. Of course, since you know all your own ratings that really isn't a privacy problem is it? [9]

In 2008, Narayanan and Shmatikov [9] used user rating information publicly available from the IMDb database to de-anonymize this dataset. They used ratings and dates to link Netflix subscribers to IMDb accounts:

> With 8 movie ratings (of which 2 may be completely wrong) and dates that may have a 14-day error, 99% of records be uniquely identified in the dataset. [9]

**Social Network Deanonymization** In 2011, the machine-learning competition platform `kaggle.com` offered a challenge to the research community to predict links (future interactions) in social networks.

> The contest dataset was created by crawling a large online social network and partitioning the obtained edge set into a large training set and a smaller test

> set of edges augmented with an equal number of fake edges. Challenge entries were required to be probabilistic predictions on the test edge set. Node identities in the released data were obfuscated to prevent cheating. [8]

The data was taken from a partial crawl of the Flickr social photo sharing network. In the same year, Narayanan et al. [8] did a partial crawl of Flickr on their own, "were able to successfully de-anonymize 64.7% of the test edge-set", applied some basic machine-learning to the remaining test edge-set and subsequently won the competition.

**Web-History Deanonymization** Web trackers and advertisers assure internet users that data about their browsing habits is only kept in anonymized form, with any identifying information removed (which is questionable in itself, see AOL search disaster).

In 2017, Jessica Su et al. [10] hypothesized that social media users are more likely to view web pages that are linked in their social media stream than users who don't see these links on social media. This information is often publicly available (see the list of followers on twitter) and so this raised the question if an anonymized browsing history can be de-anonymized using this information. From their paper:

> We formalize this intuition by specifying a model of web browsing behavior and then deriving the maximum likelihood estimate of a user's social profile. [...] To gauge the real-world effectiveness of this approach, we recruited nearly 400 people to donate their web browsing histories, and we were able to correctly identify more than 70% of them. [...] Finally, since our attack attempts to find the correct Twitter profile out of over 300 million candidates, it is - to our knowledge - the largest-scale demonstrated de-anonymization to date.[10]

## 2 Database anonymization, $k$-anonymity and improvements

Following her de-anonymization of medical data in 2000, Sweeney suggested in 2002 to use the measure of $k$-anonymity to provide database privacy [12].

**Definition 1** (Table, records and attributes)**.** *Let $T = (t_1, \dots, t_m)$ be a table with records $t_1, \dots, t_n$ over attributes $A_1, \dots A_n$. This means that every record $t \in \{t_1, \dots, t_n\}$ is of the form*

$$t = (a_1, \dots, a_n),$$

*with $a_i \in A_i$. When $Q \subseteq \{A_1, \dots, A_n\}$ is a set of attributes, we write $T[Q]$ to denote the projection of $T$ to the table that only contains the attributes in $Q$.*

**Definition 2** (Quasi-identifier)**.** *Let $U$ be a set of entities (people) that have records in $T$, $f_c$ be a function that maps an entity $u \in U$ to the corresponding entry in $T$ and $f_g$ be a function*

| Nr. | ZIP Code | Age | Disease |
|---|---|---|---|
| 1 | 47677 | 29 | Heart Disease |
| 2 | 47602 | 22 | Heart Disease |
| 3 | 47678 | 27 | Heart Disease |
| 4 | 47905 | 43 | Flu |
| 5 | 47909 | 52 | Heart Disease |
| 6 | 47906 | 47 | Cancer |
| 7 | 47605 | 30 | Heart Disease |
| 8 | 47673 | 36 | Cancer |
| 9 | 47607 | 32 | Cancer |

Table 2: Original dataset (from [6])

| Nr. | ZIP Code | Age | Disease |
|---|---|---|---|
| 1 | 476** | 2* | Heart Disease |
| 2 | 476** | 2* | Heart Disease |
| 3 | 476** | 2* | Heart Disease |
| 4 | 4790* | $\geq 40$ | Flu |
| 5 | 4790* | $\geq 40$ | Heart Disease |
| 6 | 4790* | $\geq 40$ | Cancer |
| 7 | 476** | 3* | Heart Disease |
| 8 | 476** | 3* | Cancer |
| 9 | 476** | 3* | Cancer |

Table 3: 3-anonymous dataset (from [6])

*that links records in $T$ to entities $U$. We say a set of attributes $Q$ is a pseudo-identifier, if there exists an identity $u \in U$ so that*

$$f_g(f_c(u)[Q]) = u,$$

*that is that $Q$ uniquely identifies one entity $u$.*

**Definition 3** ($k$-anonymity)**.** *Let $T$ be a table with attributes $A_1, \ldots, A_n$ and let $Q$ be a quasi-identifier of $T$. We say table $T$ satisfies $k$-anonymity, iff each sequence of values in $T[Q]$ appears with at least $k$ occurrences in $T[Q]$.*

*All values in $T$ that share the same values in $T[Q]$ are said to be in the same equivalence class.*

Table 2 shows an example of a patient database. The quasi-identifier for this database is $\{$Zip Code, Age$\}$. Using this quasi-identifier, Table 3 shows an altered table that satisfies 3-anonymity.

## 2.1 $\ell$-diversity

Machanavajjhala et al. remarked in 2007 [7] that $k$-anonymity may still leak information about individuals. They observed two possible attacks:

**Homogenity Attack on $k$-anonymity** Consider Table 3 again. In the first 3-anonymous set of patients there is no need to identify one particular person, since the attribute one wants to protect is the same for each one in this anonymity set (Heart Disease).

**Background Knowledge Attack on $k$-anonymity** Again consider Table 3 and suppose you know that someone is in the last 3-anonymous set and has a-priori low risk for heart

| Equivalence class | Nr. | Virus Test-result | Aggregate |
|---|---|---|---|
| 1 | 1 | + | 50:50 (+,-) |
| 1 | ... | ... | |
| 1 | i | + | |
| 2 | i+1 | − | 99:1 (+,-) |
| 2 | ... | ... | |
| 2 | j | + | |
| 3 | j+1 | − | 1:99 (+,-) |
| 3 | ... | ... | |
| 3 | k | + | |

Table 4: Shows three equivalence classes. The second and third class have the same 2-diversity with every metric defined in [7], but being in the second class would be considered more sensitive than in the third class.

disease (for example due to external demographic information). Then you can conclude with high probability that this person is more likely to have cancer.

In order to address these proplems the authors propose to concentrate on the attributes one wants to protect (called sensitive attributes) and suggest a new metric, $\ell$-diversity.

**Definition 4** (Informal: Sensitive Attributes)**.** *An attribute is called sensitive, if an adversary must not be allowed to discover the value of that attribute for any individual in the dataset.*

**Definition 5** (Informal: $\ell$-diversity Principle)**.** *An equivalence class has $\ell$-diversity if there are at least $\ell$ "well-represented" values for the sensitive attributes. A table $T$ has $\ell$-diversity if every equivalence class of the table has $\ell$-diversity.*

The definition of $\ell$-diversity contains the informal term "well-represented values". The authors do not propose one particular definition for this term, but rather provide multiple ideas how one could understand it. The simplest one ensures that there are at least $\ell$ distinct values for sensitive attributes in each equivalence class.

## 2.2 $t$-closeness (brief)

Li et al. remarked in [6] in the same year that $\ell$ diversity does not take into account that different values for the same attribute may be considered more or less sensitive. The $\ell$-diversity model does also not take into account that some values may be semantically similar.

**Skewness Attack on $\ell$-diversity** The problem with the term "well-represented" in $\ell$-diversity is that it is difficult to define if a particular probability distribution is sensitive, or not. Consider Table 4. The data shows three different equivalence classes. If one would

| Equivalence class | Nr. | Virus test-result |
|---|---|---|
| 1 | 1 | gastric ulcer |
| 1 | 2 | gastritis |
| 1 | 3 | stomach cancer |
| 2 | 4 | gastritis |
| 2 | 5 | flu |
| 2 | 6 | bronchitis |
| 3 | 7 | bronchitis |
| 3 | 8 | pneumonia |
| 3 | 9 | stomach cancer |

Table 5: Shows a 3-diverse table, where the first equivalence class discloses stomach-related problems.

discover that one particular person is in one equivalence class, $\ell$-diversity should promise that nothing about the sensitive information is disclosed about this person. However, in this example being part of the second equivalence class means that the probability is higher that one has a positive test result, than in the third class.

**Similarity Attack on $\ell$-diversity** While Table 5 is 3-diverse based on a simply counting-metric, the first equivalence class only contains stomach-related problems. So being in this equivalence class will leak information to due the semantic similarity of the attribute values.

In order to address these kind of attacks the authors of [6] propose the notion of $t$-closeness, which ensures that the distribution of values in each equivalence class does not differ by much with respect to the whole table.

**Definition 6** (Informal: $t$-closeness Principle)**.** *An equivalence class is said to have t-closeness if the distance between the distribution of a sensitive attribute in this class and the distribution of the attribute in the whole table is no more than a threshold t. A table is said to have t-closeness if all equivalence classes have t-closeness.*

One major drawback of this approach is that small values for $t$ greatly reduce the utility of the entire table for data analysis. If one wants to discover correlations between attributes in equivalence classes, the notion of $t$-closeness directly influences what can be learned from such a table.

# 3 $(\varepsilon, \delta)$-differential privacy

Cynthia Dwork followed a slightly different approach and, together with other researchers, developed *differential privacy* in the timeframe from 2004 to 2006 [1]. In previous attempts

we have seen that background knowledge allows for different kinds of attacks, and it seems very difficult to estimate the amount of background knowledge an adversary has.

The goal of differential privacy is to provide a mathematically provable guarantee about what an adversary can learn, with respect to any amount of background knowledge the adversary might possess. In order to do that, it quantifies the influence one particular participant has in the resulting dataset.

The first basic result is that at long as there is some utility about the data published, there is *no perfect* protection of privacy in the cryptographic "perfectly secret" sense. Her paper from 2006 provides a formal argument that we skip here, but also gives an easy to understand analogy:

> Suppose one's exact height were considered a highly sensitive piece of information, and that revealing the exact height of an individual were a privacy breach. Assume that the database yields the average heights of women of different nationalities. An adversary who has access to the statistical database and the auxiliary information "Terry Gross is two inches shorter than the average Lithuanian woman" learns Terry Gross' height, while anyone learning only the auxiliary information, without access to the average heights, learns relatively little [1].

The definition of differential privacy thus attempts to give a measure on the *additional amount* of information (measured in the parameters $\varepsilon$ and $\delta$) that is disclosed about one individual by that individual participating in the database. Of course, one still has to decide which values for $\varepsilon$ and $\delta$ are acceptable.

**Definition 7** ($(\varepsilon, \delta)$-differential privacy)**.** *A randomized function $K$ gives $(\varepsilon, \delta)$-differential privacy if for all data sets $D_1$ and $D_2$ differing on at most one element, and all $S \subseteq Range(K)$,*

$$\Pr[K(D_1) \in S] \leq \exp^{\varepsilon} \Pr[K(D_2) \in S] + \delta.$$

The randomized response technique we have seen before can be proven to give a $(\ln(3), 0)$ differential privacy guarantee (see tutorials). And as of now, differentially private guarantees are already provided by some hightly practical implementations. For example, see the paper about RAPPOR [4], a technique implemented in the data collection factily in the Google Chrome Web Browser, that uses this randomized response technique in order to give a differential privacy guarantee.

But differential privacy has also found more applications besides protecting privacy, for example to solve practical problems in the application of statistics or machine learning [2]. See the book by Cynthia Dwork and Aaron Roth [3] for a more detailed introduction and discussion.

## References


[1] Cynthia Dwork. Differential privacy. In *33rd International Colloquium on Automata, Languages and Programming, part II (ICALP 2006)*, volume 4052, pages 1–12, Venice, Italy, July 2006. Springer Verlag.

[2] Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. Guilt-free data reuse. *Commun. ACM*, 60(4):86–93, March 2017.

[3] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

[4] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pages 1054–1067. ACM, 2014.

[5] Philippe Golle. Revisiting the uniqueness of simple demographics in the us population. In *Proceedings of the 5th ACM Workshop on Privacy in Electronic Society*, WPES '06, pages 77–80, New York, NY, USA, 2006. ACM.

[6] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, pages 106–115. IEEE, 2007.

[7] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):3, 2007.

[8] Arvind Narayanan, Elaine Shi, and Benjamin IP Rubinstein. Link prediction by de-anonymization: How we won the kaggle social network challenge. In *Neural Networks (IJCNN), The 2011 International Joint Conference on*, pages 1825–1834. IEEE, 2011.

[9] Arvind Narayanan and Vitaly Shmatikov. Robust de-anonymization of large sparse datasets. In *Security and Privacy, 2008. SP 2008. IEEE Symposium on*, pages 111–125. IEEE, 2008.

[10] Jessica Su, Ansh Shukla, Sharad Goel, and Arvind Narayanan. De-anonymizing web browsing data with social networks. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1261–1269. International World Wide Web Conferences Steering Committee, 2017.

[11] Latanya Sweeney. Simple demographics often identify people uniquely. *Health (San Francisco)*, 671:1–34, 2000.

[12] Latanya Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.

[13] World Economic Forum. Personal data: The emergence of a new asset class, 2011.