Towards an in-depth understanding of privacy parameters for randomized sanitization mechanisms

Baptiste Olivier
Orange Labs
baptiste.olivier@orange.com

Tony Quertier
Orange Labs
tony.quertier@orange.com

***Abstract*—Differential privacy, and close other notions such as $d_{\mathcal{X}}$-privacy, is at the heart of the privacy framework when considering the use of randomization to ensure data privacy. Such a guarantee is always submitted to some trade-off between the privacy level and the accuracy of the result. While a privacy parameter of the differentially private algorithms leverages this trade-off, it is often a hard task to choose a meaningful value for this numerical parameter. Only a few works have tackled this issue, and the present paper's goal is to continue this effort in two ways. First, we propose a generic framework to decide whether a privacy parameter value is sufficient to prevent from some pre-determined and well-understood risks for privacy. Second, we instantiate our framework on mobility data from real-life datasets, and show some insightful features necessary for practical applications of randomized sanitization mechanisms. In our framework, we model scenarii where an attacker's goal is to de-sanitize some data previously sanitized in the sense of $d_{\mathcal{X}}$-privacy, a privacy guarantee close to that of differential privacy. To each attack is associated a meaningful risk of data disclosure, and the level of success for the attack suggests a relevant value for the corresponding privacy parameter.**

I. Introduction

As big data processing is now a reality, more and more statistical information is extracted from rich databases. This straightforwardly paves the way to new valuable applications for transport, health, research and business [1] [2], but it also yields many severe privacy concerns. Indeed, many personal (and sometimes private) information can be retrieved from large datasets: some mobility patterns of an individual permit to guess his home location, work location, to infer the most (and the least) visited places during a trip... In such situations, there is clearly a threat for the individual privacy. Even worse, individual private information may be retrieved from sanitized datasets, usually performing data correlation on multiple datasets coming from multiple sources [3] [4] [5] [6].

Sanitization mechanisms aim at processing datasets in order to release information with a certain privacy guarantee. The latter is usually defined by some privacy definition, associated with a privacy level depending on some parameter(s). One well-known privacy guarantee is *k-anonymity* [7] and asserts that the quasi-identifiers attributes of an individual are indistinguishable from those of $k-1$ others individuals. For k-anonymity, parameter k leverages the privacy level. Regarding randomized sanitization mechanisms, differential privacy is by far the most studied property [8]. The notion of ϵ-differential privacy provides the guarantee that two neighbor datasets (differing from only a single individual) are ϵ-close one from each other in distribution. Parameter ϵ, a positive numerical value, determines the level of privacy.

A privacy level parametrized by a single numerical value seems attractive at a first glance, but it appears to be highly non-trivial when deploying differentially private algorithms in practice. For instance, a single privacy risk (e.g. re-identification) can be observed in two different scenarii (different datasets, different queries), while a common value ϵ for the privacy parameter correspond to two distinct levels of privacy. Even in the case of a single scenario, there is today no ad-hoc protocol to choose the value ϵ. In particular, it is not clear how to associate a comprehensive notion of risk (re-identification, well-targeted statistics inference...) to this parameter value.

In this paper, we provide an end-to-end system to choose meaningful (for privacy concerns) and useful (for statistical utility) values for privacy parameters of a sanitized mechanism (e.g. ϵ for differential privacy). We believe that a natural way to choose privacy parameters in a secure manner is to understand to which extent a chosen value prevents from well-known *privacy* attacks on datasets (re-identification, inference...). Our model aims at modelizing such attacks, and for doing so involves different parties: users who publish their data in some sanitized form, that we call sanitized profiles; data miners who extract interesting statistical features from users sanitized profiles; and an attacker whose goal is to perform privacy attacks, using published sanitized profiles and possibly some side information.

We instantiate our system on mobility data extracted from Call Detail Records. Our user-level sanitization process is designed to produce profiles satisfying $d_{\mathcal{X}}$-privacy [9] [10], a privacy guarantee based on randomized techniques and designed to provide a certain level of obfuscation for each user's profile. The latter level depends on privacy parameters used to define the $d_{\mathcal{X}}$-privacy guarantee. Our system provides insightful results regarding the use of $d_{\mathcal{X}}$-privacy in practice, and in particular the choice of relevant values for the associated privacy parameters.

A. Our main contributions

Here is a list of the main contributions in the present work:

- We propose a model to choose privacy parameters for practical use of sanitization mechanisms, and in particular those using randomization. The model confronts sanitized profiles with both statistical constraints, and well-known privacy attacks. Instantiating this system on real datasets permits to understand more clearly the choice of privacy parameters for randomized sanitization techniques such as $d_{\mathcal{X}}$-privacy, when applied on some mobility data. To our knowledge, this is the first time that the security related to parameters of $d_{\mathcal{X}}$-privacy is studied.

- We provide an instantiation of our system on mobility data, extracted from Call Detail Records. We find enlighting results regarding the admissible ranges of privacy parameters.
- We design new mechanisms to achieve $d_{\mathcal{X}}$-privacy in the case where each profile is represented by a vector. Although these mechanisms are highly inspired by well-known techniques introduced in the context of differential privacy, we provide a new parametrization adapted for $d_{\mathcal{X}}$-privacy (see the choice of subset $\mathcal{X} \subset \mathbb{R}^m$ in Section III). We show that this new privacy parameter can provide interesting benefits for a $d_{\mathcal{X}}$-private mechanism, when it is well-chosen.

B. *Organization of the paper*

In Section II, we describe the overview of our system. We discuss in more details randomized sanitization mechanisms in Section III. The adversarial model is described in Section IV, while some instantiations of attacks are provided in Section V. In Section VI, we expose our experimental results and analyze them. Sections VII and VIII are devoted to the related works to ours and the conclusion.

II. Our model for enhancing the application of sanitized mechanisms

In this section, we explain the model of our system. We have instantiated this framework for mobility features extracted from real Call Detail Records. As easily seen from the description below, our approach generalizes to many other applications.

A. *Users profiles*

We will consider only some of the information related to users data in Call Detail Records (CDRs). Each Call Detail Record contains, among other attributes, the following tuple (user id, POI id, time), where a Point Of Interest (POI) corresponds to the location of an antenna, hence serving as an approximation of user's location. Such information extracted from the CDRs of a user is further represented as a vector of numerical values, that we call the user's *profile*. Since there are many possible representations for users profiles, we will only focus on some natural profiles, defined in the examples below.

The set of users will be indexed by $[1, n]$ in a model involving n users, and the user with index i will be referred as u_i. The set of POIs will be indexed by $[1, m]$, m being the total number of POIs in our datasets.

Example 1. (boolean profile)
An important family of profiles we will use is that of *boolean profiles*. The latter encompass information described by means of boolean predicates.

More formally, a *boolean profile* is a profile $b = (b_j)_j$ such that for all j, $b_j = 0$ or $b_j = 1$ depending on a boolean condition.

In the sequel, we will use the following definition of boolean profiles b_i: $b_{ij} = 1$ if and only if user u_i has visited POI j during the observation period. Then vector $b_i = (b_{ij})_j$ encodes the presence or not of user u_i at some places.

Example 2. (count profile)
Count profiles (or also histograms in the literature) are more precise than boolean profiles. They are obtained by extracting frequency information as follows

$$c_i = (c_{ij})_j$$

where c_{ij} is the number of times user u_i has visited POI j during the observation period.

If the context does not precise if boolean or count profiles are at stake, we will use the notation $a_i = (a_{ij})_j$ for user u_i's profile. The notation $a_i^T = (a_{ij}^T)_j$ will be used if we want to specify the observation period T during which the profile is computed.

As will be proved in the paper, profiles may be highly sensitive regarding users privacy. For instance, a boolean profile b_i may be used as a fingerprint to re-identify user u_i. In worst cases, not only they permit to re-identify the corresponding user, but they also allow to infer private personal information about a user.

B. *Sanitization of users profiles*

In the current paper, we assume that each user performs the sanitization task locally (on his/her device), and then publishes a sanitized version of his/her profile. The sanitized version of user u_i's profile $a_i = (a_{ij})_j$ will be denoted by $\tilde{a}_i = (\tilde{a}_{ij})_j$. Sanitized profile $\tilde{a}_i$ will be also called *fake profile* in the sequel.

Applications to sanitize data locally already exist (LocLok [11], Location Guard [12], fake GPS locations applications...) and their number will certainly grow in the coming years. The main obvious reason for a user to apply them is to protect his/her privacy. But such a guarantee has a cost, reflecting the usual trade-off between data privacy and statistical utility: for instance, completely fake locations (such as for the application Fake GPS) make impossible the use of additional services based on locations (LBS). A user may also share his/her data for purposes related to health, research advances, while protecting his/her privacy. In such a case also, some minimal accuracy is required.

The situations described above require advanced sanitization mechanisms, capable to propose a parametrizable trade-off between the privacy level and the accuracy of the profile. For that reason, we chose $d_{\mathcal{X}}$-privacy as the targeted privacy guarantee for our model (see Section III for details).

C. *Extracting statistical features from profiles*

The motivation for publishing (sanitized) users profiles is to derive some interesting statistical features about users behaviours. In our model, this is the role of the *statistician*, whose goal is to compute global statistics about the population. The statistician performs the computations on sanitized profiles, and so the accuracy of the results will depend on the level of privacy used in the sanitization phase (that is on the privacy parameters values). We introduced the statistician in our model in order to bound the range of privacy parameters values to those that can provide potentially meaningful results for some data-mining purposes.

Here we illustrate on two examples relevant statistical objectives that can be computed from boolean and count profiles from CDRs (see Examples 1 and 2).

Example 3. *Let $(b_i)_{1\leq i\leq n}$ be n boolean profiles on CDRs, as described in Example 1. We define $f_1 : [1, m] \to \mathbb{R}$ such that $f_1(j)$ is the number of visitors on POI j during the observation period, that is*

$$f_1(j) = \sum_{1\leq i\leq n} b_{ij}.$$

Since the previous quantity depends on the collection of profiles $(b_i)_i$, we will write $f_1((b_i)_i, j)$ instead of $f_1(j)$ to clarify the context if necessary.

Example 4. *Let $(c_i)_{1\leq i\leq n}$ be n count profiles on CDRs, as described in Example 2. We define $f_2 : [1, m] \to \mathbb{R}$ such that $f_2(j)$ is the average frequency of visits on POI j during the observation period, that is*

$$f_2(j) = \frac{1}{n} \times \sum_{1\leq i\leq n} c_{ij}.$$

D. Attacks on profiles

The main motivation when designing our system was to introduce the role of an attacker, referred as $\mathcal{A}$ in the sequel. As said before, $\mathcal{A}$ is assumed to perform some well-understood attacks on sanitized data, such as re-identification or inference of the most visited POI, and then associate a level of risk (re-identification rate, ...) to the sanitization process at stake. As the quality of the sanitization mechanism depends on the privacy parameters values, we directly relate a privacy parameter value to a tractable level of risk.

In the attacks, we will provide to the attacker $\mathcal{A}$ an access to published sanitized profiles $(\tilde{a}_{ij})_{i\in I, j\in J}$ (for some subsets $I \subset [1, n]$, $J \subset [1, m]$), together with some auxiliary information. The auxiliary information is modelled as a collection of *global* statistics on the population of users at stake (see Section V for more details). Side information may be available to some malicious eavesdropper in real-life situations: data leakage, exposure of personal data in public places...

E. How the different roles are combined

Given some sanitized mechanism, the overall goal of our system is the following: for each choice of privacy parameters, and for each identified risk for privacy, associate a level of success for the attack instantiating that risk.

In a first phase, sanitized profiles are obtained from users profiles, using a pre-determined sanitization mechanism. As shown in Figure 1, the sanitization mechanism is parameterized by some privacy parameters. Note that while these profiles are supposed to be published with the users contents, there

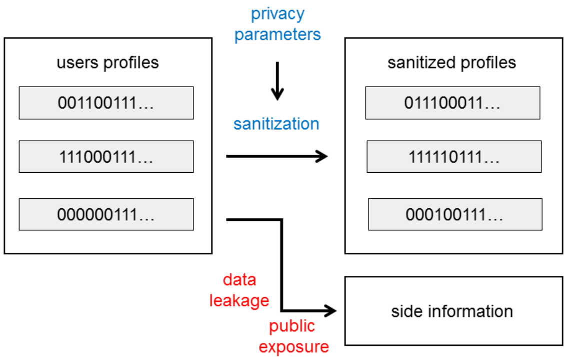

Fig. 1. To sanitize his/her profile, each user is asked to choose privacy parameters values.

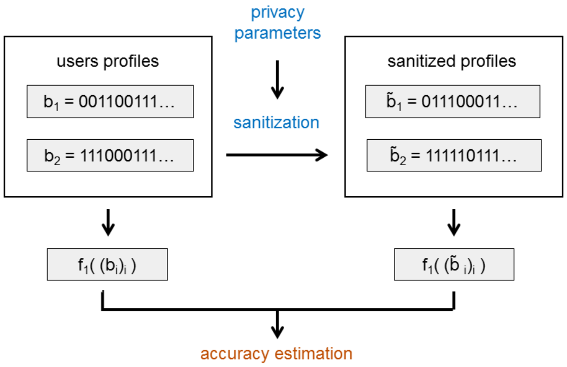

Fig. 2. Comparing the same statistics over non-sanitized profiles and over sanitized profiles, a statistician decides the range of privacy parameters which is reasonable for statistical purposes.

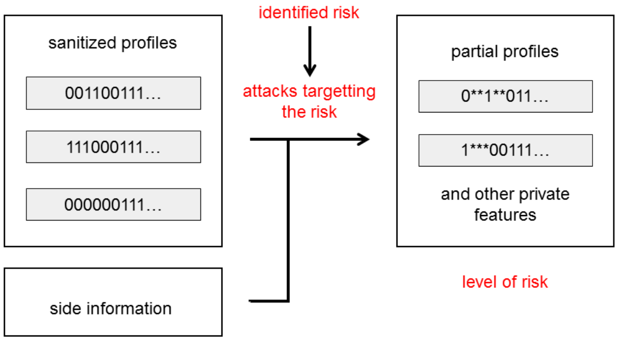

Fig. 3. Each attack is associated to some well-known risk (re-identification, inference of important features). An attack succeeds with more or less success, depending on the choice of privacy parameters.

may be other auxiliary information available to a malicious de-sanitizer, on which the user has less control: public exposure (social networks, ...), data leakage from the true profiles.

A second phase confronts the sanitization mechanism to the statistician. Outputs are validated against some utility metrics: statistics over true profiles are compared to statistics over sanitized profiles. Then sanitization mechanism is deployed only for privacy parameters lying in some admissible range, in which sanitized profiles provide *reasonably* useful statistics. This phase is described in Figure 2.

During a third phase described in Figure 3, given sanitized profiles and side information as parameters, the system provides a level of privacy risk related to some well-known threat. Indeed, this phase simulates the attacker's point of view: having some purpose against users privacy, modelized by the identified risk (e.g. re-identification), an attacker exploits both side information and sanitized profiles in order to perform privacy attacks. Such an attack will succeed with some rate success, which finally will be used to estimate the level of risk. An important point is that the nature of the attack, and so the associated success rate, are tractable information which can be handled by a standard management of privacy risks.

Our system combines the three previous steps together, and provides a mapping of privacy parameters to the success rate of a well-known attack on privacy. The rest of the paper is devoted to the instantiation of this generic system:

- Phase 1 is performed on boolean and count profiles described in Examples 1 and 2, with d_χ-private sanitization mechanisms depending on privacy parameters $(\chi_i, \epsilon_i)_i$ (see the next section for explanations);
- Phase 2 will be validated by means of statistics introduced in Examples 3 and 4;
- Phase 3 will be instantiated on several identified risks with a panel of attacks (e.g. re-identification, guess of most visited place), providing numerical values of attacks success rates.

III. Sanitization mechanisms

First, we introduce and discuss the privacy guarantee used in the paper, that is d_χ-privacy. Then we provide sanitization mechanisms satisfying d_χ-privacy for both boolean and count profiles.

A. About d_χ-privacy

In all this paper, data sanitization will be achieved by means of randomized mechanisms. The main privacy guarantee at stake, called d_χ-privacy, is closely related to the well-known notion of *differential privacy*. The concept of d_χ-privacy was first introduced as *geo-indistinguishability* in [10] to manage location obfuscation for planar data points. We extend the scope of these obfuscation techniques to the case of general vectors (extension from $\mathbb{R}^2$ to $\mathbb{R}^m$), with the objective to apply these techniques to users profiles (see Section II). We will denote by $P(A)$ the probability that an event A occurs.

Definition 5. *([10]) Let χ be a subset of $\mathbb{R}^m$, and let d_χ be a metric on χ. A randomized mechanism $M : \chi \to \mathbb{R}^m$ is said to be d_χ-private if for all $a, a' \in \chi$, and all $z \in \mathbb{R}^m$, we have*

$$\frac{P(\ M(a) = z)}{P(\ M(a') = z)} \leq e^{d_\chi(a,a')}.$$

When this condition is satisfied, mechanism M is also said to be *geo-indistinguishable* [9], abreviated as $Geo - I$ in the subsequent literature. The terminology d_χ-privacy is more adapted for our use than geo-indistinguishability, since the latter refers explicitly to the initial use-case introduced in [9], that is the obfuscation of spatial coordinates. In the current paper, we will use d_χ-privacy for our more general notion of profiles, that can instantiate spatial coordinates, but not only, as shown by Examples 1 and 2.

The idea of d_χ-privacy is that a mechanism satisfying such privacy guarantees should satisfy some continuity constraints: two close (in the sense of metric d_χ) profiles should impose some closeness on the corresponding distributions after applying the mechanism. This concept was introduced with differential privacy, where the closeness was measured by the differential between two datasets. Indeed, it has already been noticed [13] that the choice $d_\chi = \epsilon \times d$ is simply a reformulation of ϵ-differential privacy, in the case where $d(a, a')$ is the Hamming distance between two user profiles a, a'. Recall that the Hamming distance is given by the ℓ_1-norm when restricted to the space $\{0, 1\}^m$.

In the current paper, we will consider another point of view for privacy. We will apply Definition 5 to a single user profile, in order to measure to what extent the user protects his/her privacy by means of obfuscation techniques. For doing so, we will use the following pseudo-metric on profiles in $\mathbb{R}^m$.

Definition 6. ***(metric** $d_{\chi,\epsilon}$) Let $\epsilon > 0$ and $\chi \subset \mathbb{R}^m$ be privacy parameters. For two profiles $a, a' \in \mathbb{R}^m$, metric $d_{\chi,\epsilon}$ is defined by*

$$d_{\chi,\epsilon}(a, a') = \epsilon \times |a - a'|_{1,\chi}\ ,$$

where $|.|_{1,\chi}$ is the ℓ_1-norm restricted to the coordinates in χ, that is

$$|a|_{1,\chi} = \sum_{j\in\chi} |a_j| \text{ for all } a \in \mathbb{R}^m.$$

Since we focus our study on the privacy parameters values, we should use the terminology $d_{\chi,\epsilon}$-privacy in place of d_χ-privacy in the current paper, and in particular for Definition 5. Note also that for simplicity in notations, we denote by χ the subset of coordinates in $[1, m]$, and the corresponding coordinate subspace in $\mathbb{R}^m$ as well.

In Definition 5, the closeness in distributions depends on the privacy level captured by the pseudo-metric $d_{\chi,\epsilon}$. A more *tight* pseudo-metric $d_{\chi,\epsilon}$ induces a stronger level of privacy. With our choice of pseudo-metric, this tightness can be leveraged in two ways: with the privacy parameter ϵ as for differential privacy, but also with a relevant choice of subset $\chi \subset \mathbb{R}^m$.

Indeed, restricting the privacy constraint to some subspace $\chi \subset \mathbb{R}^m$ has non-negligible effects on the privacy model, as well as on the accuracy of the results. Typically in the case of the current paper, subset χ will stand for either the complete set of POIs, or some *neighborhood* of the subset of POIs visited by the user. The privacy constraint holds only for coordinates appearing in subset χ, which will be distinct from $\mathbb{R}^m$ when χ will be some neighborhood of the user's POIs. As a consequence of Definition 5, a randomized mechanism may satisfy d_χ-privacy only if coordinates in $\mathbb{R}^m \backslash \chi$ are deterministic. Since only coordinates in χ may be obfuscated, such a restricted distance yields a weaker notion of privacy: even in the case where the user has not visited POIs in $\mathbb{R}^m \backslash \chi$, it leaks precisely the fact that they were not visited. More generally, for $\chi_1 \subset \chi_2 \subset \mathbb{R}^m$, d_{χ_1}-privacy is weaker than d_{χ_2}, in the sense that the former leaks more information about profiles than the latter. Another interesting remark is that a d_{χ_1}-private mechanism is also d_{χ_2}-private, since the coordinates in $\chi_2 \backslash \chi_1$ cannot be obfuscated. Thus the knowledge of the minimal set χ for which a sanitization mechanism satisfies d_χ-privacy encodes some level of privacy guaranteed by the latter mechanism.

Our sanitization mechanisms model the situation where the user himself decides the level of privacy. We stress that such a system would require to suggest some levels of privacy to the user, but the important point for our work is to understand to which extent a *by-user* privacy design is relevant for personal data protection. A user may want to calibrate the privacy level, a numerical value, in order to reach a certain privacy guarantee on his/her personal information: create some uncertainty on the shops he/she has visited, on the home/work place, randomize the frequency of visits to some particular places... However, it is a non-trivial task to associate a tractable meaning of privacy to a numerical value of the privacy parameters, namely to the values of ϵ and χ in our model. In the current approach, we will propose a panel of attacks on sanitized results, in order to fill the gap between the semantic meaning of privacy, and numerical privacy parameters.

B. Sanitization of boolean profiles at the user scale

To achieve the d_χ-privacy guarantee for boolean profiles, we will use some well-known mechanism introduced in the context of differential privacy. The mechanism first appeared as the so-called *randomized response* mechanism [14], which was brought into the field of differential privacy in [15], [16]. Given a collection of boolean profiles $a = (a_{ij})_{ij} \in \{0,1\}^{nm}$, a private version $M(a) = \tilde{a} \in \{0,1\}^{nm}$ is obtained by adding to each component a_{ij} a Bernoulli random variable whose parameter depends on the privacy parameters.

Since in our model, the user applies his/her own sanitization mechanism, we adopt a slightly different point of view. Given user u_i's privacy parameters ϵ_i, χ_i, we define the following randomized mechanism M_i for user u_i's boolean profile $b_i = (b_{ij})_j$:

$$M_i(b_i) = (b_{ij} + d_{ij})_{ij}$$

where $(d_{ij})_{ij}$ are Bernoulli independent random variables with parameter $p_i \in [0,1]$, for $j \in \chi_i$, and $d_{ij} = 0$ for $j \notin \chi_i$. The following general result relates the Bernoulli parameter p_i to the privacy parameter ϵ_i.

Proposition 7. *Let $p_i \in [0,1]$, $\epsilon_i > 0$. Assume that the following inequalities hold:*

$$\frac{1}{1+e^{\epsilon_i}} \leq p_i \leq \frac{e^{\epsilon_i}}{1+e^{\epsilon_i}}.$$

Then mechanism M_i defined above is d_{χ_i,ϵ_i}-private.

A clever choice of the subset χ_i is crucial in order to improve both accuracy and consistency of results. Indeed, when $\chi_i = \mathbb{R}^m$, the previous mechanism perturbates any POI with the same amount of noise (in average). But for natural reasons (geographical proximity, transport design, ...), some pairs of POIs are more likely to be crossed by a single user than others. This fact is what we call *POIs affinity* in the sequel. Compared to a technique that would perturbate only the *closest* POIs (in terms of the affinity) from the user's profile, the naive uniform pertubation (with the same Bernoulli parameter for all coordinates in the profile) is obviously a disaster regarding the usefulness of the result. But less obvious is the fact that uniform noise addition can also harm the privacy. Indeed, the affinity between POIs is likely to be some public information in many cases, and choosing randomly a fake POI among *all* the available POIs may certainly result in inconsistencies in the fake profile. Such inconsistencies are additional information for an attacker: discarding inconsistent POIs in the fake profile yields a higher probability to guess the POIs that are both in the real and the fake profiles.

To produce more consistent results during the sanitization phase, we propose a choice of subsets $(\chi_i)_i$ with the general idea to exploit POIs affinity, with the objective to avoid inconsistent noise addition to the profiles. The notion of affinity we consider in our experiments is based on the global number of profiles shared by a pair of POIs. There are other ways to describe the affinity between POIs: for instance, one could consider the *location* affinity, where the affinity depends on the geographical distance between two POIs. Indeed, a user is more likely to *jump* from a POI to another when the two POIs locations are close.

Definition 8. *(***Affinity between POIs***)* The *affinity* is a matrix of non-negative numbers $(\text{Aff}_{j_1 j_2})_{1 \leq j_1, j_2 \leq m}$.

Given the boolean profile p_i of a user u_i, the *affinity of POI j relative to profile* b_i is given by $\text{Aff}(i,j) = \sum_{j_1, b_{ij_1}=1} \text{Aff}_{j_1 j}$.

The affinity matrix $(\text{Aff}_{j_1 j_2})_{1 \leq j_1, j_2 \leq m}$ is interpreted as follows: the larger the value $\text{Aff}_{j_1 j_2}$, the larger the probability for a user visiting j_1 to visit also j_2. An attacker could exploit the information contained in matrix Aff to detect inconsistencies in noisy profiles, for instance a large number of fake values i, j such that $\text{Aff}_{i,j}$ is small. We will provide a concrete instantiation of matrix $(\text{Aff}_{j_1 j_2})_{1 \leq j_1, j_2 \leq m}$ in Section V.

Now we describe a more interesting choice of subset χ_i than just $\chi_i = \mathbb{R}^m$. User u_i decides a number n_i of POIs to add to his boolean profile p_i. Let $\text{Aff}_1 \geq \text{Aff}_2 \geq ... \geq \text{Aff}_m$ be the sorted list of affinities $(\text{Aff}(i,j))_{1 \leq j \leq m}$ relative to the profile b_i. Then we set

$$\chi_i = \{\ j \mid b_{ij} = 1\ \} \cup \{\ j \mid \text{Aff}(i,j) = \text{Aff}_k \text{ for some } k \leq n_i\ \}.$$

The subset χ_i above will be denoted $\chi_i = N(b_i, n_i)$ (for *neighborhood* of b_i up to n_i values). Hence using the valuable information contained in the affinity matrix, user u_i may apply mechanism M_i with a more interesting choice of χ_i, obfuscating only POIs in a neighborood of the real POIs in the boolean profile. As will be shown in experiments, a correct choice of subset n_i and χ_i can provide an advantageous trade-off between privacy and utility.

C. Sanitization of count profiles at the user scale

From CDRs, a much more precise user profile than boolean profile can be obtained. Instead of only considering the information that a POI was visited or not by some user, we can count the number of visits for each POI, and derive interesting frequency statistics. This information is encoded in the *count profiles*, often called histograms in the literature.

The most natural approach to sanitize histograms in the sense of differential privacy is to add random noise to each count. For our mechanism, we chose to apply Laplacian mechanism, post-processed by a rounding step (to have an integer-valued output). Another way to reach the same goal is to use the geometric mechanism, as in [17].

We will use the notation $[\alpha]$ for the integer part of a number α. Now given user u_i's privacy parameters ϵ_i, χ_i, the sanitization mechanism $\overline{M}_i$ for count profile $c_i = (c_{ij})_j$ is defined as follows:

$$\overline{M}_i(c_i) = (\max(0, [c_{ij} + d_{ij}])_j,$$

where the following conditions for $(d_j)_j$ hold: $(d_{ij})_{j \in \chi_i}$ are independent Laplacian random variables of standard deviation $\frac{1}{\sqrt{2} \times \epsilon}$; and $d_{ij} = 0$ if $j \notin \chi_i$. The maximum and the rounding steps are used in order to provide consistent results for counts, that is to obtain non-negative integer values at the end of the processing.

Proposition 9. *Let $\epsilon_i > 0$ and $\chi_i \subset \mathbb{R}^m$ be privacy parameters. Then mechanism $\overline{M}_i$ is d_{χ_i,ϵ_i}-private.*

As in the case of boolean profiles, one can play with the parametrization over χ_i in order to gain flexibility on privacy. It would be possible to define some notion of affinity (see the previous section) that takes into account the frequency information, rather than only the predicates from boolean profiles. However, for simplicity in the presentation of our model, we chose to apply the same process for choosing χ_i as for boolean profiles. In other words, we will apply mechanism $\overline{M}_i$ for $\chi_i = N(b_i, n_i)$, where b_i is u_i's boolean profile, and n_i some privacy parameter chosen by u_i.

IV. Description of the adversarial model on several scenarii

In our model, each profile is a vector of m coordinates, where each coordinate corresponds to some information associated to some POI. Assume also that we have n users involved. Hence the collection of all users profiles is modelled as a matrix $A = (a_{ij})_{i \leq n, j \leq m}$ of size $n \times m$, where row i $a_i = (a_{ij})_j$ is user u_i's profile.

We assume that the collection of sanitized versions of the profiles is made public, in the form of a matrix $\tilde{A} = (\tilde{a}_{\pi(i)j})_{i \leq n, j \leq m}$ of size $n \times m$, where $\pi : [1, n] \to [1, n]$ is a permutation, and each row $(\tilde{a}_{\pi(i)j})_j$ is a sanitized version of user $u_{\pi(i)}$'s profile. The permutation π's role is just to release an unordered list of (fake) profiles. Hence, no information about π is known to the attacker a priori, one of whose goal is precisely to correctly guess (part of) the map π^{-1}, that is to re-identify users from their fake profiles.

For our experiments, a fake profile $(\tilde{a}_{ij})_j$ will be a sanitized version of a boolean or count profile, with some d_χ-privacy guarantees. The sanitization process will be instantiated by one of the mechanisms introduced in Section III. Now we provide models to describe two scenarii that may happen in real-life, and that may strongly harm users privacy, even in the case of sanitized data.

A. Scenario 1: data leakage during a bounded time slot

The first scenario models what could happen when some data leakage occurs during some period T_1, and other data related to the same users, but on a later period T_2, is sanitized and published. When correlating information from periods T_1 and T_2, one should naturally observe a correlation between the seriousness of the leakage at T_1, and the required privacy level at T_2 in order to provide some pre-determined privacy guarantee.

A data leakage situation as above may happen in real-life. As data becomes more and more valuable, data leakage scandals occur more and more often. Nowadays, it is a well-known fact that a large amount of data is sold on the black market or obtained by data leakage ([18],[19]. Probably less harming at the global scale, but also very destructive for privacy at the user scale, the data leakage may come from the user itself. Indeed, in the case where the user decides his/her privacy level, it is very tempting to lower the privacy level during some periods (potential periods T_1) in order to benefit from the complete accuracy necessary for services provided by LBS or other services providers. For instance, some applications that are designed to protect geo-locations (e.g. Location Guard [12]) could be shut down (intentionally or not) at some moments. We will show that the situation described here strongly harms the potential use of such applications.

Our model is as follows. The attacker $\mathcal{A}$ learns about leaked information at T_1, modelized as a matrix $A^{T_1} = (a_{ij}^{T_1})_{i \in I, 1 \leq j \leq m}$ of some of the *true profiles* observed during period T_1. Moreover, sanitized data $\tilde{A}^{T_2}$ is published, and so it is also known to $\mathcal{A}$. Recall that we have $\tilde{A}^{T_2} = (\tilde{a}_{\pi(i)j}^{T_2})_{1 \leq i \leq n, 1 \leq j \leq m}$ is the collection of all sanitized profiles observed during period T_2.

Then attacker $\mathcal{A}$'s goal is to infer more information than expected about the *true* profiles observed at T_1. For instance, $\mathcal{A}$ could guess the level of privacy used by some user u_i, which may be a sensitive information on its own. He could also infer new information for profiles $a_i^{T_2}$, and even for those users u_i whose profiles were not leaked (a_i for $i \notin I$).

B. Scenario 2: partial profiles leakage

As for scenario 1, we will test the robustness of d_χ-privacy depending on the privacy parameters values, by measuring the inference potential of the adversary during a single time slot T. In scenario 2, the adversary is assumed to have partial knowledge about some of the true profiles ([20],[21]). With the knowledge of (complete) fake profiles, the attacker strives to learn new information about the missing items of the involved profiles.

There are potentially an important threat for user's privacy caused by such partial data leakage. Indeed, the information that a user has visited a particular place (shop, market, ...), represented as a POI in our model, is extremely valuable for marketing purposes. While it seems not so harming when considering a single POI, such practices lead to strong privacy issues if an entity collects the data from multiple *providers* (shops, ...) and correlate them. Then, partial profiles may be retrieved, with accuracy depending on the number of POIs observed by providers. As for scenario 1, a partial leakage can also happen when a user decides to control his/her privacy on some POIs: the user could decide to protect his/her geolocations only on some particular places (home, workplace, ...).

In our model, a partial profiles leakage can be modelled in two ways. In our scenario 2.1, we assume that the attacker $\mathcal{A}$ knows a submatrix $A_{(J_i)_i} = (a_{ij})_{i \in [1,n], j \in J_i}$ of matrix A, where for each i, $J_i \subset [1,m]$ is a subset of the set of POIs $[1,m]$. For $i \in [1,n]$, the set J_i represents the POIs involved in the information leakage for user u_i. For simplicity, we will assume some uniformity on the leakage: we assume that $J_i = J$ for all i, and for some subset $J \subset [1,m]$ of the POIs. In other words, we assume that the leakage involves the same POIs for each user. We will denote $A_J = A_{(J_i)_i}$ in that case. Scenario 2.2 models the attacker's side information by the following partial matrix: $A_I = (a_{ij})_{i \in I, 1 \leq j \leq m}$ for some subset $I \subset [1,n]$ of users. This illustrates what could happen when the data of some (but not all) users is leaked.

As said before, $\mathcal{A}$ also knows $\tilde{A}$, and will strive to correlate information from $\tilde{A}$ and A_J (or A_I), in order to recover new features about the missing values of true profiles (for instance values in $A_{[1,m] \setminus I}$).

V. Development of attacks for each scenario

In the configuration of scenarii in Section IV and for sanitization mechanisms from Section III, we provide details about algorithms performed by the attacker to *de-sanitize* the data. Each attack is associated to some well-understood risk (re-identification, inference of important features), and will be assigned a level of success depending on the sanitization technique which is used (and in particular on privacy parameters).

A. Re-identification

The first privacy risk that we evaluate is re-identification. Concretely, for a given privacy parameter, we estimate the number of fake profiles that are *sufficiently close* to their corresponding real profiles. To formalize this, we need to introduce some definitions. Recall that a_i stands for the real profile of user u_i, and $\tilde{a_i}$ for its sanitized version. Here, d is assumed to be the distance obtained from ℓ_1-norm on profiles.

Definition 10. *((δ, M)-success for fake profiles) Let $\delta > 0, M > 0$. The set $profilesGuess(i, \delta)$ is defined as follows:*

$$profileGuess(i, \delta) = \{ \ j \mid d(a_j, \tilde{a_i}) \leq \delta \ \}$$

We say that an attack on fake profile $\tilde{a_i}$ is a (δ, M)-success if $i \in profileGuess(i, \delta)$ and $|profileGuess(i, \delta)| \leq M$.

The intuition behind the previous definition is that in case of success, an attacker has guessed the correct profile, up to $M - 1$ other profiles. For small values of M, it becomes easy to guess the real profile, using partial auxiliary information.

We will discuss later an ad-hoc choice of parameter δ. Regarding parameter M, we allow parameter values $M > 1$ since randomized mechanisms may yield different concurrent profiles in the set $profileGuess(i, \delta)$. Even for a small amplitude of noise, it can occur that a_i is not one of the closest profiles to $\tilde{a_i}$ (typically, in a case where many real profiles are close one from each other).

Now we propose an evaluation method to understand to what extent such profiles guesses succeed. Experimented for different values of privacy parameter, it provides more insights on the privacy risks, and the privacy level of sanitization mechanisms at stake.

Algorithm 11.

```
Algorithm **re-identification**
Parameters: $\delta$, $M$, $n$ the number of profiles
. For all $i \leq n$, compute sets $guessProfiles(i, \delta)$
. Compute the following re-identification rate
$r(\delta, M, n) = \frac{|\{ \ i \mid \text{ attack on } \tilde{a_i} \text{ is a } (\delta, M)-\text{success } \}|}{n}$
```

B. De-randomization: exploiting inconsistencies in randomized results

As already discussed in Section III, a sanitized mechanism may add noise on some inconsistent locations, simply because the design of the mechanism allows to create non-realistic random values. The most obvious situation may happen when creating a *fake* location very far from the usual visiting locations of a user.

From auxiliary information (data leakage, wifi hotspot, ...), the attacker $\mathcal{A}$ may collect information about a population (for us, the collection of profiles), and compare it to sanitized profiles to find inconsistencies. Here we model this auxiliary information in terms of global statistics about the whole collection of profiles. More precisely, we compute what we call the *affinity* between POIs, that is an indicator estimating the probability to have two POIs in the same profile. We introduced this notion in Definition 8, and we provide now a concrete realization for it.

In our work, affinity is estimated using *real profiles* extracted from our CDRs dataset. A real de-sanitization attack would make use of some public information to estimate this value. For all j_1, j_2, we define our affinity indicators Aff_{j_1, j_2} as follows:

$$\text{Aff}_{j_1, j_2} = \frac{|\{ \ i \mid b_{ij_1} = 1 \text{ and } b_{ij_2} = 1 \ \}|}{n}.$$

With our choice of affinity, two POIs are *close* if they *both* appear in a relatively large number of profiles. Note that matrix $(\text{Aff}_{j_1 j_2})_{j_1 j_2}$ may be computed from count profiles too, since $b_i = 1$ if and only if $c_i \neq 0$ for user u_i.

Algorithm 12.

```
Algorithm **discard inconsistencies**
Parameters: profile id $i$, affinity thresholds $\tau, N$,
number of profiles $n$, affinity matrix $(\text{Aff}_{j_1 j_2})_{j_1 j_2}$
. For $j$ such that $\tilde{a_{ij}} \neq 0$, tag $j$ as an inconsistency
if $\text{Aff}_{jj'} < \tau$ for at least $N$ values of $j'$ such that $\tilde{a}_{ij'} \neq 0$
. Create a new profile $\overline{a_i}$ by discarding all inconsistencies
from profile $\tilde{a}_i$
```

As τ goes smaller and N larger, inconsistencies are harder to find, and a suitable choice is required in order to optimize the attack. Algorithm 12 can be used together with Algorithm 11 to define an algorithm **re-identification on consistent profiles**: first, apply Algorithm 12 to each profile; then perform Algorithm 11 by replacing fake profiles $\tilde{a_i}$ by their corresponding consistent profiles $\overline{a_i}$. We will denote $\overline{r}(\delta, M, n, \tau, N)$ the identification rate obtained by this process.

In the previous analysis, we only have exploited partial information from matrix $(\text{Aff}_{j_1 j_2})$. Indeed, we focused on the detection of *wrong* fake POIs in a profile by considering small values of $\text{Aff}_{j_1 j_2}$. An analogous analysis on large values of $\text{Aff}_{j_1 j_2}$ would provide insights of possible locations that disappeared during the sanitization process.

C. De-randomization: denoising profiles

Here we assume that the attacker possesses partial knowledge on some true profiles. This can be the case for both scenarii 1 and 2. In particular, comparing a true (even partial) profile of a user to its sanitized version permits to accurately estimate the level of noise in some cases. For boolean profiles, it is then possible to guess the true number of POIs in the user's profile. In the case of scenario 2, the following algorithms should be understood with $T_1 = T_2 = T$.

Algorithm 13.

```
Algorithm guess Bernoulli Parameter
Parameters: profile id $i$, true boolean profiles $(b_{ij}^{T_1})_{j \in J}$ for subset $J \subset [1, m]$
. Compute $\alpha_i = \frac{1}{|J|} \times \sum_{j \in J} \mid \tilde{a}_{ij}^{T_2} - a_{ij}^{T_1} \mid$
. Estimate Bernoulli parameter as $\tilde{p}_i = \alpha_i / |J|$
```

Note that for count profiles, the attacker collects more information. He certainly can retrieve the amount of noise added to each count.

Algorithm 14.

```
Algorithm guess Laplacian Parameter
Parameters: profile id $i$, true count profiles $(c_{ij}^{T_1})_{j \in J}$ for subset $J \subset [1, m]$
. Estimate the Laplacian parameter
$\frac{1}{\tilde{\epsilon}_i} = \frac{1}{\sqrt{2} \times |J|} \times \sum_{j \in J} \mid \tilde{c}_{ij}^{T_2} - c_{ij}^{T_1} \mid$
```

Accurate results of both of the above algorithms are supported by the law of large numbers, and should produce accurate estimations whenever profiles $a_i^{T_1}$ are sufficiently close to profiles $a_i^{T_2}$. These algorithms induce in turn natural attacks to infer new knowledge in our scenarii.

For the case of boolean profiles, knowing both the Bernoulli parameter p_i and a fake profile $\tilde{b}_i$, it is possible to *bound* the set of possible candidates for true profile b_i, with a high level of confidence. Indeed, the following formula holds (see the appendix for a justification):

$$P(\ d(b_i, \tilde{b}_i) \leq \delta\ ) = \sum_{k \leq \delta} \binom{|\chi_i|}{k} \times p_i^k \times (1 - p_i)^{|\chi_i| - k}.$$

If the attacker has some idea about the size of user u_i's profile, the value $|\chi_i|$ can be estimated from the profile $\tilde{b}_i$ and the value p_i. Then the formula above can be exploited to decide a reasonable value δ for Algorithm 11 for instance. To do so, probabilities $P(\ d(b_i, \tilde{b}_i) \leq \delta\ )$ can be computed successively for increasing values of δ, until the result reaches a sufficient level of confidence: then the corresponding value δ should be a relevant candidate for computing $profileGuess(i, \delta)$.

D. Guess important features in profiles

Count profiles are much more precise representations than boolean profiles, and so their sanitized versions are more likely to leak information. For instance, the higher the count is on a POI, the more likely the user has visited this POI. Moreover, if a user has visited many times a POI during periods T_1 and T_2, attacker can certainly infer that this POI is part of the profile, and possibly an important place (home, work location ...).

Definition 15. *Let ϵ_i, χ_i be the privacy parameters for mechanism $\overline{M}_i$. The set of most significative POIs in sanitized count profile $\tilde{c}_i$ is defined as follows:*

$$poiGuess(i) = \{\ j \in \tilde{c}_i \mid \tilde{c_{ij}} = \max_{j' \in \chi_i} \tilde{c_{ij'}}\ \}.$$

The next algorithm searches the most significative POI among the best candidates for true profile c_i.

Algorithm 16.

```
Algorithm guess most significative POI
Parameters: profile id $i$, $\delta$, $n$
. Compute $P_{i,\delta} = profileGuess(i, \delta)$
. For $j \in P_{i,\delta}$, compute $poiGuess(j, \delta)$
. Compute the set $\overline{P}_{i,\delta} = \cap_j poiGuess(j, \delta)$
. Compute the following rate
$\tilde{r}(\delta, n) = \frac{|\{\ i \mid i \in \overline{P}_{i,\delta}\ \}|}{n}$.
```

With the information contained in $\overline{P}_{i,\delta}$, and a semantic knowledge on the POIs in it, the attacker can infer precise information about the user. As an example, geolocation data of the POIs could be used to know the semantic properties of the most important POI. Some open-source tools such as Open Street Map provide such semantic information about locations: residential zone, roads,... The attacker could then deduce if some *significative POI* is a workplace, a home, or a shop.

VI. Experiments

To illustrate the different scenarii, we realized experimental evaluations of the described attacks. Then we analyse the experimental results to provide insights about the choice of privacy parameters in d_χ-private sanitization processes.

A. Settings

All the algorithms were implemented in Scala on a personal computer with 2.30 GHz Intel i5 CPU and 8 GB RAM Memory. We use a Call Detail Records dataset from a large mobile phone provider to model our system. POIs are defined to be antennas cells co-located in some region around some big city, so that our dataset contains exactly $m = 29$ POIs and more than 100 000 users. In order for profiles to have sufficiently many information, we filtered our dataset and considered only the profiles having visits on at least 6 distinct POIs during a day.

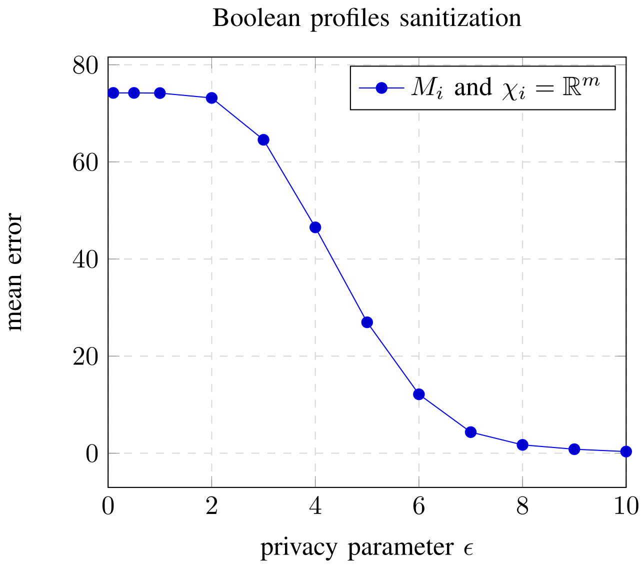

Fig. 4. The mean error is $1/m \times \sum_{1 \leq j \leq m} f_1(j)$. It should be compared to the average number of visitors on a POI, which is 25 here.

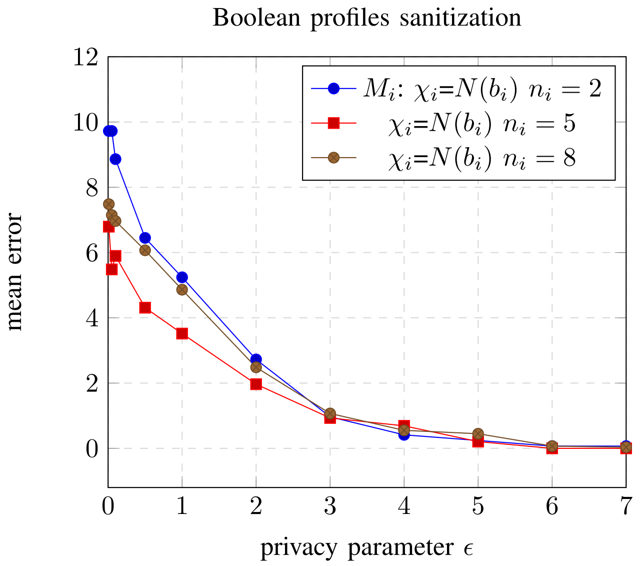

Fig. 5. The mean error is $1/m \times \sum_{1 \leq j \leq m} f_1(j)$. It should be compared to the average number of visitors on a POI, which is 25 here.

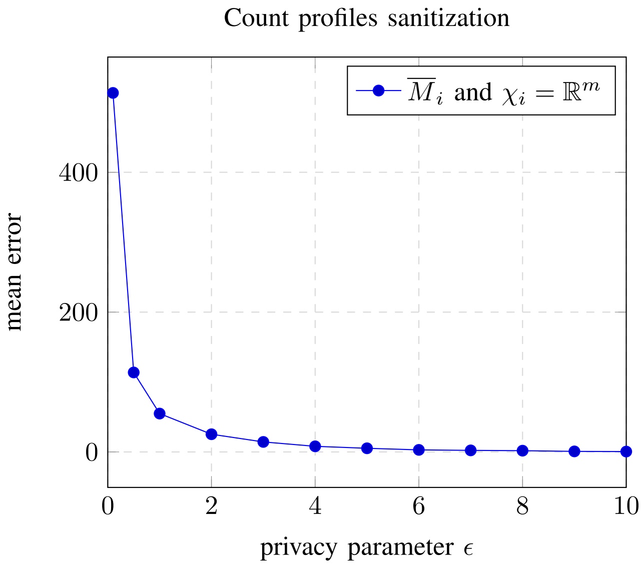

Fig. 6. The mean error is $1/m \times \sum_{1 \leq j \leq m} f_2(j)$. It should be compared to the average number of visits on a POI, which is 98 here.

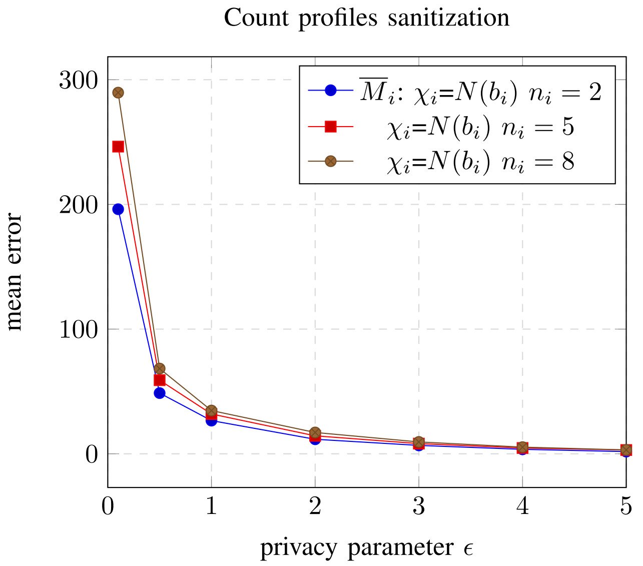

Fig. 7. The mean error is $1/m \times \sum_{1 \leq j \leq m} f_2(j)$. It should be compared to the average number of visits on a POI, which is 98 here.

Here are some information about the parameters we chose to conduct the experiments:

- We consider subsamples of $n = 100$ users to conduct our experiments. We consider an observation period T equivalent to a whole day. For scenario 2.2, we used a subsample of $|J| = 50$ users. In the case of scenario 1, T_1 represents the first half of a day (from 00:00 to 11:59), and T_2 the second half (from 12:00 to 23:59).
- Even if a *by-user* choice of privacy parameters is possible in our system, we chose uniform values for n_i, χ_i and ϵ_i (that is common values for each $i \in [1, n]$: $\epsilon_i = \epsilon$...), in order to facilitate the comprehension in analysis.

We chose to restrict our experiments to restricted samples of 100 users ($n = 100$ with our notations) for several reasons. First, this enforces a more difficult task from the sanitization point of view, and tends to prevent from finding more threatening privacy risks: as was observed in many prior works, sanitization would be made easier with larger values of n, since overlaps over profiles are more likely to happen, which *hides users in the crowd* while preserving the statistics accuracy. Second, attack scenarii need to be realistic. In the case of partial information leakage (see in particular our scenarii 2.1 and 2.2 in Section IV), the attacker's ability to collect personal information from users should be reasonably bounded: for instance, while it may be possible for the attacker to retrieve all the users who visited the same wifi hotspot, the number

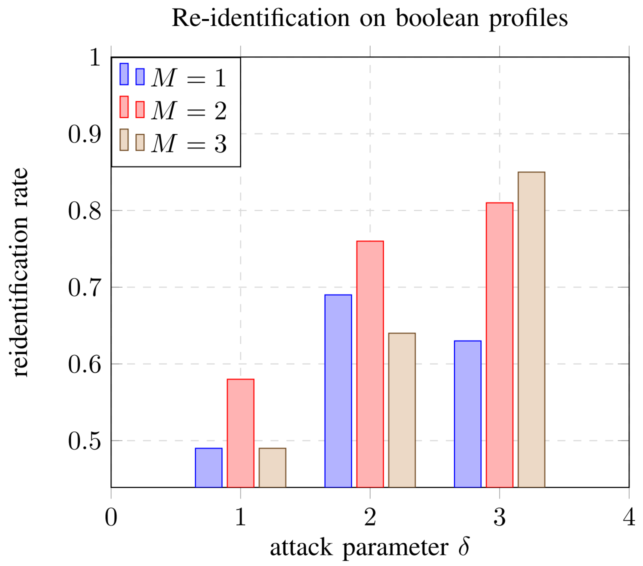

Fig. 8. Re-identification on sanitization mechanism M_i with $\epsilon = 7$ and $\chi_i = \mathbb{R}^m$.

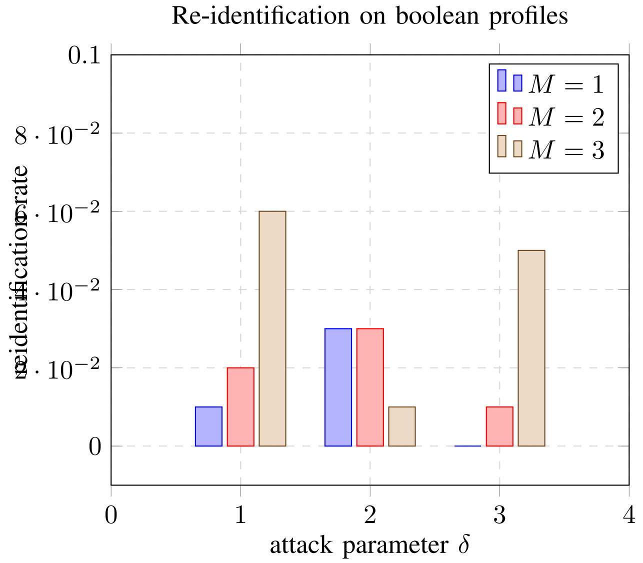

Fig. 9. Re-identification on sanitization mechanism M_i with $\epsilon = 0.1$ and $\chi_i = N(b_i, n_i)$, $n_i = 5$.

of such users is inherently bounded since visiting this place depends on the user's habits.

The global purpose of our experiments is to associate, for each attack described in Section V, suggestions on how to apply (if applying is reasonable) d_χ-private mechanisms on boolean and count profiles. As exposed previously in the paper, the first step of our system consists in deciding an admissible range for privacy parameters regarding global statistics utility (statistician's view). In a second step, we instantiate various attacks on the sanitized profiles (attacker's view).

B. Admissible range for privacy parameters

For boolean profiles, the *mean error* we consider is the average over all POIs of the error between the computation of

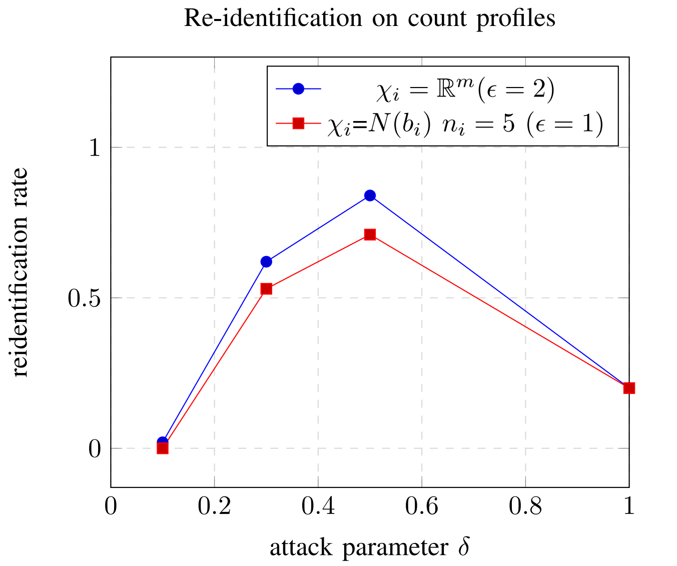

Fig. 10. Re-identification on sanitization mechanisms $\overline{M}_i$, using attack parameter $M = 1$.

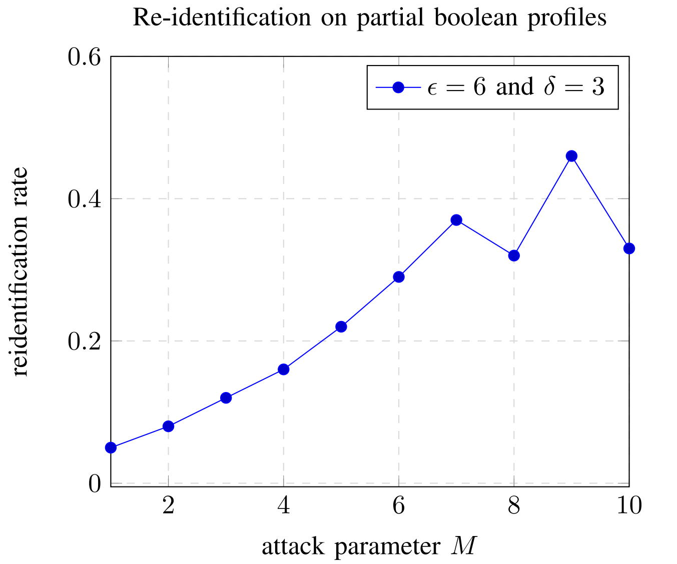

Fig. 11. Sanitization mechanism M_i with $\epsilon = 6$ and $\chi_i = \mathbb{R}^m$. Then reidentification given only partial profiles with 15 POIs.

f_1 on true profiles and its computation on fake profiles (see Section II), that is

$$mean\ error = \frac{1}{m} \times \sum_{1 \le j \le m} |f_1((b_i)_i, j) - f_1((\tilde{b}_i)_i, j)|.$$

The latter quantity should be *small* compared to the average of f_1 over all POIs, whose value is 25 for the considered sample of users.

Figures 4 and 5 display the trade-off privacy/utility for mechanism M_i applied to boolean profiles. When used with parameter $\chi_i = \mathbb{R}^m$, mechanism M_i shows to be useful for values ϵ in the range $[6, +\infty[$. In particular, values lower than 6 should not be considered, since they completely destroy the information from the statistician point of view. This is

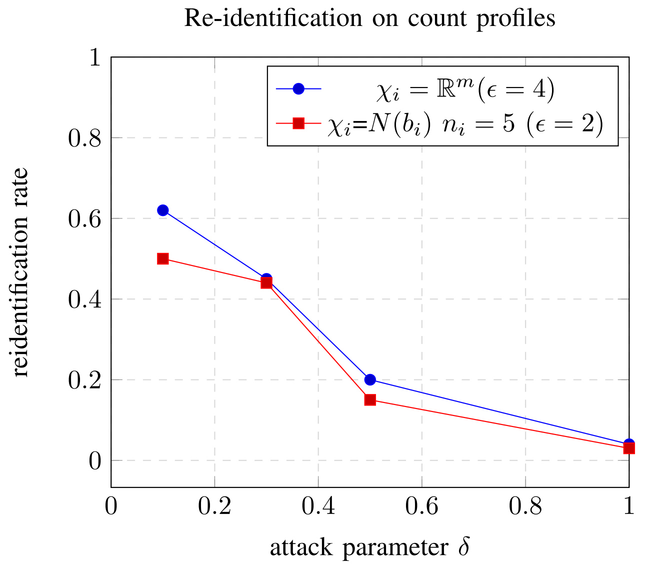

Fig. 12. Sanitization mechanism $\overline{M}_i$ and reidentification given only partial profiles with 15 POIs ($M = 1$).

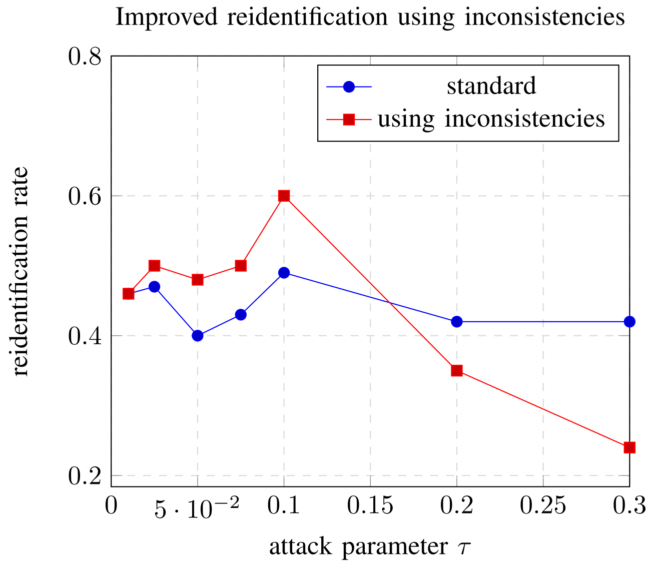

Fig. 13. Sanitization mechanism M_i with $\epsilon = 6$, $\chi_i = \mathbb{R}^m$. Reidentification attacks with $(M, \delta) = (2, 2)$ using or not inconsistencies.

to be compared to the range $[0.1, +\infty[$, admissible for M_i parametrized by $n_i = 5$ and $\chi_i = N(b_i, n_i)$. We warn the reader that even if a value $\epsilon = 0.1$ seems much more appealing from the privacy aspect than $\epsilon = 6$, the restriction of χ_i from $\mathbb{R}^m$ to $N(b_i, n_i)$ has also consequences over privacy. We will clarify this comparison using our system, through some attacks from Section V on both mechanisms.

For count profiles, we adopt the same protocol using f_2 in place of f_1, and $\overline{M}_i$ in place of M_i. Hence we have:

$$mean\ error = \frac{1}{m} \times \sum_{1 \leq j \leq m} |f_2((b_i)_i, j) - f_2((\tilde{b}_i)_i, j)|.$$

Here the quantity to be compared with, that is $\frac{1}{m} \sum_{1 \leq j \leq m} f_2(j)$, is equal to 98 for our sample.

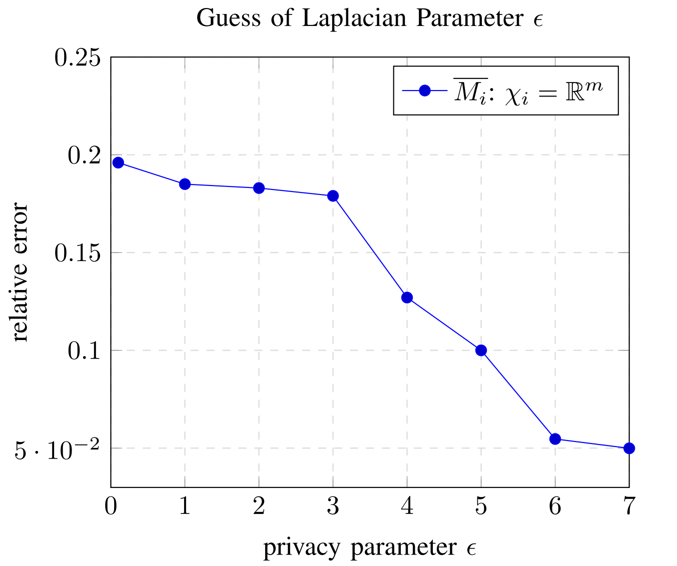

Fig. 14. The relative error is $\frac{|\tilde{\epsilon} - \epsilon|}{\epsilon}$ where $\tilde{\epsilon}$ is computed with algorithm guess Laplacian parameter.

Figure 6 shows that range $[2, +\infty[$ seems admissible for ϵ values, in order to have reasonable utility for statistic f_2. By contrast with boolean profiles, the restriction from $\chi_i = \mathbb{R}^m$ to $\chi_i = N(b_i, n_i)$ (and *small* n_i) has much less impact on utility in the case of count profiles. in fact, some similar range $[1, +\infty[$ may be chosen. This is completely normal, and results from the design of algorithm $\overline{M}_i$: while twisting 0 to 1 and 1 to 0 radically defaces a boolean profile, the amount of noise added by Laplacian mechanism may be relatively small compared to counts in a count profile.

C. Re-identification

As illustrated by Figures 8 and 9, the choice of attack parameters M and δ seems to be not straightforward in general. It is far from being clear how an attacker could design an optimal choice for (M, δ). But since the attacker's strategy is not known a priori, the worst case should be considered for designing the sanitization mechanism. Our system permits to compute attacks over many pairs (M, δ) and thus to choose the one having the best reidentification rate $r(\delta, M, n)$. Recall also that values of M are supposed to be sufficiently small, since otherwise it is useless for the reidentification performed by the attacker (too many candidates). Here experiments were conducted in scenario 2.2, if not explicitly precised.

Worst-case for M_i and $\chi_i = \mathbb{R}^m$ occurs with a reidentification rate $r(\delta, M, n) \sim 0.9$ (meaning 90 % of the population was re-identified), which is far from being admissible from the security point of view. On the contrary, as shown in Figure 9, the choice of $\chi_i = N(b_i, n_i), n_i = 5$ leads to much more interesting results for $\epsilon = 0.1$, that is a reidentification rate $r(\delta, M, n)$ close to 0.05.

Experiments on reidentification over a single observation period T were also conducted for sanitization mechanism $\overline{M}_i$ on count profiles. As was the case for the utility, mechanism $\overline{M}_i$ performs similarly with $\chi_i = \mathbb{R}^m$ or $\chi_i = N(b_i, n_i)$ for some small n_i. This point was already explained in the previous subsection. A much more interesting fact is that privacy is unreasonably harmed (more than 80 % of reidentification) when using such mechanisms. Indeed, count profiles contain very precise information about the user, and so provides quite easily a strong fingerprint to reidentify him/her. The use of such mechanisms should be avoided, in any case where a leakage scenario as in Section IV could happen.

We have also measured to what extent a side information composed of *partial profiles* (in the sense of scenario 2.1) may be exploited by the attacker. To do so, we provide the attacker half of the true profiles, that is the profiles restricted to 15 POIs. Then attacker attempts to maximize $r(\delta, M, n)$ combining both partial profiles, and sanitized profiles. It is reasonable in realistic scenarii to assume that the attacker $\mathcal{A}$ knows the POIs on which the partial profiles are defined. Hence, as a pre-processing step, $\mathcal{A}$ restricts also the sanitized profiles to these 15 POIs. Then $\mathcal{A}$ applies Algorithm 11, some of whose results are displayed in Figure 11 for the case of boolean profiles and mechanism M_i. The graph shows that it is much more difficult to re-identify with partial information on the profiles, as expected (compare to Figure 4). However, it is worth noticing also that if we allow the attacker to make less precise approximations for $profileGuess(i, \delta)$ (that is letting M grow larger), the attack may harm the privacy to some unreasonable level (40 % of *almost* re-identification in the worst case). For mechanism $\overline{M}_i$ and reidentification from partial count profiles, results in Figure 12 displays a 50 % reidentification rate for the best attacker's strategy.

D. De-randomization

For experiments to validate our system, side information is modelled as a matrix $(\text{Aff}_{j_1 j_2})_{j_1 j_2}$ defined in Section V. Figure 13 proves that $(\text{Aff}_{j_1 j_2})_{j_1 j_2}$ may be exploited to improve reidentification attacks, for sufficiently small values of the *inconsistency threshold* τ and suitable value of the minimal number of inconsistencies N ($N = 8$ for the graph in Figure 13). Indeed, small values of τ may provide 10 % more reidentification in the sample.

Such a non-negligible improvement on the attack should suggest to relativise any result regarding the privacy provided by (any) sanitization mechanism. Improved attacks may always be designed if a sufficient amount of side information is available to the attacker. As briefly suggested in Section V, even matrix $(\text{Aff}_{j_1 j_2})_{j_1 j_2}$ could be exploited further to gain information on the true profiles. It is clear than adding other sources of side information may have disastrous impacts on the d_χ-private mechanisms at stake in the current paper.

Fortunately (for users privacy), the choice of τ and N is highly non-trivial to perform an attack. It is unlikely that the attacker could guess the optimal values for attack parameter. However, our goal is to validate some *worse case* for privacy, defined in terms of attack parameters τ, N, against which users should be protected.

Figure 14 reflects to what extent the privacy parameter ϵ_i can be deduced from sanitized profiles and some of the true profiles. Algorithm 14 performs particularly well (in the case of scenario 2.2), and as expected better as the value ϵ increases. This corroborates the well-admitted idea in the security community that the security parameters should be made public.

scenario	privacy parameters	$\tilde{r}(0.3, 100)$
scenario 1	$\chi_i = \mathbb{R}^m$, $\epsilon = 6$	0.76
scenario 1	$\chi_i = N(b_i, n_i)$, $n_i = 5$, $\epsilon = 3$	0.70
scenario 2.1	$\chi_i = \mathbb{R}^m$, $\epsilon = 4$	0.99
scenario 2.1	$\chi_i = N(b_i, n_i)$, $n_i = 5$ $\epsilon = 2$	0.95
scenario 2.2	$\chi_i = \mathbb{R}^m$, $\epsilon = 2$	1.0
scenario 2.2	χ_i=$N(b_i, n_i)$, $n_i = 3$, $\epsilon = 1$	0.85

E. Guess on important features

The following array shows to what extent the most important POI in a true count profile c_i can be guessed from its corresponding sanitized version $\overline{M}_i(c_i)$. We consider this experiment with privacy parameters that *optimize* the privacy level in the admissible range of privacy values (regarding utility).

To be protected against such an inference attack, mechanism $\overline{M}_i$ tuned with a restricted neighborhood $N(b_i, n_i)$ (n_i small) seems to be a better option. However, the attack success rate $\tilde{r}(0.3, 100)$ is too large to consider $\overline{M}_i$ a sufficient protection against that risk.

F. Concluding remarks about our experimental results

Several concluding remarks can be drawn from our experimental evaluation. First, when performing mechanism M_i on boolean profiles, the use of neighborhoods $\chi_i = N(b_i, n_i)$ for small n_i values permits to outperform the case $\chi_i = \mathbb{R}^m$, both with regards to privacy and utility. The benefits of such improvements for mechanisms $\overline{M}_i$ are not significative.

Second, subsection on de-randomization highlights the well-known fact that auxiliary information may have disastrous impact on the privacy guarantee provided by sanitization mechanisms. A valuable approach from the sanitizer point of view would be to measure the impact of the amount of side information on the attack success rate. We did not address this issue in the current paper, but it would be very interesting to integrate such parametrizations in our system.

Experiments also showed that the use of mechanism $\overline{M}_i$ should be avoided, whenever any scenario from Section IV could be met in reality. We do not pretend that the use of d_χ-private mechanisms on fine-grained data such as count profiles is useless. Such mechanisms, once completely defined through their privacy parameters, are inherently associated to some risk regarding specific scenarii of attacks. The current work suggests that the latter risks should be well understood before applying these sanitization mechanisms in practice. In other words, privacy applications require some management of risks, made in interaction with a suggestion on privacy parameters for each possible and reasonable risk. Our system helps to fill the current gap between the choice of privacy parameters and the management of risks: for some situations (e.g. sanitization $\overline{M}_i$ before statistical task f_2), our system proves that there is no appealing trade-off between privacy and accuracy of the statistics; for others (e.g. sanitization M_i, $\epsilon = 6$ and statistical task f_1), the system provides a level of privacy risk (which can be considered as reasonable or not) together with the privacy parameters to achieve this level.

Finally, we think that randomized mechanisms on count profiles would have a better success if sanitization is designed for sanitizing a group of users with some privacy guarantee. This was for instance the initial goal of differential privacy: similar users could be aggregated together to produce a *group profile*, that could in turn be sanitized in the sense of differential privacy.

VII. Related work

Here we discuss the closest works to ours, already appeared in the literature:

- **Tuning the privacy parameter ϵ in differential privacy:**
  In [17], Naive Bayes classifiers are used to produce prediction functions that map quasi-identifiers tuples to their (supposed) corresponding sensitive attribute. Experiments are provided on histograms sanitized in a differentially private manner. Letting the privacy parameter ϵ vary, the author highlights interesting features about differentially private mechanisms: even *small* value of ϵ ($\epsilon = 0.01$) yields non-trivial attacks, but the corresponding differentially private mechanism certainly offers much more protection (avoiding to guess sensitive attributes from quasi-identifiers) than raw histograms. Our approach follows the same line as the work [17], but only from a high-level point of view: we follow the principle of designing attacks in order to understand the limits and the benefits of randomized mechanisms. In particular, we broaden the scope of privacy from differential privacy to $d_{\mathcal{X}}$-privacy. Also, our attacks are fundamentally different since we have no notion of quasi-identifiers in our data representation.
  Differential privacy guarantee only asserts that the participation of a single individual in a statistical game is not noticeable, and does not pretend to hide the information of a single individual (which can be deduced entirely from the data of other individuals in some cases). In [22], a new privacy notion is introduced, called ρ-differential identifiability, and designed to measure the ability of an adversary (via the privacy parameter ρ) to infer the presence of a single individual in a database, from some sanitized results. Authors of [22] show relationships between ρ-differentially identifiable mechanisms and ϵ-differentially private mechanisms, and that their new notion can be used to provide an upper *safe* bound for parameter ϵ. In fact, the same authors have already discussed the choice of meaningful (in terms of tractable privacy) upper bounds on ϵ in some prior work [23]. Our current work also provides such upper bounds, but directly by designing re-identification or inference attacks, and without requiring additional contextual parameters (see parameter $|\psi|$ in [22]).
  The paper [24] proposes a model to choose relevant values for ϵ, depending on two interacting parties, the data analyst (for us, the statistician) and the user. This issue was not deeply addressed in the current paper, while the issue of risk assessment was not tackled in [24]. Hence our works are complementary and both point of views should be considered to produce a more complete model.
- **Understanding the differential privacy guarantee:**
  A more theoretical approach aims at understanding privacy through the introduction of other privacy notions, such as mutual-information differential privacy($MI - DP$) in [25]. A clear advantage by doing that is to view the differential privacy picture by means of a better known theory (here information theory). However, it still locate the choice of the privacy parameter ϵ in a theoretical domain, which can be difficult to grasp for non-specialists.
- **About de-sanitization attacks on mobility data:**
  In [26], a summary of the various attacks on mobility data is provided in Figure 1, with for each attack, the re-identification rate: [27] [28] [29] [30] [31] [32] [33] [34]. In most attacks against mobility data, attackers use a period of training and the objective is re-identification. In our case we do not restrict ourselves to such a configuration (which is scenario 1 with reidentification attacks). We propose a panel of different attacks, and other possible scenarii (scenarii 2.1 and 2.2). As an example close to our work, authors of [35] show that they can re-identify at least 63 % of the users from a sanitized dataset with geo-indistinguishability guarantees. As is the case for the current paper, such attacks do not break the geo-indistinguishability guarantee, whose goal is more to obfuscate precise locations rather than avoiding re-identification. But such attacks permit to understand to what extent geo-indistinguishability protects user's privacy.

VIII. Conclusion

In this paper, we addressed the issue related to the choice of privacy parameters values in randomized sanitization mechanisms. We propose a model that instantiates well-known attacks on sanitized data. First, our model provides a better understanding of the privacy level hidden behind the parameters values. Second, it allows any party with sufficiently many data to make relevant suggestions concerning the choice of such parameters values, in order to optimize the trade-off between privacy and utility. We applied our framework on mobility data extracted from Call Detail Records, and we have provided meaningful insights about $d_{\mathcal{X}}$-privacy, related sanitization randomized mechanisms, and their applications on mobility data sanitization.

For the current work, we decided to study the privacy parameters involved in $d_{\mathcal{X}}$-private sanitization processes. Although we chose to apply $d_{\mathcal{X}}$-privacy to protect privacy at the user-scale, $d_{\mathcal{X}}$-privacy can also provide privacy guarantees among a large set of users, as was the case for prior applications of ϵ-differential privacy. Group profiles may be introduced instead of users profiles, in order to twist from a user-scale privacy to a *global* scale (over many users). Once group profiles are correctly defined, our system and experiments will apply easily to provide more understanding of the privacy parameters ϵ and δ, when using ϵ-differential privacy or (ϵ, δ)-differential privacy. We postponed this study to a further work.

References

[1] T. B. Murdoch and A. S. Detsky, "The inevitable application of big data to health care," *Jama*, vol. 309, no. 13, pp. 1351–1352, 2013.

[2] G.-H. Kim, S. Trimi, and J.-H. Chung, "Big-data applications in the government sector," *Communications of the ACM*, vol. 57, no. 3, pp. 78–85, 2014.

[3] A. Korolova, "Privacy violations using microtargeted ads: A case study," in *Data Mining Workshops (ICDMW), 2010 IEEE International Conference on*. IEEE, 2010, pp. 474–482.

[4] A. Narayanan and V. Shmatikov, "De-anonymizing social networks," in *Security and Privacy, 2009 30th IEEE Symposium on*. IEEE, 2009, pp. 173–187.

[5] J. He, W. Chu, and Z. Liu, "Inferring privacy information from social networks," *Intelligence and Security Informatics*, pp. 154–165, 2006.

[6] E. Zheleva and L. Getoor, "To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles," in *Proceedings of the 18th international conference on World wide web*. ACM, 2009, pp. 531–540.

[7] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[8] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography Conference*. Springer, 2006, pp. 265–284.

[9] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: Differential privacy for location-based systems," in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*. ACM, 2013, pp. 901–914.

[10] K. Chatzikokolakis, M. E. Andrés, N. E. Bordenabe, and C. Palamidessi, "Broadening the scope of differential privacy using metrics," in *International Symposium on Privacy Enhancing Technologies Symposium*. Springer, 2013, pp. 82–102.

[11] Y. Xiao and L. Xiong, "Protecting locations with differential privacy under temporal correlations," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2015, pp. 1298–1309.

[12] K. Chatzikokolakis, C. Palamidessi, and M. Stronati, "Location guard: location privacy for the rest of us."

[13] ——, "Constructing elastic distinguishability metrics for location privacy," *Proceedings on Privacy Enhancing Technologies*, vol. 2015, no. 2, pp. 156–170, 2015.

[14] C. Dwork, A. Roth *et al.*, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014.

[15] M. Alaggan, S. Gambs, and A.-M. Kermarrec, "Blip: non-interactive differentially-private similarity computation on bloom filters," in *Symposium on Self-Stabilizing Systems*. Springer, 2012, pp. 202–216.

[16] Ú. Erlingsson, V. Pihur, and A. Korolova, "Rappor: Randomized aggregatable privacy-preserving ordinal response," in *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*. ACM, 2014, pp. 1054–1067.

[17] G. Cormode, "Personal privacy vs population privacy: learning to attack anonymization," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2011, pp. 1253–1261.

[18] "Call detail records leak by private firms indicate corporate rivalry, snooping," http://indiatoday.intoday.in/story/call-records-leak-delhi-crime-branch/1/711849.html, 2016.

[19] "Cdr leak case: Visa verification company bls management solution may lose contract," http://indiatoday.intoday.in/story/cdr-leak-case-bls-management-solution/1/714450.html, 2016.

[20] "Call details leak: Women pay to spy on their husbands; spies turn double agents, extortionists," http://www.hindustantimes.com/mumbai-news/call-details-leak-women-paid-to-spy-on-their-husbands -spies-turn-double-agents-extortionists/story-ivaDzWywn8GLc1DBSkcqlO.html, 2016.

[21] "Mumbai: Two pvt detectives held for snooping on call records, selling details to clients," http://indianexpress.com/article/cities/mumbai/two-pvt-detectives-held-for-snooping-on-call-records -selling-details-to-clients-4505064/, 2016.

[22] J. Lee and C. Clifton, "Differential identifiability," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2012, pp. 1041–1049.

[23] ——, "How much is enough? choosing ε for differential privacy," in *International Conference on Information Security*. Springer, 2011, pp. 325–340.

[24] J. Hsu, M. Gaboardi, A. Haeberlen, S. Khanna, A. Narayan, B. C. Pierce, and A. Roth, "Differential privacy: An economic method for choosing epsilon," in *Computer Security Foundations Symposium (CSF), 2014 IEEE 27th*. IEEE, 2014, pp. 398–410.

[25] P. Cuff and L. Yu, "Differential privacy as a mutual information constraint," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 43–54.

[26] S. Gambs, M.-O. Killijian, and M. N. del Prado Cortez, "De-anonymization attack on geolocated data," *Journal of Computer and System Sciences*, vol. 80, no. 8, pp. 1597–1614, 2014.

[27] Y. De Mulder, G. Danezis, L. Batina, and B. Preneel, "Identification via location-profiling in gsm networks," in *Proceedings of the 7th ACM workshop on Privacy in the electronic society*. ACM, 2008, pp. 23–32.

[28] H. Zang and J. Bolot, "Anonymization of location data does not work: A large-scale measurement study," in *Proceedings of the 17th annual international conference on Mobile computing and networking*. ACM, 2011, pp. 145–156.

[29] C. Y. Ma, D. K. Yau, N. K. Yip, and N. S. Rao, "Privacy vulnerability of published anonymous mobility traces," *IEEE/ACM Transactions on Networking (TON)*, vol. 21, no. 3, pp. 720–733, 2013.

[30] J. Freudiger, R. Shokri, and J.-P. Hubaux, "Evaluating the privacy risk of location-based services," in *International Conference on Financial Cryptography and Data Security*. Springer, 2011, pp. 31–46.

[31] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, and J.-P. Hubaux, "Quantifying location privacy," in *Security and privacy (sp), 2011 ieee symposium on*. IEEE, 2011, pp. 247–262.

[32] M. Srivatsa and M. Hicks, "Deanonymizing mobility traces: Using social network as a side-channel," in *Proceedings of the 2012 ACM conference on Computer and communications security*. ACM, 2012, pp. 628–637.

[33] Y.-A. De Montjoye, C. A. Hidalgo, M. Verleysen, and V. D. Blondel, "Unique in the crowd: The privacy bounds of human mobility," *Scientific reports*, vol. 3, p. 1376, 2013.

[34] J. Unnikrishnan and F. M. Naini, "De-anonymizing private data by matching statistics," in *Communication, Control, and Computing (Allerton), 2013 51st Annual Allerton Conference on*. IEEE, 2013, pp. 1616–1623.

[35] V. Primault, S. B. Mokhtar, C. Lauradoux, and L. Brunie, "Differentially private location privacy in practice," *arXiv preprint arXiv:1410.7744*, 2014.

Appendix

A. Privacy proofs

Proposition 7 Let $r \geq 0$. For all boolean profile $y = (y_i)_i$, and for all boolean profiles b, b' such that $d(b, b') = r$, we have to show

$$\frac{P(b + d = y \bmod 2)}{P(b' + d = y \bmod 2)} \leq e^{\epsilon \times r}.$$

For $i \in \chi$, we have $P(b_i + d_i = y_i \bmod 2) = P(d_i = y_i + b_i \bmod 2)$. The latter quantity is equal to p if $y_i + b_i = 0$, and $1 - p$ if $y_i + b_i = 1$.

Since by assumption $d(b,b')=r$, y_i+b_i differ from $y_i+b'_i$ on exactly r indices $i\in\chi$. It follows that

$$\frac{P(b+d=y \bmod 2)}{P(b'+d=y \bmod 2)} \leq \max(\frac{p}{1-p}, \frac{1-p}{p})^r.$$

Then a sufficient condition for the required inequality to hold is the following:

$$p\leq e^{\epsilon}\times(1-p) \text{ and } 1-p\leq e^{\epsilon}\times p.$$

The proposition is proved. □

Proposition 9 Since both maximum and rounding are post-processing not depending on a particular instance of dataset, it suffices to prove that the mechanism $c\rightarrow c+d$ satisfies d_{χ}-privacy. Let $c=(c_j)_j, c'=(c'_j)_j$ be two profiles. For all $y\in\mathbb{R}^m$, we have

$$\log(\frac{P(c_j+d_j=y_j)}{P(c'_j+d_j=y_j)})=\epsilon\times|c_j-c'_j|$$

The result follows from the independence of the random variables $(d_j)_j$, and by summing over $1\leq j\leq m$. □

B. Missing justifications

Now let $M_i(b_i)=\tilde{b}_i$ be the sanitized version of boolean profile b_i under mechanism M_i introduced in Section III, with privacy parameters p_i and $\chi_i\subset\mathbb{R}^m$. Let also $\delta>0$. We justify the following formula:

$$P(\ d(b_i,\tilde{b}_i)\leq\delta\ )=\sum_{k\leq\delta}\binom{|\chi_i|}{k}\times p_i^k\times(1-p_i)^{|\chi_i|-k}.$$

Indeed, we have

$$\begin{aligned}P(\ d(b_i,\tilde{b}_i)\leq\delta\ ) &= P(\ \sum_{j\in\chi_i}|b_{ij}-\tilde{b}_{ij}|\leq\delta\ )\\ &= \sum_{k\leq\delta}P(\ \sum_{j\in\chi_i}|b_{ij}-\tilde{b}_{ij}|=k\ )\\ &= \sum_{k\leq\delta}\binom{|\chi_i|}{k}\times p_i^k\times(1-p_i)^{|\chi_i|-k}\end{aligned}$$

since independent random variables $(|b_{ij}-\tilde{b}_{ij}|)_j$ are Bernoulli variables of parameter p_i.