DATA, PRIVACY AND THE INDIVIDUAL

DIFFERENTIALLY PRIVATE DATA SETS

JORDI SORIA-COMAS
UNIVERSITAT ROVIRA I VIRGILI

NOVEMBER 2019

DIFFERENTIALLY PRIVATE DATA SETS: METHODS, LIMITATIONS AND MITIGATION STRATEGIES

Jordi Soria-Comas

Universitat Rovira i Virgili

ABSTRACT

Data set releases are the most convenient way to make data available for secondary use: in principle, they allow analysts to carry out any data analysis task (e.g., exploratory data analysis). However, data set releases are a great threat to privacy. This is the issue that privacy preserving data publishing (PPDP) aims to address. Among the available sanitization methods, differential privacy (DP) stands out for the strong privacy guarantees it offers. The fact that DP offers protection regardless of the side information available to intruders is very convenient in the current landscape (pervasive data collection and many untrusted data controllers). However, such strong guarantees have a downside: the information loss we incur when using DP is likely to be large. As a result, there is no standard methodology to generate DP data sets and the use of DP for PPDP is rather limited. In this work, we review the main approaches used in the generation of DP data sets (i.e., histograms, and record aggregation and masking), and describe the advantages and the limitations of each of these approaches in terms of computational cost and information loss. Next, we describe some of the strategies that have been proposed to mitigate the previously described limitations. Among these, we highlight two common strategies: to increase the privacy budget, and to use a relaxed version of DP. Using large privacy budgets is common; however, it has an important downside: DP itself becomes meaningless. Using relaxed versions of DP allows us reduce the information loss while keeping reduced but meaningful privacy guarantees.

Reference to this paper should be made as follows:
Soria-Comas, J. (2019) "Differentially Private Data Sets: Methods, Limitations and Mitigation Strategies", *Data, Privacy and the Individual*.

INTRODUCTION

Differential privacy (DP, [5]) is well-known for the strong privacy guarantees it provides. One remarkable feature of DP is that its privacy guarantees hold regardless of the side knowledge available to attackers. This is very convenient in the current landscape: data collection is pervasive and it is unfeasible to make well-grounded assumptions about the side knowledge available to attackers. However, DP has been criticized for the information loss it implies [1, 16]. Such criticisms reveal that some simple data analyses that were usually considered safe (e.g. the publication of magnitude tables traditionally done by national statistical institutes) require a thorough masking under DP. There has been significant controversy around these criticisms, which is the result of two opposing views: those who favor utility over privacy (and disregard DP for being too strict, for instance, by not making assumptions about the side knowledge available to intruders) and those who favor privacy over utility (and claim that privacy must be guaranteed regardless of any drawbacks).

DP was designed to protect data subjects' privacy in database queries: the response to any query should be similar regardless of the presence or absence of any subject in the data set. The extent to which queries are allowed to differ is the privacy budget. We can spend the privacy budget in a single query or split it among several queries.

The proposal of DP as a privacy definition for queries (rather than for data sets) was motivated by several results showing that, unless a thorough masking is done, the publication of data sets is likely to result in privacy breaches [4, 7]

DP assumes the presence of a trusted party who holds the database and responds to the queries submitted by database users in a privacy-preserving manner. By limiting the protection to the queries posted by database users, the entire privacy budget can be spent on data analysis tasks that matter. However, there is an issue: once the privacy budget is exhausted, no more queries can be answered. If the number of queries that we expect is large, we may need to allocate a small privacy budget to each query, which reduces the accuracy of the responses. Some techniques have been proposed to mitigate this issue, such as the median mechanism [15] and the sparse vector technique [10]. However, these techniques offer only limited improvements. The generation of DP data sets remains the only option if we want to allow for unlimited data analysis. However, when generating DP data sets, we should keep in mind that accuracy guarantees are likely to be very limited [5].

The generation of DP data sets is conceptually simple: a DP data set is a DP response to a query whose outcome is the whole data set. In practice, the generation of DP data sets is complex because we not only want to enforce DP but also to incur in as small information loss as possible. To illustrate this point, let us consider a naive DP data set generation method that directly masks each record to make it DP. Since DP hides the presence or absence of any subject in the data set, each record must be thoroughly masked, leading

to a large information loss. To avoid this issue, the generation of DP data sets is based on queries whose result is mostly unaffected by the presence or absence of one record. However, even in the latter case, the generation of data sets with strict DP guarantees remains feasible only for low complexity data domains.

Going back to the protection offered by DP, we notice that, in the scenario proposed by DP, data subjects are not protected from the data collector. Indeed, DP assumes that the data collector is trusted. This assumption comes in handy because it allows for simpler sanitization mechanisms that incur in less information loss. However, it is often unrealistic in the current landscape. Local DP is an alternative to DP that is capable of offering strong DP privacy guarantees in the presence of an untrusted data collector. In local DP, each data subject has to protect her data locally before submitting it to the untrusted data collector. Usually in local DP the masking is adjusted to the data analysis that will be performed afterwards. This makes local DP less flexible than plain DP. Alternatively, to allow for arbitrary data analysis tasks, we can generate a DP data set by processing the local DP data records received. However, local DP being more restrictive than plain DP, the range of available methods is narrower and, thus, the information loss higher.

In this work, we review the main approaches used in the generation of DP data sets both in the presence of trusted and untrusted data collectors. We describe the limitations of each of these approaches in terms of computational cost and information loss. Next, we describe some strategies that have been used to deal with the previous limitations: the use of large privacy budgets (which makes DP meaningless) and the use of relaxations of DP (which allow us to reduce the information loss while keeping meaningful privacy guarantees).

PRELIMINARIES

Differential Privacy

Differential privacy [5] was originally proposed as a way to limit disclosure risk in database queries. In this setting, the differentially private sanitization mechanism sits between users submitting queries and the database controller answering them. To preserve the privacy of data subjects, the sanitization mechanism must guarantee that the impact that each data subject has on the query result is limited (according to an ϵ parameter).

ϵ-Differential privacy. A randomized function κ gives ϵ-differential privacy (or ϵ-DP) if, for all data sets $D1$ and $D2$ that differ in one record (a.k.a. neighbor data sets), and all $S \subset Range(\kappa)$, we have $Pr(\kappa(D1) \in S) \leqslant \exp(\epsilon)Pr(\kappa(D2) \in S)$.

Given a query f, the goal is to find a function κf that satisfies ϵ-DP and approximates f as closely as possible. The value of $\kappa f(D)$ is then returned as a privacy preserving replacement of $f(D)$.

A significant advantage of DP over alternative privacy models is that the protection offered by DP is independent of the side information available to intruders. This has made DP very popular among the research community. On the other hand, its deployment in real-world applications is rather limited. Except for a number of well-behaved applications (queries that are stable to the modification of one record), the impact of DP on the query responses may be large.

The Laplace mechanism is the most common way to attain DP for numerical queries. It masks the query output by adding a random noise whose magnitude is proportional to the sensitivity of the query (that is, the maximum variability of the query result between neighbor data sets).

Laplacian mechanism. Let f be a query function with values in $\mathbb{R}$. The randomized function $\kappa f = f + \text{Laplace}(0, (\Delta f)/(\epsilon))$, where $\Delta f = \max\|f(D1) - f(D2)\|1$ is the sensitivity of f, isϵ-DP.

Privacy in Data Set Releases

We consider a data set in which each record refers to a different data subject. Traditionally, privacy in data set releases has been tackled by the following two approaches:

Anonymity. It should not be possible to link records in the released data set to a specific subject.

Confidentiality. Access to the released data should not reveal confidential information about any specific subject.

DP is formulated as a privacy model for database queries. As such, the privacy guarantees that it offers are, in principle, unrelated to anonymity and confidentiality: the probability of getting a specific DP data set must be similar between original data sets that differ in any one record. However, DP privacy guarantees are readily interpretable in terms of anonymity and confidentiality. DP weakens the link between subjects and records because the probability of any given DP data set is similar regardless of the presence or absence of any subject in the original data set. Similarly, DP weakens the link between subjects and their confidential data.

Differential Privacy and the GDPR

The purpose of the GDPR is to protect data subjects by placing limits to the processing of their personal information. According to the GDPR, personal information is any information that concerns an identified or identifiable person. The GDPR is guided by

some general principles. Among them, it requires personal data to be collected only for specified, explicit and legitimate purposes and not further processed in a manner that is incompatible with those purposes. Moreover, the GDPR recognizes several rights that data subjects have over their data. For instance, data subjects have the right to access, rectify, and delete their data.

The purpose of the GDPR is not to thwart data analysis but rather to make sure that data analysis is compatible with the privacy of data subjects. For this reason, the GDPR does not foreclose the execution of arbitrary data analysis tasks (other than the one that triggered the data collection). It rather describes the conditions under which such data analyses can be conducted.

As the focus of the GDPR is on personal data, anonymization is the primary technique to fulfil its requirements. A piece of information is anonymous when it does not relate to an identified or identifiable natural person or when it has been transformed in such a manner that the data subject is not or no longer identifiable. However, contrary to other privacy regulations, the GDPR does not describe how the anonymization must be done. DP is one among a variety of approaches that can be used to anonymize data sets.

Alternatively, the GDPR allows to carry out processing tasks on personal data for purposes other than those that triggered data collection provided that some conditions are met. Among these, further processing for statistical purposes is allowed provided that it is not incompatible with the purpose that triggered the data collection. By using DP to protect the results of data analyses we make sure that only statistical analyses return valid results: the noise required to attain DP in queries that concern a single individual is large, which makes the result unreliable.

GENERATION OF DP DATA SETS

Although DP was initially proposed in a query-response setting, attempts to publish DP data sets (non-interactive setting) took place soon after its inception [11]. In the non-interactive setting, the trusted data collector generates a DP data set and releases it. Then, data users can freely analyze the released data set.

DP can handle data set releases seamlessly: a DP data set is simply a DP answer to a query that asks for the whole data set. However, in practice, the generation of utility preserving DP data sets is a complex topic. The main difficulty lays on finding a low sensitivity query that returns a good enough approximation of the original data set. Among the variety of methods that have been proposed for the generation of DP data sets, there are two main approaches: histograms [23, 24, 25], and record aggregation and masking [19, 17, 21, 20]. In this section, we describe them and explain their limitations.

DP Data Sets Based on Histograms

Histograms are a common approach to generate DP data sets. Essentially, we partition the domain of the data set into a set of bins and give DP counts of the records contained within each of these bins. The motivation for using histograms is their low sensitivity: adding or removing a record from a data set changes the count of a single bin by one. Hence, adding a small amount of noise to the count of each bin suffices to attain differential privacy.

Although attaining DP for a given histogram query is straightforward, the generation of DP data sets based on histograms presents significant difficulties. These are related to the information loss that results from approximating the original data set as a DP histogram. Let us describe the two sources of error:

Partitioning error. Records that are different may belong to the same partition bin. Thus, by representing original records in terms of histogram bin, we lose some of the information in the records. The amount of information loss depends on the granularity of the set of bins.

Error due to DP counts. To attain DP we need to mask the bin counts by adding some random noise.

Ideally, we would like to minimize both types of error; however, this is not possible. To reduce the partitioning error, we have to increase the granularity of the bins. By doing so, we reduce the counts associated to bins, which in turn increases the impact of adding a fixed amount of noise to them.

In the rest of this section, we describe some histogram-based methods to generate DP data sets and explain their limitations. Notice, however, that providing a comprehensive survey of them is beyond the scope of this work. More details can be found in [9]. Methods based on histograms can be broadly classified into two categories depending on the presence or absence of a target set of histogram bins. When a target set of bins is given beforehand, the focus goes solely into minimizing the error due to DP counts.

Baseline method. This is the most basic method. It matches the generation of DP histograms previously described: it adds independent $\text{Laplace}(1/\epsilon)$ distributed noise to the count of each partition set [5]. When using the baseline method, the absolute error in range queries (queries that ask for the sum of counts of all bins included in a given range of values) has order $\mathcal{O}(\sqrt{k})$, where k is the number of bins included in the given range. This makes the method appropriate only for small sets of bins.

To avoid the error in the counts to grow as $\mathcal{O}(\sqrt{k})$, we need to reduce the number of bins that we need to aggregate to compute the count of a given range. Some of the approaches that have been proposed to address this issue are:

Hierarchical method [14]. Rather than computing a single histogram, the hierarchical method computes h histograms, each of them with a different granularity. At the top level of the hierarchy, we have a single bin that covers the whole data domain D. Each other level of the hierarchy is a refinement of the bin set in the previous level. DP counts are then computed for each of the bin sets in the hierarchy, and the counts combined to get a DP count for any given range query. The advantage of the hierarchical method is that (by combining partition sets from different levels), each range query can be answered based on no more than $2(b-1)h$ partition sets, where b is the branching factor and h is the height of the tree. Assuming that the privacy budget is evenly distributed across all the partitions, the worst case absolute error has order $O((\log k)^{3/2})$. The previous description of the hierarchical method is based on a single attribute. Although it can be applied to data sets with an arbitrary number of attributes, the advantage of using it decreases as the number of dimensions increases. This is a major drawback as nowadays most data sets have a large number of attributes.

Query selection method [2]. The purpose of this method is to generate a DP data set that is optimal in answering a given family of count queries. It starts with a prefixed (e.g. uniformly distributed) DP data set and selects the query whose result has the greatest deviation with respect to the actual data set. Then the DP data set is adjusted to the actual query result.

When a target set of bins does not exist, the DP data set generation approach has to propose one. In the bin selection, we must account for both the partitioning error and the error due to DP counts. The most basic method is to partition the data domain into a set of equal sized bins that are independent of the data set. The main problem with this method is that dense and scarce regions of the data domain are equally treated. This is not optimal. In dense regions, we can afford smaller bins (which reduce the partitioning error) while still keeping a reasonable error due to the DP counts. On the contrary, in scarce regions, we must use coarser bins to keep the error associated to the DP counts within a reasonable level. Ideally, we would like to adjust the granularity of the bins to the actual data set. However, to satisfy differential privacy, any binning decision must be done in a differentially-private way. In general, this means that a part of the privacy budget must be used to define the partition and, consequently, less privacy budget can be spent in the DP counts.

The use of histograms is appropriate for simple data sets (with few attributes and attribute domains limited to a small set of categorical values). In more complex data sets (such as those with moderate to large number of attributes that can be either categorical or continuous) the use of histograms presents severe limitations: for a fixed granularity in each attribute, the number of histogram bins grows exponentially with the number of attributes, which has a severe impact on both computational cost and accuracy. Computational issues derive from the fact that we have to compute and store the counts for a large number of bins. Accuracy issues result from having a large number of scarcely

populated bins. For instance, along the lines of [25], let us consider a data set with 1 million records and 10 attributes, each of them with 10 possible values. The average count in each bin is 10^{-4}, whereas the average absolute noise introduced by a Laplace noise to attain ϵ-DP for count queries is $1/\epsilon$. For sensible values of ϵ (e.g. $\epsilon = 1$), the error introduced in the count of each cell exceeds its actual value by a large margin, making the reported values meaningless.

Some dimensionality reduction techniques, such as [25,23], have been proposed to mitigate the effect of the number of attributes, but gains are limited.

DP Data Set based on Record Aggregation and Masking

Assume that, given a data set D, we want to generate $D\epsilon$ a ϵ-DP version of D. For each record $r \in D$, let $Ir(D)$ be the query that returns r. We can think of the data set D as the collected answers to the queries $Ir(D)$ for $r \in D$, and we can generate $D\epsilon$ by collecting ϵ-DP responses to $Ir(D)$ for $r \in D$. Such a naive procedure to generate a DP data necessarily leads to a large information loss: the purpose of DP is to make sure that individual records do not have any significant effect on query responses, which means that the accuracy of the responses to $Ir(D)$ must necessarily be low.

To make perturbative masking viable for the generation of the DP data set, we have to reduce the sensitivity of the queries used. We need a shift from Ir to queries that ask for aggregated or statistical information. As the latter queries depend on several records, they are more stable to changes in one record. In DP terms, they are less sensitive and, thus, less masking is needed to attain DP.

In this section, we describe several methods that use microaggregation. Microaggregation [3] is a well-known technique for controlling disclosure risk in data set releases that works in two stages:

First, the set of records in a data set is clustered in such a way that: i) each cluster contains at least k records; ii) records within a cluster are as similar as possible.

Second, each record within each cluster is replaced by a representative of the cluster, typically the average record.

This line of work was started in [19]. Queries Ir were replaced by $Ir \circ M$, where M is a microaggregation algorithm. The introduction of the microaggregation functions has two effects on the accuracy. On the one hand, there is an approximation error that results from replacing Ir by $Ir \circ M$. On the other hand, $Ir \circ M$ has reduced sensitivity, which lowers the amount of noise required to attain DP. For this approach to pay off the reduction in the noise required to attain DP must compensate the error due to microaggregation. In general, we observe the following trends: when the cluster size is small, increasing it yields a significant reduction in the sensitivity at the cost of a small increase in the

microaggregation error; when the cluster size is large, increasing it barely reduces the sensitivity, while the microaggregation error continues to increase.

Contrary to histogram-based approaches, the previously described microaggregation approach does not have computational time and space issues when the number of attributes grows. However, it is a well-known fact that clustering algorithms have a poor behavior when the number of attributes grows: for a large number of attributes, all the points tend to be far apart from each other, which leads to a poor set of clusters.

LOCAL DP DATA SETS

In the absence of a trusted party, data subjects must mask their data themselves before submitting it to the data collector. As each record is very sensitive, a thorough masking must be applied. Thus, if data collector merely collects masked records, the information loss in the resulting data set will be large. To mitigate this issue, the masking must be done in a way that allows the data collector to get a more accurate estimate the distribution of the original data.

Randomized response is used in [8] to generate local DP data sets. Randomized response [22] is a mechanism that respondents to a survey can use to protect their privacy when asked about the value of sensitive attribute (e.g. did you take drugs last month?). The interesting point is that the data collector can still estimate the empirical distribution of the true answers of the respondents from the randomized responses. Randomized response is usually applied on each attribute independently. However, by doing so the relation between attributes is lost. [8] proposes a mechanism to give estimates of the marginal distributions of each attribute while keeping the residual dependency attributes that is present in the randomized data set.

RELAXING PRIVACY GUARANTEES

For PPDP to make sense, we should be able to get valid statistical results out of the protected data set. With current DP data generation techniques, this is feasible only for simple data sets (few attributes with few categories). It is unclear if further research in DP will allow us to tackle complex data sets effectively. In this section, we describe some methods to generate DP-like data sets that are capable of dealing with more complex data sets. The approach in all these methods is the same: relax the privacy guarantees in exchange for reduced information loss.

Increase ϵ

When using DP, we need to fix the privacy budget ϵ, which determines the level of protection that we get. Ideally, ϵ should be small. Values such as 0.1, $\ln 2$, or even 1 are

considered safe. In practice, the values of ϵ used to generate DP data sets are usually much larger.

As a paradigmatic case, we discuss the use of DP in Apple. It is known that most software companies collect and analyze data from their customers, and it is also known that some customers may not feel confident with that practice. To reassure customers, Apple claims that they protect privacy by using DP. However, to keep the utility of the data, they are forced to use large values forϵ. The actual ϵ used depends on the particular system. In iOS 10, researchers have found out there is a daily ϵ of 14. Such a value is so large that DP become meaningless. In DP terms, $\epsilon = 14$ means that the inclusion of a single record in the data set can change the probability of the output of a given analysis task by 1,200,000. Moreover, such ϵ is a daily budget. It becomes 28 after two days, 42 after three days, and so on.

It would be easy to claim that Apple fails to provide any privacy protection. However, we should differentiate between the theoretical privacy guarantees (which are lacking) and the real privacy protection. DP is a worst case privacy model: privacy will be protected even if the attacker knows everything except the record associated to the a target data subject. To protect data subjects against such a strong attacker, the collected data must be thoroughly masked, even for large privacy budget. Thus, for the most common case of not so strong attackers, privacy is likely to be well protected. Having said this, we would like to remark that using a large ϵ to keep the utility of the data seems an inappropriate way to proceed. To get meaningful privacy guarantees, it is better to target another privacy model that is compatible with our data utility requirements.

Dimensionality Reduction Techniques

PPDP becomes increasingly difficult when the number of attributes grows. Although we focus on DP, this is also the case in other privacy models that seek to offer protection at the record level. In DP terms, this is easily understandable by recalling that DP limits the amount of information that is revealed about each data subject. Therefore, if the number of attributes grows, the amount of information that can be revealed about each attribute is reduced.

Although this problem is unavoidable, there have been some attempts to mitigate it. In Section 3.1↑, we discussed some methods that aim at increasing the accuracy of DP histograms by reducing the dimensionality. In this section, we focus on the method described in the work by Domingo-Ferrer et al. [8], which deals with dimensionality reduction in the context of record-masking based DP data sets. A peculiarity of this method is that it makes some interesting observations about the privacy risks associated to data set transformations to avoid the need of applying DP in the dimensionality reduction steps. Strictly speaking this method is not DP; however, the rationale underlying it seems convincing enough.

The proposed method is based on principal component analysis (PCA). PCA uses an orthogonal transformation to convert a set of observations of possibly correlated attributes into a set of values of linearly uncorrelated attributes called principal components. This transformation is defined in such a way that principal components are ordered by descending variance. That is, the first principal component accounts for as much of the variability in the data as possible, and each succeeding component in turn accounts for as much of the variability in the data as possible under the constraint that it is orthogonal to the preceding components. Using this representation of the data, dimensionality reduction is very simple. We just have to drop some principal components starting from the last one (which is the least informative one).

If we want strict DP, the PCA transformation should be done in a DP way. However, to avoid spending a part of the privacy budget in the PCA transformation, [8] argues that there is no need to make the principal components DP. The reason is that PCA is used as an internal transformation that is undone before releasing any data. As the principals components are not released, we don't face the risk associated to them.

The advantage of the reduced data set is that it has less attributes, which allows us to reduce the amount of masking required to attain DP. Of course, the dimensionality reduction steps incur in some information loss, but this cost is expected to be smaller than the gain that results from the reduced making.

Differential Privacy at the Attribute Level

When the number of attributes becomes large, dimensionality reduction techniques become ineffective. One paradigmatic case is that of collaborative filtering (CF) recommender systems. Such systems are used to recommend items to a subject based on her preferences and on the preferences of similar subjects. CF needs to store the known preferences for the target set of users and items. Thus, the amount of attributes stored about each subject is really large (one attribute for each of the target items). Although dimensionality reduction techniques are employed in collaborative filtering, they are insufficient to make DP feasible.

The well-known work by McSherry and Mironov [12] on DP collaborative filtering uses the following relaxation of DP. Rather that enforcing ϵ-DP at the record level, the goal is to enforce ϵ-DP at the attribute level. That is, for a given subject, her preferences about any given item are ϵ-DP. This is not ϵ-DP at the record level, but rather ϵm-DP, where m is the number of attributes in the data set. Obviously, as m is likely to be large, the protection of DP at the record level is meaningless.

Protecting an Alternative Data Set

The aggregation and masking approach to generate DP data sets fails when the number of attributes is large. The reason is that, as the number of attributes grows, the distance

between records increases. This limits our ability to conduct a meaningful record aggregation, which is at the heart of the aggregation and masking approach.

To be able to use microaggregation when the number of attributes is large, [20] proposes to work independently with each attribute. That is, we get a DP version of each attribute by microaggregating and masking it. Then attributes are put together to generate the DP data set. However, this approach has a caveat: it requires us to switch the target of protection from the original data set D to the microaggregated data set D'. In other words, given the original data set D, we generate a microaggregated version D'. Once we have D', we discard D and focus on the generation of a DP version of D'. This approach departs from DP. The rationale underlying it is that, as D' contains less information than D, a DP version of D' should be safe.

Other Relaxations of DP

The literature is full of relaxations of DP. All these relaxations seek to reduce the information loss by lowering the privacy guarantees that subjects receive. Although they are general relaxations that do not target PPDP, they can be applied to data set releases.

(ϵ, δ)-indistinguishability [13], which allows some additional margin δ to the requirements in DP.

(ϵ, δ)-probabilistic differential privacy [11], which allows DP conditions to be broken with probability δ. In other words, the probability that the adversary gains significant information about an individual is, at most, δ.

(μ, τ)-concentrated differential [6], which, similarly to the previous case, allows DP conditions to be broken with a probability that is determined by μ and τ.

ϵ-individual differential privacy [18], which relaxes DP by limiting the privacy guarantees to the actual data set (rather than guaranteeing that DP guarantees hold for any arbitrary data set that can be completely unrelated to the actual data set).

CONCLUSIONS

DP is recognized for the strong privacy guarantees that it provides. It was formulated in a query-response scenario, but sticking to such a scenario is problematic because it limits the amount of data analyses that can be run on the data. Each data analysis consumes a part of the available privacy budget ϵ; when the privacy budget is exhausted no more data analyses are possible. To avoid this limitation, we generate a DP data set, which can be subsequently analyzed without further restrictions.

The generation of DP data set is a complex topic. It is feasible to generate accurate DP data sets when the data domain is simple (e.g. a small number of categorical attributes,

each of them with a small number of categories). Additionally, some techniques have been proposed that allow us to deal effectively with slightly more complex data sets (e.g. a moderate number of attributes). However, existent techniques fail when the complexity of the data set grows. Either because the computational cost becomes large or because the information loss grows significantly.

To be able to generate DP-like data set for more complex data domains, some relaxations of DP have been proposed. In Section 5↑, we describe some of them: to increase the privacy budget, to apply dimensionality reduction techniques, to apply DP at the attribute level and to protect an alternative data set, among others.

REFERENCES

[1] J. Bambauer, K. Muralidhar, R. Sarathy. Fool's Gold: an illustrated critique of differential privacy. 2013.

[2] A. Blum, K. Ligett, A. Roth. A learning theory approach to noninteractive database privacy. Journal of the ACM, 60(2):1—25, 2013.

[3] J. Domingo-Ferrer, J. M. Mateo-Sanz. Practical data-oriented microaggregation for statistical disclosure control. IEEE Transactions on Knowledge and Data Engineering, 14(1):189—201, 2002.

[4] C. Dwork. A firm foundation for private data analysis. Communications of the ACM 54(1): 86—95, 2011.

[5] C. Dwork, F. McSherry, K. Nissim, A. D. Smith. Calibrating noise to sensitivity in private data analysis. Theory of Cryptography, Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006, pp. 265—284.

[6] C. Dwork, G. N. Rothblum. Concentrated differential privacy. CoRR, abs/1603.01887, 2016.

[7] C. Dwork, A. Smith, T. Steinke, J. Ullman, S. Vadhan. Robust Traceability from Trace Amounts. 2015 IEEE 56th Annual Symposium on Foundations of Computer Science, Berkeley, CA, 2015, pp. 650-669.

[8] J. Domingo-Ferrer, R. Mulero-Vellido, J. Soria-Comas. Multiparty Computation with Statistical Input Confidentiality via Randomized Response. Privacy in Statistical Databases, PSD 2018, Valencia, Spain, September 26-28, 2018, pp.175—186.

[9] N. Li, M. Lyu, D. Su, W. Yang. Differential privacy: From theory to practice. Synthesis Lectures on Information Security, Privacy, & Trust, Morgan & Claypool, 2016.

[10] M. Lyu, D. Su, N. Li. Understanding the Sparse Vector Technique for Differential Privacy. Proc. VLDB Endow., 10(6):637—648, 2017.

[11] A. Machanavajjhala, D. Kifer, J. Abowd, J. Gehrke, L. Vilhuber. Privacy: theory meets practice on the map. Proceedings of the 2008 IEEE 24th International Conference on Data Engineering, 2008, pp. 277—286.

[12] F. McSherry, I. Mironov. Differentially Private Recommender Systems: Building Privacy into the Netflix Prize Contenders. Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2009, pp. 627—636.

[13] K. Nissim, S. Raskhodnikova, A. Smith. Smooth sensitivity and sampling in private data analysis. Proceedings of the thirty-ninth annual ACM symposium on Theory of computing, 2007, pp. 75—84.

[14] W. Qardaji, W. Yang, N. Li. Understanding Hierarchical Methods for Differentially Private Histograms. Proc. VLDB Endow., 6(14):1954—1965, 2013.

[15] A. Roth, T. Roughgarden. Interactive Privacy via the Median Mechanism. Proceedings of the 42th ACM Symposium on Theory of Computing, 2010, pp.765—774.

[16] R. Sarathy, K. Muralidhar. Evaluating Laplace noise addition to satisfy differential privacy for numeric data. Transactions on Data Privacy, 4(1):1—17, 2011.

[17] D. Sánchez, J. Domingo-Ferrer, S. Martínez, J. Soria-Comas. Utility-preserving differentially private data releases via individual ranking microaggregation . Information Fusion, 30:1 - 14, 2016.

[18] J. Soria-Comas, J. Domingo-Ferrer, D. Sanchez, D. Megias. Individual differential privacy: a utility-preserving formulation of differential privacy guarantees. Trans. Info. For. Sec., 12(6):1418—1429, 2017.

[19] J. Soria-Comas, J. Domingo-Ferrer, D. Sánchez, S. Martínez. Enhancing data utility in differential privacy via microaggregation-based k-anonymity. The VLDB Journal, 23(5):771—794, 2014.

[20] J. Soria-Comas, J. Domingo-Ferrer. Differentially private data publishing via optimal univariate microaggregation and record perturbation. Know.-Based Syst., 153(C):78—90, 2018.

[21] J. Soria-Comas, J. Domingo-Ferrer. Differentially private data sets based on microaggregation and record perturbation. MDAI, 10571:119—131, 2017.

[22] S. L. Warner. Randomized Response: A Survey Technique for Eliminating Evasive Answer Bias. Journal of the American Statistical Association, 60(309):63—69, 1965.

[23] Y. Xiao, L. Xiong, C. Yuan. Differentially private data release through multidimensional partitioning. In Secure Data Management (Jonker, Willem and Petković, Milan, ed.). Springer Berlin Heidelberg, 2010.

[24] J. Xu, Z. Zhang, X. Xiao, Y. Yang, G. Yu. Differentially private histogram publication. Proceedings of the 28th IEEE International Conference on Data Engineering:32—43, 2012.

[25] J. Zhang, G. Cormode, C. M. Procopiuc, D. Srivastava, X. Xiao. PrivBayes: private data release via bayesian networks. Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data:1423—1434, 2014.