Profile-based Privacy for Locally Private Computations

Joseph Geumlek and Kamalika Chaudhuri
University of California, San Diego
Email: {jgeumlek, kamalika}@cs.ucsd.edu

***Abstract*—Differential privacy has emerged as a gold standard in privacy-preserving data analysis. A popular variant is local differential privacy, where the data holder is the trusted curator. A major barrier, however, towards a wider adoption of this model is that it offers a poor privacy-utility trade-off.**

In this work, we address this problem by introducing a new variant of local privacy called *profile-based privacy*. The central idea is that the problem setting comes with a graph G of data generating distributions, whose edges encode sensitive pairs of distributions that should be made indistinguishable. This provides higher utility because unlike local differential privacy, we no longer need to make every pair of private values in the domain indistinguishable, and instead only protect the identity of the underlying distribution. We establish privacy properties of the profile-based privacy definition, such as post-processing invariance and graceful composition. Finally, we provide mechanisms that are private in this framework, and show via simulations that they achieve higher utility than the corresponding local differential privacy mechanisms.

A full version of this paper is accessible at: https://arxiv.org/abs/1903.09084

I. Introduction

Increasing amounts of sensitive data are being collected and analyzed, and consequently the need for privacy preserving data analysis has grown. Differential privacy has emerged as the gold standard of privacy-preserving data analysis, and a version, known as local differential privacy [1], has been used in many applications such as Google's RAPPOR [2] and Apple's iOS data collection. This local model consists of users privatizing their own data before submission to a data curator. Due to the robustness of differential privacy under further computation, this model protects privacy regardless of the trust in the curator, now or in the future.

However, a major barrier for the local model is the undesirable utility sacrifices of the submitted data. A local differential privacy implementation achieves much lower utility than a similar method that assumes trusts in the curator. Strong lower bounds have been found for the local framework [1], leading to pessimistic results that necessarily require massive amounts of data to achieve both privacy and utility.

In this work, we address this challenge by proposing a new restricted privacy definition, called profile-based privacy. The central idea relies on specifying a graph G of *profiles* or data generating distributions, where edges encode sensitive pairs of distributions which should be made indistinguishable. Our framework does not require that all features of the observed data be obscured; instead only the information connected to identifying the distributions must be perturbed. This offers privacy by obscuring data from sensitive pairs of profiles while side-stepping the utility costs of local differential privacy, where every possible pair of observations must be indistinguishable. As a concrete example, suppose we have commute trajectories from a city center to four locations - A, B, C and D, where A and B are north and C and D are south of the center. Profile based privacy can make the trajectories that originate in A and B and those that originate in C and D indistinguishable. This offers better utility than local differential privacy, which would make every trajectory indistinguishable, while still offering finer grained privacy in the sense that an adversary will only be able to tell that the commuter began north or south of the city center.

We show that profile-based privacy satisfies some of the beneficial properties of differential privacy, such as post-processing invariance and certain forms of graceful composition. We provide new mechanisms in this definition that offer better utility than local differential privacy, and conclude with theoretical as well as empirical evidence of their effectiveness.

II. Related Work

Our proposed definition is related to two commonly used privacy frameworks: the generalized Pufferfish privacy framework [3], and geoindistinguishability [4]. Like our definition, Pufferfish presents an explicit separation of sensitive and insensitive information with distributional assumptions. However, we focus on a local case with distributional secrets, while the existing Pufferfish literature targets value-dependent secrets in a global setting. Our definition is also similar to geoindistinguishability, but our work does not require an explicit metric and applies more readily to a discrete setting.

Our methods also resemble those seen under maximal-leakage-constrained hypothesis testing [5]. The maximal-leakage framework also employs a distribution-focused mechanism design, but solves a different problem. Our work aims to prevent identifying distributions while preserving and identifying observed values where possible. The maximal-leakage setting inverts this goal, and protects the observed values while maximizing the detection of hypotheses on the distributions. This distinction in goal also applies with respect to the distributional privacy framework [6]. Finally, our work can also be viewed in relation to information theoretic definitions dues to the deep connections present from differential privacy [7].

III. Preliminary: Privacy definitions

We begin with defining local differential privacy – a prior privacy framework that is related to our definition.

Definition 1: A randomized mechanism $\mathcal{A} : \mathcal{X} \to \mathcal{Y}$ achieves ϵ-local differential privacy if for every pair (X, X') of individuals' private records in $\mathcal{X}$ and for all outputs $y \in \mathcal{Y}$ we have:

$$\Pr(\mathcal{A}(X) = y) \leq e^{\epsilon} \Pr(\mathcal{A}(X') = y). \tag{1}$$

Concretely, local differential privacy limits the ability of an adversary to increase their confidence in whether an individual's private value is X versus X' even with arbitrary prior knowledge. These protections are robust to any further computation performed on the mechanism output.

IV. Profile-based Privacy Definition

Before we present the definition and discuss its implications, it is helpful to have a specific problem in mind. We present one possible setting in which our profiles have a clear interpretation.

A. Example: Resource Usage Problem Setting

Imagine a shared workstation with access to several resources, such as network bandwidth, specialized hardware, or electricity usage. Different users might use this workstation, coming from a diverse pool of job titles and roles. An analyst wishes to collect and analyze the metrics of resource usage, but also wishes to respect the privacy of the workstation users. With local differential privacy, any two distinct measurements must be rendered indistinguishable. Under our alternative profile-based framework, a choice exists to protect user identities instead of measurement values. This shifts the goal away from hiding all features of the resource usages, and permits measurements to be released more faithfully when not indicative of a user's identity.

B. Definition of Profile-based Differential Privacy

Our privacy definition revolves around a notion of profiles, which represent distinct potential data-generating distributions. To preserve privacy, the mechanism's release must not give too much of an advantage in guessing the release's underlying profile. However, other facets of the observed data can (and should) be preserved, permitting greater utility than local differential privacy.

Definition 2: Given a graph $G = (\mathcal{P}, E)$ consisting of a collection $\mathcal{P}$ of data-generating distributions ("profiles") over the space $\mathcal{X}$ and collection of edges E, a randomized mechanism $\mathcal{A} : \mathcal{X} \times \mathcal{P} \to \mathcal{Y}$ achieves (G, ϵ)-profile-based differential privacy if for every edge $e \in E$ connecting profiles P_i and P_j, with random variables $X_i \sim P_i$ and $X_j \sim P_j$, and for all outputs $y \in \mathcal{Y}$ we have:

$$\frac{\Pr(\mathcal{A}(X_i, P_i) = y)}{\Pr(\mathcal{A}(X_j, P_j) = y)} \leq e^{\epsilon}. \tag{2}$$

Inherent in this definition is an assumption on adversarial prior knowledge: the adversary knows each profile distribution, but has no further auxiliary information about the observed data X. The protected secrets are the identities of the source distributions, and are not directly related to particular features of the data X. Put another way, the adversarial goal here is to distinguish P_i versus P_j, rather than any fixed X versus X' pair in local differential privacy. These additional assumptions in the problem setting, however, permit better performance. By not attempting to completely privatize the raw observations, information that is less relevant for guessing the sensitive profile identity can be preserved for downstream tasks.

The flexible specification of sensitive pairs via edges in the graph permits privacy design decisions that also impact the privacy-utility trade-off. When particular profile pairs are declared less sensitive, the perturbations required to blur those profiles can be avoided. Such decisions would be impractical in the data-oriented local differential privacy setting, where the space of pairs of data sets is intractably large.

The profile-based differential privacy framework exists as an inverse to the goals seen in maximal-leakage-constrained hypothesis testing [5], where hypotheses serve a similar role to our profiles. While they focus on protecting observation-privacy and maintaining distribution-utility, we focus on maintaining observation-utility and protecting distribution-privacy. Both settings are interesting and situational.

C. Discussion of the Resource Usage Problem

This privacy framework is quite general, and as such it helps to discuss its meaning in more concrete terms. Let us return to the resource usage setting. We'll assume that each user has a personal resource usage profile known prior to the data collection process. The choice of edges in the graph G has several implications. If the graph has many edges, the broad identity of the workstation user will be hidden by forbidding many potential inferences. However, this protection does not require all the information about resource usage to be obscured. For example, if all users require roughly the same amount of electricity at the workstation, then electrical usage metrics will not require much obfuscation even with a fully connected graph.

A more sparse graph might only connect profiles with the same job title or role. These sensitive pairs will prevent inferences about particular identities within each role. However, without connections across job titles, no protection is enforced against inferring the job title of the current workstation user. Thus such a graph declares user identities sensitive, while a user's role is not sensitive. When users in the same role have similar usages, this sparser graph will require less perturbations of the data.

One important caveat of this definition is that the profile distributions must be known and are assumed to be released a priori, i.e. they are not considered privacy sensitive. If the user profiles cannot all be released, this can be mitigated somewhat by reducing the granularity of the graph. A graph consisting only of profiles for each distinct job role can still encode meaningful protections, since limiting inferences on job role

can also limit inferences on highly correlated information like the user's identity.

The trade-off in profile granularity is subtle, and is left for future exploration. More profiles permit more structure and opportunities for our definition to achieve better utility than local differential privacy, but also require a greater level of a priori knowledge.

V. Properties

Our privacy definition enjoys several similar properties to other differential-privacy-inspired frameworks. The post-processing and composition properties are recognized as highly desired traits for privacy definitions [3].

a) Post-Processing: The post-processing property specifies that any additional computation (without access to the private information) on the released output cannot result in worse privacy. Following standard techniques, our definition also shares this data processing inequality.

Observation 3: If a data sample X_i is drawn from profile P_i, and $\mathcal{A}$ preserves (G, ϵ)-profile-based privacy, then for any (potentially randomized) function F, the release $F(\mathcal{A}(X_i, P_i))$ preserves (G, ϵ)-profile-based privacy.

b) Composition: The composition property allows for multiple privatized releases to still offer privacy even when witnessed together. Our definition also gets a compositional property, although not all possible compositional settings behave nicely. We mitigate the need for composition by focusing on a local model where the data mainly undergoes one privatization.

Profile-based differential privacy enjoys additive composition if truly independent samples X are drawn from the same profile. The proof of this follows the same reasoning as the additive composition of differential privacy.

Observation 4: If two independent samples X_1 and X_2 are drawn from profile P_i, and $\mathcal{A}_1$ preserves (G, ϵ_1)-profile-based privacy and $\mathcal{A}_2$ preserves (G, ϵ_2)-profile-based privacy, then the combined release $(\mathcal{A}_1(X_1, P_i), \mathcal{A}_2(X_2, P_i))$ preserves $(G, \epsilon_1 + \epsilon_2)$-profile-based privacy.

A notion of parallel composition can also be applied if two data sets come from two independent processes of selecting a profile. In this setting, information about one instance has no bearing on the other. This matches the parallel composition of differential privacy when applied to multiple independent individuals.

Observation 5: If two profiles P_i and P_j are independently selected, and two observations $X_i \sim P_i$ and $X_j \sim P_j$ are drawn, and $\mathcal{A}_1$ preserves (G, ϵ_1)-profile-based privacy and $\mathcal{A}_2$ preserves (G, ϵ_2)-profile-based privacy, then the combined release $(\mathcal{A}_1(X_i, P_i), \mathcal{A}_2(X_j, P_j))$ preserves $(G, \max\{\epsilon_1, \epsilon_2\})$-profile-based privacy.

However, this framework cannot offer meaningful protections against adversaries that know about correlations in the profile selection process. For example, consider an adversary with knowledge that profile P_k is always selected immediately after either P_i or P_j are selected. An edge obscuring P_i versus P_j will not prevent the adversary from deducing P_k in the next round. This matches the failure of differential privacy to handle correlations across individuals. The definition also does not compose if the same observation X is reprocessed, as it adds correlations unaccounted for in the privacy analysis. Although such compositions would be valuable, it is less important when the privatization occurs locally at the time of data collection.

Placing these results in the context of reporting resource usage, we can bound the total privacy loss across multiple releases in two cases. Additive composition applies if a single user emits multiple independent measurements and each measurement is separately privatized. When two users independently release measurements, each has no bearing on the other and parallel composition applies. If correlations exist across measurements (or across the selection of users), no compositional result is provided.

VI. Mechanisms

We now provide mechanisms to implement the profile-based privacy definition. Before getting into specifics, let us first consider the kind of utility goals that we can hope to achieve. We have two primary aspects of the graph G we wish to exploit. First, we wish to preserve any information in the input that does not significantly indicate profile identities. Second, we wish to use the structure of the graph and recognize that some regions of the graph might require less perturbations than others.

A. The One-Bit Setting

We begin with a one-bit setting – where the input to the mechanism is a single private bit – and build up to the more general discrete setting.

The simplest case is when we have two profiles i and j represented by Bernoulli distributions P_i and P_j with parameters p_i and p_j respectively. Here, we aim to design a mechanism $\mathcal{A}$ that makes a bit b drawn from P_i or P_j indistinguishable; that is, for any $t \in \{0, 1\}$, with $b_i \sim P_i$ and $b_j \sim P_j$,

$$\frac{\Pr(\mathcal{A}(b_i, P_i) = t)}{\Pr(\mathcal{A}(b_j, P_j) = t)} \leq e^{\epsilon}. \tag{3}$$

A plausible mechanism is to draw a bit b' from a Bernoulli distribution that is independent of the input bit, but this discards all the information from the input.

We instead use a mechanism that flips the input bit with some probability $\alpha \leq 1/2$. Lower values of α improve the correlation between the output and the input. The flip-probability α is obtained by solving the following optimization problem:

$$\begin{aligned} \min \quad & \alpha \\ \text{subject to} \quad & \alpha \geq 0 \\ & \frac{p_i(1-\alpha) + (1-p_i)\alpha}{p_j(1-\alpha) + (1-p_j)\alpha} \in [e^{-\epsilon}, e^{\epsilon}] \\ & \frac{(1-p_i)(1-\alpha) + p_i\alpha}{(1-p_j)(1-\alpha) + p_j\alpha} \in [e^{-\epsilon}, e^{\epsilon}]. \end{aligned} \tag{4}$$

When $p_i = 0$ and $p_j = 1$ (or vice versa), this reduces to the standard randomized response mechanism [8]; however, α

may be lower if p_i and p_j are closer – a situation where our utility is better than local differential privacy's.

The mechanism described above only addresses two profiles. If we have a cluster of profiles representing a connected component of the profile graph, we can compute the necessary flipping probabilities across all edges in the cluster. To satisfy all the privacy constraints, it suffices to always use a flipping probability equal to the largest value required by an edge in the cluster. This results in a naive method we will call the One Bit Cluster mechanism, directly achieves profile-based privacy.

Theorem 6: The One Bit Cluster mechanism achieves (G, ϵ)-profile-based privacy.

B. The Categorical Setting

The One Bit Cluster mechanism has two limitations. First, it applies only to single bit settings and Bernoulli profiles, and not categorical distributions. Second, by treating all pairs of path-connected profiles similarly, it is overly conservative; when profiles are distant in the graph from a costly edge, it is generally possible to provide privacy with lesser perturbations for these distant profiles.

We now show how to address both. Addressing the first limitation is relatively straight-forward, and is done by picking additional constraints, as well as a (possibly) domain specific objective that maximizes a measure of fidelity between the input and the output. To address the second, we use ideas inspired by the smoothed sensitivity mechanism in differential privacy [9]. However, rather than smoothly calibrating perturbations across the entire space of data sets, a profile-based privacy mechanism needs only to smoothly calibrate over the specified profile graph, which is a tractable task.

Specifically, suppose we have k categorical profiles each with d categories; we introduce kd^2 variables to optimize, with each profile receiving a $d \times d$ transition matrix. To keep track of these variables, we introduce the following notation:

- $P_i, \ldots, P_k$: a set of k categorical profiles in d dimensions encoded as a vector.
- $A^1, \ldots, A^k$: A set of d-by-d transition matrix that represents the mechanism's release probabilities for profile i. $A^i_{j,k}$ represents the (j,k)-th element of the matrix A^i.
- $P_i A^i$ represents the d dimensional categorical distribution induced by the transition matrix A^i applied to the distribution P_i.
- In an abuse of notation, $P_i A^i \leq e^\epsilon P_j A^j$ is a constraint that applies element-wise to all components of the resulting vectors on each side.

With this notation, we can express our optimization task:

$$\min_{A^1,\ldots,A^k} \max(\text{off-diagonal entries of } A^1, \ldots, A^k) \tag{5}$$

$$\text{subject to } \forall i \in [n] \forall j \in [d] \forall k \in [d]: \quad 0 \leq A^i_{j,k} \leq 1$$

$$\forall i \in [n] \forall j \in [d]: \quad \sum_{k=1}^{d} A^i_{j,k} = 1$$

$$\forall (P_i, P_j) \in E: \; P_i A^i \leq e^\epsilon P_j A^j, \quad P_j A^j \leq e^\epsilon P_i A^i.$$

To address the tractability of the optimization, we note that each of the privacy constraints are linear constraints over our optimization variables. We further know the feasible solution set is nonempty, as trivial non-informative mechanisms achieve privacy. All that is left is to choose a suitable objective function to make this a readily solved convex problem.

To settle onto an objective will require some domain-specific knowledge of the trade-offs between choosing which profiles and which categories to report more faithfully. Our general choice is a maximum across the off-diagonal elements, which attempts to uniformly minimize the probability of any data corruptions. This can be further refined in the presence of a prior distribution over profiles, to give more importance to the profiles more likely to be used.

We define the Smooth Categorical mechanism as the process that solves the optimization (5) and applies the appropriate transition probabilities on the observed input.

Theorem 7: The Smooth Categorical mechanism achieves (G, ϵ)-profile-based privacy.

C. Utility Results

The following results present utility bounds which illustrate potential improvements upon local differential privacy; a more detailed numerical simulation is presented in Section VII.

Theorem 8: If $\mathcal{A}$ is a mechanism that preserves ϵ-local differential privacy, then for any graph G of sensitive profiles, $\mathcal{A}$ also preserves (G, ϵ)-profile-based differential privacy.

An immediate result of Theorem 8 is that, in general and for any measure of utility on mechanisms, the profile-based differential privacy framework will never require worse utility than a local differential privacy approach. However, in specific cases, stronger results can be shown.

Observation 9: Suppose we are in the single-bit setting with two Bernoulli profiles P_i and P_j with parameters p_i and p_j respectively. If $p_i \leq p_j \leq e^\epsilon p_j$, then the solution α to (4) satisfies $\alpha \leq \max\{0, \frac{p_j - e^\epsilon p_i}{2(p_j - e^\epsilon p_i) - (1 - e^\epsilon)}, \frac{p_i - e^\epsilon p_j + e^\epsilon - 1}{2(p_i - e^\epsilon p_j) + e^\epsilon - 1}\}$.

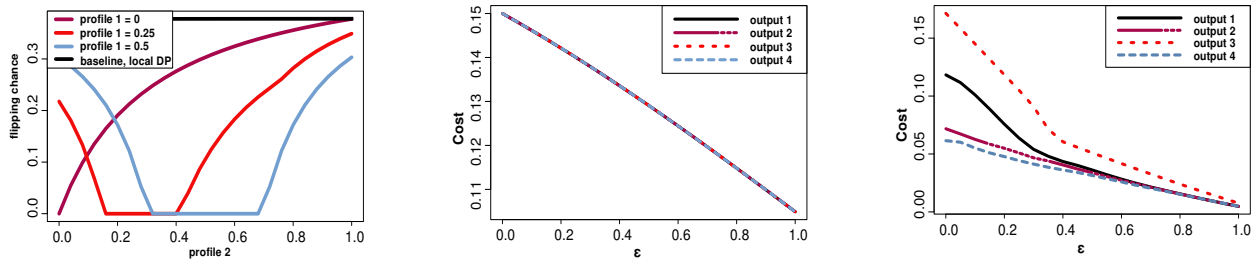
Observe that to attain local differential privacy with parameter ϵ by a similar bit-flipping mechanism, we need a flipping probability of $\frac{1}{1+e^\epsilon}$, while we get bounds of the form $\frac{1}{1+(1+\frac{e^\epsilon - 1}{p_j - e^\epsilon p_i})}$. Thus, profile based privacy does improve over local differential privacy in this simple case. The proof of Observation 9 follows from observing that this value of α satisfies all constraints in the optimization problem (4).

VII. Evaluation

We next evaluate our privacy mechanisms and compare them against each other and the corresponding local differential privacy alternatives. In order to understand the privacy-utility trade-off unconfounded by model specification issues, we consider synthetic data in this paper.

A. Experimental Setup

We look at two experimental settings – Bernoulli-Couplet, and Categorical-Chain.



(a) Bernoulli-Couplet, Our Method and Baseline.

(b) Categorical-Chain, Baseline (Local differential privacy). All 4 curves overlap.

(c) Categorical-Chain, Our Method.

Fig. 1. Experimental results in various settings. In all figures, lower is better.

a) Settings: In Bernoulli-Couplet, the profile graph consists of two nodes connected by a single edge $G = (\mathcal{P} = \{a, b\}, E = \{(a, b)\})$. Additionally, each profile is a Bernoulli distribution with a parameter p. In Categorical-Chain, the profile graph comprises of three nodes connected into a chain $P_1 - P_2 - P_3$. Each profile however, corresponds to a 4-dimensional categorical distribution, instead of a Bernoulli.

TABLE I
CATEGORICAL-CHAIN PROFILES

P_1	0.2	0.3	0.4	0.1
P_2	0.3	0.3	0.3	0.1
P_3	0.4	0.4	0.1	0.1

b) Baselines: For Bernoulli-Couplet, we use Warner's Randomized Response mechanism [8] as a local differentially private baseline, as well as our One Bit Cluster Mechanism. For Categorical-Chain, the corresponding local differentially private baseline is the K-ary version of randomized response, against our Smooth Categorical mechanism.

B. Results

Figure 1(a) plots the flipping probability for Bernoulli-Couplet as a function of the difference between profile parameters p. We find that as expected, as the difference between the profile parameters grows, so does the flipping probability and hence the noise added. However, in all cases, this probability stays below the corresponding value for local differential privacy – the horizontal black line – thus showing that profile-based privacy is an improvement.

Figures 1(b)-1(c) plot the utility across different outputs in the Categorical-Chain setting. We illustrate its behavior through a small setting with 3 profiles, each with 4 categories. We can no longer plot the entirety of these profiles, so at each privacy level we measure the maximum absolute error for each output. Thus, in this setting, each privacy level is associated with 4 costs of the form given in (6). This permits the higher fidelity of profile-irrelevant information to be seen.

$$cost_j = \max_{i \in [n]} |P^i_j - (P^i A^i)_j| \tag{6}$$

Our experiments show the categories less associated with the profile identity have lower associated costs than the more informative ones. However, the local differential privacy baseline fails to exploit any of this structure and performs worse.

VIII. CONCLUSION

In conclusion, we provide a novel definition of local privacy – profile based privacy – that can achieve better utility than local differential privacy. We prove properties of this privacy definition, and provide mechanisms for two discrete settings. Simulations show that our mechanisms offer superior privacy-utility trade-offs than standard local differential privacy.

Acknowledgements. We thank ONR under N00014-16-1-261, UC Lab Fees under LFR 18-548554 and NSF under 1804829.

REFERENCES

[1] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Local privacy and statistical minimax rates," in *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*. IEEE, 2013, pp. 429–438.

[2] 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.

[3] D. Kifer and A. Machanavajjhala, "A rigorous and customizable framework for privacy," in *Proceedings of the 31st ACM SIGMOD-SIGACT-SIGAI symposium on Principles of Database Systems*. ACM, 2012, pp. 77–88.

[4] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: Differential privacy for location-based systems," *arXiv preprint arXiv:1212.1984*, 2012.

[5] J. Liao, L. Sankar, F. P. Calmon, and V. Y. Tan, "Hypothesis testing under maximal leakage privacy constraints," in *Information Theory (ISIT), 2017 IEEE International Symposium on*. IEEE, 2017, pp. 779–783.

[6] M. F. Balcan, A. Blum, S. Fine, and Y. Mansour, "Distributed learning, communication complexity and privacy," in *Conference on Learning Theory*, 2012, pp. 26–1.

[7] W. Wang, L. Ying, and J. Zhang, "On the relation between identifiability, differential privacy, and mutual-information privacy," *IEEE Transactions on Information Theory*, vol. 62, no. 9, pp. 5018–5029, 2016.

[8] S. L. Warner, "Randomized response: A survey technique for eliminating evasive answer bias," *Journal of the American Statistical Association*, vol. 60, no. 309, pp. 63–69, 1965.

[9] K. Nissim, S. Raskhodnikova, and A. D. Smith, "Smooth sensitivity and sampling in private data analysis," in *Proceedings of the 39th Annual ACM Symposium on Theory of Computing, San Diego, California, USA, June 11-13, 2007*, 2007, pp. 75–84.