PSI (Ψ):
a Private data Sharing Interface[*]
(WORKING PAPER)

Marco Gaboardi[†] James Honaker [‡] Gary King [§] Jack Murtagh[¶]
Kobbi Nissim[‖] Jonathan Ullman[**] Salil Vadhan[††]

with contributions from
Nabib Ahmed, Andreea Antuca, Brendan Avent, Connor Bain, Victor Balcer, Jessica Bu,
Mark Bun, Stephen Chong, Fanny Chow, Vito D'Orazio, Anna Gavrilman, Benjamin Glass,
Caper Gooden, Paul Handorff, Raquel Hill, Allyson Kaminsky, Chan Kang, Murat Kuntarcioglu,
Vishesh Karwa, George Kellaris, Michael Lackner, Jack Landry, Hyun Woo Lim, Giovanni Malloy,
Nathan Manohar, Dan Muise, Marcelo Novaes, Ana Luisa Oaxaca, Sofya Raskhodnikova, Grace Rehaut,
Ryan Rogers, Or Sheffet, Adam D. Smith, Thomas Steinke, Clara Wang, Haoqing Wang,
Remy Wang, David Xiao, and Joy Zheng

November 30, 2016

Abstract

We provide an overview of the design of PSI ("a Private data Sharing Interface"), a system we are developing to enable researchers in the social sciences and other fields to share and explore privacy-sensitive datasets with the strong privacy protections of differential privacy.

---

[*] This work is part of the "Privacy Tools for Sharing Research Data" project at Harvard, supported by NSF grant CNS-1237235 and a grant from the Sloan Foundation. This is a working paper describing a vision for work that is still in progress, and is therefore authored by the leadership of the efforts. Future and accompanying publications that emphasize specific technical contributions will be authored by team members responsible for those contributions.

[†] Department of Computer Science and Engineering, University at Buffalo, SUNY. Work done in part while at the University of Dundee, UK and visiting the Center for Research on Computation & Society, John A. Paulson School of Engineering & Applied Sciences, Harvard University. `gaboardi@buffalo.edu`.

[‡] `james@hona.kr`; `http://hona.kr`

[§] Albert J. Weatherhead III University Professor, Harvard University, Institute for Quantitative Social Science. `king@harvard.edu`; `http://GaryKing.org`

[¶] John A. Paulson School of Engineering & Applied Sciences, Harvard University. `jmurtagh@seas.harvard.edu`

[‖] Department of Computer Science, Georgetown University, *and* Center for Research on Computation & Society, John A. Paulson School of Engineering & Applied Sciences, Harvard University. `kobbi.nissim@georgetown.edu`.

[**] College of Computer and Information Sciences, Northeastern University. Work done in part while affiliated with the Center for Research on Computation & Society, John A. Paulson School of Engineering & Applied Sciences, Harvard University. `jullman@ccs.neu.edu`

[††] Center for Research on Computation & Society, John A. Paulson School of Engineering & Applied Sciences, Harvard University. Work done in part while visiting the Shing-Tung Yau Center and the Department of Applied Mathematics at National Chiao-Tung University in Taiwan. Also supported by a Simons Investigator Award. `salil@seas.harvard.edu`.

1 The Problem

Researchers in all empirical fields are increasingly expected to widely share the data behind their published research, to enable other researchers to verify, replicate, and extend their work. Indeed, data-sharing is now often mandated by funding agencies [53, 52, 28] and journals [44, 29, 65]. To meet this need, a variety of open data repositories have been developed to make data-sharing easier and more permanent. The index from the *Registry of Research Data Repositories* surpassed 1500 different repositories in April 2016 [5], many of which are narrow in scope. The largest general purpose repositories include those that use the open-source Dataverse platform [15, 38], CERN's Zenodo, and the commercial Figshare [60], and Dryad [67] repositories.

However, many of the datasets in the social and health sciences contain sensitive personal information about human subjects, and it is increasingly recognized that traditional approaches such as stripping "personally identifying information" are ineffective at protecting privacy, especially if done by a lay researcher with no expertise in deidentification. This leads to two problems, one for privacy and one for utility:

1. There are numerous data sets, such as surveys, that have been "deidentified" via traditional means and increasingly are being deposited in publicly accessible data repositories. As the literature has repeatedly shown, it is likely that many subjects in these surveys can be reidentified by attackers with a moderate amount of background information, and thus their privacy may not be sufficiently well-protected.

2. There are numerous other data sets that researchers do not make available at all, or only with highly restrictive and time-consuming provisions. Such provisions can include a review by the original data depositor—who may no longer be accessible—and/or an Institutional Review Board (IRB), and a lengthy negotiation between institutions on the terms of use.

Thus, an important problem is to develop and deploy methods that can be used to offer greater privacy protections for datasets of the first type, ideally at little or no cost in utility[1], and enable the safe sharing of datasets of the second type.

Differential privacy [23] offers an attractive approach to addressing this problem. Indeed, it provides a formal mathematical framework for measuring and enforcing the privacy guarantees provided by statistical computations. Consider a randomized data analysis over a dataset x resulting in an output y. Informally, differential privacy requires that if we change any one individual's data in x, then the distribution of y does not change much. So, each individual's data is hidden from an adversary that views the output y. Differential privacy gives a quantitative way to measure and control the changes in y. The level of privacy protection that differential privacy can offer is described in terms of two privacy parameters ϵ and δ; the smaller they are, the closer the distributions for two databases that differ in one individual are, and hence the greater the level of privacy. To achieve this greater level of privacy protection, a differentially private algorithm will generally inject a greater amount of random "noise" into a statistical computation, thereby yielding less "accurate" results.

Using differential privacy enables us to provide wide access to statistical information about a dataset without worries of individual-level information being leaked inadvertently or due to an adversarial attack. There is now both a rich theoretical literature on differential privacy and numerous efforts to bring differential privacy closer to practice (discussed in detail in Section 4). However, none of the past work simultaneously meets all of our desiderata for such a system:

---

[1] Even traditional de-identification techniques have been found to have a significant negative impact on utility [17].

- Accessibility by non-experts: researchers in the social sciences should be able to use the system to share and explore data with no involvement from experts in data privacy, computer science, or statistics.

- Generality: the system should be applicable and effective on a wide variety of heterogeneous datasets, as opposed to being tailored for a particular data source or domain.

- Workflow-compatibility: the system should fit naturally in the workflow of its users (e.g. researchers in the social sciences), and be positioned to offer clear benefits (e.g. more access to sensitive data or less risk of an embarrassing privacy violation) rather than being an impediment.

Our Contribution: PSI

In this paper, we provide an overview of PSI ("a Private data Sharing Interface"), a system we are developing to enable researchers in the social sciences and other fields to share and explore privacy-sensitive datasets with the strong privacy protections of differential privacy. It is designed to achieve all of the desiderata mentioned above (Accessibility for Non-Experts, Workflow-compatibility, and Generality). Unique features of PSI include:

- None of its users, including the *data depositors* who have privacy-sensitive data sets they wish to share and the *data analysts* who seek to analyze those datasets, are expected to have expertise in privacy, computer science, or statistics. Nevertheless, PSI enables them to make informed decisions about the appropriate use of differential privacy, the setting of privacy parameters, the partitioning of a privacy budget across different statistics, and the interpretation of errors introduced for privacy.

- It is designed to be integrated with existing and widely used data repository infrastructures, such as the Dataverse project [15, 38], as part of a broader collection of mechanisms for the handling of privacy-sensitive data, including an approval process for accessing raw data (e.g. through IRB review), access control, and secure storage. Consequently, PSI can initially be used to *increase* the accessibility of privacy-sensitive data, augmenting rather than replacing current means for accessing such data, thereby lowering the adoption barrier for differential privacy.

- Its initial set of differentially private algorithms were chosen to include statistics that have wide use in the social sciences, and are integrated with existing statistical software designed for lay social science researchers, namely the Zelig [13] package in R and the TwoRavens [34] graphical data exploration interface.

Outline. Before formally introducing differential privacy in Section 3, we provide a motivating story in Section 2 to illustrate one of the scenarios in which PSI may be useful. In Section 5, we discuss more generally the incentives for the use of differential privacy and PSI in the sharing of research data. Section 6 describes the different kinds of actors that can use PSI and its expected workflow. Section 8 describes the privacy budgeting interface and Section 9 outlines the different statistics a data analyst can release through the system. Section 10 describes the concrete architecture of PSI and its implementation while Section 12 discusses security considerations for the system. In Section 13, we describe the results of some experiments we have performed using PSI. Section 4 describes related works and Section 14 contains some final remarks.

A preliminary prototype of PSI is available at `http://privacytools.seas.harvard.edu/psi`. It does not yet incorporate all of the planned features described in this paper, as a number of them are still under development. The purpose of this paper is to describe the *design* of PSI, and initiate a discussion about the choices made and possible alternatives.

2 A Motivating Story

Consider a social scientist who is studying the relationship of health status to political participation. Since the rules surrounding health insurance have been a recent focus of political debate, the researcher is interested to see if individuals with recent health concerns have become more engaged in the political process. She hypothesizes that out of self interest, individuals with major illnesses may be more likely to vote, volunteer time or contribute money to campaigns, because access to health insurance, and rules surrounding prior medical conditions are a major topic of recent law and current political campaigns. However, the researcher recognizes several competing alternate hypotheses. Persons coping with illness may have less available time, and thus instead become *less* engaged in political acts. Perhaps family members whose children or relatives are ill would be those to become engaged. Or perhaps there is some interactive effect where these effects are only present in low income households, since low income voters are more likely be effected by the new markets and subsidies for health insurance, or only persons who live in states that accepted federally subsidized health insurance exchanges.

She searches the catalog of the tens of thousands of datasets archived in a Dataverse data repository and finds some handful that have some promise of containing information to test her hypothesis. Some are broad surveys of attitudes and behavior that contain hundreds of questions across very many domains,[2] so may contain only a couple questions on voting turnout or a couple questions on health status. She might also find more focused studies that survey patient populations, with richer questions about their medical issues and questions to judge the impacts on their lives and opinions, or longitudinal studies that revisit these participants repeatedly over a long time scale. She might also find detailed time diary studies, where respondents agree to provide extensive recordings of how they spend their time each day, spaced with periodic surveys, that might even include biological surveys of cortisol and other hormone levels from saliva.

The broad surveys are generally publicly available. Even here, however, geographic variables such the state of residence, are only available in a special version that is closed to the public.[3] The focused studies may or may not be publicly available depending on the original data depositor's wishes, agreements with the depositor's Institutional Review Board (IRB), and whether the data touches on vulnerable protected populations (such as children, felons, minorities, the disabled) or data for which there are federal regulations (such as health care and student records). The time diary studies will almost certainly be closed to the public, because of their rich description of individuals and potential for reidentification.

In summary, as the datasets become richer, and more likely to directly test between her hypotheses, they are more likely to be closed to public access. So our researcher can use available public data and test the broadest implications of her theory with somewhat crude proxy variables for the items of interest. Or she can apply to gain access to sensitive data that can allow her to

---

[2] For example the General Social Survey, National Election Study, or Cooperative Congressional Election Study.

[3] For example, to get the state of residence of respondents, the General Social Survey requires a signed contract with the researcher, a thorough description of the research to be conducted on the data, a fee of $750, approval from the researcher's IRB, and construction of a data protection plan that generally requires non-networked computers set up in a room with secured limited access [8].

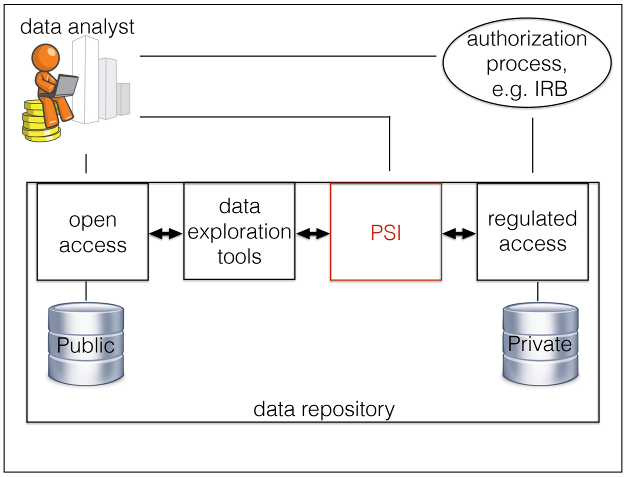


Figure 1: Overview of a data analysis repository using PSI.

directly test all of her hypotheses in nuance. Each application delays the research project, is costly in terms of researcher time and IRB resources, and commonly requires participation or approval by the original data depositor and relevant IRBs, all of which in turn may need to be facilitated by communication through the repository staff curators. However, even if a researcher is committed to applying for access to private data, the codebook and other public descriptions can be insufficient to judge which of the potential closed datasets contain the best or most relevant data for the researcher's purposes. Many of these IRB applications may turn out to be costly lost efforts when once made available the researcher finds there are insufficient observations of the type required, or the variables are not coded in the manner expected, or the time period is wrong or for other reasons the desired hypothesis can not be tested on the archived data. Our PSI system is built to allow more immediate exploratory access to these closed files and to reduce the wasted effort of researchers, data depositors, repositories and IRBs coming from applications to access to datasets that eventually prove to not be useful to the applicant.

The situation we have described above is schematically represented in Figure 1. When a data analyst has access to a data repository infrastructure like Dataverse, they currently have only two options: either using the publicly available data which are offered with open access, or going through an authorization process that may be lengthy and costly. With PSI there will be a valuable alternative: accessing the sensitive data for data exploration. To enhance this opportunity, PSI is designed to be naturally integrated with both the data repository infrastructure and data explorations tools.

3 Differential Privacy

Differential privacy is a formal mathematical framework for measuring the privacy guarantees provided by statistical computations. Consider an algorithm M that takes a dataset x as input and performs a randomized computation to produce an output y. Informally, differential privacy requires that if we change any one individual's data in x, then the distribution of y does not change much. Intuitively, this means that each individual's data is hidden from an adversary that views the output y.

To make this intuition precise, we need to define what we mean by "one individual's data," and provide a measure of how much the distribution of y is allowed to change. For the former, a typical choice is to consider datasets x that consist of n records, where we think of each record as consisting of one individual's data, and the sample size n is public (not sensitive information). We call two datasets x and x' *neighbors* if they agree in all but one record (i.e. x' is obtained from x by changing one individual's data). Then the formal definition of differential privacy is as follows:

Definition 3.1 (Differential Privacy, [23, 22]) *For parameters $\epsilon \geq 0$ and $\delta \in [0,1]$, a randomized algorithm M is (ϵ, δ)-*differentially private *if for every two neighboring datasets x, x' and every set S of outputs,*

$$\Pr[M(x) \in S] \leq e^{\epsilon} \cdot \Pr[M(x') \in S] + \delta,$$

where the probabilities are taken over the randomization of the algorithm M.

The level of privacy protection is governed by the two *privacy parameters* ϵ and δ; the smaller they are, the closer the distributions of $M(x)$ and $M(x')$ are, and hence the greater the level of privacy. Typically, ϵ is taken to be a small constant such as 0.1, whereas δ is taken to be very small, like 2^{-30}.

The way that differentially private algorithms for statistical analysis are often designed is by carefully introducing a small amount of random noise into non-private algorithms for the same analyses. The more noise that is introduced, the greater the level of privacy protection (i.e. a smaller ϵ and/or δ). However, less noise produces a more accurate and useful analysis. Thus differentially private algorithms offer a privacy-utility tradeoff.

By now, there is a large literature giving differentially private algorithms for a wide variety of data analysis tasks. Often, these algorithms are accompanied by a theoretical analysis showing that their performance converges to that of the non-private algorithm as the sample size n tends to infinity. However, such asymptotic performance guarantees do not necessarily translate to good performance at a specific finite sample size, and thus a great deal of work remains to be done to engineer differentially private algorithms to be useful in practice.

In addition, one typically does not want to run just one analysis on a dataset, but rather a large collection of analyses. Fortunately, differential privacy satisfies a variety of *composition theorems* showing that the privacy protection degrades gracefully when we run many differentially private algorithms. For example:

Lemma 3.2 (Basic Composition [23, 22]) *Let $M_1, \ldots, M_k$ be randomized algorithms where M_i is (ϵ_i, δ_i) differentially private for $i = 1, \ldots, k$. Then the algorithm $M(x) = (M_1(x), \ldots, M_k(x))$ that runs each of the M_i's using independent coin tosses is $(\sum_i \epsilon_i, \sum_i \delta_i)$ differentially private.*

Since the δ parameter in differential privacy is generally taken to be negligibly small, the effect of this summing is generally insignificant, so we will focus on the ϵ parameter. If we want to achieve a global, overall level of privacy protection given by ϵ_g, we can think of ϵ_g as a "privacy budget" to be spent on different analyses M_i we want to run. We can spend more of this budget on a specific analysis M_i (i.e. make ϵ_i larger), but this will consume more of our budget, leaving less for the other analysis if we want to ensure that $\sum_i \epsilon_i \leq \epsilon_g$.

There are better bounds on the composition of differentially private algorithms than the simple summing bound given above [24, 36, 49], but they still have the same budget-like effect—a larger ϵ_i (i.e. higher accuracy, lower privacy) for one computation requires reducing the ϵ_j's for other computations in order to maintain the same overall level of privacy.

4 Related work

Most of the previous work to bring differential privacy to practice can be partitioned into the following categories:

- *Programming languages and systems:* here the goal is to make it easier for users to write programs that are guaranteed to be differentially private, either by composition of differentially private building blocks [43, 55, 31], using general frameworks such as "partition-and-aggregate" or "subsample-and-aggregate" [51] to convert non-private programs into differentially private ones [59, 47], or by formal verification from scratch [9]. On one hand, these methods provide much more generality than we seek—our target users are not programmers, and it will already be very useful to provide them with a small, fixed collection of differentially private versions of statistical computations that are common in the social sciences. On the other hand, most of these tools do not provide much guidance for a lay user in deciding how to partition a limited privacy budget among many statistics or analyses he or she may want to run, or how to interpret the noisy results given by a differentially private algorithm.

  In contrast to the other tools mentioned above, GUPT [47] does enable a user to specify fine-grained accuracy goals and automatically converts these into privacy budget allocations, in a similar spirit to our privacy budgeting tool (described in Section 8). However, GUPT is limited to differentially private programs obtained via the subsample-and-aggregate framework, whereas our tool has no such restriction, and can be extended to include arbitrary differentially private algorithms. Moreover, our tool allows the privacy budget allocation to be interactively adjusted by users, and supports optimal composition theorems for differential privacy [49].

- *Optimization for specific data releases:* there have been several successful applications of differential privacy to very specific and structured sources of data like commuter patterns [40], mobility data [45], client-side software data [27], and genome-wide association studies [14]. Here differential privacy experts carefully optimize the choice of differentially private algorithms and the partitioning of the privacy budget to maximize utility for the particular data source. In the context of a broad data repository in the social or health sciences, the collection of data sources and the structure of the datasets is too heterogenous to allow for such optimization. And it is not scalable to have a differential privacy expert manually involved in each instance of data sharing.

- *Optimization and evaluation of specific algorithms:* there is a vast literature on the design of differentially private algorithms for specific data analysis tasks, including substantial experimental work on comparing and optimizing such algorithms across a wide range of datasets. As an example, the recent work on DPBench [32] provides a thorough comparison of different algorithms and different ways of optimizing them. Such work is complementary to ours. Algorithms that perform well in such evaluation are natural candidates to add to our library of differentially private routines, but such evaluation does not address how to budget the privacy allocated to this one algorithm against many other analyses one might want to run on the same dataset or more generally how to enable lay users to make appropriate use of differential privacy. Moreover, our use case of a general-purpose social science data repository guides the choices of which algorithms to implement, the measures of accuracy, and the methods for evaluation, as discussed in later sections.

There are also a number of deployed systems that provide query access to sensitive data, using heuristic approaches to protect privacy. These include systems for querying clinical health data [39, 41], education data [4], genomic data [63], and Census data [1]. However, the lack of rigorous privacy guarantees raises a genuine risk, as illustrated by attacks on the Israeli Census query system [69], on genomic data [33, 26] and more generally on releases of aggregate statistics [18, 25]. (Some of the aforementioned systems address this concern by limiting access to a more trusted set of users.)

5 Incentives for use

Differential privacy has sometimes been critiqued for its cost in utility (coming from the noise introduced in statistics), thus one might wonder what would motivate researchers to use it in place of the current data-sharing ecosystem. We see at least three different scenarios in which differential privacy can provide a clear benefit over current approaches.

- ("DP works great") In some circumstances, the results of differentially private analyses are virtually indistinguishable from non-private analyses. Currently, this tends to be the case when the number n of samples is large, the data is low-dimensional, and the analyses to be performed are relatively simple and few in number. In such cases, the greater privacy protections of differential privacy come essentially for free. As both theoretical and applied work on differential privacy advances and data gets "bigger" (n gets larger), we can expect an increasingly large set of data-sharing circumstances to fall in this scenario.

- ("Access is wide") When we wish to make sensitive data available to an extremely wide community (for example, when allowing public access), we should be increasingly concerned about attacks from individuals with malicious intent. Such adversaries can include ones who have extensive knowledge about a particular data subject that can be exploited as background information. Thus, the strong protections of differential privacy, which remain meaningful regardless of an adversary's background information, are attractive.

- ("Data is currently unavailable") For data that is currently unavailable except possibly through restrictive and time-consuming provisions, *any* useful statistical information that differential privacy can offer is a benefit to utility, even if it does not fall in the "DP works great" category. In particular, differential privacy can offer the possibility of rough, exploratory analysis to determine whether a dataset is of sufficient interest to go through the process of applying for access to the raw data.

The architecture of PSI is designed to support all three of these scenarios.

In the near term, we expect the third scenario, namely enabling exploratory analysis of data that is currently unavailable, to be the one where PSI is most frequently used. In this scenario, PSI can provide a clear utility benefit, can be applied with the modest sample sizes that are common in social science, and does not require an extensive library of highly optimized and sophisticated differentially private algorithms. However, PSI is extensible to incorporate such a library in the future, and we hope that eventually it will be used more often in the other two scenarios as well, providing high-utility and privacy-protective access to data that is currently shared in a less safe manner [16].

In the future, another potential incentive for the use of a differentially private data analysis system like PSI is the automatic protection that differential privacy provides against false discovery, allowing analysts to perform adaptive data exploration (without "preregistration") and still have confidence that the conclusions they draw are statistically valid [21, 10, 57].

We note that sometimes researchers do not wish to share their data, and are only using privacy as an excuse. A system like PSI can help eliminate the excuse. Still, other external incentives may be needed (such as from the research community, funding agencies, or journals) to encourage sharing of data.

On exploratory analysis

Since it is our initial goal for the use of PSI, we elaborate on what we mean by supporting "exploratory data analysis." This term generally refers to a wide-ranging set of techniques to empirically learn features of data by inspection, and familiarize oneself with the nature of the data, or discover apparent structure in the data [64]. It is inspection and discovery not driven by theory or modeling. In our setting of a social science data repository, we envision at least two uses for exploratory analysis. For lay-users (e.g. members of the general public), exploratory analysis can be a way to satisfy curiosity and discover interesting facts for situations where a statistically rigorous analysis may not be necessary (e.g. for a high-school project). For a social science researcher, the goal of exploratory analysis can be to determine which of the many datasets in the repository are of most interest, so that the researchers only invest their time and effort in applying for raw access to those datasets. Any final analyses they wish to perform and publish could then be done on the raw data, not through the differentially private interface. This more modest near-term goal for PSI compensates for the fact that we cannot perform the kinds of optimizations that might be done if we had a differential privacy expert involved in each instance of data sharing.

6 Actors and Workflow

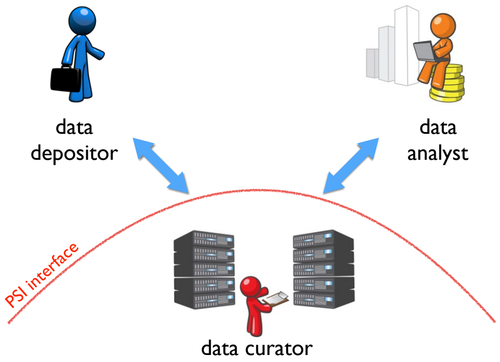


Figure 2: PSI Actors and Workflow

We have three different kinds of actors in PSI: data depositors, data curators, and data analysts. Each of them has a different role and different requirements. We represent them and their interaction schematically in Figure 2. The threat model we consider, in terms of trust for different actors, is described in Table 1. We now detail the roles and the expected expertise for each of them.

Data depositors. These are users that come to deposit their privacy-sensitive dataset in a data repository, and may wish to make differentially private access to their dataset available. Based on the interaction with the data depositor, the system will gather basic information about the dataset (e.g. the types and ranges of the variables), set the overall privacy parameters, select an initial set of differentially private statistics to calculate and release, and determine how the remaining privacy budget will be partitioned among future data analysts.

Data depositors are the ones with the initial ethical and/or legal responsibility for protecting the privacy of their data subjects, and they (or their institutions) may be liable if they willfully violate their obligations. Thus, they can be trusted to follow instructions (if not onerous or confusing) and answer questions truthfully to the best of their knowledge. On the other hand, they cannot be assumed to have expertise in differential privacy, computer science, or statistics, so any questions that involve these areas need to be explained carefully.

Data curators. These are the data-repository managers that maintain the hardware and software on which PSI runs and the accompanying data repository infrastructure (e.g. Dataverse) and associated statistical tools (e.g. Zelig and TwoRavens). They are trusted, and indeed may also have legal obligations to ensure that the repository does not violate the privacy protections it claims to offer through tools such as PSI. Data curators can be assumed to have expertise in IT systems administration and data stewardship [30] and archiving [6], and can be trained to have at least a modest background in statistics and differential privacy. But they are few in number, and cannot be actively involved in most instances of data sharing or data exploration. Thus PSI needs to be sufficiently automated to enable data depositors and data analysts to safely use it on their own.

Data curators would also be responsible for deciding whether to accept new differentially private routines into the library used by PSI and correcting bugs or security flaws found in existing routines. These can be difficult tasks even for experts in differential privacy. Thus, in a future version of the system, it would be of interest to minimize the amount of trusted code, and have tools to formally verify the remaining components (both original components and later contributions), along the lines of the programming languages tools described in Section 4.

Data analysts. These are users that come to access sensitive datasets in the repository, often with the goal of data exploration as discussed in Section 5. They will have access to all of the differentially private statistics selected by the data depositor, as well as the ability to make their own differentially private queries (subject to staying within the overall privacy budget, as discussed more below).

We envision at least two tiers of trust for data analysts. PSI can make access available to a very wide community of analysts (e.g. the general public), in which case the analysts are considered completely *untrusted*. Alternatively (or in addition), we can restrict to a set of analysts that are identifiable (e.g. as registered users of the data repository), with some accountability (e.g. through their verified affiliation with a home institution). Such analysts may be considered as *semi-trusted*, as we can assume that most of them will follow basic terms of use to not abuse the system in certain ways. Specifically, we will assume that semi-trusted users will not collude to compromise privacy, and will not create phony accounts. (This will enable us to provide greater utility for such users, as discussed in Section 8.)

Actors	Level of trust	DP expertise
data curators	trusted	modest
data depositors	trusted	none
data analysts (restricted)	semi-trusted	none
data analysts (general public)	untrusted	none

Table 1: Actors and their level of trust and required expertise.

7 Pedagogical Materials

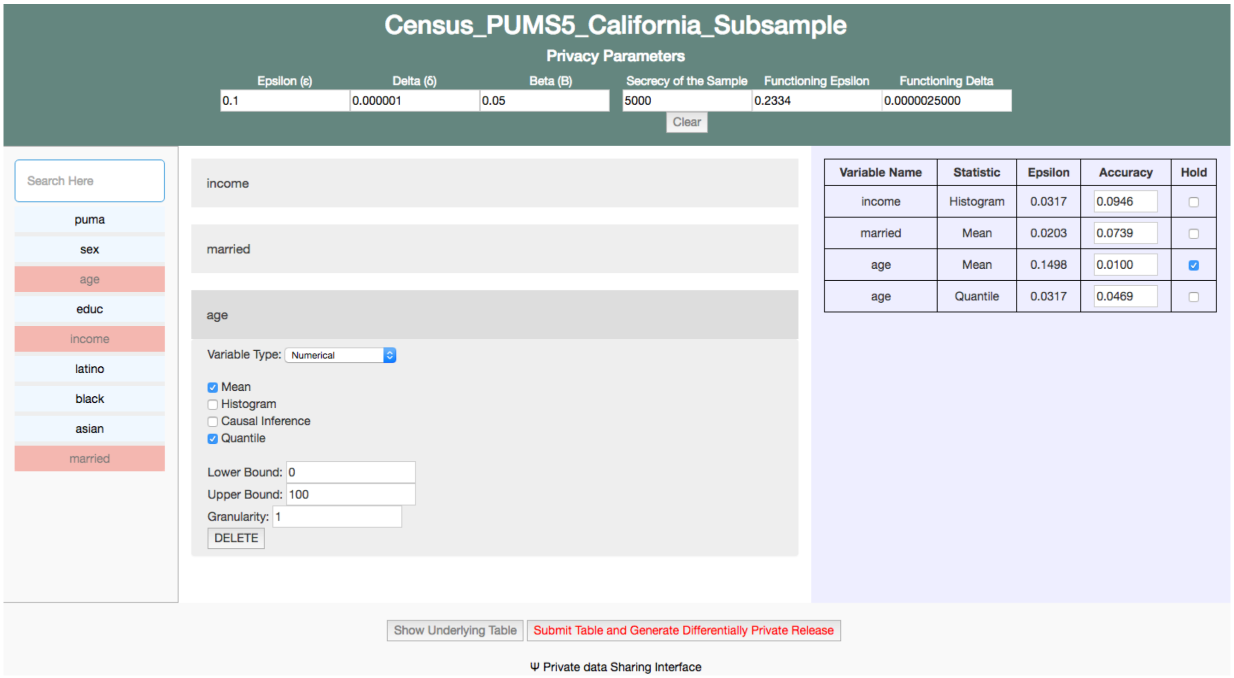
In order to enable PSI to be used by empirical researchers without expertise in privacy, computer science, or statistics, we have prepared pedagogical materials explaining differential privacy in an intuitive but accurate manner, with a minimum of technical terminology and notation. These materials are meant to be sufficient for data depositors and data analysts to understand and make appropriate choices in using PSI, such as those described in the forthcoming sections. Data depositors require more background material than data analysts, as the former are concerned with the privacy protections afforded to their data subjects, whereas the latter only need to understand the impact of the system on their analyses (namely, that results will be less accurate or statistically significant than would be obtained on the raw data, and that there is a limited "budget" of queries that they can perform).

Relevant extracts of the pedagogical materials will be offered to users of PSI at each decision point, and can also be included when describing data-sharing plans to Institutional Review Boards (IRBs). In addition, members of our team have started to develop rigorous arguments showing that differential privacy should be deemed to satisfy certain legal obligations of privacy protection, which can also be used to reassure data depositors, data curators, and IRBs that differential privacy is a sufficiently strong form of protection. For example, the combined legal and technical analysis in [50] provides an argument that, when applied to educational data, differentially private computations satisfy the requirements of the Family Educational Rights and Privacy Act of 1974 [2].

As discussed in Section 6, we assume that data curators have expertise in IT systems administration and data stewardship, and at least a modest background in statistics and differential privacy. Thus, they do not need any specialized pedagogical materials other than a thorough documentation of the system.

8 Privacy Budget Management

One of the challenges in enabling non-experts to use differential privacy is that it can be difficult to understand the implications of different selections of the privacy parameters (namely ε and δ), both in terms of privacy and utility, especially when these need to be distributed over many different statistics to be computed. To address this issue, PSI is designed to expose these implications to the user, in easy-to-understand terms, and is accompanied by a variety of simple explanations of differential privacy and its parameters that are shown to the user at relevant times. We have developed a privacy budgeting tool that guides users through judicious choices of global privacy parameters, lets users select statistics to release, automatically distributes the privacy budget across the chosen statistics, and exposes the resulting privacy-accuracy tradeoffs (see Figure 3.)



Census_PUMS5_California_Subsample

Privacy Parameters

Epsilon (ε)	Delta (δ)	Beta (B)	Secrecy of the Sample	Functioning Epsilon	Functioning Delta
0.1	0.000001	0.05	5000	0.2334	0.0000025000

age

Variable Type: Numerical

☑ Mean
☐ Histogram
☐ Causal Inference
☑ Quantile

Lower Bound: 0
Upper Bound: 100
Granularity: 1

Variable Name	Statistic	Epsilon	Accuracy	Hold
income	Histogram	0.0317	0.0946	☐
married	Mean	0.0203	0.0739	☐
age	Mean	0.1498	0.0100	☑
age	Quantile	0.0317	0.0469	☐

Figure 3: PSI privacy budgeting interface.

Global privacy parameters

The data depositor, who carries the initial responsibility for protecting the privacy of her data subjects, is charged with setting the overall ("global") privacy parameters ϵ_g, δ_g for her dataset (seen at the top left of Figure 3). To enable this choice, we provide intuitive (but accurate!) explanations of the meaning of each of these privacy parameters, and give recommended settings based on the level of sensitivity of a dataset (e.g. corresponding to an institution's established research data security levels, such as [3] or the similar categories in the DataTags system that integrates with PSI [62]). δ_g is easily explained as the probability of arbitrary leakage of information, like the probability of an adversary breaking an encryption scheme, and thus should be set to be extremely small, like 2^{-30}. For the main privacy parameter, ϵ_g, we explain it with a table comparing an adversary's posterior belief that a data subject has a sensitive trait to the posterior belief had the subject opted out of the study. PSI also confirms with the data depositor that each individual subject's data corresponds to one row of the uploaded dataset (so that the per-row protections of differential privacy translate to per-subject protections).

Secrecy of the sample

The data depositor is asked whether the dataset is a random sample from a larger population, and whether the choice of this sample has been kept confidential. If so, a useful lemma in differential privacy known as "secrecy of the sample" allows for an effective savings in the privacy parameters corresponding to the ratio of sizes between the dataset and the larger population. This means that correspondingly greater utility can be provided for the same level of privacy protection. (To account for the fact that, in practice, population samples are typically not perfectly random, the depositor is instructed to conservatively estimate the overall population size.)

Lemma 8.1 (Secrecy of the sample [37, 61]) *Let M be an (ϵ, δ)-differentially private algorithm for datasets of size n. Let M' be a randomized algorithm that takes as input a dataset D of size $m \geq n$, and then runs M on a dataset D' obtained by selecting a uniformly random subset of D's records of size n. Then, M' is $((e^{\epsilon} - 1) \cdot (n/m), \delta \cdot (n/m))$-differentially private.*

In the application of this lemma in PSI, D' represents a dataset that is being deposited in the repository, D represents a larger population from which D' was (randomly) drawn, and M represents the differentially private statistics computed by PSI on D'. Note that in typical applications of differential privacy, ϵ is a small constant and therefore $(e^{\epsilon} - 1) \cdot n/m \approx \epsilon \cdot n/m$. In concrete applications, especially in the social sciences, this lemma permits large savings in the privacy budget. For this reason, we integrate this property in the budgeting interface (See Figure 3).

Budgeting among different statistics

Once global privacy parameters have been determined, users can select variables from their dataset (from the left-hand panel of the budgeting interface, Figure 3) and choose statistics to release about those variables from PSI's library of differentially private algorithms. At this stage, there is still the challenge of how the global privacy parameters should be distributed among the different statistics to be computed. That is, for each statistic to be computed, we need to select privacy parameters (i.e. set ϵ_i and δ_i for statistic i) and then apply composition theorems to ensure that globally, we achieve (ϵ_g, δ_g) differential privacy.

This leaves the question of how a user should select individual privacy parameters ϵ_i (and δ_i). The larger the value of ϵ_i is taken, the more utility we obtain from the i'th statistic, but this leaves less of the global privacy budget remaining for the other statistics. Since some statistics a user is computing may be more important than others, and different differentially private algorithms have different privacy-utility tradeoffs, the "best" use of the privacy budget is likely to involve a non-uniform distribution of the ϵ_i's.

To enable users to determine this partition without requiring that they be privacy experts, PSI automatically assigns initial privacy parameters to each chosen statistic. Similarly to GUPT [47], PSI then exposes the privacy-accuracy tradeoffs to the user (see the summary table in the right-hand panel of Figure 3.) Rather than adjusting the individual privacy parameters ϵ_i, the user can instead modify the "accuracy" that will be obtained for different selected statistics (presented as, for example, the size of 95% confidence intervals; see further discussion in the next section). For each differentially private algorithm in PSI, there are accompanying functions that translate between the privacy parameters and a measure of accuracy (also depending on other metadata, such as the range of variables involved and the dataset size n). These functions are used by the privacy budgeting tool to translate the accuracy bounds into individual privacy parameters and ensure that the global privacy parameters are not exceeded.

Optimal composition

To ensure that we get the most utility out of the global privacy budget, we use the Optimal Composition Theorem of [49], which in fact was developed for the purpose of our privacy budget tool. This theorem characterizes the optimal value for the global privacy budget ϵ_g (for each possible $\delta_g \in [0,1)$) when composing k algorithms that are (ϵ_i, δ_i) differentially private.

Theorem 8.2 (Optimal Composition Theorem, [49]) *Let $M_1, \ldots, M_k$ be randomized algorithms where M_i is (ϵ_i, δ_i) differentially private for $i = 1, \ldots, k$ and let $\delta_g \in [0,1)$. Then the algorithm $M(x) = (M_1(x), \ldots, M_k(x))$ that runs each of the M_i's using independent coin tosses is (ϵ_g, δ_g) differentially private for the least value of ϵ_g satisfying the following inequality:*

$$\frac{1}{\prod_{i=1}^{k}(1+e^{\epsilon_i})} \cdot \sum_{S \subseteq \{1,\ldots,k\}} \max\left\{ e^{\sum_{i \in S} \epsilon_i} - e^{\epsilon_g} \cdot e^{\sum_{i \notin S} \epsilon_i}, 0 \right\} \leq 1 - \frac{1-\delta_g}{\prod_{i=1}^{k}(1-\delta_i)}$$

While the Basic Composition Theorem gives an upper bound on the degradation of privacy under composition, the above theorem is optimal in the sense that for every set of privacy parameters, (ϵ_i, δ_i) for $i \in \{1, \ldots, k\}$ and δ_g there exists a set of algorithms $M_1, \ldots, M_k$ that are (ϵ_i, δ_i) differentially private, respectively, whose composition achieves (ϵ_g, δ_g) differential privacy *exactly*.

For even moderate values of k, the optimal composition theorem can provide substantial savings in the privacy budget over the other composition theorems in differential privacy. In an effort to maximize utility for users, the budgeting interface uses an implementation of Theorem 8.2 to apportion a global epsilon value across several statistics. Since the Optimal Composition Theorem is infeasible to compute in general, we use an efficient approximation algorithm that still outperforms the alternative composition theorems [49].

Budgeting among different actors.

Recall that the selection of differentially private statistics to be computed is done both by the data depositor, who selects an initial set of statistics that will be shared among all analysts that access the dataset, and by individual data analysts, who may be carrying out novel explorations of their own conception. The privacy budgeting tool described above is designed to support both types of actors (with slightly different settings for each to reflect their different roles and level of trustworthiness). The data depositor is tasked with deciding how much of the global privacy budget ϵ_g to reserve for future data analysts. For example, if the data depositor uses up ϵ_d units of privacy for the statistics she chooses to release, then at least $\epsilon_a = \epsilon_g - \epsilon_d$ units of privacy will be left for the future analysts. (ϵ_a might actually be larger, since composition theorems for differential privacy can in some cases give better bounds than simply summing the privacy parameters.)

In the case of semi-trusted data analysts (who we assume will not collude, as discussed in Section 6), PSI provides *each* analyst a *per-user* privacy budget of ϵ_a. In the case of completely untrusted analysts, we share ϵ_a among all future analysts. This model is more conservative with respect to privacy protection, and thus may be appropriate when analysts do not have the sufficient accountability or the data is highly sensitive (e.g. with life-or-death or criminal implications). The downside of the more conservative model is that it is vulnerable to a denial-of-service attack, where the first few data analysts, intentionally or inadvertently, deplete the entire privacy budget, leaving future analysts unable to make any queries. This can be partly mitigated by rate-limiting the use of the privacy budget and by sharing all statistics computed publicly. It is also possible to reserve part of the privacy budget for untrusted analysts and part for trusted analysts, with each part being treated as described above.

Budgeting for Interactive and Adaptive Queries

An additional subtlety in privacy budgeting comes from the fact that data analysts may choose their privacy parameters (ϵ_i, δ_i) *adaptively*, depending on the results of previous queries. In such a case, it is natural to try to use composition theorems as *privacy filters* [58] — for example, the k'th query would be allowed only if its privacy parameters (ϵ_k, δ_k) do not cause the inequality of the Optimal Composition Theorem (Thm. 8.2) to be violated. Unfortunately, as shown in [58], this strategy does not in general yield (ϵ_g, δ_g) differential privacy overall. However, more restrictive bounds, such as Basic Composition (Thm. 3.2), do yield valid privacy filters. Consequently, once PSI incorporates interactive queries for data analysts (a feature still in development), the Optimal Composition Theorem (and its approximations) will only be used within (non-adaptive) *batches* of queries; to compose across different batches, we will use a valid privacy filter (such as Basic Composition).

9 Differentially Private Algorithms

9.1 Choice of Statistical Procedures

While PSI is designed to be easily extensible so as to incorporate new algorithms from the rapidly expanding literature, the initial set of differentially private algorithms in PSI were chosen to support the most necessary statistics that are needed to provide immediate utility for social science research and data exploration. Specifically, we include:

- Univariate descriptive statistics, such as means, quantiles, histograms, and approximate cumulative distribution functions. From some of these, post-processing can also provide additional descriptive statistics at no additional privacy cost.

- Basic statistical estimators, for inference about the population from which a dataset was sampled. We have selected some of the most widely used statistical inference procedures in social science, such as difference-of-means testing for causal inference, hypothesis tests for the independence of categorical variables, and low-dimensional covariance matrices from which can be extracted correlations, least-squares regressors, and principal components.

- Transformations for creating new features (variables) out of combinations of already existing ones. These allow the previously described procedures to be leveraged to do more sophisticated computations on a broader range of questions.[4]

These choices are also motivated in part by the data exploration tools that PSI integrates with, and which we expect our data analysts to use. In particular, the TwoRavens graphical data exploration tool (`http://2ra.vn`) provides descriptive statistics for each variable in a dataset, as well as graphical illustrations of its empirical distribution (e.g. a histogram or a probability density function) [20]. PSI replaces these with the differentially private descriptive statistics it computes. TwoRavens, together with the R package Zelig that it integrates with (`http://zeligproject.org`), also provides a unified, user-friendly interface for running a wide variety of statistical models. We

---

[4] For example, the (empirical) covariance between two attributes can be estimated by estimating the mean of a new attribute that is the product of the two original attributes (as well as the means of the original attributes), or the mean of a variable in a subpopulation can be computed from the mean of the product of that variable with a binary indicator for the subpopulation of interest, and the mean of the indicator.

have chosen to initially implement differentially private versions of statistical inference procedures that are widely used in social science and where the differentially private algorithms are sufficiently simple and well-understood to give good performance at finite sample sizes.

9.2 Measuring Accuracy

The choice of accuracy measure, and how to represent it to the users, is important both in the privacy budgeting tool as well as for data exploration by data analysts, who need to know how to interpret the noisy statistics provided by differential privacy.

For descriptive statistics, we have determined that 95% confidence intervals are the simplest and most intuitive way to represent the noise introduced by differential privacy. For many of the basic differentially private algorithms for descriptive statistics (such as the Laplace mechanism), a theoretical worst-case analysis is also indicative of typical performance, so we can use this to calculate the a priori privacy-accuracy translation needed in the privacy budgeting tool.

For statistical inference procedures, the accuracy (e.g. size of a confidence interval obtained) is necessarily data-dependent, even without privacy. (For example, using a t-test for mean estimation gives a confidence interval of size that depends on the empirical variance of the data.) When incorporating such methods, PSI uses *conservative* confidence intervals, meaning that it ensures that the differentially private confidence interval includes the true value with probability *at least* .95. Intuitively, we account for the noise introduced by differential privacy by making the confidence intervals larger — this ensures that analysts do not draw incorrect conclusions from the differentially private statistics (but more analyses may come out inconclusive, as we explain to users of the system). And to provide the *a priori* accuracy bounds needed by the privacy budgeting tool, we use “rules of thumb” based on experimental evaluation: given n, ϵ, the number of variables, etc.

10 Software Architecture

We have implemented a preliminary prototype of PSI. This prototype is not yet ready for use by the general public and it does not yet incorporate all of the planned features described in this paper. However, it already includes the key ideas we described. In this section we will describe the current implementation.

10.1 Metadata

Archival data for a research study are commonly stored on repositories as an original data file, and a complementary meta-data file. The original data file contains the raw numeric values of observations in the dataset.[5] The meta-data file contains auxiliary information about the dataset that increases its ability to be reused by researchers; this might include text descriptions of the variables, summary statistics, provenance [12] and numerical fingerprints for validation[6]. The largest repositories have shared standards for how this meta-data file should be constructed [66, 11], so that catalogs of data can be built across repositories [42, 56], and software utilities can be reused and deployed across different institutions [68].

---

[5] These are typically stored either in the file format the original researcher used, or converted to a standardized file format the repository works with. Many different statistical packages have their own specific data format, which may duplicate some of the separate meta-data we describe.

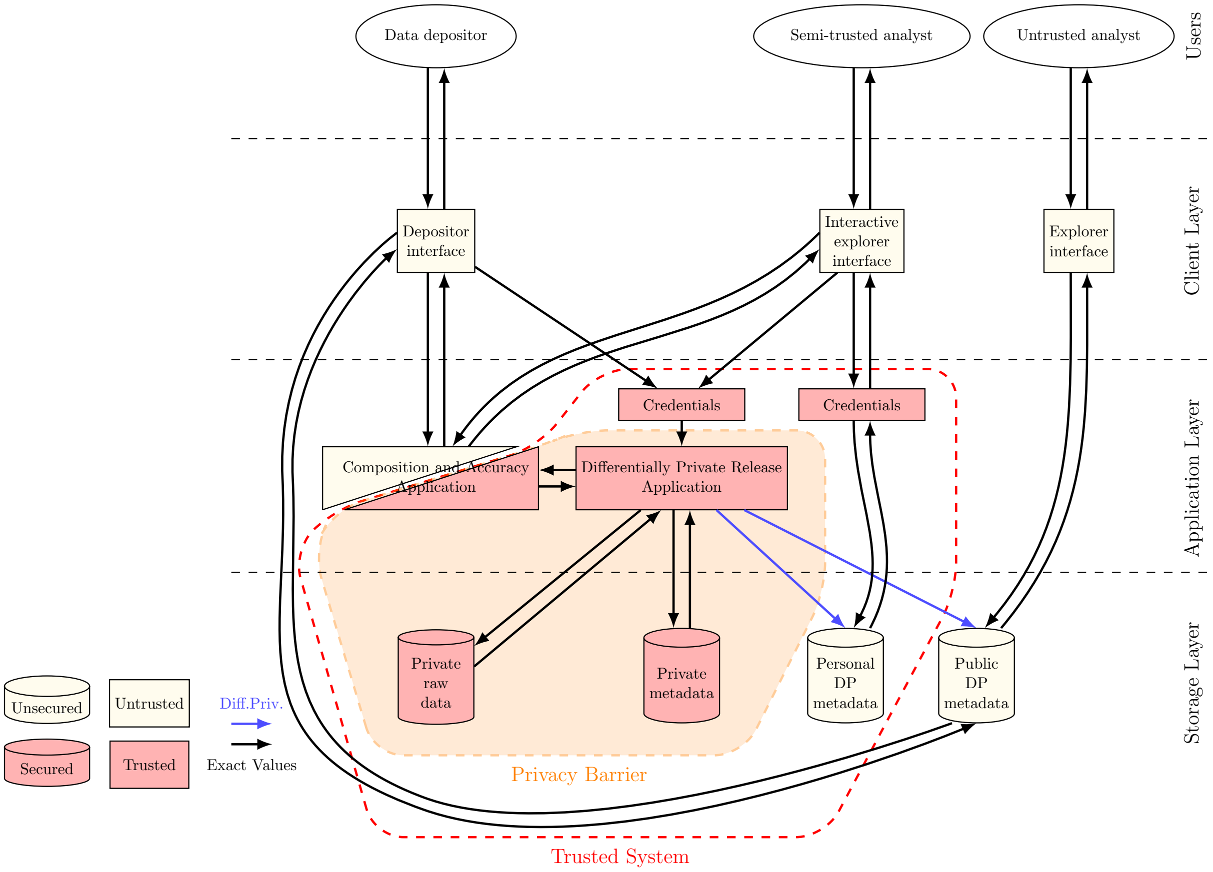


Figure 4: *Architecture diagram.*

Some of the information that gets recorded in the metadata we consider public, such as the names and text descriptions of the meanings of the variables and the sample size. Some of the metadata, such as variable-level summary statistics, contains private information, even if aggregated. Thus if the dataset contains private information, we consider its metadata to also be a private file that could potentially leak information. It is compliant with the shared standards, however, for metadata to have missing or empty fields, so we can construct a reduced version of the private metadata, that only contains public information. To this we can add differentially private versions of certain summary statistics, and still distribute the metadata file for public use, so long as the total privacy budget after composition (see section 8) of these statistics is below the appropriate global parameter. We call this the *public metadata*.

The bottom of Figure 4 shows the private raw data, its accompanying private metadata, and the public metadata, residing in a storage layer in our system. Surrounding them, are the application layer tools for differential privacy, which run on a remote server.[6] The differentially private algorithms, the accuracy estimates, and the budgeting coordinated by the composition theorem, each discussed in the previous sections, are all implemented in the R programming language, which is widely used in the statistics and quantitative social science communities [54]. We describe how they interlink below, as we trace out user's interaction with the system. We expect to distribute all of

---

[6] We run the R code on a remote Apache server, as an rApache application [35].

these routines as an R package for easy reuse within the R environment (independently of Dataverse and TwoRavens). In addition to this code on the server, there are client layer interfaces (written as thin HTML Javascript GUI's) that allow different types of users to interact with the system, but no direct access to the raw data. We now describe our different key users (the same as introduced in section 6), and how their respective interfaces interact with the larger system, in turn.

10.2 Depositor Interaction

At the time of budgeting the *depositor interface*, as for example in Figure 3, allows the *data depositor* to construct a list of statistics they would like to release on the dataset. This interface has no direct access to either the data or computations on the data; whenever the page requires a new computation,[7] it copies the contents of the current page[8] to a remote application that uses differential privacy composition theorems to re-partition the privacy budget among the current set of statistics (by scaling all of the ϵ_i's by the largest multiplicative factor that stays within the global privacy budget), and recalculates the corresponding accuracies. This remote process then recomputes and returns an updated list of privacy parameters and accuracy estimates associated with each selected statistic. The frontend interface then rewrites the summary table in the right-hand panel of Figure 3 with these newly provided values, and waits for more actions from the user until another round of computation is required. The backend composition process is memoryless, in the sense that no past version of the page persists or is stored, but every request of the backend begins an entirely new set of budgeting and accuracy computations. For this reason, the connection between the frontend and backend does not have to be persistent.

When the depositor has finalized the list of statistics she wishes to make available, together with their respective privacy parameters, a table containing the chosen statistics and their associated metadata and privacy parameters is then submitted to another separate remote release application that then computes all the differentially private statistics requested. This release tool checks the composition of the request with a trusted version of the composition application, which means that code to this point does not have to be trusted, so long as the global ϵ can be verified. This is detailed on Figure 4 as a split box, representing that there are two instances of the same code, one listening and replying to client requests, which does not have to be trusted, and another copy has to be trusted, but that only interacts with the backend, and has no web connection so is easier to protect. The release tool is the only process that has access to the raw data which sits in secure storage in the Dataverse repository. The application that calculates the differentially private releases does not reply to the depositor interface. The architecture diagram in Figure 4, shows the directions of communication between every piece of the system and one can trace out from this that any path from the raw data to any data analyst (or even the data depositor), has to pass through the differentially private channel from this application to the release of a differentially private value written to a metadata file.

---

[7] As when the metadata for a statistic is completed, or a statistic is deleted, or when an accuracy value, or any global parameter is edited.

[8] As a JSON file, by means of an HTTP POST to the composition application running remotely as an rApache process.

10.3 Analyst Interaction

The differentially private statistics that are generated are released in a file of metadata associated with the securely archived data. Everything in this metadata file can be made available for public browsing. In Figure 4, we show a *untrusted public analyst* who does not need to prove any credentials, able to access the public metadata file with the differentially private releases. The public analyst can use the public metadata file in whatever manner they prefer. However, since all this information is written in the repository metadata standards, a difficult to read XML file, we provide an *explorer interface* that presents the information in a more easily interpretable graphical form, using a modified version of the TwoRavens software [34], described in the next section. This is a statistical platform that allows users to explore data in repositories by means of their metadata, so is a good match for this application where only the metadata is available to the user.

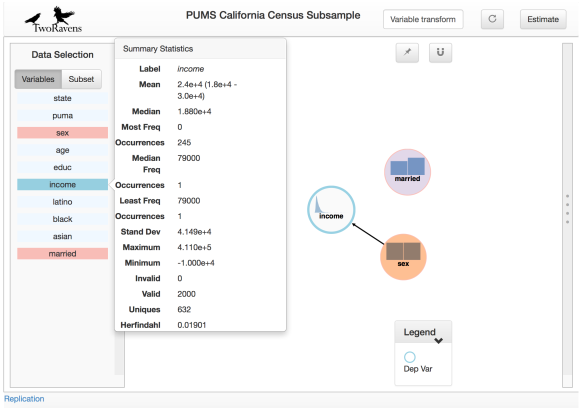
In addition, we provide another tier of access to *semi-trusted* users. These are users for which the depositor has granted a user-specific privacy budget ϵ_a from which they can generate additional differentially private releases, beyond those included in the public release. We expect these users will have some distinct university or research affiliation which can be verified by credentials and agree to terms of use.[9] Their explorer interface includes both the exploratory ability of the untrusted analyst interface, and the budgeting ability of the depositor interface. Again, these users can construct a list of statistics they would like to release, partitioning their personal ϵ_a budgets among them, by assistance of the composition application. When they have a batch of statistics whose accuracies they find useful, they are submitted to the differentially private release function which checks that the composition of statistics meets the available budget using a trusted copy of the composition application. This may occur in several adaptive rounds, as they learn things about the data that inform their further exploration, until their budget is exhausted. Each semi-trusted user has their own personal metadata file. This starts out with the same information in the public metadata, but each release adds the additional differentially private releases that have been paid from that user's personal ϵ_a budget. Only the semi-trusted user has access to this metadata file, by means of their credentials, and specifically in the terms of use we are trusting they will accord to, they have agreed not to share these values in collusion with other users (as discussed in the trust model in Section 6).

11 Exploration Interface

As described in the previous section, all released differentially private values are written to metadata files, either public files or files belonging to only one user. These files can be used by the permitted analyst in whatever manner they prefer, but we provide in our system a user friendly manner to read the information stored in the metadata. The TwoRavens platform for statistical inference (`http://2ra.vn`) is an interface that allows users, at all levels of statistical expertise, to browse data on repositories, explore summary statistics and build statistical models on those datasets by means of directed graphs [34, 20]. The interface is a browser-based, thin client, with the data remaining in an online repository, and the statistical modeling occurring on a remote server. The data remains in the repository and never goes to the browser, rather the statistical exploration is achieved by remote statistical processing and moving the correct metadata to the browser. This architecture works well with the PSI system since it relies solely on metadata, and we have been adapting

---

[9] For example, Dataverse verifies members of certain universities by *Shibboleth* [48] using the Security Assertion Markup Language (SAML) protocol.



Summary Statistics

Label	*income*
Mean	2.4e+4 (1.8e+4 - 3.0e+4)
Median	1.880e+4
Most Freq	0
Occurrences	245
Median Freq	79000
Occurrences	1
Least Freq	79000
Occurrences	1
Stand Dev	4.149e+4
Maximum	4.110e+5
Minimum	-1.000e+4
Invalid	0
Valid	2000
Uniques	632
Herfindahl	0.01901

Figure 5: *Explorer graphical user interface for inspecting differentially private released values, adapting TwoRavens platform.*

some of the graphs and summary tables available to convey to the user the additional uncertainty inherent in dealing with differentially private releases from noisy mechanisms, for example, providing confidence intervals for differentially private values, and histograms and density plots that represent the uncertainty in the graphed values due to noise. We have additionally been integrating our budgeting and transformation tools to be accessed by the appropriate semi-trusted users, through this interface.

12 Security

We feel that our highest priority is evaluating whether the design of PSI is useful for its potential user community. For this reason, we did not evaluate all the security vulnerabilities of the current prototype implementation and we plan to address them in a future version, before it is used to handle highly sensitive data.

Nevertheless, the PSI design addresses several of the security and side-channel issues that have been raised in the literature about implementations of differential privacy [31, 47]. We will discuss in this section our solutions.

12.1 Timing, state and privacy budget attacks

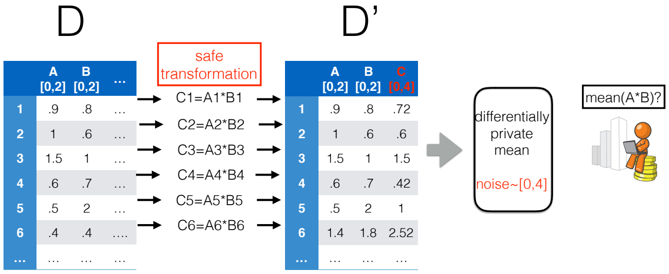
Haeberlen et al. [31] analyze the possible attacks to a differential privacy system working in a centralized scenario similar to the one we described in Section 6. In their scenario, data analysts are allowed to submit arbitrary analyses to the differential privacy system and the system is responsible for running these analyses if they pass some formal requirements guaranteeing differential privacy and if there is still some budget left. Even if these formal requirements guarantee differential privacy, this model is prone to three main kinds of side channel attacks:

Timing attacks The data analysis may leak information about an individual using a timing (or any other covert) channel.

State attacks The data analysis may leak information about an individual through an observable change in the application state, for instance by using a global variable.

Privacy budget attacks The data analysis may leak information about an individual by running a subanalysis that fails because of lack of privacy budget.

Most of these attacks can be implemented only if data analysts are allowed to submit arbitrary data analyses. In PSI a data analyst can only select built-in differentially private data analysis and run variable transformations before them. Using only built-in differentially private data analysis prevents all these attacks at data analysis time. For instance, there is no risk of a privacy budget attack since queries cannot run subanalyses that can exhaust the privacy budget. Nevertheless, data analysts can submit to PSI variable transformations that can create new features by combining existing ones as we discussed in Section 9, and these are at risk of timing attacks and state attacks.



	A [0,2]	B [0,2]	...
1	.9	.8	...
2	1	.6	...
3	1.5	1	...
4	.6	.7	...
5	.5	2	...
6	.4	.4	....
...	...	...	...

	A [0,2]	B [0,2]	C [0,4]
1	.9	.8	.72
2	1	.6	.6
3	1.5	1	1.5
4	.6	.7	.42
5	.5	2	1
6	1.4	1.8	2.52
...	...	...	...

Figure 6: *Workflow schema for safe variable transformations.*

12.2 Safe variable transformations

An important property of differential privacy is closure under post-processing.

Lemma 12.1 (Post-processing, [23]) *Let M be a (ϵ, δ)-differentially private randomized algorithm from*[10] *X^n to Y, and f be an arbitrary (possibly randomized) map from Y to Z. Then, the composition of M and f, denoted $f \circ M$ is a (ϵ, δ)-differentially private algorithm from X^n to Z.*

This property guarantees that the result of a differentially private data analysis can be released without further privacy concerns. In particular, the result of a differentially private computation can be safely given as input to any other data analyses.

The situation is more involved when variable transformations occur before applying a differentially private mechanism. Indeed, if we fix a differentially private algorithm M from X^n to Y and we arbitrarily pre-process its input dataset with an arbitrary map f from Z^n to X^n we can break its guarantee. As a simple example, consider a differentially private mechanism that approximately releases the fraction of the individuals with a particular feature B in a database with n records, and a map f that returns a database with n records with the feature B in the case John Doe is in the database, and that removes all the elements with feature B, otherwise. When n is sufficiently large, a data analyst observing the result of $M \circ f$ can determine with high probability whether John Doe is in the database or not. Fortunately, there is a class of important transformations that preserves differential privacy: per-row transformations.

---

[10] We use X^n to describe the possible set of databases with n records of type X.

Lemma 12.2 (Per-row transformations) *Let $M(x_1, \ldots, x_n)$ be a (ϵ, δ)-differentially private randomized algorithm*[11] *from X^n to Y, and f be a map from Z to X. Then, the composition of f and M, denoted $M(f(x_1), \ldots, f(x_n))$ is a (ϵ, δ)-differentially private randomized algorithm from Z^n to Y.*

This kind of transformation can be very useful in practice. For instance, as we mentioned before, the (empirical) covariance between two attributes can be estimated by estimating the mean of a new attribute that is the product of the two original attributes (as well as the means of the original attributes), or the mean of a variable in a subpopulation can be computed from the mean of the product of that variable with a binary indicator for the subpopulation of interest, and the mean of the indicator.

However, one must be careful in using this lemma. Indeed, if the input data is Z^n one must consider the possible change of attributes in X^n when reasoning about the differentially private algorithm M. Let see this with an example. Consider the case where we want a differentially private estimate of the mean of a new attribute C that is the product of the two original attributes A, B. A differentially private algorithm for computing the mean must choose noise that is proportional to the range of the attribute. Suppose that we know the range of A and B is $[a, b]$ for $a, b \geq 0$. When we choose the noise for M we need to reason about the range of C which is not $[a, b]$ but it is $[a^2, b^2]$ instead.

In order to allow only transformations that are *safe* in the sense discussed above, PSI requires the data curators and the data analysts to provide the ranges of each variable before and after the transformations and enforces them at runtime, i.e. the differentially private algorithms, truncate values that are outside the specified range. This guarantees the correct use of the principle formalized in Lemma 12.2 and so privacy is preserved. To support the design of transformations PSI uses a restricted domain-specific language and an automated program analysis tracking variable ranges. The workflow of variable transformations is described in Figure 6. Starting from the private dataset D, a variable transformation generates a new private dataset D', containing the same individuals as D but with potentially new variables, on which the differentially private algorithm is run. In this example, similar to the discussion above, the transformation creates a new attribute C as the product of A and B. This is performed per-row and the program analysis forwards the information about the range from the inputs (in this example the range for both A and B is $[0, 2]$) to the newly generated variable (in this example the range for C is then $[0, 4]$). This range is provided to the user who can decide to keep it or to use a different range. The differentially private algorithm will then enforce this range and add noise proportional to it.

The language for variable transformations is restricted to allow only statistical operations that combine, transform or separate variables in a value independent way. This prevents high-level timing attacks — ones where the timing leakage is intentional — even if it doesn't prevent fine-grained timing analysis on numerical computations, as we will discuss below. Moreover, to protect against state attacks, the language for variable transformations only allows access to locally defined variables. The program analysis is based on a flow sensitive type system that is used to guarantee that information about the changes in the ranges of variables are propagated to the output.

Summing up, our approach of separating variable transformations from the differentially private data analysis (whose code is not accessible by the data analyst) guarantees protection against privacy budget attacks. The use of a domain specific language further protects against state attacks

---

[11] We make here explicit the fact that M is a function of the records $x_1, \ldots, x_n$ of the input dataset.

and (high-level) timing attacks. Finally, the enforcement of the variable ranges at runtime prevents the misuse of the variable transformations. To help the user decide the range for each variable, the domain specific language uses a program analysis propagating range information from the input to the output.

12.3 Floating-point rounding attack

There is a last kind of attack that PSI is protected from: the floating-point rounding attack identified by Mironov [46]. The idea of this attack is to exploit the irregularities in floating-point implementations of some basic algorithms like the Laplace mechanism. When the output is numeric, differential privacy requires every output to be feasible, i.e. being returned with some probability for every input, and outputs to have similar probabilities when the inputs differ by an individual. Mironov showed instead that naive implementations of differential privacy lead to results that are concentrated on subsets of outputs. Even worse, it can be the case that for neighboring databases some outputs may only be possible under one of the two databases. This allows adversaries to distinguish the output distributions of the two database with certainty and violate differential privacy.

The solution proposed by Mironov is to use the snapping mechanism [46, 19] which essentially tosses out the least significant bits of the differentially private floating-point outputs using a combination of clamping and rounding procedures. This mechanism is also effective when the mechanism is instantiated with imperfect randomness. PSI implements this mechanism for sampling from the Laplace distribution at runtime. The use of this mechanism is transparent to the data analyst and the data depositors and the loss in accuracy with respect to the idealized Laplace mechanism is relatively small.

12.4 Fine-grained side channels attacks

Side channel attacks are in general difficult to prevent. We discussed before how the use of built-in differentially private primitives and a domain specific language for variable transformations can help in mitigating timing channels. Nevertheless, the current implementation may still be prone to fine grained attacks like the one by [7] exploiting time leakages due to floating points computations. We expect these kinds of attacks to be further mitigated by the fact that PSI is only accessed remotely and so some of these fine grained observations are absorbed by delays in the communication. We expect that by using an execution environment where statistical operations have value-independent cost, which can be achieved by padding thanks to the restricted setting, by using some of the proposed mitigations [7], and by having PSI only accessed remotely we can prevent further vulnerabilities. Nevertheless, we leave a complete evaluation of these vulnerabilities to future version of our prototype.

13 Empirical Evaluation

We expect a valuable evaluation of PSI to be by means of extensive user experiments with the kind of social scientists who would actually be using the system. Nevertheless, we have experimentally evaluated all of the differentially private algorithms implemented in PSI using a combination of real and synthetic data.

So far, we have performed these main kinds of experimental evaluations:

1. experiments aiming at confirming the feasibility of releasing several statistics with a given budget,
2. experiments replicating studies from the social science literature.

Experiments on the combined release of statistics The goal of this category of experiment was to answer the question "can we release basic statistics for all the variables with a fixed budget and with a good accuracy?"

To address this we have analyzed several datasets of different size (with n as small as 10^3 and as big as 10^6) available in Dataverse. The overall goal was to release all the univariate statistics currently implemented in PSI under different values of the budget for ϵ (in the range $[0.01, 1]$) with fixed δ (set at 2^-20) and varying the secrecy of the sample assumption (with values $1\%, 3\%, 5\%, 100\%$). We have considered different splits of the privacy parameters among the different statistics, and we have experimented using the optimal composition theorem and the basic composition theorem. We have also used different accuracy measures to capture different characteristics of the different data: mean absolute error, mean relative error, mean squared error, root of mean squared error, ℓ_1, ℓ_2 and ℓ_∞ norm.

From this experience we learnt that in many situations we can provide differentially private results for all the univariate statistics with a non-trivial accuracy. For datasets with sample size 100,000 we were able to release several univariate statistics (mean, histograms, and CDF) for all the variables (∼50 attributes), with mean relative error $\leq 10\%$, with global $\epsilon = 0.1$ and global delta 2^{-20}. As expected, these results have shown some variability depending on the setup of the parameters, e.g. larger dataset sizes and larger values of epsilon give better accuracy, as well as on the error metric used to measure accuracy. Nevertheless, the experiments we performed met some of the expectations set in Section 5. Besides, this step helped us optimizing the code of the different statistics increasing the scalability of the analysis, e.g. the release of several univariate statistics (mean, histograms, and CDF) on datasets with milions of entries and ∼50 variables takes less than 10sec.

Replication of social science studies Replicating the results of published works is a higher bar for PSI than the actual initial goal, which is to support data exploration (for determining whether one should apply for access to raw data). Nevertheless, we created a corpus of 60 datasets from quantitative social science by finding datasets on repositories and reaching out to authors of studies. Our goal was to find datasets that could be publicly released, but whose topics and structure closely resembled those that would ordinarily be closed due to the inclusion of sensitive data. These give us a variety of types and sizes of datasets from which we can benchmark the performance of differentially private statistics, while releasing in comparison the true dataset values. From this corpus, we also chose seven studies which had published articles or reports using simple statistical methods that we could emulate using our available differentially private statistics. Unsurprisingly, the results had once again a large variability depending on the choice of the parameters, the data and the statistics. However, this evaluation has been very important to better understand how to provide tools that are of direct use to applied researchers in the social sciences. As a simple example, many studies release statistics about the overall data as well as statistics about subset of their data, e.g. a subset of the population with a distinguished feature. We plan to explore these functionalities in future releases of PSI.

14 Conclusions and future work

We have presented the design of PSI (a Private data Sharing Interface), a system we are developing to help researchers in the social sciences explore privacy-sensitive datasets under differential privacy. PSI is designed to be accessible to users that have no expertise in privacy, computer science, or statistics, and to be integrated with other tools from the standard workflow of data analysts.

We intend to evaluate PSI by means of extensive user experiments with the kinds of social scientists who would actually be using the system. Such experiments would test whether the users are able to appropriately set privacy parameters and interpret the results given by differential privacy, whether the system is sufficiently easy and efficient to use for practicing social scientists, and whether the system satisfies our goals for data exploration. On the latter point, we want to know whether users make the same decisions about which datasets are of most interest for investigating a particular research question as they would if they could view the raw data.

References

[1] American factfinder. http://factfinder.census.gov/.

[2] Family Educational Rights and Privacy Act (FERPA), 20 U.S.C. § 1232g; 34 C.F.R. Part 99.

[3] Harvard university data classification table. http://policy.security.harvard.edu/.

[4] Naep tools on the web. Technical Report NCES 2011-460, U.S. Department of Education, 2011. https://nces.ed.gov/nationsreportcard/about/naeptools.aspx.

[5] re3data.org reaches a milestone & begins offering badges, 2016. `http://www.re3data.org/2016/04/re3data-org-reaches-a-milestone-begins-offering-badges/`.

[6] Micah Altman, Margaret O. Adams, Jonathan Crabtree, Darrell Donakowski, Marc Maynard, Amy Pienta, and Copeland H. Young. Digital preservation through archival collaboration: The data preservation alliance for the social sciences. *The American Archivist*, 72(1):170–184, 2009.

[7] Marc Andrysco, David Kohlbrenner, Keaton Mowery, Ranjit Jhala, Sorin Lerner, and Hovav Shacham. On subnormal floating point and abnormal timing. In *2015 IEEE Symposium on Security and Privacy, SP 2015, San Jose, CA, USA, May 17-21, 2015*, pages 623–639, 2015.

[8] NORC at the University of Chicago. Obtaining gss sensitive data files.

[9] Gilles Barthe, Boris Köpf, Federico Olmedo, and Santiago Zanella Béguelin. Probabilistic relational reasoning for differential privacy. In John Field and Michael Hicks, editors, *Proceedings of the 39th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2012, Philadelphia, Pennsylvania, USA, January 22-28, 2012*, pages 97–110. ACM, 2012.

[10] Raef Bassily, Kobbi Nissim, Adam Smith, Thomas Steinke, Uri Stemmer, and Jonathan Ullman. Algorithmic stability for adaptive data analysis. In *48th Annual Symposium on the Theory of Computing (STOC'16)*, June 2016. To appear. Preliminary version available at http://arxiv.org/abs/1511.02513.

[11] Grant Blank and Karsten Boye Rasmussen. The data documentation initiative the value and significance of a worldwide standard. *Social Science Computer Review*, 22(3):307–318, 2004.

[12] James Cheney, Stephen Chong, Nate Foster, Margo Seltzer, and Stijn Vansummeren. Provenance: a future history. In *Proceedings of the 24th ACM SIGPLAN conference companion on Object oriented programming systems languages and applications*, pages 957–964. ACM, 2009.

[13] Christine Choirat, James Honaker, Kosuke Imai, Gary King, and Olivia Lau. Zelig: Everyone's statistical software (version 5), 2015.

[14] A community assessment of privacy preserving techniques for human genomes. Xiaoqian jiang and yongan zhao and xiaofeng wang and bradley malin and shuang wang and lucila ohno-machado and haixu tang. *BMC Medical Informatics and Decision Making*, 14(Suppl 1)(S1), 2014.

[15] Mercè Crosas. The dataverse network: An open-source application for sharing, discovering and preserving data. *D-Lib Magazine*, 17:1–2, 2011. doi:1045/january2011-crosas.

[16] Mercè Crosas, Gary King, James Honaker, and Latanya Sweeney. Automating open science for big data. *The ANNALS of the American Academy of Political and Social Science*, 659(1):260–273, 2015.

[17] Jon P. Daries, Justin Reich, Jim Waldo, Elise M. Young, Jonathan Whittinghill, Andrew Dean Ho, Daniel Thomas Seaton, and Isaac Chuang. Privacy, anonymity, and big data in the social sciences. *Communications of the ACM*, 57(9):56–63, September 2014.

[18] Irit Dinur and Kobbi Nissim. Revealing information while preserving privacy. In *PODS*, pages 202–210, 2003.

[19] Yevgeniy Dodis, Adriana López-Alt, Ilya Mironov, and Salil P. Vadhan. Differential privacy with imperfect randomness. In *Advances in Cryptology - CRYPTO 2012 - 32nd Annual Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2012. Proceedings*, pages 497–516, 2012.

[20] Vito D'Orazio and James Honaker. *A User Guide to TwoRavens: An overview of features and capabilities*, 2016.

[21] Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. The reusable holdout: Preserving validity in adaptive data analysis. *Science*, 349(6248):636–638, 2015.

[22] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *EUROCRYPT*, pages 486–503, 2006.

[23] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In 3^{rd} *Theory of Crypt. Conf.*, pages 265–284, 2006.

[24] Cynthia Dwork, Guy N. Rothblum, and Salil P. Vadhan. Boosting and differential privacy. In *FOCS*, pages 51–60, 2010.

[25] Cynthia Dwork, Adam D. Smith, Thomas Steinke, and Jonathan Ullman. Hiding in plain sight: A survey of attacks on private data. Manuscript, April 2016.

[26] Yaniv Erlich and Arvind Narayanan. Routes for breaching and protecting genetic privacy. *Nature Reviews Genetics*, 15(6):409–421, 2014.

[27] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. RAPPOR: randomized aggregatable privacy-preserving ordinal response. In Gail-Joon Ahn, Moti Yung, and Ninghui Li, editors, *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, Scottsdale, AZ, USA, November 3-7, 2014*, pages 1054–1067. ACM, 2014.

[28] National Science Foundation. Award & administration guide (AAG) chapter vi.d.4, 2014.

[29] Sergiu Gherghina and Alexia Katsanidou. Data availability in political science journals. *European Political Science*, 12(3):333–349, Sep 2013.

[30] Alyssa Goodman, Alberto Pepe, Alexander W. Blocker, Christine L. Borgman, Kyle Cranmer, Merce Crosas, Rosanne Di Stefano, Yolanda Gil, Paul Groth, Margaret Hedstrom, David W. Hogg, Vinay Kashyap, Ashish Mahabal, Aneta Siemiginowska, and Aleksandra Slavkovic. Ten simple rules for the care and feeding of scientific data. *PLoS Comput Biol*, 10(4):1–5, 04 2014.

[31] Andreas Haeberlen, Benjamin C. Pierce, and Arjun Narayan. Differential privacy under fire. In *Proceedings of the 20th USENIX Security Symposium*, August 2011.

[32] Michael Hay, Ashwin Machanavajjhala, Gerome Miklau, Yan Chen, and Dan Zhang. Principled evaluation of differentially private algorithms using dpbench. In *SIGMOD*, 2016.

[33] Nils Homer, Szabolcs Szelinger, Margot Redman, David Duggan, Waibhav Tembe, Jill Muehling, John V Pearson, Dietrich A Stephan, Stanley F Nelson, and David W Craig. Resolving individuals contributing trace amounts of dna to highly complex mixtures using high-density snp genotyping microarrays. *PLoS genetics*, 4(8):e1000167, 2008.

[34] James Honaker and Vito D'Orazio. Statistical modeling by gesture: A graphical, browser-based statistical interface for data repositories. In *Extended Proceedings of ACM Hypertext 2014*. ACM, 2014.

[35] Jeffrey Horner. *rApache: Web application development with R and Apache.*, 2013.

[36] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. The composition theorem for differential privacy. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 1376–1385, 2015.

[37] Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? *SIAM J. Comput.*, 40(3):793–826, 2011.

[38] Gary King. An introduction to the dataverse network as an infrastructure for data sharing. *Sociological Methods and Research*, 36:173–199, 2007.

[39] H. J. Lowe, T. A. Ferris, P. M. Hernandez, and S. C. Weber. STRIDE ?- an integrated standards-based translational research informatics platform. *AMIA Annual Symposium Proceedings*, pages 391–395, 2009.

[40] Ashwin Machanavajjhala, Daniel Kifer, John M. Abowd, Johannes Gehrke, and Lars Vilhuber. Privacy: Theory meets practice on the map. In *Proceedings of the 24th International Conference on Data Engineering, ICDE 2008, April 7-12, 2008, Cancún, México*, pages 277–286, 2008.

[41] Andrew J. McMurry, Shawn N. Murphy, Douglas MacFadden, Griffin Weber, William W. Simons, John Orechia, Jonathan Bickel, Nich Wattanasin, Clint Gilbert, Philip Trevvett, Susanne Churchill, and Isaac S. Kohane. SHRINE: enabling nationally scalable multi-site disease studies. *PLoS ONE*, 8(3), 2013.

[42] Katherine McNeill. Interoperability between institutional and data repositories: A pilot project at mit. 2007.

[43] Frank McSherry. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2009, Providence, Rhode Island, USA, June 29 - July 2, 2009*, pages 19–30, 2009.

[44] R. Michael Alvarez and Jonathan N. Katz. Editors' note. *Political Analysis*, 24(2):131, 2016.

[45] Darakhshan J. Mir, Sibren Isaacman, Ramón Cáceres, Margaret Martonosi, and Rebecca N. Wright. DP-WHERE: differentially private modeling of human mobility. In Xiaohua Hu, Tsau Young Lin, Vijay Raghavan, Benjamin W. Wah, Ricardo A. Baeza-Yates, Geoffrey Fox, Cyrus Shahabi, Matthew Smith, Qiang Yang, Rayid Ghani, Wei Fan, Ronny Lempel, and Raghunath Nambiar, editors, *Proceedings of the 2013 IEEE International Conference on Big Data, 6-9 October 2013, Santa Clara, CA, USA*, pages 580–588. IEEE, 2013.

[46] Ilya Mironov. On significance of the least significant bits for differential privacy. In *the ACM Conference on Computer and Communications Security, CCS'12, Raleigh, NC, USA, October 16-18, 2012*, pages 650–661, 2012.

[47] Prashanth Mohan, Abhradeep Thakurta, Elaine Shi, Dawn Song, and David E. Culler. GUPT: privacy preserving data analysis made easy. In K. Selçuk Candan, Yi Chen, Richard T. Snodgrass, Luis Gravano, and Ariel Fuxman, editors, *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2012, Scottsdale, AZ, USA, May 20-24, 2012*, pages 349–360. ACM, 2012.

[48] RL Morgan, Scott Cantor, Steven Carmody, Walter Hoehn, and Ken Klingenstein. Federated security: The shibboleth approach. *Educause Quarterly*, 27(4):12–17, 2004.

[49] Jack Murtagh and Salil P. Vadhan. The complexity of computing the optimal composition of differential privacy. In *Theory of Cryptography - 13th International Conference, TCC 2016-A, Tel Aviv, Israel, January 10-13, 2016, Proceedings, Part I*, pages 157–175, 2016.

[50] Kobbi Nissim, Aaron Bembenek, Alexandra Wood, Mark Bun, Marco Gaboardi, Urs Gasser, David O'Brien, Thomas Steinke, and Salil Vadhan. Bridging the gap between computer science and legal approaches to privacy. 2016.

[51] Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. Smooth sensitivity and sampling in private data analysis. In *Proc.* 39^{th} *STOC'07 ACM*, 2007.

[52] National Institute of Health. NIH notice NOT-OD-03-032: Final NIH statement on sharing research data., 2003.

[53] Office of Management and Budget. OMB memorandum M-13-13: Open data policy – managing data as an asset., 2013.

[54] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2016.

[55] Jason Reed and Benjamin C. Pierce. Distance makes the types grow stronger: A calculus for differential privacy. In *ICFP*, 2010.

[56] Robin Rice. Disc-uk datashare project: Final report. 2009.

[57] Ryan M. Rogers, Aaron Roth, Adam Smith, and Om Thakkar. Max-information, differential privacy, and post-selection hypothesis testing. *CoRR*, abs/1604.03924, 2016.

[58] Ryan M. Rogers, Aaron Roth, Jonathan Ullman, and Salil P. Vadhan. Privacy odometers and filters: Pay-as-you-go composition. *CoRR*, abs/1605.08294, 2016.

[59] Indrajit Roy, Srinath T. V. Setty, Ann Kilzer, Vitaly Shmatikov, and Emmett Witchel. Airavat: Security and privacy for mapreduce. In *Proceedings of the 7th USENIX Symposium on Networked Systems Design and Implementation, NSDI 2010, April 28-30, 2010, San Jose, CA, USA*, pages 297–312. USENIX Association, 2010.

[60] Jatinder Singh. Figshare. *Journal of Pharmacology and Pharmacotherapeutics*, 2(2):138, 2011.

[61] Adam Smith. Differential privacy and the secrecy of the sample. `https://adamdsmith.wordpress.com/2009/09/02/sample-secrecy/`.

[62] Latanya Sweeney, Mercè Crosas, and Michael Bar-Sinai. Sharing sensitive data with confidence: The datatags system. *Technology Science*, 2015.

[63] Kimberly A. Tryka, Luning Hao, Anne Sturcke, Yumi Jin, Zhen Y. Wang, Lora Ziyabari, Moira Lee, Natalia Popova, Nataliya Sharopova, Masato Kimura, and Michael Feolo. Ncbi???s database of genotypes and phenotypes: dbgap. *Nucleic Acids Research*, 42(D1):D975–D979, 2014.

[64] John Wilder Tukey. *Exploratory Data Analysis*. Addison-Wesley, 1977.

[65] Sven Vlaeminck and Lisa-Kristin Herrmann. Data policies and data archives: A new paradigm for academic publishing in economic sciences? In B. Schmidt and M. Dobreva, editors, *New Avenues for Electronic Publishing in the Age of Infinite Collections and Citizen Science: Scale, Openness and Trust*, pages 145–155.

[66] Stuart Weibel, John Kunze, Carl Lagoze, and Misha Wolf. Dublin core metadata for resource discovery. Technical report, 1998.

[67] Hollie C White, Sarah Carrier, Abbey Thompson, Jane Greenberg, and Ryan Scherle. The dryad data repository: A singapore framework metadata architecture in a dspace environment. *Universitätsverlag Göttingen*, page 157, 2008.

[68] Mark D Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E Bourne, et al. The fair guiding principles for scientific data management and stewardship. *Scientific data*, 3, 2016.

[69] Amitai Ziv. Israel's 'anonymous' statistics surveys aren't so anonymous. *Haaretz*, 7 January 2013.