Differential Privacy, Firm-level Data and the Binomial Pathology

David D. Clark
CSAIL
MIT
Cambridge, MA
ddc@csail.mit.edu

Simson Garfinkel
Harvard John A. Paulson
School of Engineering and Applied Sciences
Cambridge, MA
simson@acm.org

KC Claffy
CAIDA
UC San Diego
La Jolla, CA
kc@caida.org

Abstract—Differential Privacy (DP) is a privacy enhancing technology (PET) that is being increasingly used and called for by policymakers in the US and Europe, but not well-suited for protecting corporate proprietary information that are used to produce aggregate industry-wide statistics. We elucidate this scenario with an example of cybersecurity management data, and consider an alternative approach that relies on a pragmatic assessment of harm to add noise to the data.

Index Terms—Public Policy Issues, Data Sharing, Privacy

I. Introduction

Differential Privacy (DP) is a privacy-enhancing technology (PET) that is increasingly used to protect data about individuals. Here, we explore its use for protecting proprietary corporate information that might be used to produce aggregate industry-wide statistics. We provide a worked example that will provide a gentle introduction to DP and shows how recasting the statistics that are to be produced allows the analyst to improve the trade-off between statistics that are both useful and that protect the interests of data providers, who simultaneously wish to benefit from industry-wide statistics while limiting the risk of revealing their proprietary information.

Some firms are willing to support research by providing sensitive and potentially damaging confidential data provided that the results cannot be used to make inferences that point back to those supplying the data. For example, companies may be willing to answer confidential surveys about their experience with cybercrime or malware, provided that statistics are only published in aggregate. Network providers may be willing to provide details regarding their circuit capacity, utilization, and packet loss. However, Dwork and Roth's Fundamental Law of Information Recovery [1] states that every use of confidential data to produce public statistics releases information increases the chance that confidential data can be accurately reconstructed.

DP, despite being a powerful PET, poorly addresses some kinds of reputational harms that firms might suffer following the release of some kinds of aggregate statistics. However, changing the specific questions that are asked of the confidential data—the query set—may make it easier to both apply DP and reason about possible harms. Readers may find this especially relevant, given various national efforts to advance the use of DP and other privacy enhancing technologies. [2]

Our worked example here uses synthetic cybersecurity management data, specifically the number of patched systems in an enterprise. However, our argument is applicable to a wide range of proprietary data. For example, it could apply to counts of ransomware attacks, or counts of systems, or to the packet loss of high-speed internet links, or any other statistic that one might wish to protect with DP that exhibits the mathematical properties that we present here. We do this by avoiding the traditional mathematical framing of DP as much as possible and instead focus on the utility of the released statistics in an intuitive way based on visualizations of simulated results.

Finally, we consider an alternative approach for performing privacy-preserving data analysis that is inspired by DP, but where the amount of noise added is derived based on an empirical assessment of harm.

II. Background

There are three distinct and complex challenges when trying to protect proprietary data at the firm level. First, since the kinds of harms that firms and individuals can suffer are fundamentally different, firms have fundamentally different concerns regarding the disclosure of proprietary information than individuals have regarding the disclosure of their personal data. Second, the sample size (the number of firms providing data) is often small. Third, in general, there are considerably more differences in measurable characteristics between firms than between people, resulting in more variability between samples and more outliers. All of these pose challenges for DP.

There are also practical challenges in using DP. First, DP is formally described in terms of a mathematical abstraction called privacy loss, typically represented by the greek letter epsilon (ϵ), which maps to some relative potential improvement in capability that an attacker may enjoy as a result of a data release, but does not map to absolute increases in harm. While the ultimate goal of DP is to minimize harm that data subjects might experience as a result of providing data, the potential for actual harm must take into account the context of the query—something that DP generally does not do.

There are a number of ways to think about and use DP, but the simple version we use in this paper is to imagine that there is a database with a number of records, each with a fixed number of fields. Each record represents confidential data from a different entity. The goal is to produce a useful statistic for release while providing some degree of protection for the confidential data.

DP provides protection by adding a degree of noise to the result of each query against the confidential records. This noise makes it difficult to reconstruct one or more of the true, confidential values of any record or combination of records. Equivalently, DP limits the ability of an attacker (which we will call the data hacker) to ascertain if data from a particular entity are or are not included in the confidential dataset.

In general, DP mechanisms fall into three broad modes of operation:

1) Local model. Noise is added to every element of every record of the entire database, after which the entire noisy database can be used for any number of statistical operations without further privacy loss. Indeed, once noised, the entire noisy database can be publicly released. Local model requires comparatively high levels of noise to achieve significant privacy protection, which limits the usefulness of this approach.
2) Central model. A trusted data curator receives confidential data, computes statistics, and adds noise to each result. Multiple queries that address the same records increase the overall privacy loss that data subjects experience.
3) Central model with synthetic data. The trusted curator performs queries on the data to produce a noisy statistical model, which it then uses to generate synthetic data. This data can be used or published without additional privacy loss. The challenge with this approach is creating synthetic data that have sufficient fidelity and accuracy. In practice, this is an open research problem.

Here we explore the use of DP solely in the trusted curator mode, avoiding the high levels of noise required by the local model and the immaturity of methods to generate synthetic data.

The goal of DP is to assure that an analysis of a database containing an individual's confidential data should not differ by more than a small amount from an analysis of a similar database that does not contain the individual's data.

DP uses a parameter ϵ to quantify what we mean by "a small amount." If $\epsilon = 0$, there should be no difference, which means that queries on the database can have no relationship to the data stored in the database. If $\epsilon = \infty$, then any difference is acceptable. In practice, $\epsilon = \infty$ allows a query to precisely release any value in the database, or even the entire database.

We are interested in the range $0 < \epsilon < \infty$, where there is a trade-off between the accuracy of the output and the amount of privacy loss incurred. The higher the accuracy, the more privacy loss.

If the intended use of the data requires more accuracy, or alternatively if the data do not require so much protection, then less noise can be used, and there is more privacy loss and more accurate statistics.

III. A simple example of harms from querying firm-level data

Consider the following simple example:

> 100 firms using a widely used operating system each complete a survey reporting the fraction of systems they are running that have been upgraded to the latest security patch. Each of these 100 reports consists of a single number between 0.0 and 1.0 and is stored in database D that is operated by the trusted curator. Our goal is to get a sense, industry-wide, of whether firms are keeping their systems up-to-date with respect to security patches.

Once the trusted curator receives the reports, the curator computes one or more queries on the confidential data and publishes the result to the public.

One obvious query of public interest would be the average of the values. In practice, we might wish to weigh each sample based on the size of the firm, but for this simple example, we assume a query that solely computes the mean of the values returned from each firm. This statistic will eventually be translated to a headline on a website, such as: "Customer Survey finds only 90% of Systems Properly Patched."

Our first question is whether releasing the mean of these values (with no noise added) can cause harm to the firms that provided the inputs. What if the mean is 0.5? This might imply that the contributing firms have all upgraded half of their systems to the latest patch level. Alternatively it might be that precisely half of the firms reported patching all of their systems, and half reported patching none.

A. The Data Hacker

In the specific case above, unless we assume that the data hacker knows the statistic for 99 of the firms and is attempting to learn the data for the firm that remains, it is unlikely that revealing that the average is 0.5 will harm any one firm. (We return to this assumption in Section IV-C.)

However, this harmless situation may not hold with other averages. What if the mean is 0%? Then it would have to be true (from the math) that each of the firms returned the value 0.0 as their firm's response to the query. No firm has upgraded any of their systems. The release of the average would cause reputational harm to all of the contributing firms. Note that if the average had been 100%, the firms might be very happy to reveal that result. The actual harm (or the potential for actual harm) depends on both the result and the context of the query, not the underlying math.

We term this harm the binomial pathology, drawing an analogy to the binomial theorem, in that there are many ways to take 50 balls out of an urn with 100 balls in it (without replacement), but ignoring order, there is only one way to take out 0 or all 100. As the returned value of the query gets closer to the minimum or maximum of the possible range for the mean, there are fewer and fewer combinations of data values that can yield that specific result. Thus, the potential for harm is data dependent.

The binomial pathology can arise in other contexts. Consider a census block where the average age is 45. There could be much younger and much older people contributing to that average, so we learn little about them as individuals. But if the average is 85, it is a good guess that most of the people in that block are likely to be at least over 65, because no individual is likely to be over 130. The data hacker, seeing that the average age is 85, can only guess about a given individual in the census block, but a guess can be good enough to cause harm. Also, the guess can become arbitrarily more accurate with additional public data—for example, learning that a couple living on the block married just before the husband was drafted to serve in the Korean War.

B. Privacy loss vs. harm

While the ultimate goal of DP is to prevent harm to the data subject, the protection provided by DP is defined not by the possible harm of a data release, but by the maximum amount of privacy loss that can result.

It is the relative privacy loss that is data independent: the absolute protection depends upon the global data context. A given amount of privacy loss will be more damaging in the hands of a data hacker who has substantial knowledge about the world in which the data subjects reside.

DP's protection is defined by the degree that the potential harm caused by the result of a query is independent of whether the individual's data were considered when evaluating the query. The classic DP example is a query that tries to establish a link between smoking and cancer. If that linkage is accepted, a smoker might see their health or life insurance rates go up. The smoker is harmed by the result of the query, but not because of privacy loss: it made no difference whether the smoker's confidential data were in the database or not. So the difference between the harm suffered whether or not the smoker's data were considered is zero, which is why the approach is called differential privacy. The smokers in the data were harmed, but so were the smokers not in the data. [3]

In our security example above, if we publish that the average patch rate is 0.0, the firms are individually harmed, but so is the broader community: it will be guilt by association.

This kind of harm may not be acceptable in the case of corporate confidentiality. If we seek voluntary release of data (as opposed to data release that is compelled by regulation or law), the fear of this kind of harm may cause firms to refuse to release data.

For example, corporations may fear that making confidential data available to produce industry-wide statistics may help create a body of evidence that will be used to regulate the industry. This is a harm that DP is not designed to mitigate, because this is a harm outside of DP's definition of privacy loss. [4] But recognizing this limitation, can adding noise to a result contribute to the mitigation of this sort of harm?

IV. Adding noise, in the DP way

DP protects privacy by adding noise to the result of each query, creating uncertainty for a data hacker attempting to learn the contents of the confidential database.

There are many approaches for adding noise that are consistent with DP; here we use the Laplace Mechanism, which adds noise drawn from a Laplace distribution with zero mean to the result of each query. The zero mean assures that the noise added to the true answer is equally likely to be positive or negative, so that there is no implicit bias added to the query results. The magnitude of the noise added (the width of the Laplace distribution) is determined by two factors: ϵ (discussed above), and a factor called sensitivity. While ϵ gets all the attention in discussions of DP, the concept of sensitivity is equally significant, as the amount of noise added is a function of both.

A. Sensitivity

DP sensitivity (Δf) is the maximum amount that a query result (in this case, mean) can change if the data associated with the unit of protection—typically a single database record—is changed or removed. (Here we ignore the subtle difference between removing a record and changing it; see Kifer and Machanavajjhala [5] for a full discussion.)

The sensitivity of a query is based not on the actual values in the current database, but on the theoretical maximal impact that a single record change could cause for the universe of all possible databases. In fact, it is an error to use the contents of a particular database to compute query sensitivity: it must be inferred from the range of values that might be in the database. In our example, with 100 samples between 0 and 1, the maximum impact a single firm could have would be the situation where $(x_1 ... x_{99}) = 0$ and $x_{100} = 1$. In this case, the mean will either be $\frac{x_{100}}{100}$, or .01. or else 0, if x_{100} is changed to 0. The global sensitivity S is thus .01.

B. The Laplace noise function

The zero-mean Laplace distribution is defined as follows: for a possible value x of noise to be added to the true value, the probability of adding that noise value is

$$P = \frac{1}{2b} exp(-abs(\frac{x}{b}))$$

where b is defined as $\frac{S}{\epsilon}$, and S is the sensitivity.

Most papers that introduce differential privacy include a plot of the Laplace distribution; ours appears below, in figure 1. The Y-axis height of the red line indicates the probability that a single value drawn from this Laplace distribution will result in the value indicated on the X-axis. For this distribution with a mean of 0.5 and a scale of 0.01, the most probable value is 0.5, and 95% of the values will be between 0.47 and 0.53. These values correspond to using the Laplace Mechanism with an $\epsilon = 1.0$ and sensitivity $\Delta f = 0.01$ to add noise to a value of 0.5.

Note that while it appears that 0.50 is the most probable value, a value close to 0.5 is vanishly improbable: fewer than 10% of the values are between 4.999 and 0.501

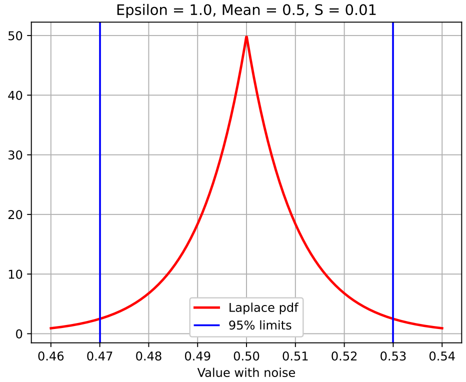

Fig. 1. The probability distribution function (PDF) of possible noisy statistics for a mean of 0.5; the total area under the red line is equal to 1.0; 95% of the values appear between the two blue lines.

We would like to derive the required level of noise (or other possible mitigations) from an assessment of the real potential for harm, rather than an abstract notion of privacy loss based on ϵ. In practice, it is difficult to quantify the potential for harm. Instead, many data analysts take the reverse approach: they review the amount of noise that a given level of ϵ will add to the statistic and decide if the result is still useful. If not, the amount of noise is decreased; if so, the amount of noise is increased. Eventually the analyst finds the maximum amount of noise that can be added while still allowing the statistic to be fit-for-use. This emphasis on utility may dominate an assessment of harm.

In our example, in which the query is designed to reveal the fraction of systems patched to the latest release, the privacy protected value will be between 0.47 and 0.53 with high probability, and between 0.45 and 0.55 with almost absolute certainty. Many people consider $\epsilon = 1.0$ to be a high amount of privacy protection, and in this case it happens to produce what we consider to be a useful answer! But we maintain that this would not be the case if the industry mean were an extreme value.

C. The worst-case assumption

DP's definition causes it to make a worst-case assumption about the prior knowledge of the data hacker, short of knowing the actual value that DP is trying to protect. That is, it assesses an upper-bound of the potential loss of privacy that might result from a data release, independent of what the data hacker's prior knowledge or computational capabilities. So consider the case of a data hacker that happens to know the actual answer for 99 of the firms, and wants to learn the answer for the final firm. If no noise is added to the answer, then the hacker can easily reverse the computation of the mean and derive the answer for that firm. Thus, the data curator decides to protect the result with DP.

Assume that the mean of the known 99 values was 0.5. If the remaining (unknown) value is 0.0, the true mean of all the values will be 0.495. If the remaining value is 1.0, the true mean would be 0.505. (Note that we have just recomputed the Global Sensitivity in this case–the difference is 0.01.) In these two extreme cases, what would the Laplace distributions be for the noisy answer with an epsilon of 1?

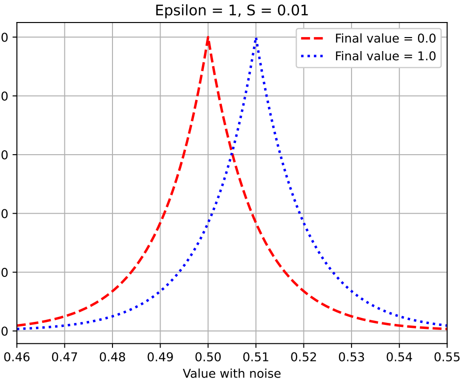

Fig. 2. PDF graphs showing the range of possible noisy results depending on actual value of the 100th data item being 0.0 (red) or 1.0 (blue), assuming that the other 99 data items are all 0.0.

As before, the data hacker faces a 95% certainty that the returned value is ±0.03 from the actual answer. The hacker does not know from which distribution (anywhere between the lowest and highest pictured in Figure 2) the returned value came. All the hacker sees is a single number. If that number happened to be .5, the value is equally likely to have come from the lowest and the highest alternative, so the hacker has learned nothing. However, if the answer were (for example) .48, it is much more likely that this result was from the distribution on the left. In other words, the hacker cannot guess the true value of the final value, knowing the other 99 values, but may be able (for some published noisy results) to guess that the remaining value is "lowish" or "highish." Whether this degree of guess is harmful is a policy question that must be answered from the actual context: it is not a question of math.

One question we might ask is whether we need to address the actual worst case. A hacker that knows all but 2 of the values would learn essentially nothing from the range of possible noisy answers. If we allow ourselves to relax the worst case assumption in assessing the potential for actual harm, we may get a more realistic assessment of what a data hacker can actually learn. Once again, this is a policy question that the mathematical foundation of DP allows us to ask, but does not answer.

V. Addressing the binomial pathology

The previous illustrations showed the distribution of added noise if the actual mean was 0.5. What if the true mean was really 0.0—that is, what if all of the firms had patched none of their systems? Figure 3 shows the resulting distribution of noisy answers.

Here, the hacker can reasonably infer that there is a 95% probability that less than 3% systems are patched. The hacker cannot know that the actual value was 0.0, but even knowing that it is highly likely that the value is less than 3% may be enough to cause harm. Would this

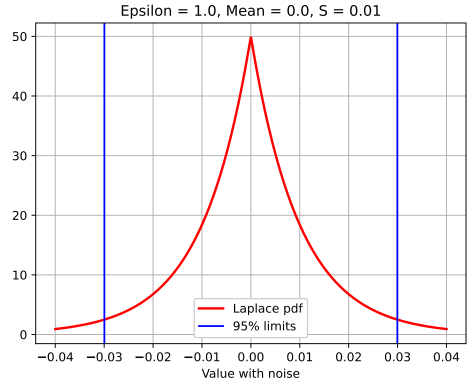

Fig. 3. PDF of possible noisy results if the true mean of the 100 values is 0.0.

degree of uncertainty allow any single firm that had contributed data to plausibly claim that while the overall number was really low, they had actually done a good job? Probably not.

Note that the reported value after protection may be less than zero or greater than 1. That is, the trusted curator might declare that -10% of all companies are fully patched. In this case, all parties (both legitimate data analysts and the data hacker) would understand that this cannot be a true value, and that it was either be the result of a small number (such as 0) being treated with a small negative value from the Laplace distribution, or else (with lower probability) a larger value (such as .5, or even 1.0) being treated with an even more negative value from the Laplace distribution.

To minimize the public's confusion, the organization producing the protected statistics might resolve to only report values ≥ 0 and ≤ 1.0 This robs downstream data users (and hackers) of some information, but lessens the chance that the reporting agency is open to ridicule. (This exact problem faced the US Census Bureau in its use of differential privacy for the 2020 Census; it resolved the problem by publishing two sets of statistics: one set having only non-negative integer counts, and a second set, the so-called noisy measurements file, containing negative and fractional numbers.)

How about an organization that has not contributed to the dataset? If the names of the firms are themselves confidential, an organization could claim that their data was not included in the computation, but there is no way to prove this. DP cannot help this organization. Even if the organization can establish through some kind of audit that its data truly was not included, that organization will still likely suffer reputational harm because it will be tarred with the same brush as the poorly performing organizations that did participate.

These sorts of harms are not the harms that DP is designed to prevent. This is like the case of smoking and cancer. Some reputational harm may attach to firms of the sort surveyed, whether or not they were in the sample.

VI. Small samples exacerbate DP privacy loss

What if there were only 10 firms in the database, rather than 100? In that case, the global sensitivity S would be 10 times greater, and if ϵ were still 1, the distribution of possible results would look like Figure 4.

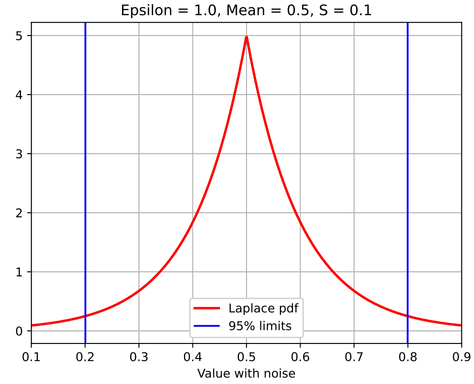

Fig. 4. PDF of the protected (noisy) statistic if the true mean of the 10 values is 0.5.

This amount of noise, given a global sensitivity of .1, significantly decreases the utility of any published results.

How can we improve the utility of the result? One obvious answer is to decrease the amount of noise, which is accomplished by increasing ϵ. WIth $\epsilon = 10$, the plot will be exactly the same as Figure 1. This should not be a surprise: if we increase ϵ by 10 and increase the global sensitivity by 10, the two changes cancel out. What is different is that the data hacker can now form a far more accurate hypothesis regarding the underlying confidential data (see Figure 5).

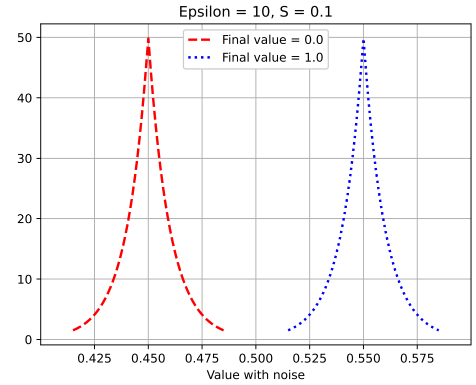

Fig. 5. Range of noisy results depending on actual value of the final data item, with 10 samples.

Because the global sensitivity has changed by a factor of 10, the curves representing the Laplace distribution for the minimum and maximum value of the one sample that the worst-case hacker does not know have moved 10 times further apart. In this case, while the hacker cannot guess an exact number for that remaining hidden value, as Figure 5 shows, the hacker can make a very good guess.

The high quality of the guess is consistent with DP's concept of privacy loss: in the context of a single query protected using the Laplace mechanism, there is little protection with $\epsilon = 10$.

VII. Making the situation better

We have identified two risks that DP was not designed to mitigate (and does so poorly): small data set sizes and the binomial pathology. Regarding the first risk, as the sample size (the number of firms) shrinks, we must rely on a pragmatic, not worst-case, analysis of the capabilities and intentions of the hacker to assess potential harm. But what can we do about the binomial pathology?

A. Sampling the data records

One approach that is used to try to protect individual entries in a database is to sample the database and compute a noisy result across a subset of the sample (here, the firms). [6] In this case, a firm can try to claim that the result may not apply to them because they may not even have been in the sample. In this simplistic example, would sampling address the privacy concerns of the firms?

Sadly, probably not. Here, the sample size works against that claim. If there were 100 firms, and the trusted agent that computes and returns the noisy answer declares that it has used the data from only 90 of them, what then? Statisticians, when looking at the behavior of firms, can often justify the assumption that the variables that define their behavior are i.i.d. In this case, using a sample of 90 to predict the behavior of a population of 100 is a highly robust statistical assumption. The more firms that are in the data, the stronger the justification for a statistical conclusion that a subsample is a robust predictor of the population. Sampling cannot help us here.

B. Change the query

Another way out of this dilemma is not to compute the mean, but to devise an alternative query that provides sufficient information for the needs of the data analyst but avoids pitfalls such as the binomial pathology. These alternative queries may also benefit from added noise, but with careful design can avoid dangerous results from low-probability data values. Indeed, this experience is common for those attempting to adopt a statistical analysis to incorporate differential privacy: frequently it is necessary not simply to add noise to the statistics that are reported, but to change the statistics that we choose to report.

One kind of query might be some form of quantile. For example, the query might be: "what fraction of the firms have patched more than 50% of their systems." This is essentially a histogram with two bins, and a count for each.

Of course, none of the firms might have patched more than 50% of their systems, so 100 would be in the lower bin, and none of them in the upper bin. Would this outcome represent a harm to an individual firm? None of the firms have patched more than 50%–that fact is known about each individual firm. But the firms might find this degree of reputational loss to be tolerable, since all the others are in the same boat. And any single firm could argue that they had done 49% of their systems.

With DP we protect histograms by adding noise independently to each bin's count. If there are 90 firms with fewer than 50% of their systems patched and 10 firms with more than 50% of their systems patched, the DP computation might add noise of +2.5 to the first number and -1.3 to the second number, with the result of 92.5 firms in the firm bin and 8.7 firms in the second. Of course, the reporting organization might choose to round these numbers.

Small counts are still a problem, however. If there is a bin with 100 firms in it, adding or subtracting one or two as we add noise does not greatly change the utility. For a bin with one firm in it, adding or subtracting one or two, which might make the result zero or negative, is potentially a huge loss in the precision (and utility) of that small bin size. If we created more bins, the expected number of firms in each bin would be smaller, so the degree of uncertainty for the results would increase. If there were only 10 firms in the data, and we split them up into more than a very few bins, the added noise would render the results less useful.

C. Add noise based on the actual data

If it is necessary to use a query (such as mean) that has a low-probability data disclosure pathology, we could consider abandoning the logic of DP and adding noise (or more noise) only when the actual data triggers the pathology. To do so steps completely outside the philosophy of DP, because the fact that additional noise has been added to a particular result (which has to be disclosed) itself reveals important facts about the data. In our example with 100 firms, the trusted agent could add additional noise as the true value of the result approaches 0.0. This would prevent a data hacker from making a precise guess, but would still make obvious that the number was unfortunately low. Such ad hoc systems are difficult to analyze and are brittle if there exists external data that can be used to undo their protection mechanisms: it was the analysis of such schemes and the dissatisfaction with them that led to the development of DP.

Once the trusted curator has committed to releasing the query of a mean, there are no easy ways to mask the pathological outcomes. Refusing to return a result itself reveals something about the data. Adding lots of noise based on the data reveals something about the data, which is why DP requires that the trusted curator make the decision about how much noise to add before looking at the confidential data. This is an example of the "Fienberg Problem." [7] The US Census Bureau addressed this problem by tuning its DP system for the 2020 Census using data from the 2010 Census.

It is critical to disclose the amount of noise that a trusted curator has added to a result, both to inform the legitimate data analyst and as well a possible data hacker. A frustrating harm can occur when an hacker does not understand how the added noise has limited the validity of his conclusion, and publishes an unjustified conclusion, causing reputational harm that could have been prevented had the hacker been aware of the added noise. Showing some form of error bars on an answer may be a way to make the point forcefully. However, the error bars must not be presented in a way that reveals anything further about the actual data.

VIII. Conclusions

The astute reader may have observed that we could have written this paper from a different starting point, with a title such as: “The Hidden Perils of Computing a Mean,” and gotten much of the way through the development without even mentioning DP. We chose this course through the material both to introduce the basic ideas of DP, and to point out that there are queries on specific kinds of datasets that must lead to either poor privacy or poor utility outcomes even for systems that implement DP. Kifer et. al’s award winning paper explores this specific tradeoff in greater depth. [5].

Practitioners seeking to deploy DP to real-world situations must be concerned with both the setting of ϵ and in developing queries that provide sufficient utility and address the risk of addressing firm-specific information. Such details are frequently not discussed in introductory texts on DP, but they are beginning to appear in practical guides, such as NIST’s recently published draft Guidelines for Evaluating Differential Privacy Guarantees. [8] Dwork recommends that organizations maintain an “epsilon registry” to for both internal accounting and public accountability. [9] Benthall et al propose an approach for integrating DP with the contextual integrity privacy model. [10]

We believe that practitioners deploying DP should start not with a desired of ϵ, but instead start with an assessment of harm, and tune the amount of noise to be added based on that assessment. After determining the amount of noise, it may be useful to compute and report ϵ to assist others in assessing the amount of privacy protection.

David D. Clark is a senior research scientist at the MIT Computer Science and AI Lab, the Massachusetts Institute of Technology, Cambridge, Massachusetts, 02139, USA. His research focuses on the design of the Internet and the forces shaping its future; Internet security; and the societal, economic, and regulatory implications of the Internet. Clark received his Ph.D. from the Massachusetts Institute of Technology, Cambridge, Massachusetts, USA. He is a Fellow of IEEE and of ACM. Contact him at ddc@csail.mit.edu.

K. Claffy is a Founder and Director of the Center for Applied Internet Data Analysis (CAIDA), a Resident Research Scientist of the San Diego Supercomputer Center at UC, San Diego, and an Adjunct Professor in the Department of Computer Science and Engineering at UC, San Diego. Her research interests span Internet topology, routing, security, economics, future Internet architectures, and policy. She leads CAIDA research and infrastructure efforts in Internet cartography, aimed at characterizing the changing nature of the Internet’s topology, routing and traffic dynamics, economics, and investigating the implications of these changes on network science, architecture, infrastructure security and stability, and public policy. She has been a member of the Security and Stability Advisory Committee since 2003. She has been at SDSC since 1991. She received the Ph.D. degree in computer science from UC San Diego, La Jolla, CA, USA. Contact her at kc@caida.org.

Simson Garfinkel is the Chief Scientist of BasisTech, a technology accelerator in Somerville, MA. He is currently working on projects at the intersection of AI, privacy, data management and digital forensics. He was previously the Senior Computer Scientist for Data Confidentiality and Access at the US Census Bureau, where he helped deploy differential privacy for the 2020 Census, and an Associate Professor of Computer Science at the Naval Postgraduate School. Garfinkel received his Ph.D. in Computer Science from the Massachusetts Institute of Technology, Cambridge, Massachusetts, USA. He is a Fellow of IEEE, AAAS and ACM. Contact him at simsong@alum.mit.edu

References

[1] C. Dwork and A. Roth, “The Algorithmic Foundations of Differential Privacy,” Foundations and Trends in Theoretical Computer Science, vol. 9, pp. 211–407, Aug. 2014. Publisher: Now Publishers, Inc.

[2] T. DeBlanc-Knowles, J. Joshi, N. Lefkovitz, and K. McCall-Kiley, “National Strategy to Advance Privacy-Preserving Data Sharing and Analytics,” tech. rep., NCO NITRD, Mar. 2023.

[3] C. Dwork, F. McSherry, K. Nissim, and A. Smith, “Differential Privacy: A Primer for the Perplexed,” (Tarragona, Spain), Oct. 2011.

[4] C. Dwork, F. McSherry, K. Nissim, and A. Smith, “Calibrating Noise to Sensitivity in Private Data Analysis,” in Theory of Cryptography (S. Halevi and T. Rabin, eds.), Lecture Notes in Computer Science, (Berlin, Heidelberg), pp. 265–284, Springer, 2006.

[5] D. Kifer and A. Machanavajjhala, “No free lunch in data privacy,” in Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data, SIGMOD ’11, (New York, NY, USA), p. 193–204, Association for Computing Machinery, 2011.

[6] B. Balle, G. Barthe, and M. Gaboardi, “Privacy Amplification by Subsampling: Tight Analyses via Couplings and Divergences,” in Advances in Neural Information Processing Systems, vol. 31, Curran Associates, Inc., 2018.

[7] C. Dwork and J. Ullman, “The Fienberg Problem: How to Allow Human Interactive Data Analysis in the Age of Differential Privacy,” Journal of Privacy and Confidentiality, vol. 8, Dec. 2018. Number: 1.

[8] J. Near, D. Darais, N. Lefkovitz, and G. Howarth, “Guidelines for Evaluating Differential Privacy Guarantees,” Tech. Rep. NIST Special Publication (SP) 800-226 (Draft), National Institute of Standards and Technology, Dec. 2023.

[9] C. Dwork, N. Kohli, and D. Mulligan, “Differential Privacy in Practice: Expose your Epsilons!,” Journal of Privacy and Confidentiality, vol. 9, Oct. 2019. Number: 2.

[10] S. Benthall and R. Cummings, “Integrating Differential Privacy and Contextual Integrity,” in Proceedings of the Symposium on Computer Science and Law, CSLAW ’24, (New York, NY, USA), pp. 9–15, Association for Computing Machinery, Mar. 2024.