Privacy in Practice: The Feasibility of Differential Privacy for Telemetry Analysis

Trey Scheid
tscheid@ucsd.edu

Tyler Kurpanek
tkurpane@ucsd.edu

Bradley Nathanson
bnathanson@ucsd.edu

Christopher Lum
cslum@ucsd.edu

Yu-Xiang Wang
yuxiangw@ucsd.edu

Abstract

This research investigates the implementation implications of differential privacy mechanisms for telemetry data analysis, with a focus on real-world applications. We show, through examples, how knowledge of fundamental privacy-preserving techniques, including randomized response and the Laplace mechanism, is enough to protect sensitive information while maintaining analytical utility. We privatize data tasks from 4 applied research works using Intel telemetry data which encompasses multiple statistical tasks, from user-level rate analysis to logistic regression classification. The study utilizes various (ϵ, δ) budgets (using AutoDP) for precise privacy loss measurement and to quantify the inherent tradeoff between privacy and utility. By demonstrating the feasibility of differential privacy in production environments, we provide a roadmap for organizations seeking to enhance their privacy practices.

Website: https://trey-scheid.github.io/privacy-in-practice/
Code: https://github.com/Trey-Scheid/privacy-in-practice

Table of Contents

1 Introduction

The implementation of differential privacy in production environments presents significant challenges in balancing privacy guarantees with analytical utility. This research addresses these challenges by developing practical privacy-preserving mechanisms for existing telemetry analysis tasks while maintaining the usefulness of their systems. We integrate various differential privacy mechanisms including: Gaussian Composition, the ExponentialLaplace mechanism, and __ to protect sensitive information in telemetry data. Our chosen data tasks encompasses multiple statistical tasks, from user-level rate analysis to logistic regression classification correlation. By evaluating the trade-offs between privacy guarantees and analytical accuracy in production settings, we provide evidence and direction for organizations looking to enhance their privacy practices.

1.1 Motivation

Despite the growing importance of privacy-preserving data analysis Pew Research Center (2019), many practitioners perceive differential privacy implementation as complex and challenging, others note how results only come with dissatisfactory ϵ-level guarantees Bonawitz et al. (2022). Besides the sub-optimality of some DP methods, this perception stems from human difficulties: the mathematical complexity of privacy definitions, the need to carefully calibrate privacy parameters, and concerns about reduced utility (managing the trade-off) Ponomareva et al. (2023). A survey by Smith et al. found that only 23% of data scientists felt confident implementing differential privacy mechanisms in their workflows **?**. Researchers are working to democratize, demystify, and improve usability around Differential Privacy Ponomareva et al. (2023). Recent developments have significantly lowered these barriers to entry. Tools like Google's Privacy on Beam [1], Microsoft's SmartNoise [2], and various open-source libraries like AutoDP[3] provide accessible frameworks for implementing differential privacy. These tools abstract away much of the underlying complexity while maintaining rigorous privacy guarantees. In addition, educational resources and practical tutorials have emerged to guide practitioners through implementation challenges [4].

This research builds upon these recent developments by providing a practical demonstration of differential privacy mechanisms in telemetry data analysis. By implementing privacy-preserving techniques for existing tasks, we aim to show that differential privacy can be seamlessly integrated into production systems without significant utility loss. Our work focuses on two key objectives: privatizing existing telemetry analysis tasks and evaluating the privacy-utility tradeoffs in production settings.

[1] https://codelabs.developers.google.com/codelabs/privacy-on-beam
[2] https://smartnoise.org
[3] https://pypi.org/project/autodp/
[4] https://desfontain.es/blog/friendly-intro-to-differential-privacy.html

1.2 Differential Privacy

Differential privacy by Dwork and Roth (2014) is a framework for data privacy that gives a mathematical guarantee that the information for each individual (record or user) in the dataset is protected[5]. The core idea is to introduce random noise into algorithms so that the data of any individual does not significantly affect the overall result and therefore is not recoverable or identifiable.

Mathematically, a mechanism M is considered (ϵ, δ) differentially private if for all datasets D and D' which differ by at most 1 element when $\mathbb{P}[M(D) \in S] \leq e^{\epsilon}\mathbb{P}[M(D') \in S)] + \delta$ where ϵ and δ are privacy parameters and S is a query solution set. Smaller ϵ and δ imply stronger privacy guarantees. Differential privacy is a *property* of algorithms, not datasets; it is a method that ensures private results to a high degree of probability (whether that is a trained model or a noisy dataset).

This definition applies to data anonymization, but does not cover methods for transparency, use, access or security. By pursuing this property for common data tasks we aim to create a solution which once implemented achieves similar results but removes the need for data access and security. Some settings the predictions themselves are important, but sensitive, meaning a mechanism without anonymity is unusable!

1.3 Telemetry Data

The Intel Data Collection and Analysis (DCA) team derives insights from over 39 million systems! This includes any of their hardware installed in personal, corporate and IoT devices (collected only with consent). Through their partnership with the University of California San Diego's Halicioğlu Data Science Institute, at the foundation of Intel Lab's Telemetry Center of Excellence, they permit study of possibly sensitive device information to develop solutions that benefit the whole ecosystem [6]. Faculty and Intel researchers have published many white papers using the database since the CoE inception in 2020. We selected 4 of interest and found their core data science methods Table 1.

Differential privacy methods and guarantees are attractive for many domains. Telemetry is the remote data transfer of automated system measurements. As people use technology everyday their machines track usage diagnostics which are used by hardware and software manufacturers to reduce bugs and increase efficiency. System usage information is recorded at regular intervals and usually results in massive quantities of measurements. The identifiability of the specific machine or user of an event is a concern regardless of PIID tags. Dinur Nissim Reconstruction and linkage attacks can be used to recover or reconstruct the original information: the source Dinur and Nissim (2003). This is a breach of privacy for a user which depending on the sensitivity of the information can be concerning. For example, personal laptops may send diagnostics to Intel given that the user opts in to the program [Intel telemetry].

[5] Gadotti et al. (2024) has an in depth explanation of interpretations and attacks.

[6] https://community.intel.com/t5/Blogs/Tech-Innovation/Data-Center/Intel-Labs-Investment-in-Telemetry-Center-of-Excellence-Produces/post/1460669

Table 1: Data Tasks

Data Task	Code	Paper	Citation
Conditional Probabilities	COND_PROB	Exploration of CPU Error Dependencies and Prediction	Kwasnick (Unpublished)
KMeans Clustering	KMEANS	PC Health Impact White Paper	Ryan et al. (Unpublished)
Lasso Regression	LASSO	Power Consumption Patterns in Intel's Telemetry Data: China Burns 2x Energy that of the US	Cheon (Unpublished)
Logistic Regression Significance Tests	LR_PVAL	Product Health Insights Using Telemetry	Su et al. (2024)

We use a secure research database shared be Intel Corporation with consent of its users to generate real results....

1.3.1 Errors

In our paper, we will analyze two different types of errors. The Machine Check Architecture, or MCA, will detect an error and label it as either corrected or uncorrected. A corrected error means the system can observe and correct a detected error. Correction mechanisms include single error correction, double error correction, and more. An uncorrected error is one that was detected but not corrected or there was a computation delay long enough that the MCA treated it as an interrupted computation. Kwasnick (Unpublished)

1.3.2 Hardware Power [not done yet]

Another concept is power, this is the rate of energy consumption by the device. Our analysis is on CPU's produced by Intel and AMD. Kwasnick (Unpublished)

1.4 Applications [not done yet]

Telemetry is one narrow domain which privacy is a concern, many other types of data require sensitive handling and sharing practices. For example the US Census [7].

We will get into the specifics of each task, however note that each one can be applied to datasets about any interest. probabilities of political party affiliation, Lasso/kmeans for gene identification, or correlation for __.

[7] need citation

1.5 Related Work

We are building on a previous analysis on differentially private mechanisms for Logistic Regression Scheid et al. (2024). The paper investigates how privacy affects different mini-batch stochastic gradient descent algorithms for logistic regression classification. It is shown that privacy affects the batch size for optimal performance.

2 Methods

2.1 Data Structure

The first sub-task was data ingestion and processing. One key feature of telemetry data is the volume; the research database was already a processed version of raw signals from devices, aggregated and merged for usefulness and practicality. This left us with a schema containing 10's of tables and cryptic metrics, highlighting the importance of documentation by our fellow researchers to help us replicate their work.

The tables once loaded to disc could be processed with SQL queries and filtered and processed with more basic operations such as aggregating by group and finding statistics. These would make up our featurized datasets ready for Python analysis.

2.2 LR_PVAL: Private Correlation (via Logistic Regression Coefficient)

This paper[8] seeks to identify whether a certain variable is disproportionately present for a certain outcome. More specifically, it takes a close look at two variables, max temperature on a day and whether a corrected error was present on that day. They would take one of those two variables and train a logistic regression model with maximum likelihood estimation to predict whether an uncorrected error was present. From the model, they use the coefficient of the variable and make a hypothesis test whether that variable is equal to zero.

For our implementation, we focused only on whether there were corrected errors on a day, and not the variable max temperature on a day. We add privacy to the model by using DP-SGD when training the logistic regression model, where the hypothesis test is then private by means of post-processing.

2.3 LASSO Private Regression (via DP-Frank-Wolfe)

will add lots of detail about lasso, then talk about adapting franke-wolfe to be differentially private.

[8] needs citation

2.4 KMEANS Private Clustering (DP-Lloyd's)

K-Means clustering (Lloyd's Algorithm) is applied to group devices based on similarities in their usage patterns. The method leverages Z-scores for standardizing the usage data and calculates L1 distances between weekly usage patterns to identify trends over time. Lloyd's Algorithm clusters devices by assigning them to centroids based on their usage patterns, recalculating the centroids as the mean of assigned points after each iteration.

Differentially Private Lloyd's Algorithm (DP-Lloyd's)[9] modifies the standard K-Means clustering by adding Laplacian noise during the iterative centroid update step to ensure privacy. It introduces noise to both the sum of coordinates and the count of points within clusters, with the amount of noise controlled by the number of iterations and the sensitivity of the data.

2.4.1 Z-score (Additive Noise)

As Z-score is computed before performing K-means clustering,

$$Z = \frac{X - \mu}{\sigma}$$

One can privatize this clustering task by simply adding Laplacian noise to the Z-scores, though the privacy gurantee and performance between the two methods, DP-Lloyd's and Additive Noise are likely to be different.

$$Z_{\text{private}} = \frac{X - \mu}{\sigma} + \text{Lap}\left(\frac{\Delta f}{\epsilon}\right)$$

Δf is the global sensitivity of the Z-score computation,
ϵ is the privacy parameter.

2.5 COND_PROB: Private Conditional Probability Release (Laplace Mechanism)

In order to replicate the paper, were are going to apply differential privacy methods to the processes outlined in the paper. In the paper, the authors first separate each GUID (user) into the number of corrected errors observed during a set time period. They then created a histogram where the x-axis was the number of corrected errors observed and the y-axis was the percentage of GUID's that observed an uncorrected error. This is the process that we are aiming to privatize.

We are releasing a percentage for each bin in the histogram, and in order to guarantee privacy, we must add noise to to both the numerator and denominator where the numerator is the number of GUID's that contained an uncorrected error (number of 1's) and the

[9] linkneeded

denominator is the number of GUID's total. However we can not release the number of GUID's in each corrected error count as that would violate differential privacy so instead we are just going to add noise to number of GUID's that contained an uncorrected error as well as the number of GUID's that did not contain an uncorrected an error (number of 1's + number of 0's) so that we have a private denominator.

$$P(Uncorrected) = \frac{\text{GUID's containing an Uncorrected Error}}{\text{Total GUID's}}$$

We are going to apply the Laplace mechanism to to release this percentage privately. The Laplace mechanism adds noise drawn from the Laplace distribution to the output of a function. B is the scale parameter and ∇f is the sensitivity of the function f.

$$noise \sim Lap(b)$$
$$Lap(x|b) = \frac{1}{2b} e^{-\frac{|x|}{b}}$$
$$b = \frac{\Delta f}{\epsilon}$$

The sensitivity of the function is defined as the maximum possible absolute change in the output of the function due to the change in a single user's data. Since we are dealing with a percentage (and we are considering the worst case), this change can be at most 1 user, which corresponds to a sensitivity of 1 (in terms of the scale of the count, not the percentage itself). This is because, in the worst case, the change is a single user being added or removed, and the total number of GUIDs is assumed to be large enough that the effect of one user's change on the output is not too significant.and ∇f is the privacy parameter.

$$\Delta f = \max_{D,D'} ||f(x) - f(x')|| = 1$$

D and D' are neighboring datasets

2.6 Utility

Each method has different measures of success. However, they can all be roughly translated to a proportion representing utility compared to the best model (whether that is the non-private baseline or better).

SHOULD WE ADD THIS TO EACH METHOD INDIVIDUALLY?

3 Results

3.1 All Tasks: Privacy vs Utility

INSERT COMBINED PLOT

3.2 COND_PROB

3.3 LR_PVAL

3.4 KMEANS

3.5 LASSO

4 Discussion

4.1 Interpretation

4.2 Limitations

5 Conclusion

5.1 Summary

5.2 Impact

5.3 Future Direction

6 Contributions

6.1 Author Contributions

: T.S. focused on task22 LASSO Regression to highlight the exploratory capabilities of private data while implementing a previously theoretical framework (Franke-Wolfe). C.L. implemented the algorithms in ... B.N. analyzed the experimental results ... T.K. analyzed the experimental results ... Y.W. supervised the research and provided guidance on the mathematical foundations. All authors contributed to writing and reviewing the manuscript.

6.2 Task Details

Trey Scheid

- Replication of
- Implementation of non-private franke-wolfe lasso regression
- Ethics considerations webpage
  Todo: Implementation of private franke-wolfe lasso regression

Tyler Kurpanek

Bradley Nathanson

Christopher Lum

Yu-Xiang Wang

- Concept ideation
- Data Access
- Provided guidance on the mathematical foundations
- Proofing and editing all content

6.3 Acknowledgements

We would like to recognize the support of our instructor, Yu-Xiang Wang, for his guidance and feedback throughout the project. We would also like to thank the teaching staff Umesh Bellur and Shriniwas Kulkarni for their support and feedback. The tasks database was a foundational part of our work and was created by another student researcher: Qiyu Li.

We also would like to thank the authors of the papers we referenced in our literature review. Their work was instrumental in our understanding of the topic and the development of our project. Our understandings of differential privacy has been built on the work of many researchers in the field such as: __, __, __, and __. Especially those which engaged in discussion with us about the field (Smith, Ulman, Guatam et al.). We are grateful for their contributions.

References

Bonawitz, Kallista, Peter Kairouz, Brendan Mcmahan, and Daniel Ramage. 2022. "Federated learning and privacy." *Commun. ACM* 65 (4), p. 90–97. [Link]

Cheon, Seung Hyun. Unpublished. "Power Consumption Patterns in Intel's Telemetry Data: China Burns 2x Energy that of the US."

Dinur, Irit, and Kobbi Nissim. 2003. "Revealing information while preserving privacy." In *Proceedings of the Twenty-Second ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. New York, NY, USA Association for Computing Machinery. [Link]

Dwork, Cynthia, and Aaron Roth. 2014. "The Algorithmic Foundations of Differential

Privacy." *Foundations and Trends® in Theoretical Computer Science* 9 (3–4): 211–407. [Link]

Gadotti, Andrea, Luc Rocher, Florimond Houssiau, Ana-Maria Creţu, and Yves-Alexandre de Montjoye. 2024. "Anonymization: The imperfect science of using data while preserving privacy." *Science Advances* 10 (29), p. eadn7053. [Link]

Kwasnick, Robert. Unpublished. "Exploration of CPU Error Dependencies and Prediction."

Pew Research Center. 2019. "Americans and Privacy: Concerned, Confused and Feeling Lack of Control Over Their Personal Information." report, Pew Research Center. [Link]

Ponomareva, Natalia, Hussein Hazimeh, Alex Kurakin, Zheng Xu, Carson Denison, H. Brendan McMahan, Sergei Vassilvitskii, Steve Chien, and Abhradeep Guha Thakurta. 2023. "How to DP-fy ML: A Practical Guide to Machine Learning with Differential Privacy." *Journal of Artificial Intelligence Research* 77, p. 1113–1201. [Link]

Ryan, Jacob M., Shuangquan Feng, Marie McCusker, Benjamin L Smarr, Rayan Saab, and Virginia de Sa. Unpublished. "PC Health Impact White Paper."

Scheid, Trey, Tyler Kurpanek, Christopher Lum, Bradley Nathanson, and Yu-Xiang Wang. 2024. "Exploring Tradeoffs in Differential Privacy: An Empirical Study of Logistic Regression on Telemetry Data." [Link]

Su, Fei, Robert Kwasnick, John Holm, William Penner, Hermann Gartler, Josh Boelter, Yufei Zhou, Bijan Arbab, and Michael Rothberg. 2024. "Product Health Insights Using Telemetry." *IEEE Design Test* 41 (4): 56–64. [Link]

Appendices

A.1 Project Proposal

A.1.1 Problem Statement

Telemetry data is important to privatize as it encodes personally identifiable information which could be used to discover sensitive information. This data is collected from various IT devices, from satellites to personal computers. For our project, the telemetry data includes hardware and software performance metrics, monitoring, and errors.

We will privatize 22 analysis tasks for the Intel telemetry dataset, ensuring a reasonable privacy budget (). We will implement mechanisms that balance data utility and privacy, ensuring sensitive information is protected, and allocate a reasonable privacy budget (), a parameter that governs the trade-off between accuracy and privacy.

One example of a task is to predict CPU failure. This would require a privatized logistic regression model that predicts the probability of a failure from 0-1. The model would analyze data such as CPU temperature, usage patterns, error logs, or other performance indicators. If non-privatized, this model could expose this data, as a malicious individual could do a reconstruction attack, a method to reconstruct the training data by repeatedly querying the model with various synthetic inputs. The attacker could query this model with different sets of CPU-related inputs, and, over time, the attacker could gain information such as the CPU temperature threshold for an error to occur, or whether certain system configurations have a distinct failure pattern.

A.1.2 Methods

Our methodology for privatizing the 22 telemetry analysis tasks will employ multiple privacy mechanisms, such as the exponential mechanism and the Laplace mechanism, with AutoDP serving as our core privacy accounting tool. For each analysis task, we will first evaluate the sensitivity of the computation and determine the optimal privacy mechanism to maintain utility while satisfying privacy requirements. The implementation process re-

quires careful privacy budget allocation across multiple components of each analysis to ensure the total privacy loss remains within acceptable bounds.

The evaluation of each privatized implementation will involve a comprehensive comparison with non-private baselines to document the privacy-utility tradeoff. This includes analyzing performance metrics before and after applying privacy mechanisms, measuring accuracy degradation at various privacy budget levels, and considering computational efficiency challenges specific to telemetry data analysis. AutoDP will help quantify the privacy guarantees and guide the noise calibration process throughout implementation.

Each privatized task will be thoroughly documented with implementation details, privacy guarantees, and performance metrics. This documentation will include privacy budget allocation strategies, noise mechanism selection rationale, and practical guidelines for future implementations. The goal is to create a comprehensive resource demonstrating how different privacy mechanisms can be effectively applied to various telemetry analysis scenarios while maintaining practical utility and ensuring strong privacy protections.

A.1.3 Deliverable

The privatized analysis tasks will be stored and shared in a public repository, (without release of source data from Intel). This is our primary contribution, to offer tools in a privatized manner. In collaboration with the accessible programs, we will publish a website that will serve to educate our peers on differential privacy. The variety of analysis tasks done in the telemetry domain can be generalized and applied to many types of data; therefore, descriptions of privacy algorithms, their motivations, and limitations can teach practitioners new methods for their own tasks.

The Intel data as mentioned is not public (due to the customer privacy and proprietary nature). Therefore our data processing, tasks, and report will include only some metrics of performance and data quality (size, distribution, features, etc). For the information we can share, we will compare the performance of the task with that of the non-private baseline. This gives analysts a sense of the utility-privacy tradeoff in each application.

A.1.4 Impact

By implementing differential privacy across telemetry we will create a significant impact by maintaining data confidentiality. This project will establish novel approaches to common tasks enabling hardware manufacturers to analyze system performance data while preserving strong privacy guarantees. This advances the field by demonstrating how to maintain data utility while protecting sensitive information in real-world applications.

The research contribution includes documenting privacy-utility trade-offs and establish-

ing guidelines for privacy budget allocation across multiple analysis tasks. Our work will demonstrate practical privacy considerations in telemetry analysis while protecting users' participation in datasets. The methodologies developed can be adapted by other researchers working with sensitive telemetry data.

A.1.5 Success Criteria

The success of this project is dependent on a few factors. The first two are team collaboration and schedule adherence. There are many tasks that can be privatized and there may be unique challenges for each (hence the value in sharing these!). With one-quarter complete with group work on our privatized logistic regression paper, our group is confident in our communication, task management, and problem-solving abilities. Paired with our mentor Yu-Xiang Wang, an expert in the field of differential privacy, and a seasoned professor, we are equipped to find innovative and theoretically founded methods for privatizing data tasks.

The other requirements for this project rely on data access and task availability. The Intel data is proprietary, and we have signed agreements to use the data for research, however strict access and usage terms have not been given to us yet. Previous students have worked with the contact/program at Intel successfully and we are reassured by them that we will have a usable telemetry dataset by the start of the quarter. Similarly, there is a set of non-privatized tasks completed on this dataset by previous data scientists, their work is the foundation which we will build off of to show utility is possible even with privacy. These projects were successful implementations on the specific dataset we will have access to, this pairing therefore will continue to bear fruit as we privatize the tasks and compare baselines.

Lastly, although we have not reviewed the dataset and tasks yet (no access), the intel program is sharing genuine telemetry information from devices with given consent as part of their program. Additionally, this HDSI-Intel partnership has been cooperating since 2020 and HDSI has used hundreds of terabytes of information.

A.2 Additional Results

example

A.3 Training Details

example

A.4 Additional Figures