Google DP vs. OpenDP: Empirical Comparison of Differential Privacy Libraries

Stilyan Penchev
Supervisor(s): Dr. Zeki Erkin, Dr. Roland Kromes
[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 22, 2025

Name of the student: Stilyan Penchev Final project course: CSE3000 Research Project
Thesis committee: Dr. Zeki Erkin, Dr. Roland Kromes, Dr. Xucong Zhang

An electronic version of this thesis is available at http://repository.tudelft.nl/.

Abstract

This research looks at two open-source tools for differential privacy: Google's Differential Privacy Library and the OpenDP Library. The main aim of this study is to test them side-by-side and observe how they compared quantitatively. Specifically, the focus is on their implementations of the Laplace noise mechanism. It measured their computational performance, scalability and the accuracy of their results (utility). These experiments were done for basic Count and Sum queries on synthetic structured datasets, using different privacy settings (ϵ) and dataset sizes. The experiments using the Laplace mechanism showed some clear differences. GoogleDP was consistently faster than OpenDP for both Count and Sum queries and this speed advantage was especially noticeable with larger datasets. The choice of ϵ did not really change how long either library took to run. When it came to utility, both libraries showed the expected pattern: more privacy (lower ϵ) meant less accurate results. The accuracy differences between the two were small. For Sum queries OpenDP gave results with slightly better utility, especially on large datasets and when ϵ was small (meaning more privacy). This paper offers current, data-supported information about the practical pros and cons when using these libraries for these common DP operations with the Laplace mechanism. A full comparison including the Gaussian mechanism could not be completed in this study due to the fact that PyDP (GoogleDP's python wrapper) had only the Laplace mechanism implemented. This remains an important task for future work.

1 Introduction

The modern world runs on data. Analyzing large, sensitive collections of information (from health records and financial details to how people behave online) offers great promise for improving society and the economy in areas like medical discovery, city planning, and personalized services [8]. But there's a major hurdle: this data frequently includes personal details (PII) or other private information. If this information gets out or is not handled carefully, serious privacy breaches can occur. Just taking out names is not enough, people can often still be identified by linking supposedly anonymous data with other public information, or private facts can be figured out, as research on re-identification has shown [14, 9]. Getting useful knowledge from data while also protecting individual and organizational privacy is a balancing act. Solving this problem is more than just a technical issue, it's vital for a few reasons. It's a core part of using data responsibly, making sure to follow tough laws and rules like Europe's GDPR [4] or HIPAA in the United States [15]. It's also about doing the right thing ethically by protecting the people whose data is involved, which helps make sure the public is still willing to share information for good causes [1]. In the end, keeping this balance is essential for building and keeping public faith in data-reliant systems. If that trust is lost, it can seriously harm innovation and how well society accepts data-heavy technologies [13].

To tackle these privacy challenges, Differential Privacy (DP) [2] has become a key method. It provides a strong, mathematically proven way to guarantee privacy when data is analyzed. To put DP into practice, various software libraries have been created. This study focuses on two notable open-source options: Google's Differential Privacy Library [6] and the OpenDP Library [11].

Both of these libraries provide the essential tools for DP, notably the Laplace and Gaussian

mechanisms, which work by adding carefully controlled noise to query results to keep individual data safe. Even though they both try to offer similar levels of privacy protection, the way they are built internally, their underlying design ideas, and how they perform in terms of speed and handling larger datasets can be quite different. For people working with data and for researchers, figuring out these practical differences is important when choosing the best tool for a specific job. Because these libraries are always being updated, a fresh empirical comparison is especially useful.

This paper seeks to provide a quantitative comparison of these two libraries. Specifically, the research question addressed is:

How do the Google Differential Privacy Library and the OpenDP Library compare in terms of computational performance, scalability, and the utility of their implemented Laplace and Gaussian mechanisms when applied to Count and Sum queries on structured datasets?

To fully address this main question, the following sub-questions are investigated:

1. What are appropriate and measurable quantitative metrics for evaluating computational performance, scalability and utility loss?
2. Based on empirical experiments, how do the Google DP and OpenDP libraries quantitatively compare in terms of computational time and memory usage for Count and Sum queries with Laplace and Gaussian noise across a range of ϵ and δ values?
3. How does the computational performance of each library's Laplace and Gaussian mechanism implementations for Count and Sum queries scale as the size of the input dataset increases?
4. How does the utility (accuracy) of the results for Count and Sum queries compare between the Google DP and OpenDP libraries when applying their respective Laplace and Gaussian mechanisms across a range of ϵ and δ values?

The primary contribution of this research is a quantitative comparison of the Google Differential Privacy Library and the OpenDP Library focusing on computational performance, scalability and utility. GoogleDP was consistently faster than OpenDP for both Count and Sum queries, and it handled larger datasets more efficiently. The specific privacy setting (ϵ) did not make much difference to how fast either library ran. Both libraries showed the expected trade-off: more privacy meant less accurate results. For Count queries the accuracy was pretty similar between the two. For Sum queries OpenDP sometimes had a slight edge in accuracy, especially with large datasets when using strong privacy settings.

The remainder of this paper is structured as follows. Section 2 provides necessary background on Differential Privacy, the Laplace and Gaussian mechanisms and the software libraries under investigation. Section 3 details the experimental methodology used for the empirical comparison, including the setup, datasets, privacy parameters, and evaluation metrics. Section 4 presents the comprehensive empirical results from our experiments and covers the comparative analysis of computational performance, scalability, and utility for both libraries and mechanisms. Section 5 discusses responsible research. Section 6 provides a discussion of the findings, including interpretation, implications and limitations. Section 7 concludes the paper and suggests avenues for future work.

2 Background

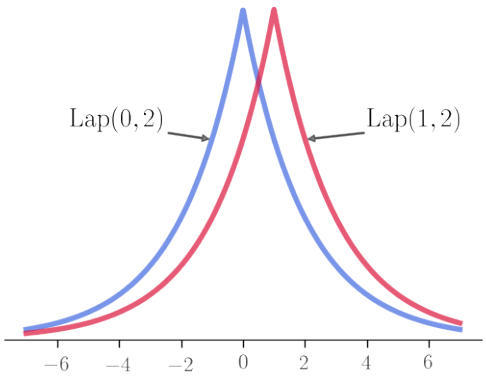
This section provides an overview of Differential Privacy (DP), the core noise-adding mechanisms relevant to this study (Laplace and Gaussian) and introduces the two software libraries under comparison: Google's Differential Privacy Library and the OpenDP Library. It also looks into existing comparative work to show where this research fits in.

2.1 Differential Privacy

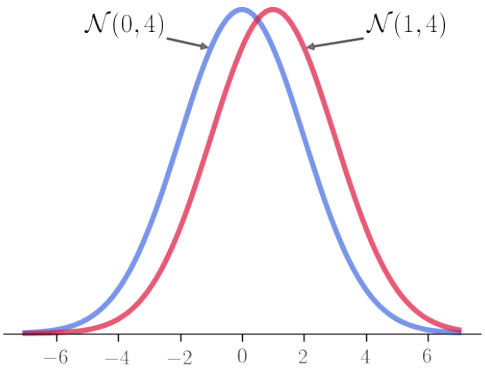
Differential Privacy (DP) has become a widely accepted standard for analyzing data while protecting privacy. It offers a mathematically solid way to prevent individuals from being re-identified within a dataset. The basic concept behind DP is that the result of any data analysis should look almost the same, whether or not one specific person's information is part of the original data. This guarantee is formally defined using two parameters: epsilon (ϵ) and delta (δ). Epsilon (often referred to as the privacy budget or privacy loss) quantifies the maximum privacy leakage allowed. A smaller ϵ implies stronger privacy. Delta (δ) represents the probability that the pure ϵ-privacy guarantee is broken; for (ϵ, 0)-DP (often called pure DP), δ is zero, while for (ϵ, δ)-DP (approximate DP), δ is a small positive value, typically much smaller than the inverse of the dataset size. The work by Dwork et al. (2006) laid the groundwork for calibrating noise to data sensitivity to achieve these guarantees [2]. Achieving DP typically involves adding calibrated random noise to the true result of a query. The amount of noise added is determined by the sensitivity of the query, how much the query's output can change due to the presence or absence of a single individual's data and the desired privacy parameters (ϵ, δ).

2.2 Noise Mechanisms: Laplace and Gaussian

The two main mechanisms for adding noise to numerical query results are the Laplace mechanism and the Gaussian mechanism. Both are central to this study [3].



(a) The Laplace Mechanism



(b) The Gaussian Mechanism

- **Laplace Mechanism:** The Laplace mechanism adds noise drawn from a Laplace distribution centered at zero. It is typically used to achieve (ϵ, 0)-differential privacy. The scale of the Laplace noise (related to its variance) is proportional to the L1 global

sensitivity of the query divided by ϵ. The L1 global sensitivity measures the maximum possible change in the L1 norm of the query output when one individual's data is modified.

- **Gaussian Mechanism:** The Gaussian mechanism adds noise drawn from a Gaussian (normal) distribution centered at zero. It is typically used to achieve (ϵ, δ)-differential privacy, where $\delta > 0$. The scale of the Gaussian noise (its standard deviation) is proportional to the L2 global sensitivity of the query and is related to both ϵ and δ. The L2 global sensitivity measures the maximum possible change in the L2 norm of the query output. The Gaussian mechanism is often preferred in scenarios allowing for approximate DP due to tighter composition properties and sometimes better utility for a given ϵ when δ is small.

2.3 Google Differential Privacy Library

The Google Differential Privacy Library, developed by Google [6], provides tools for implementing differentially private aggregations. While its core is implemented in C++, it is made accessible in Python through the PyDP wrapper (which is used in this study), and also offers interfaces in Go and Java. The library aims to provide building blocks that data practitioners can use to add differential privacy to their statistical analyses. It offers implementations for various common aggregate functions like count, sum, mean, variance, and percentiles, with support for both Laplace and Gaussian noise mechanisms. A central feature is its emphasis on bounded aggregations, where explicit lower and upper bounds for individual contributions are used to calculate global sensitivity, which is crucial for calibrating the appropriate amount of noise.

2.4 OpenDP Library

The OpenDP Library is a community-driven, open-source project initiated by Harvard University in collaboration with Microsoft and other contributors [11]. It aims to provide a trustworthy and extensible suite of tools for differential privacy. The system and its design principles were introduced by Gaboardi et al. (2020) [5]. The library is designed with a focus on correctness and usability, enabling users to construct differentially private computations from a set of core building blocks (transformations and measurements). OpenDP supports a variety of privacy mechanisms, including Laplace and Gaussian, and offers a framework for defining privacy mappings and composing private computations. The Python interface [12] is primarily used for its application.

2.5 Existing Comparative Studies and Motivation for Current Work

Google DP and OpenDP both try to simplify the use of differential privacy in practice. But how they are built, how their APIs are designed, and as a result, how they really perform in terms of speed and accuracy can differ. A few studies have already tried to compare various DP libraries. For example, Zhang et al. (2023) looked into Google DP and what was then OpenDP's SmartNoise component (which is now part of the main OpenDP system) [16]. The libraries were tested on several types of statistical queries like SUM, AVERAGE, COUNT, and HISTOGRAM. Their checks included data utility, how long operations took (runtime overhead), and memory use for different ϵ values and dataset sizes. Zhang et al. found that Google DP seemed promising for general statistics, while OpenDP's SmartNoise

did well for HISTOGRAM queries [16]. This current research builds on this existing work. It offers an updated, focused comparison by testing the current versions of the Google Differential Privacy Library and the OpenDP Library. Since these libraries are being updated frequently, having up-to-date benchmarks is important. This study focuses in on two of the most basic types of aggregate queries (Count and Sum). It provides a comparison of how the Laplace and Gaussian noise mechanisms are implemented in each library for these specific queries. By running controlled experiments that measure computational performance, scalability, and utility with exact metrics, this paper offers current, data-supported information about the practical advantages and disadvantages of using the two libraries for common DP operations.

3 Methodology

This section describes in detail the methodology used to compare the Google Differential Privacy Library and the OpenDP Library. The study focuses on their implementations of the Laplace and Gaussian mechanisms as applied to Count and Sum queries on structured datasets. The methodology covers the experimental setup, dataset characteristics, specific library configurations, privacy parameters tested and the metrics used for evaluating computational performance, scalability, and utility.

3.1 Experimental Setup and Libraries

All experiments were done on the same hardware environment: HP ZBook Power G10 laptop with an Intel Core i7-13700H CPU, 16GB of RAM and running Windows 11. This ensures that performance comparisons are attributable to library differences rather than hardware.

The following library versions were used for this study:

- **Google Differential Privacy Library (Python):** PyDP Version 1.1.4
- **OpenDP Library (Python):** Version 0.13.0

Python (Version 3.9.18) was used as the primary language for scripting the experiments and interacting with both libraries. Python libraries used include pandas for data manipulation, numpy for numerical operations, time for performance measurement and matplotlib with seaborn for generating visualizations.

3.2 Datasets

To evaluate performance, scalability, and utility, synthetic structured datasets were generated, a common practice in the empirical evaluation of differential privacy mechanisms to allow for controlled experiments and varying data characteristics [7]. Each dataset consists of a single numerical column, used for the Sum query, and implicitly for the Count query (by counting rows). Values were generated as floating point numbers uniformly chosen from the range [-100.0, 100.0]. This range defines the bounds for calculating the sensitivity of the Sum query. The datasets were generated prior to the experiments and loaded into memory for processing by each library.

Three dataset sizes were used to assess scalability: **Small:** [10,000 records], **Medium:** [100,000 records], **Large:** [1,000,000 records]

3.3 Differentially Private Queries and Mechanisms

The study focuses on two fundamental aggregate queries:

- **Count Query:** Computes the total number of records in the dataset. For DP-Count, the L1 global sensitivity ($GS_1 = 1$) is 1, as adding or removing one record changes the count by one. The L2 global sensitivity ($GS_2 = 1$) is also 1.
- **Bounded Sum Query:** Computes the sum of all values in the numerical column. To ensure known sensitivity for DP-Sum, contributions from each record were effectively bounded between $L = -100.0$ (lower bound) and $U = 100.0$ (upper bound).
  - For the **Laplace mechanism**, which uses L1 sensitivity, the sensitivity of a single bounded value contributing to the sum is $\max(|L|, |U|)$. In this setup, this is $\max(|-100.0|, |100.0|) = 100.0$. The DP libraries' sum transformations handle the aggregation of these sensitivities internally.
  - For the **Gaussian mechanism**, which uses L2 sensitivity, the sensitivity of a single bounded value contributing to the sum is $U - L$. In this setup, this is $100.0 - (-100.0) = 200.0$. Again, the DP libraries' sum transformations manage the aggregation of these sensitivities.

For each query, both the Laplace mechanism and the Gaussian mechanism were applied using the respective functionalities of each library.

3.4 Privacy Parameters (ϵ)

A range of privacy parameters was selected to evaluate the libraries under different privacy constraints:

- **Epsilon (ϵ):** Values of 0.1, 0.5, 1.0, 1.5, 2.0, 2.5, and 3.0 were tested. These represent a spectrum from strong to moderate privacy guarantees.
- **Delta (δ):** For the Gaussian mechanism, a fixed value of $\delta = 10^{-5}$ will be used for tests varying ϵ. Additionally, for a fixed ϵ, δ will be varied with values like 10^{-3}, 10^{-5}, 10^{-7} to observe its impact. For the Laplace mechanism, δ is implicitly 0.

3.5 Metrics for Evaluation

The comparison between the libraries was based on the following quantitative metrics, addressing the sub-questions:

1. **Computational Performance (Execution Time):** Measured as the average elapsed real time taken to execute the differentially private query (Count or Sum, with Laplace or Gaussian noise) over 100 independent runs for each combination of library, task, mechanism, privacy parameters, and dataset size.
2. **Scalability:** Assessed by observing how the average execution time and utility metrics (MAE) change as the dataset size increases (from Small to Medium to Large) for fixed tasks, mechanisms and selected privacy parameters.

3. **Utility (Accuracy Loss):** For each noisy query result the error was calculated relative to the true non-private result. Calculated as the mean of the absolute differences between the 100 noisy results and the true value for each parameter setting:

$$\text{MAE} = \frac{1}{N_{\text{runs}}} \sum_{i=1}^{N_{\text{runs}}} |\text{NoisyResult}_i - \text{TrueValue}| \tag{1}$$

MAE was selected for its straightforward interpretability as the average magnitude of error. This was used for both Count and Sum queries.

4 Empirical Results

This section presents the empirical results from comparing the Google Differential Privacy Library and the OpenDP Library using the Laplace mechanism for Count and Sum queries. The analysis covers computational performance versus privacy budget (ϵ), scalability of performance with dataset size, utility versus privacy budget, and scalability of utility with dataset size.

4.1 Computational Performance vs. Epsilon

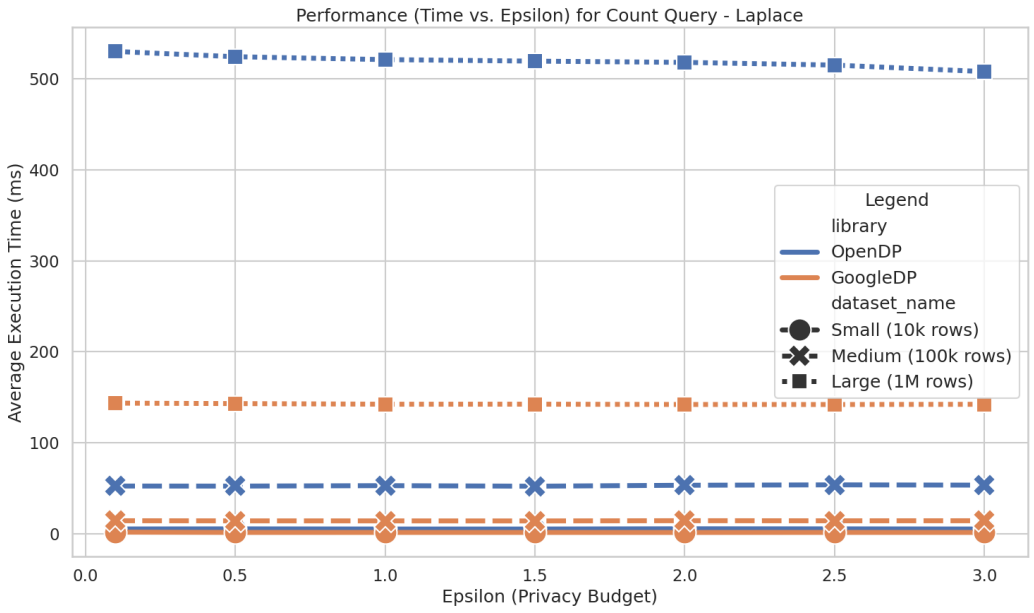


Figure 2: Performance (Time vs. Epsilon) for Count

The average execution time for performing differentially private Count and Sum queries using the Laplace mechanism was evaluated across a range of epsilon values ($\epsilon \in [0.1, 3.0]$) for Small, Medium and Large datasets.

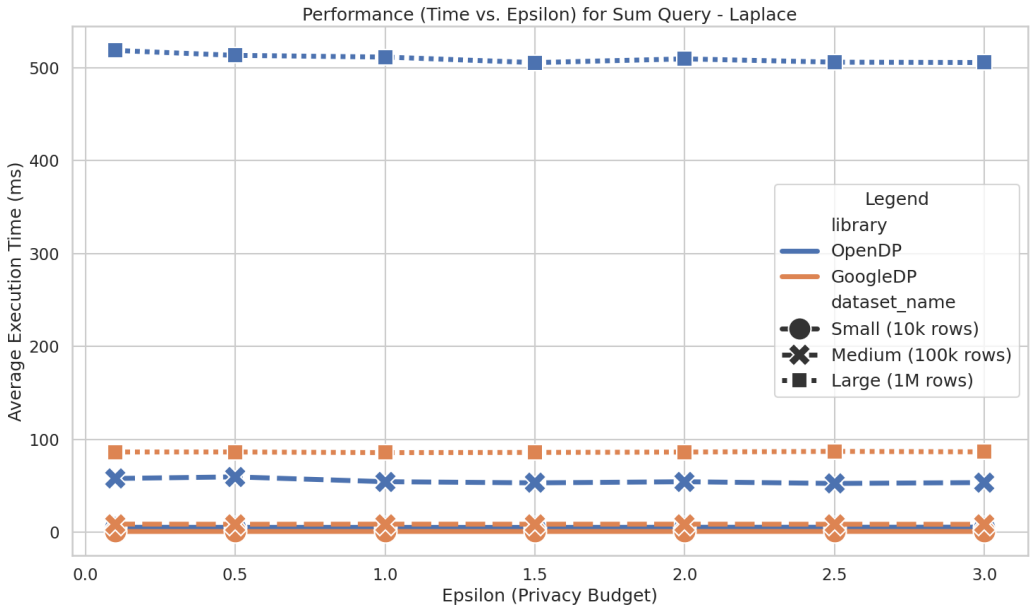


Figure 3: Performance (Time vs. Epsilon) for Sum

As illustrated in Figure 2 and Figure 3, the execution time for both libraries appears largely insensitive to changes in the epsilon value for a given dataset size and query type. This shows that the primary computational cost is associated with the data processing and noise application itself, rather than varying the scale parameter of the Laplace distribution based on epsilon, once the mechanism is constructed.

Comparing the libraries, GoogleDP consistently shows lower average execution times than OpenDP across all dataset sizes for both Count and Sum queries. For the Small dataset the performance difference is small, with both libraries executing queries under 10ms. For the Large dataset the performance gap is significant. OpenDP's execution time for the Large dataset is substantially higher (around 500-550ms) compared to GoogleDP (around 140-170ms) for both Count and Sum queries. The specific query type (Count vs. Sum) did not drastically alter these observed performance trends relative to epsilon or between libraries for a fixed dataset size.

4.2 Scalability of Performance

The scalability of computational performance was assessed by measuring the average execution time as the dataset size increased from Small to Medium to Large, for epsilon values (0.1, 1.0, 3.0). These results are presented on a log-log scale in Figure 4 and Figure 5. Both libraries show an increase in execution time with increasing dataset size, as its was expected. On the log-log scale, the performance trends for both libraries appear approximately linear which suggest polynomial scaling with dataset size.

GoogleDP demonstrates superior performance scalability compared to OpenDP. For both

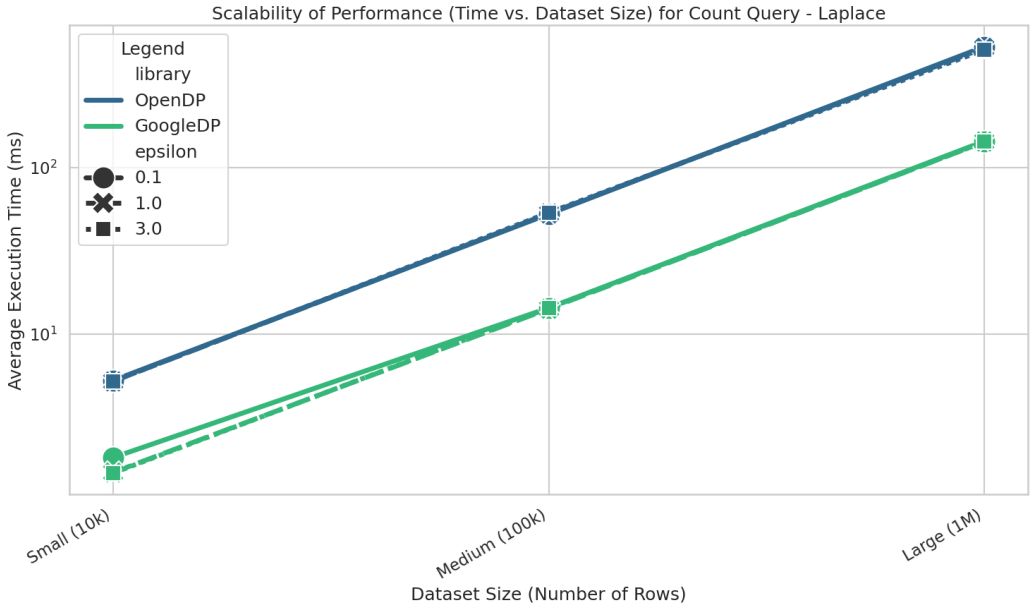


Figure 4: Scalability of Performance for Count

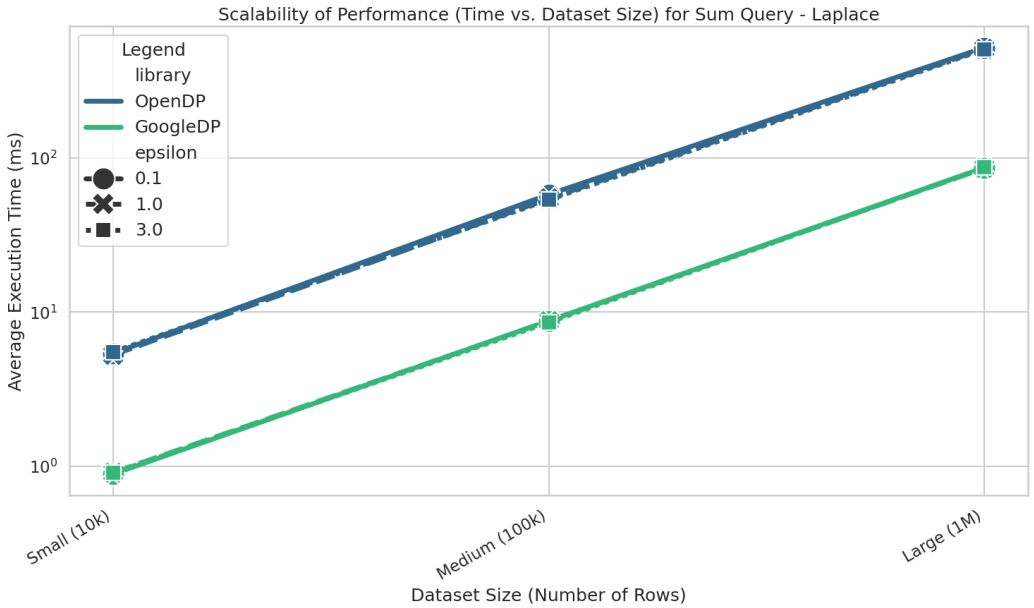


Figure 5: Scalability of Performance for Sum

Count and Sum queries GoogleDP's execution times are significantly lower than OpenDP's across all tested dataset sizes and epsilon values. The lines representing different epsilon values for each library are clustered closely together which confirms that epsilon has a minimal impact on execution time for these Laplace implementations. The performance advantage of GoogleDP becomes clearer at larger dataset sizes. For instance, in the Count query for the Large dataset, GoogleDP's time is around 10^2 ms, while OpenDP's is closer to $5{\times}10^2$, ms. Similar relative differences are observed for the Sum query.

4.3 Utility

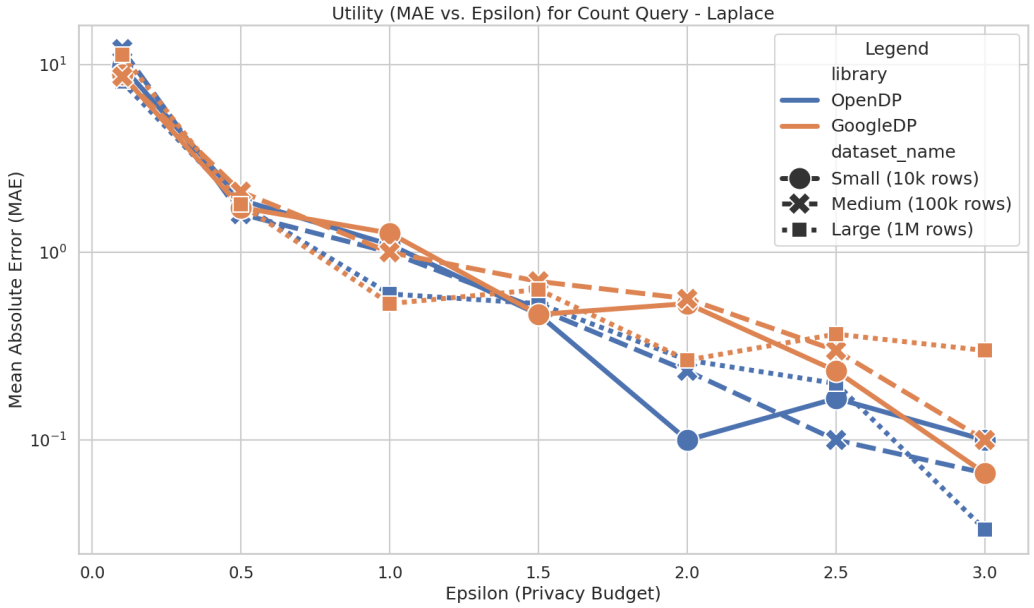


Figure 6: Utility for Count

The utility of the differentially private queries was evaluated by calculating the Mean Absolute Error (MAE) between the noisy results and the true values, across the range of epsilon values for each dataset size. Figure 6 and Figure 7 illustrate these findings, with MAE presented on a logarithmic scale.

As theoretically expected, for both Count and Sum queries, and across all dataset sizes and both libraries, the MAE generally decreases as epsilon (ϵ) increases. This demonstrates the fundamental privacy-utility trade-off: higher privacy (lower ϵ) results in more noise and thus higher error, while lower privacy (higher ϵ) allows for less noise and better accuracy. The decrease in MAE is typically steeper for smaller ϵ values and tends to flatten as ϵ grows.

For the Count query (Figure 6), library performance is nuanced. For Small and Medium datasets, GoogleDP often shows slightly lower or comparable MAE to OpenDP. For the Large dataset, OpenDP sometimes exhibits marginally lower MAE than GoogleDP, particularly at lower ϵ values (e.g., $\epsilon < 1.0$). At higher ϵ values ($\epsilon \geq 1.5$), the utility of both

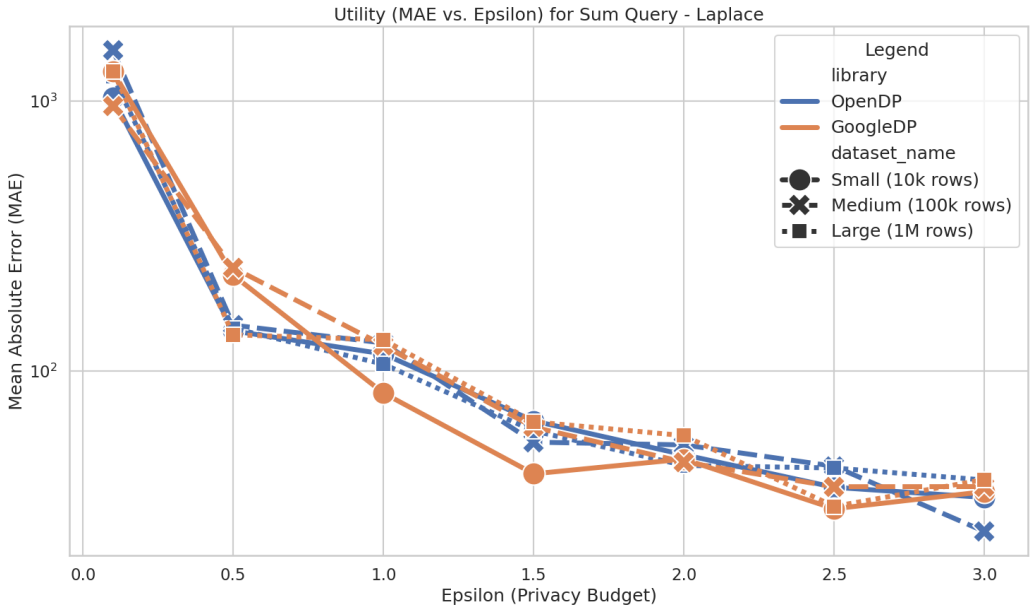


Figure 7: Utility for Sum

libraries becomes very similar for the Count query across all dataset sizes.

For the Sum query (Figure 7), which inherently has a larger sensitivity and thus higher error magnitudes, similar trends are observed. OpenDP tends to provide lower MAE for the Large dataset, especially at smaller ϵ values. For Small and Medium datasets, and at higher ϵ values for the Large dataset, the utility of both libraries is more comparable, though GoogleDP sometimes shows slightly higher MAE. The MAE for Sum queries is, as expected, orders of magnitude larger than for Count queries given the same privacy parameters and dataset size.

4.4 Scalability of Utility

The scalability of utility was examined by observing the MAE as dataset size increases, for epsilon (ϵ) values (0.1, 1.0, 3.0), as shown in Figure 8 (Scalability of Utility MAE vs. Dataset Size for Count Query - Laplace) and Figure 9 (Scalability of Utility MAE vs. Dataset Size for Sum Query - Laplace).

For the Count query (Figure 8), at higher privacy levels ($\epsilon = 1.0$ and $\epsilon = 3.0$), the MAE remains relatively low and stable, or even slightly decreases, for both libraries as the dataset size increases from Small to Large. This suggests that for a fixed, reasonable privacy budget, the absolute error of the DP Count does not significantly degrade with larger datasets. At the lowest privacy level ($\epsilon = 0.1$), the MAE is higher. For OpenDP with $\epsilon = 0.1$, MAE increases with dataset size. For GoogleDP with $\epsilon = 0.1$, MAE shows a non-monotonic behavior, decreasing from Small to Medium, then increasing from Medium to Large.

For the Sum query (Figure 9), the behavior is more complex. At $\epsilon = 0.1$, the MAE is substantial. For OpenDP, the MAE for Sum decreases as dataset size increases. For

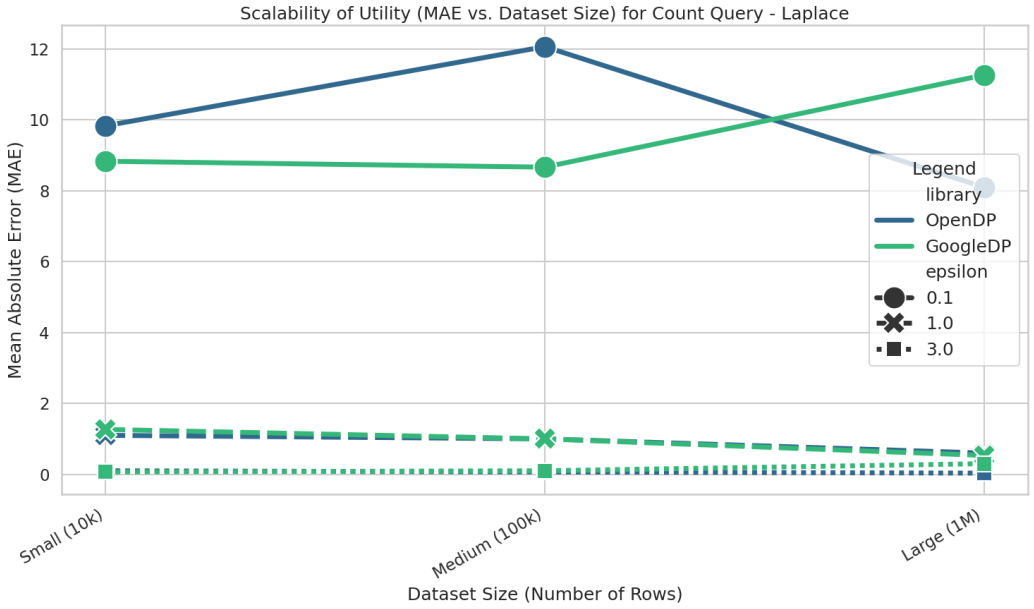


Figure 8: Scalability of Utility for Count

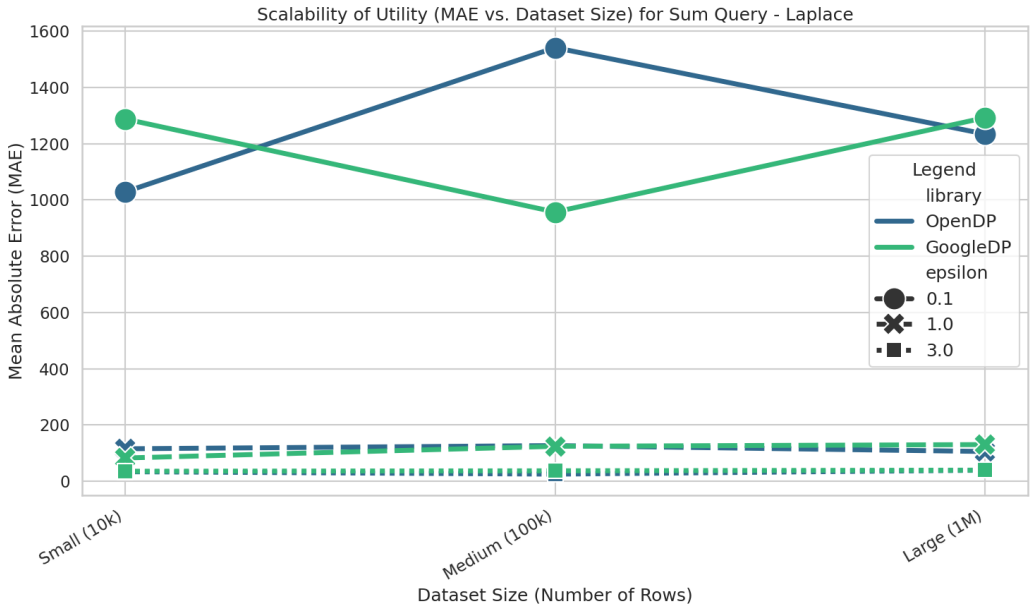


Figure 9: Scalability of Utility for Sum

GoogleDP at $\epsilon = 0.1$, MAE decreases from Small to Medium, then increases for the Large dataset. At higher privacy levels ($\epsilon = 1.0$ and $\epsilon = 3.0$), the MAE is much lower. For both libraries, the MAE for Sum at these higher ϵ values remains relatively stable or shows a slight increase as dataset size grows. OpenDP generally exhibits lower or comparable MAE to GoogleDP for the Sum query across dataset sizes, particularly at $\epsilon = 0.1$.

5 Responsible Research

This research, focused on the empirical comparison of differential privacy libraries, adheres to principles of responsible conduct, primarily concerning ethical data handling (even with synthetic data), transparency in methodology, and the reproducibility of its findings.

5.1 Ethical Considerations

The core subject of this study (differential privacy) is itself a technique designed to address ethical concerns surrounding data privacy. By evaluating tools that enable privacy-preserving data analysis, this research aims to contribute positively to the responsible use of data.

Although this study exclusively utilized synthetically generated datasets, thereby avoiding any direct handling of real individuals' personally identifiable information (PII) or sensitive attributes, the principles guiding the research are rooted in the ethical imperative to protect privacy and minimize risk to human subjects even when such risk is indirect [10]. The choice of synthetic data was deliberate to ensure that no actual privacy was at risk during the experimentation phase, allowing for a focused evaluation of the libraries' mechanisms without the ethical complexities of managing real sensitive data.

The findings of this paper are intended to help practitioners and researchers make informed decisions about which differential privacy library might best suit their needs. This, in turn, can lead to more robust and correct applications of differential privacy, ultimately better protecting individuals whose data might be analyzed using these tools in real-world scenarios. Transparency in reporting the performance and utility characteristics of these libraries, including their limitations, is an ethical obligation to the wider research and practitioner community.

5.2 Reproducibility

The reproducibility of this study's findings is supported by a transparent methodology and the use of publicly available tools and well-defined procedures. All elements ensuring reproducibility are detailed in Section 3

By providing clear details on the software, data, and experimental procedures, this study aims to allow for independent verification and replication of its results. The code is also available on GitHub at https://github.com/StiliPench/ResearchProject.

5.3 LLMs

Large Language Models (LLMs) were used as a helper for LaTeX formatting and writing refinement. A prompt of "How to reference this figure" and "How do I avoid a bigger space after a period" were used. For writing refinement a prompt of "How would you say this sentence in a more scientifically consistent way" was used to gain insight.

6 Discussion

This section interprets the empirical findings from Section 4 which compared the Google Differential Privacy Library and OpenDP Library for Count and Sum queries using the Laplace mechanism. These results are compared against initial expectations and existing research. Potential explanations for the differences between the libraries are offered and the study's limitations are acknowledged.

6.1 Interpretation and Comparison with Prior Work

The results for the Laplace mechanism consistently showed GoogleDP outperforming OpenDP in computational speed for both Count and Sum queries across all dataset sizes with the advantage being higher for larger datasets. Execution times for both libraries were largely insensitive to different ϵ values. In terms of scalability both libraries showed the expected increases in execution time with dataset size, but GoogleDP scaled better. These performance findings match the initial expectations regarding Google's optimization capabilities. The results are consistent with Zhang et al. (2023) [16] who observed GoogleDP's efficiency for general statistics with earlier library versions. This study provides an updated benchmark confirming this trend for Laplace Count and Sum.

6.2 Potential Explanations for Observed Differences

Several factors could contribute to the observed differences in performance and utility between GoogleDP and OpenDP for the Laplace mechanism:

- **Implementation and Optimization:** GoogleDP's core is implemented in C++ which is highly optimized for performance. While OpenDP also uses efficient language like Rust for its core, differences in algorithmic choices, memory management, or specific C++/Rust to Python binding overheads could lead to performance differences. PyDP acts as a wrapper and the efficiency of these bindings also plays a role.
- **Algorithmic Choices within Mechanisms:** Even for a standard mechanism like Laplace there can be small differences in how random number generation is handled, how sensitivities are precisely applied, or how floating-point arithmetic is managed, which might marginally affect both speed and the exact noise distribution, as a result influencing utility.
- **Library Architecture:** Different architectural choices might introduce performance overhead. Differences in performance could stem from GoogleDP's more direct optimization paths within its PyDP wrapper, compared to OpenDP's composable architecture which might introduce some overhead.

A definitive explanation would require a detailed code level investigation of both libraries, which is beyond the scope of this empirical study.

6.3 Limitations and Future Work Considerations

This study has several limitations. The main one is its confinement to the Laplace mechanism. The planned inclusion of the Gaussian mechanism was not done due to the lack of implementation of equivalent Gaussian Count and Sum functionalities within the tested PyDP version for GoogleDP. A comparative evaluation of Gaussian implementations across both libraries is an important area for future work. Further limitations include the focus on only Count and Sum queries, as well as the use of synthetic data and testing on a single hardware/software environment.

These limitations emphasize that while this research offers valuable specific benchmarks for Laplace implementations, more research is needed to explore the comparative aspects of these DP libraries. Future work should certainly include testing and comparing their Gaussian mechanism implementations.

7 Conclusion

This research conducted an empirical comparison of the Google Differential Privacy Library (GoogleDP) and the OpenDP Library. The study evaluated the computational performance, scalability, and utility of their Laplace mechanism implementations when applied to Count and Sum queries, aiming to provide up to date and data-backed insights for practitioners.

The findings for the Laplace mechanism indicated differences between the libraries. GoogleDP consistently showed lower execution times and better scalability for both Count and Sum queries, an advantage that was evident with larger datasets. The execution speed of both libraries was unaffected by the choice of the privacy parameter ϵ. In terms of utility (measured by Mean Absolute Error) both libraries showed the expected inverse relationship with ϵ. Increasing the privacy budget (higher ϵ) led to lower error. While utility differences were often small, OpenDP sometimes offered comparable or marginally better utility for Sum queries, especially on large datasets and under stricter privacy settings (lower ϵ).

Rgarding the main research question for the Laplace mechanism, this study concludes that GoogleDP offers a clear advantage in computational speed and performance scalability for these queries. OpenDP provides a competitive and (in some cases for Sum queries) a better utility profile. The selection between these libraries for Laplace-based Count and Sum queries may depend on whether the primary criterion is processing efficiency or achieving optimal utility for specific privacy configurations.

A major limitation of this study is the fact that comparative analysis was restricted to the Laplace mechanism. The planned evaluation of the Gaussian mechanism was not done due to the lack of implementation of equivalent Gaussian Count and Sum functionalities within the tested PyDP version for GoogleDP. Consequently, an empirical comparison of Gaussian mechanism implementations in both libraries is an essential direction for future research. Further investigations could also look into different query types, use real-world datasets and continue to assess new releases of these actively developed libraries.

References

[1] Ann Cavoukian. Privacy by design: The 7 foundational principles. Technical report, Information and Privacy Commissioner of Ontario, Canada, 2011.

[2] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography, Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006, Proceedings*, volume 3876 of *Lecture Notes in Computer Science*, pages 1–19. Springer, 2006.

[3] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

[4] European parliament and Council of the European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). Official Journal of the European Union, L 119/1, May 2016.

[5] Marco Gaboardi, James Honaker, Gary King, Daniel Kifer, Zirui Lin, Kobbi Nissim, Andrew Park, Joshua Snoke, Varun Suriyakumar, Philip Swan, Salil Vadhan, and Ellen Wu. OpenDP: A System for Deploying Differentially Private Statistical Releases. arXiv preprint arXiv:2009.03384. Also presented at Theory and Practice of Differential Privacy (TPDP) 2020 Workshop, 2020.

[6] Google. Google's Differential Privacy Library. GitHub repository.

[7] Michael Hay, Gerome Miklau, David Jensen, and Patrick Weis. Boosting the accuracy of differentially-private histograms through consistency. In *Proceedings of the VLDB Endowment*, volume 3, pages 1020–1031. VLDB Endowment, 2010.

[8] James Manyika, Michael Chui, Brad Brown, Jacques Bughin, Richard Dobbs, Charles Roxburgh, and Angela Hung Byers. Big data: The next frontier for innovation, competition, and productivity. Technical report, McKinsey Global Institute, May 2011.

[9] Arvind Narayanan and Vitaly Shmatikov. Robust de-anonymization of large sparse datasets. In *Proceedings of the 2008 IEEE Symposium on Security and Privacy*, pages 111–125. IEEE, May 2008.

[10] National Commission for the Protection of Human Subjects of Biomedical and Behavioral Research. The belmont report: Ethical principles and guidelines for the protection of human subjects of research. Department of Health, Education, and Welfare, April 1979.

[11] OpenDP Community. OpenDP - The Open-Source Differential Privacy Platform. Official Website.

[12] OpenDP Community. OpenDP Library. GitHub repository.

[13] President's Council of Advisors on Science and Technology (PCAST). Big data and privacy: A technological perspective. Technical report, Executive Office of the President, May 2014.

[14] Latanya Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.

[15] U.S. Department of Health & Human Services. Health information privacy (hipaa). Official Website of the U.S. Department of Health & Human Services.

[16] Ruobin Zhang, Ben Niu, Mohammad Al-Rubaie, and James M. Chang. Evaluating OpenDP SmartNoise and Google DP with Other Libraries for Differential Privacy. *Sensors*, 23(14):6509, jul 2023.