PUCRS

ESCOLA POLITÉCNICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
DOUTORADO EM CIÊNCIA DA COMPUTAÇÃO

HENRY CABRAL NUNES

PREDICTIVE METRIC FOR OPTIMAL BUDGET ALLOCATION IN DIFFERENTIAL PRIVACY

Porto Alegre
2024

PÓS-GRADUAÇÃO - *STRICTO SENSU*

Pontifícia Universidade Católica
do Rio Grande do Sul

PONTIFICAL CATHOLIC UNIVERSITY OF RIO GRANDE DO SUL
SCHOOL OF TECHNOLOGY
COMPUTER SCIENCE GRADUATE PROGRAM

PREDICTIVE METRIC FOR OPTIMAL BUDGET ALLOCATION IN DIFFERENTIAL PRIVACY

HENRY CABRAL NUNES

Doctoral Thesis submitted to the Pontifical Catholic University of Rio Grande do Sul in partial fulfillment of the requirements for the degree of Ph. D. in Computer Science.

Advisor: Prof. Dr. Avelino Francisco Zorzo

Porto Alegre
2024

Ficha Catalográfica

N972p Nunes, Henry Cabral

Predictive Metric for Optimal Budget Allocation in Differential Privacy / Henry Cabral Nunes. – 2024.

80 p.

Tese (Doutorado) – Programa de Pós-Graduação em Ciência da Computação, PUCRS.

Orientador: Prof. Dr. Avelino Francisco Zorzo.

1. Differential Privacy. 2. Anonimização. 3. Privacidade. 4. Dataset. 5. Estatísticas descritivas. I. Zorzo, Avelino Francisco. II. Título.

Elaborada pelo Sistema de Geração Automática de Ficha Catalográfica da PUCRS com os dados fornecidos pelo(a) autor(a).

Bibliotecária responsável: Clarissa Jesinska Selbach CRB-10/2051

HENRY CABRAL NUNES

PREDICTIVE METRIC FOR OPTIMAL BUDGET ALLOCATION IN DIFFERENTIAL PRIVACY

This Doctoral Thesis has been submitted in partial fulfillment of the requirements for the degree of Ph. D. in Computer Science of the Computer Science Graduate Program, School of Technology of the Pontifical Catholic University of Rio Grande do Sul

Sanctioned on August 29, 2024.

COMMITTEE MEMBERS:

Prof. Dr. Rodrigo Coelho Barros (PPGCC/PUCRS)

Prof. Dr. Luciano Paschoal Gaspary (PPGC/UFRGS)

Prof. Dr. Javam de Castro Machado (MDCC/UFC)

Prof. Dr. Avelino Francisco Zorzo (PPGCC/PUCRS - Advisor)

Dedico este trabalho ao meu grandioso pai.

“If you don’t know, the thing to do is not to get scared, but to learn.”
(Ayn Rand)

MÉTRICA PREDITIVA PARA ALOCAÇÃO ÓTIMA DE ORÇAMENTO EM PRIVACIDADE DIFERENCIAL

RESUMO

Neste trabalho, abordamos a questão crítica da alocação de orçamento em aplicações de Privacidade Diferencial (DP), especificamente para cenários onde estatísticas descritivas são divulgadas. Nosso principal objetivo é desenvolver uma métrica e um cenário inovadores que utilizem informações sobre o uso futuro dos dados para otimizar a distribuição do orçamento. Uma distribuição de orçamento eficaz é fundamental para melhorar a utilidade dos dados sem comprometer a privacidade, um desafio significativo no campo da DP. Identificamos e exploramos uma lacuna relacionada às interações entre consultas de DP para melhorar a utilidade dos dados. Nossa métrica é formalmente definida e demonstramos sua aplicação por meio de um cenário hipotético utilizando dados sintéticos. Os resultados indicam uma melhoria substancial na utilidade dos dados, mantendo a privacidade. Este estudo não apenas oferece uma contribuição valiosa para o campo da DP, mas também abre caminhos para futuras pesquisas e aplicações práticas em cenários do mundo real.

Palavras-Chave: Differential Privacy, Anonimização, Privacidade, Dataset, Metrica, Estatísticas descritivas.

PREDICTIVE METRIC FOR OPTIMAL BUDGET ALLOCATION IN DIFFERENTIAL PRIVACY

ABSTRACT

This work addresses the critical issue of budget allocation in Differential Privacy (DP) applications, specifically for scenarios where summary statistics are released. Our main objective is to develop a novel metric and scenario that leverages information about future data usage to optimize budget distribution. Effective budget distribution is pivotal in enhancing data utility without compromising privacy, a significant challenge in the DP field. We identify and exploit a gap related to the interactions between DP queries to improve data utility. Our metric is formally defined, and we apply it through a hypothetical scenario using synthetic data. The results indicate a substantial improvement in data utility while maintaining privacy. This study offers a valuable contribution to the DP field and opens avenues for future research and practical applications in real-world scenarios.

Keywords: Differential Privacy, Anonymization, Privacy, Dataset, Metric, Summary Statistics.

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1. INTRODUCTION

The right for an individual to separate his public from his private life is one of the most basic human rights. It is so important that it is included in Article 12 of the Universal Declaration of Human Rights: ”No one shall be subjected to arbitrary interference with his privacy, family, home or correspondence, nor to attacks upon his honor and reputation. Everyone has the right to the protection of the law against such interference or attacks.”[1]. However, because of several difficulties, the protection of this basic human right is not a trivial task.

One of the difficulties is that data about individuals are readily available. This availability is a consequence of society’s informatization. Data is collected on various information systems, and users are usually unaware of the collection process. For example, habits during our navigation on the internet or daily routine itinerary captured by a smartphone. Another case is when users voluntarily surrender information about their private lives because they are careless or do not care about their privacy. For example, information publicly available on social networks. The result of this is that a huge amount of data is readily available to multiple third parties, such as companies and governments.

Another difficulty is the inappropriate use of the data by third parties. Often, data is collected in an unauthorized way. Companies trade these data with other companies without the individual’s consent. Furthermore, data is stored in an insecure manner, which makes it vulnerable to data thievery. Finally, the data can be used illegally. Even governments of democratic countries are known to violate the privacy of their citizens [2].

Society is awakening to these and other difficulties to guarantee the right to privacy, and several initiatives have taken place. Some initiatives include new laws that restrict how individuals’ data can be used, collected, and managed. For example, the General Data Protection Regulation (GDPR) from the European Union [3], the Brazilian [4] and Indian [5] equivalents, and the UE cookie law [6], just to cite a few. Another list of initiatives is the creation of tools that protect the users’ privacy, which is also increasing significantly. A few examples include DuckDuckGo [7], Privacy Badger [8], and Brave [9]. These initiatives are important to protect individuals’ privacy, however, it is also important to recognize that data is critical to improving quality of life. Data can be used in research, mining, machine learning, and other means by companies and academia to develop new technologies,

[1] www.un.org/en/about-us/universal-declaration-of-human-rights
[2] www.theguardian.com/us-news/2020/sep/03/edward-snowden-nsa-surveillance-guardian-court-rules
[3] gdpr.eu
[4] www.serpro.gov.br/lgpd/menu/a-lgpd/o-que-muda-com-a-lgpd
[5] prsindia.org/billtrack/the-personal-data-protection-bill-2019
[6] gdpr.eu/cookies/
[7] duckduckgo.com/privacy
[8] privacybadger.org
[9] brave.com

products, and insights. This creates a conflict, while it is important to protect individuals' privacy, it is also important to make data widely available to improve our society.

This conflict attracted the attention of both academia [9] [11] and industry. The main question is whether it is possible to protect data privacy and, at the same time, keep it useful for research and other applications.

Several researchers have produced new strategies to try to solve this dilemma for different types of data, such as graphs[27] [50], and longitudinal data[46]. One of the most important is anonymization techniques for datasets, in this area, the main consolidated technique is Differential Privacy (DP)[14], which is the focus of this thesis.

However, the DP approach does not guarantee perfect privacy and data utility simultaneously. The anonymization process using the mentioned techniques will distort the data to some degree, which will incur the loss of utility in favor of privacy. Although this data utility loss can be controlled, keeping more data utility can be done at the cost of less privacy protection. The balance between privacy and utility is called the utility-privacy trade-off [14]. This trade-off is a pivotal part of this research. An analyst executing DP anonymization techniques will have a set of parameters for fine-tuning this trade-off of the anonymization process. The main objective is to provide acceptable privacy while keeping the data as usable as possible.

To support analysts in this endeavor, a diverse array of metrics and techniques exists to gauge data utility, error, and precision finely. These metrics primarily stem from the discipline of statistics. Furthermore, methodologies from machine learning and data mining, like assessing performance variations pre- and post-parameter alterations, contribute valuable insights. This approach is applicable in machine learning scenarios employing Differential Privacy (DP). Nonetheless, the metrics commonly employed tend to be broad and may not directly address the intricacies of anonymization contexts. Similarly, while the machine learning approach proves beneficial, its relevance is confined to the realm of machine learning.

Hence, a gap exists in metrics applicable to Differential Privacy (DP) to effectively measure data utility and facilitate the trade-off between privacy and utility. Our research underscores a specific gap concerning summary statistics within the framework of Global Differential Privacy. This gap pertains to understanding the interaction of anonymized queries from a DP-protected database. Leveraging this gap presents an opportunity to enhance data utility. Our research proposes addressing this gap through the introduction of a novel metric.

Thus, our research focuses on three key objectives. Firstly, we aim to illustrate that the noise introduced by DP varies in magnitude when different anonymized queries interact. In this context, noise represents a reduction in data utility, as heightened noise diminishes data precision. Secondly, we endeavor to introduce a novel metric capable of leveraging these varying noise levels. This metric aims to empower developers to enhance

data utility in their solutions while maintaining privacy standards. Thirdly, we seek to assess the effectiveness of our proposed metric in improving data utility.

1.1 Contribution

This subsection underscores the significance of the research associated with this thesis, delineating its contributions across three key points. These contributions are closely aligned with the research objectives outlined in Chapter 4, elucidating their effects on stakeholders and the context. Moreover, these contributions pave the way for future research directions, a topic thoroughly explored in Chapter 6.

The contributions are as follows:

- We have identified that in interactions involving two anonymized queries protected by Differential Privacy (DP) through mathematical operations, the resulting noise is influenced by various parameters employed in the anonymization process, notably the budget allocation. This discovery is substantiated by analyzing these interactions in basic mathematical operations, where different parameters are assessed and analyzed. These interactions offer ample opportunity for further exploration to deepen our understanding, as well as potential avenues for enhancing data utility in specific scenarios.
- We propose a metric designed for DP developers to explore the interaction of anonymized queries. Its effectiveness hinges on the developer's comprehension of the DP scenario. This represents an initial endeavor to leverage insights extracted from anonymized query interactions to enhance data utility.
- The metric is evaluated within a theoretical framework, yielding promising results indicating that its utilization can indeed bolster data utility without compromising privacy.

The thesis heavily centers on the proposed metric. This focus stems from the early stages of the research associated with this thesis, where one of the primary objectives was to explore how various metrics could enhance data utility in Differential Privacy (DP) scenarios.

1.2 Structure

Following this introduction, the subsequent chapter will provide the necessary background to understand the rest of the work. This includes key concepts of Differential

Privacy, alternative definitions, and application scenarios. The measurement of data utility, based on the noise introduced by the anonymization process of DP algorithms, is also presented. This concept is crucial for the majority of our thesis. In Chapter 3, we present a series of related works, organized by topic. Chapter 4 provides a contextualization of our research problem and presents the research question. Chapter 5 presents the main body of our research, focusing on identifying a gap in DP, proposing a metric that allows a developer to improve data utility in specific DP scenarios, and finally, evaluating the performance of such a metric. In the final chapter, we engage in a comprehensive discussion of this work and suggest further directions for research.

1.2.1 Disclaimer

ChatGPT [10] and Grammarly[11] were employed to enhance the fluidity and comprehension of the text in this work. However, their use was limited to text correction.

[10] ChatGPT.com
[11] app.grammarly.com

2. BACKGROUND

In this section, we explore one of the most significant approaches to dataset anonymization: Differential Privacy (DP). This technique encompasses several modifications and alternative definitions suited for different scenarios. Here, we will present this field's most used and important aspects. Additionally, we will briefly introduce another notable approach for dataset anonymization: Synthetic Anonymization.

2.1 Differential Privacy

DP is a definition that yields strong privacy guarantees to a dataset member. This dataset member has a strong case for denying membership in the dataset. The cause is that the amount of information learned if he participates in a dataset or not is nearly the same [15]. It makes it almost impossible for an attacker to infer membership if DP parameters are correctly set. Take a dataset for a study about smoke diseases as an example; an analyst will know the same amount of information about an individual and whether or not he belongs to such a dataset. It is a challenge to affirm whether it is a member of the dataset or not. This approach to anonymization is in contrast with other attempts to formalize privacy in datasets, *i. e.* the privacy definition revolved around the amount of information that is learned about an individual before and after accessing the dataset [29] [10]. Later, it was proved that these alternative definitions were not feasible [14].

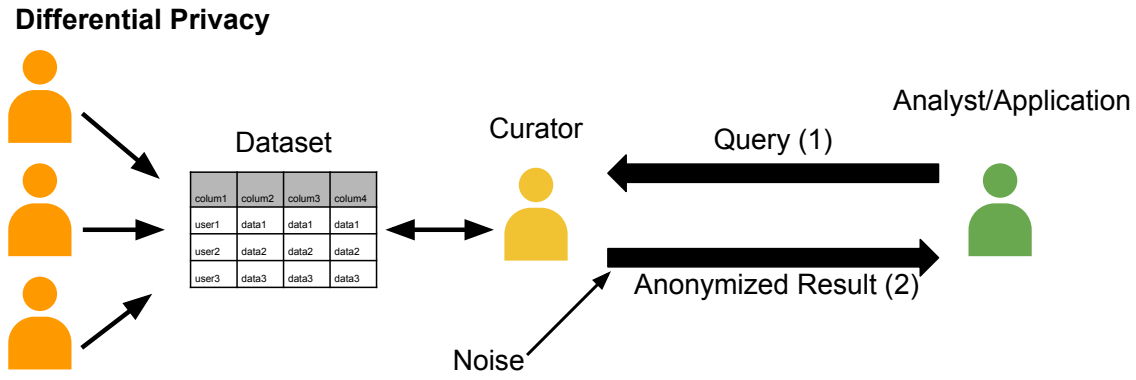


Figure 2.1 – Differential Privacy Process

The original definition of DP does not generate a transformed dataset. It depends on a trusted curator, as shown in Figure 2.1. It will process a solicited query (step 1 in Figure 2.1) made by an analyst and add noise to it before returning the answer(step 2 in Figure 2.1). Thus, the data owner will need to keep the means of processing individual queries to the dataset from the analysts' requests. This approach protects against attacks

on privacy, such as the data linkage attack [33] [45]. Also, it does not have the same vulnerabilities as other approaches for dataset anonymization, such as Synthetic Anonymity. We will discuss this approach in Section 2.7. However, another DP approach named Global Differential Privacy dismisses the need for a curator, and we will present it in Section 2.5.

The formal definition for Differential Privacy is expressed through ϵ-Differential Privacy, defined as the following.

$$\forall x.x \in range(M) \mid Pr[M(D) = x] \leq exp(\epsilon).Pr[M(D') = x]$$

In this equation, M represents a randomized algorithm, often referred to as the **Mechanism**, which operates on a dataset D. D and D' are two adjacent datasets, meaning they differ in exactly one row of data. The function *range* encompasses all possible values of x that can be produced by the mechanism M. Intuitively, this means that the probability distribution of outcomes between both datasets will have a probability distribution that differs at most ϵ. The value of ϵ is a value defined by a curator. A smaller value means better privacy.

In practical terms, whether an individual is included in dataset D or not will have minimal impact on the outcome of any query result, provided an appropriate ϵ value has been chosen.

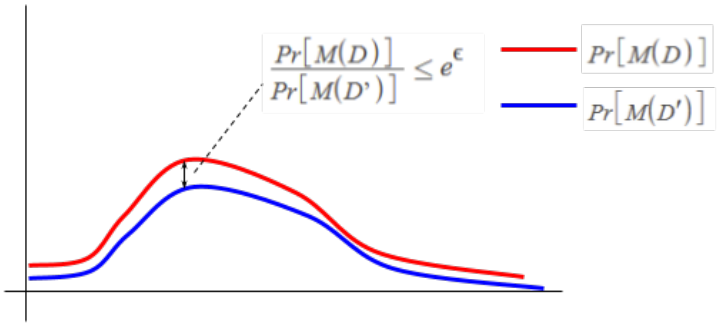


Figure 2.2 – Differential Privacy distribution

The algorithm M outputs the result of queries to the database by adding noise to the answer. For a query $f(D)$, we can define the algorithm as $M(D) = F(D) + n$, where it represents the noise n. The noise is generated based on a probability distribution, with the Laplace distribution being the canon distribution for this application, although other distributions are also applicable.

The Laplace distribution is characterized by two parameters: the scale (s), which defines the spread and shape of the distribution, and the location (μ), which specifies the distribution's central point. In DP, the location parameter (μ) is always set to zero, leaving the scale parameter (s) as the critical variable influencing the noise. The scale is calculated as $s = \frac{\text{sensitivity}}{\epsilon}$, where sensitivity represents the maximum difference in query

results between any adjacent dataset D' and the original dataset D. The parameter ϵ governs the level of privacy, as previously explained.

2.2 Properties of Differential Privacy

Differential privacy has three important properties that are fundamental to its application [16].

- **Post-Processing:** An output from a differential private mechanism is differential private independent of other processing performed on that output. Formally, if $f(x)$ satisfies ϵ-differential privacy, then any function g that uses f as input $g(f(x))$ is ϵ-differential privacy.
- **Concurrent composition or Group Privacy:** As mentioned in the previous section, removing one individual from the dataset will expose nearly the same amount of information. This concept extends to groups of individuals, although large groups will degrade the privacy guarantee. Formally, if a function $f(x)$ satisfies ϵ-differential privacy, and we partition the dataset X into disjoint sets $x_1 \cup x_2 \cup ... \cup x_n = X$. Then each release of the function $f(x)$ using as input a subset of the dataset, denoted as $f(x_k)$, also satisfies ϵ-differential privacy.
- **Sequential Composition:** It is possible to apply multiple differential privacy algorithms to the same dataset or on the result of another differential privacy algorithm. The released result maintains differential privacy, although the overall privacy guarantee may decrease. Formally, if a mechanism $f_1(x)$ satisfies ϵ_1-differential privacy, and another mechanism $f_2(x)$ satisfies ϵ_2-differential privacy. Then function $g(x)$ is $(\epsilon_1 + \epsilon_2)$-differential privacy.

  $$g(x) = (f_1(x), f_2(x))$$

  In the simplest scenario (Figure 2.3), where functions operating on the dataset $f_1(x), f_2(x), ..., f_n(x)$ do not interact with each other, each function is allocated an individual privacy budget ϵ_i. In most cases of DP, a single privacy budget ϵ is defined to be shared among all functions operating on the dataset. However, this sharing doesn't imply equal allocation; some functions may receive higher budgets than others, resulting in outputs with varying noise levels.

  Another scenario for sequential composition arises when the output of one function serves as input in the following function. This scenario represented in Figure 2.4 is called **adaptive composition**. This scenario is prevalent in machine learning

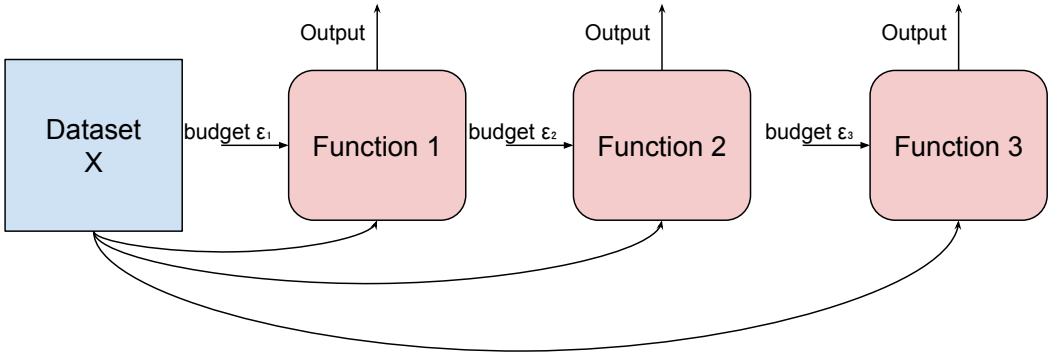


Figure 2.3 – Sequential Composition

applications, where the result of one function feeds into the next. In this setup, the budget ϵ values allocated for each function remain fixed before execution. A third approach changes that.

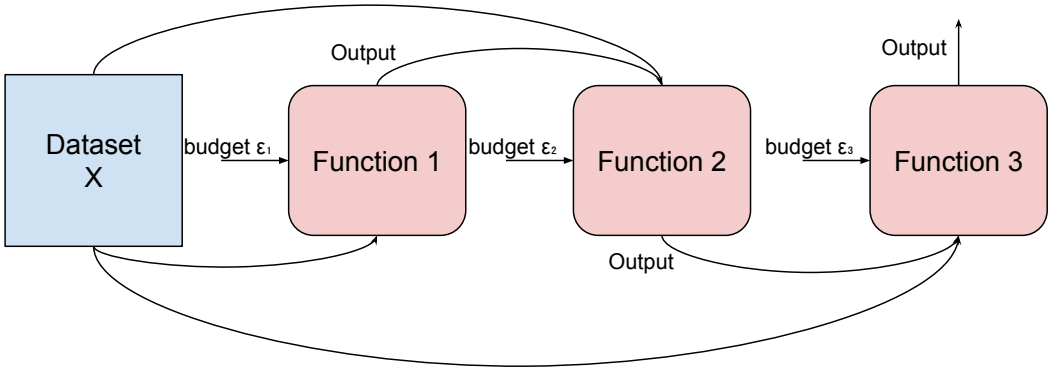


Figure 2.4 – Adaptive Composition

A third scenario of sequential composition is the **Fully adaptive composition**. It takes into account the output of a function as a parameter that impacts the allocation of the privacy budget ϵ for the following function [41]. Figure 2.5 represents this scenario.

2.3 Two challenges of DP: The Privacy-Utility trade-off and the Budget Allocation

As discussed in previous sections, ϵ is a fundamental parameter in DP scenarios. It governs the level of noise added in a mechanism. In practice, ϵ plays a dual role:

On one hand, it determines the level of privacy the mechanism provides. A lower ϵ value corresponds to higher noise, increasing privacy.

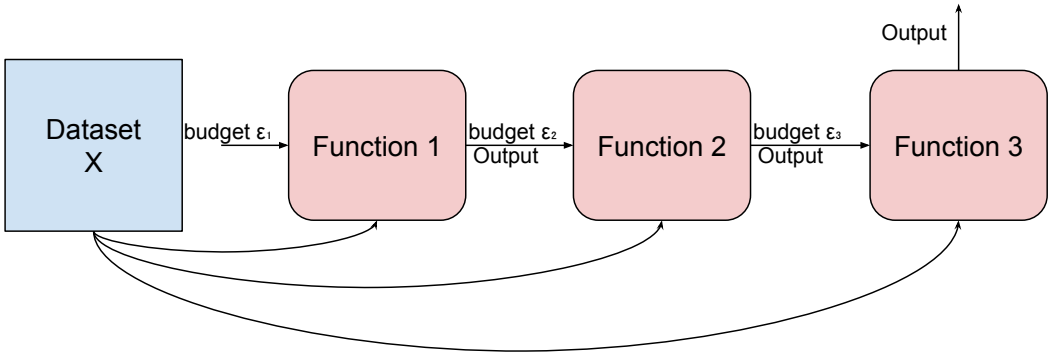


Figure 2.5 – Fully Adaptive Composition

On the other hand, ϵ also influences the utility of the mechanism's output. A higher ϵ value reduces noise, leading to more precise and valuable results.

This dilemma between privacy and utility of a mechanism is known as the **Privacy-Utility trade-off**. The ϵ is defined by the developer of the application, which needs to take into account several implications, including:

- Privacy Requirements: Legal or business-defined privacy requirements must be considered when defining ϵ.
- Application Sensitivity: Some applications have specific needs for data utility to be helpful, which influences the value of ϵ.
- Risk Tolerance: It's essential to assess your organization's risk tolerance for potential privacy breaches when determining ϵ.

Some researchers work on the definition of the ϵ value [21] [38], while others focus on explaining how this value impacts privacy [32] [30]

Another problem stemming from the parameter ϵ is its allocation between multiple mechanisms. As discussed in Section 2.2, in Sequential Composition, the ϵ value is shared among numerous functions or mechanisms, although not necessarily equally. This collective ϵ allocated across the entire application is referred to as ϵ budget.

The distribution of this budget significantly impacts the application's performance in terms of precision and accuracy. Dedicating less budget to a mechanism results in more noise and reduced utility in its output. Allocating more budget reduces noise and enhances utility. Since not all mechanisms require the same level of performance, determining how to divide the budget becomes a crucial problem.

This issue is a central focus in the field of differential privacy and is addressed in detail in Chapters 3, 4, and 5. It is worth noting that, regardless of the budget distribution, maintaining the total budget ensures consistent privacy levels, as stated in the Sequential Composition property.

2.4 Relaxations and alternative definitions to Differential Privacy

The DP described thus far represents the canonical version, known for its high privacy guarantees and strong definition. However, a drawback of the canonical version is its tendency to reduce the data utility, rendering some applications infeasible. Additionally, some applications may benefit from a modified DP definition that better suits the application's needs.

In this section, we introduce two modified DP definitions: Approximate Differential Privacy and Gaussian Differential Privacy. Afterward, we provide a summary of other variants of DP.

2.4.1 Approximate Differential Privacy

The notion of DP can be weakened to provide more flexibility, leading to the concept of Approximate Differential Privacy [16]. One of the benefits of such flexibility is the **Advanced Composition**, which is a form of composition where the error caused by the noise is less than that of the normal composition. The notation for this definition is (ϵ, δ)-Differential Privacy. The definition is as follows:

$$Pr[(M(D) \in S)] \leq exp(\epsilon).Pr[M(D' \in S)] + \delta$$

Where D and D' are two adjacent datasets. S is all subsets of the $range(M)$. The parameter δ, introduced by the curator, serves as an additional margin beyond the bounds of ϵ. When an output of the algorithm M falls within this margin, the privacy for that specific execution is compromised. As such, the δ parameter should be a small value.

It is worth noting that when $\delta = 0$, the constraint of Approximate Differential Privacy is equivalent to that of the canonical Differential Privacy. Such a case is called pure Differential Privacy.

2.4.2 Gaussian Differential Privacy

Gaussian Differential Privacy uses the Gaussian distribution as the noise source in the mechanism [12]. It falls to the f-differential privacy family, which comprises DP definitions approaching Approximate Differential Privacy. As such, it represents a relaxed version of the canonical DP. Notably, Gaussian Differential Privacy offers stronger privacy guarantees during sequential composition than Approximate Differential Privacy. Among

the f-differential privacy family, Gaussian Differential Privacy stands out as the most predominant definition.

It uses a model of attack on the DP scenario based on hypothesis testing. An attacker wants to distinguish the dataset D and its neighbor D' based on the output of a mechanism. The hypothesis testing problem is as follows.

$$H_0 : \text{The dataset used is D}$$

$$H_1 : \text{the dataset used is D'}$$

Based on this testing in the case of **Approximate Differential Privacy**, the upper bound of any testing using a significance level $0 < \alpha < 1$ is $e^{\epsilon}\alpha + \delta$. Thus, the test is powerless for small privacy values of α and δ. The problem of **Approximate Differential Privacy** is when composition is used. In that case, the privacy guarantees, based on the upper bound, are no longer clear. Gaussian Differential Privacy includes tools to reason about composition using the Central Limit Theorem.

2.4.3 Others

Numerous other alternative definitions of DP exist, as (α, ϵ)-Rényi DP [31] that is based on R´enyi divergence, and ρ-zero-concentrated DP [5], also based on R´enyi divergence but focused in trail probabilities. This is an active field of research, with ongoing work for new alternatives. This section presents a summary in Table 2.1. While the list is not exhaustive, it includes some of the more important definitions to date.

2.5 Local and Global Differential Privacy

The DP presented until now is the original version, where the mechanism adds noise to protect the privacy of members of dataset D. It still depends on a curator that will have access to the original dataset as represented in Figure 2.6 to add noise to the queries. This approach that depends on the curator to add noise to the query result to protect privacy is called **Global Differential Privacy**.

Two significant problems arise from the Global Differential Privacy approach. Firstly, the curator needs to be trusted implicitly. Since the curator has access to all the original data, any malicious action on their part could lead to data leakage from the dataset members. This reliance on a trusted curator introduces a potential vulnerability.

Secondly, even if the curator is trusted, the original dataset remains vulnerable to external attacks. While the curator may act in good faith, external threats could still com-

DP Definition	Summary
ϵ-DP [14]	The canonical definition, also known as Pure Differential Privacy, offers strong privacy guarantees but is often too restrictive for real-world applications.
(ϵ, δ)-DP [16]	Approximate Differential Privacy is a relaxation of ϵ-DP, introducing an additive term δ to balance privacy and data usability. However, measuring privacy guarantees is less clear compared to ϵ-DP.
f-DP [12]	f-Differential Privacy, a family of relaxed DP definitions, improves on Approximate Differential Privacy by enabling clear risk analysis using Hypothesis testing.
Gaussian DP [12]	A significant variant of f-Differential Privacy, utilizing the Gaussian Distribution for noise and offering enhanced risk analysis capabilities.
(α, ϵ)-Rényi DP [31]	This definition proposes a stronger relaxation of DP based on Rényi divergence, facilitating quantitative tracking of privacy loss during advanced composition.
ρ-zero-concentraded DP [5]	Similar to Rényi DP, this variant employs Rényi divergence to improve privacy loss quantification, focusing specifically on tail probabilities of privacy loss.

Table 2.1 – List of alternative DP definitions

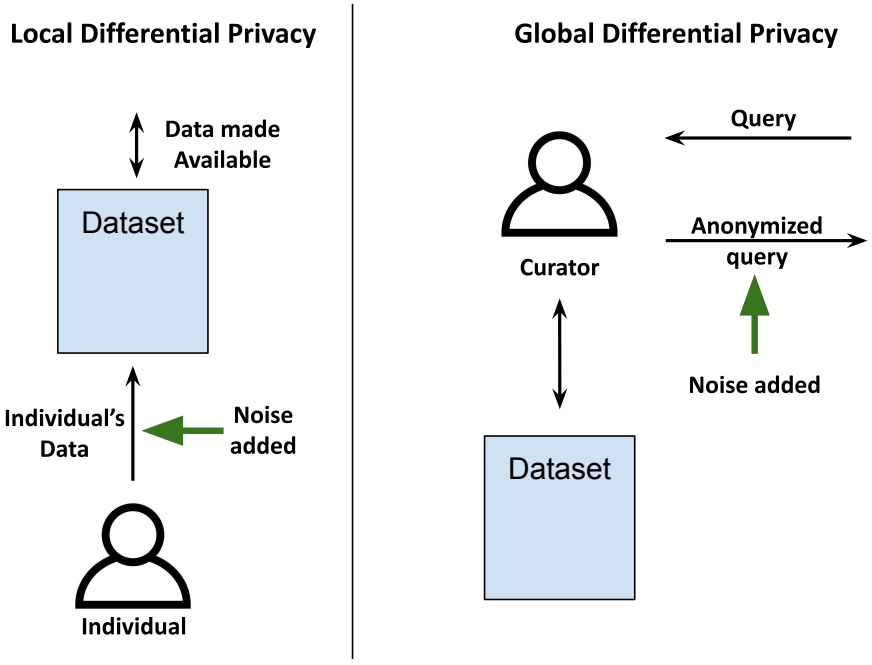


Figure 2.6 – Global and Local Differential Privacy

promise the dataset. This necessitates robust security measures to protect the dataset from unauthorized access and breaches.

An alternative approach to Differential Privacy is known as **Local Differential Privacy**. In this approach, each individual adds noise to their data before it is inserted into the dataset. This means that every participant adds noise to their data, making the dataset publicly available. The need for a trusted curator in Global Differential Privacy is eliminated.

While Local Differential Privacy addresses the issues of Global Differential Privacy, it also introduces several challenges. One challenge is the limited information sharing among participants, as each individual adds noise independently to their data. This limits the potential insights that the curator could derive from the dataset.

Another challenge is the difficulty of fine-tuning and preserving data utility. Since noise is added locally by each participant, it is impossible to analyze the dataset before noise is added, thus hindering any attempt to improve data utility based on the real values of the dataset.

Another significant challenge is enforcing participants' compliance with privacy-preserving protocols. It is essential to ensure the correct aggregation of data while maintaining the privacy of all dataset members.

Both approaches have their challenges and advantages. The choice of the better one is highly dependent on the application and its requirements. There are use cases for both Local Differential Privacy and Global Differential Privacy.

2.6 Measuring Utility

An essential aspect of Differential Privacy is the measurement of data utility. While privacy is governed by the parameter ϵ, there is no direct way to quantify data utility. Various methods are employed for this purpose

One measurement method focuses on the noise added to the original query result, which distorts the query result before it is presented to an analyst. Our work will measure data utility based on the **amount of noise added**.

The **noise added** can be quantified as the absolute value of the difference between the real result of an operation in the database $f(D)$ and the result from a mechanism $M(D)$. This quantification is essential for assessing data utility. However, due to the randomness introduced by the Laplace distribution[6] utilized in the mechanism, a significant number of measurements of the noise added (denoted as na) need to be conducted for accurate assessment.

To account for this, the resulting measurement should be weighted based on the number of measurements (p) conducted. The weighted measurement can be calculated using Equation 2.1, where p represents the number of measurements.

$$na = \frac{\sum_i^p abs(f_i(D) - M_i(D))}{p} \tag{2.1}$$

In this study, we will adopt a similar approach to measure utility. However, instead of measuring the **noise added** of a single mechanism, we will measure it for a mathematical operation that utilizes two mechanisms. For instance, consider the addition of two mechanisms: $O(D) = M_1(D) + M_2(D)$. The resulting equations are depicted in Equation 2.2, where $OO(D)$ represents the original result of the operation without the noise introduced by the mechanisms.

$$na = \frac{\sum_i^p abs(OO_i(D) - O_i(D))}{p} \tag{2.2}$$

2.7 Other anonymization technique - Synthatic Anonymization

The Syntactic Anonymity approach for dataset anonymization involves altering multiple attribute values of each individual data, resulting in an anonymized dataset, as illustrated in Figure 2.7. This alteration intends to ensure that multiple rows hold identical values. By doing so, an attacker attempting a linkage attack [44], which involves using external information to identify a target person, would need to segregate the row representing the target from other rows with the same values.

However, altering attribute values also leads to the loss of some information. Therefore, there is a concern to minimize these changes to the data in order to preserve as much information as possible.

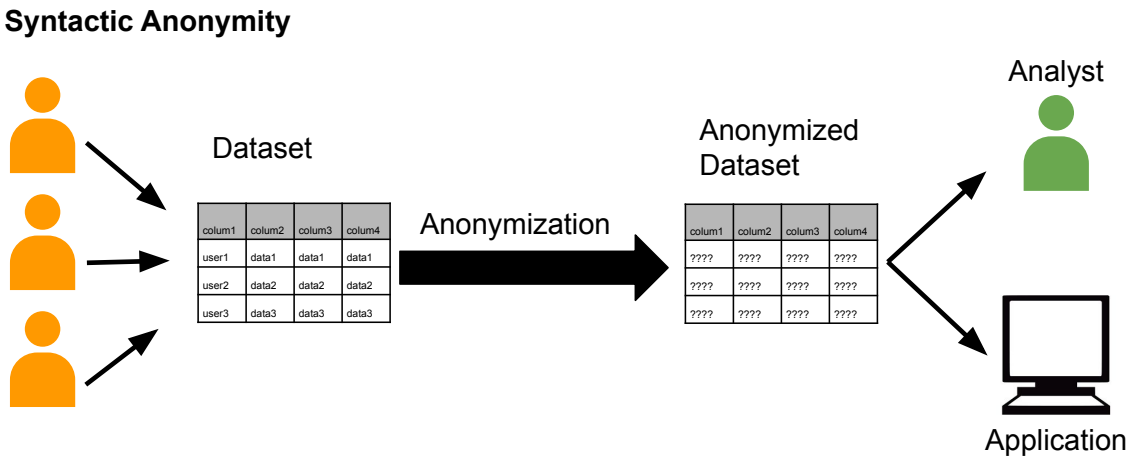


Figure 2.7 – Syntactic Anonymity Process

To change the values in a dataset, two main techniques are used:

- **Generalization:** Involves changing the original value with another one that is less specific but still holds some meaning to the original value. For instance, consider an attribute like ”sex” with values male, female. A possible generalization for this case is changing both values to ”any sex” male $\rightarrow$ any sex, female $\rightarrow$ any sex.

  For numerical values, intervals or averages can be used. For example, suppose an attribute ”salary” has the following values: 2000, 3500, 2800. It can be generalized to an interval [2000, 3500].

- **Suppression:** Involves the removal of all values for an attribute or an entire row from the dataset, rendering it unusable. While this may seem extreme, it is necessary in several cases where preserving privacy is paramount.

These techniques were employed alongside various models that offer some level of protection for anonymized data. However, it is important to note that all these models have some vulnerabilities. Nevertheless, these vulnerabilities do not render them useless, and most of them are still widely used.

In chronological order, the most important of these models are k-anonymity, l-diversity, and t-closeness. These models are closely related, each building upon its predecessor to improve and address specific vulnerabilities.

k-anonymity was the first syntactic model proposed to protect privacy [44] [43]. This model proposes that each tuple in a dataset DS must have at least k other tuples with the same quasi-identifier attribute values. The tuple must belong to an equivalence class with k tuples in it. To achieve k, a transformation on the dataset DS is applied. This transformation will use the mentioned generalization and suppression techniques to create a dataset DS', which achieves k. This model has some vulnerabilities, for example, the homogeneity attack [51] and the background knowledge attack [29]. The next model solves these problems.

l-diversity[29] was introduced to address attacks that exploit the lack of diversity in the sensitive value SD for equivalence classes in k-anonymity (known as homogeneity attack). It operates by incorporating the sensitive data into the process of creating equivalence classes, ensuring that each equivalence class contains at least l distinct sensitive values among its rows. This approach enhances protection against background knowledge attacks as well.

However, this additional requirement to distribute sensitive data among equivalence classes often necessitates more generalization, which leads to a higher loss of information.

t-closeness[24] is similar to the l-diversity in the sense that it changes the equivalence classes to accommodate different values for the sensitive attribute SD. However, unlike the l-diversity, t-closeness takes into account the original distribution of SD in the entire dataset DS. The difference, or distance, between the sensitive data distribution

SD in each equivalence class in the anonymized dataset compared to the original dataset must be no more than the parameter *t*.

2.8 Final Consideration

In this section, we highlighted the most important aspects of Differential Privacy (DP), a pivotal component of our work. DP stands out as the most significant technique for dataset anonymization due to its strong definition and rigorous mathematical properties. This background provides a sufficient foundation for understanding the work developed in this thesis. Additionally, we included a brief overview of Syntactic Anonymity, another anonymization technique, although it is not as robustly defined as DP.

3. RELATED WORK

Our work is closely tied to the allocation of the privacy budget ϵ in data anonymization processes. The allocation of this budget significantly influences the utility of anonymized data, making it a pivotal aspect of DP. Consequently, extensive research is dedicated to optimizing budget allocation. In this thesis, we explore two cases of budget allocation: firstly, general strategies for allocation, and secondly, optimal allocation tailored explicitly for Machine Learning algorithms. Given the prominence of Machine Learning in this domain, the latter receives particular emphasis. The selected studies are from 2019 or early.

3.1 General Budget Allocation Strategies

This section explores proposals for allocating the privacy budget that transcend specific scenarios. While much of the research on budget allocation focuses on particular contexts, often centered around specific machine learning algorithms, a general approach remains crucial. This encompasses proposals adaptable to virtually any scenario, including the summary statistics scenario.

The research conducted by Bai *et al.* [1] revolves around employing convergent series as a strategy for budget allocation. The authors argue that the uniform budget allocation strategy frequently falls short regarding data utility, advocating instead for the flexibility offered by convergent series. Additionally, they introduce several optimization approaches applicable within the convergent series framework and discuss countermeasures against collusion attacks [13].

The proposal introduces utilizing two distinct series: the Geometric series and the Taylor series. While the former has been previously employed for budget allocation, the latter represents a novelty in this context. The general approach for budget distribution is outlined as follows, where ϵ_i denotes the budget allocation for term i, ϵ signifies the total budget, and k_i represents the proportion of the budget allocated for term i:

$$\epsilon_i = k_i \epsilon$$

In the Geometric series, the value of k_i is determined by the formula:

$$k_i = (1 - r) r^{i-1}$$

In this context, r signifies the ratio between two consecutive terms, serving as a mechanism to govern the budget's distribution proportion. A lower value of r suggests

a swifter budget dispersion. The authors suggest that if the number of queries n to be conducted is known, the following formula yields the optimal r.

$$r = \frac{n-1}{n}$$

The Taylor series possesses two key characteristics that render it suitable for budget allocation. Its most significant attribute is nonmonotonicity coupled with positivity. In the Taylor series, k_i is computed as follows:

$$k_i = \frac{t}{(i-1)}(ln\frac{1}{t})^{i-1}$$

t defines the first term of the series and impacts the function form. The authors suggest the following formula to determine the value of t.

$$t = e^{1-[\frac{n}{2}]}$$

In both sequences, not all the budget available will be used. A calibration strategy is used to correct that. In it, the correct proportion is k_i*, which is used to calculate the budget for each query $\epsilon_i* = k_i*\epsilon$. The formula for the calibration is presented below, where $S_n = \sum_{i=1}^{n} k_i < 1$.

$$k_i* = \frac{k_i}{S_n}$$

The first proposed approach to optimizing budget allocation involves inverting the distribution. Typically, the normal distribution using series prioritizes allocating more budget to the earlier terms, resulting in reduced noise for initial queries but heightened noise for subsequent ones. However, certain applications, such as random forest machine learning algorithms, which will be discussed later in this chapter, could benefit from reversing this order. By allocating more budget to later queries and less to earlier ones, a more favorable balance of noise distribution can be achieved.

The second approach entails establishing an acceptable budget allocation threshold. This threshold represents the minimum budget required for a query, which is essential because, in certain instances, the noise generated by a mechanism may render its results unusable. However, determining an acceptable budget poses challenges due to the varying sensitivities of different queries within the dataset. Not all queries are created equal; some exhibit various levels of sensitivity to DP, necessitating distinct acceptable budget allocations.

To address this challenge, the authors propose normalizing the sensitivity of the dataset. Additionally, they present a strategy for integrating the acceptable budget as a constraint during the budget allocation process. This ensures that each query receives a

budget allocation tailored to its specific sensitivity level, thus enhancing the overall utility of the privacy-preserving mechanism.

The authors also introduce three strategies to safeguard the budget allocation strategy against collusion attacks. These strategies involve employing a Random Arrangement of the budget, utilizing probability distributions, and implementing the Laplace Mechanism. While these strategies are crucial for ensuring the robustness of the budget allocation approach, we will not consider them in this thesis, as they are beyond the primary focus of our research.

Pujol *et al.* [40] introduces another scenario of budget allocation known as the multi-analyst scenario. In this setup, multiple analysts have various queries to submit, and a privacy budget is assigned to them. Depending on the strategy employed, noise can be minimized, thereby enhancing data utility. While the paper primarily focuses on summary statistics, this approach could also be extended to other instances of budget allocation. This paper offers several contributions worth noting.

The first contribution lies in formulating three Desiderata, which are key characteristics defining the ideal budget allocation mechanism for the multi-analyst scenario. These Desiderata are:

- **Sharing Incentive:** Each analyst will experience either the same amount of error or less in their queries if they collaborate with other analysts using the mechanism. Participating in collaboration implies giving their share of the budget allocation.
- **Non-Interference:** Adding more analysts to the mechanism will not escalate the error for the analysts already participating and sharing their budget.
- **Adaptivity:** The mechanism adapts its strategy based on the input it receives from the analysts.
- **Computational Efficient:** Not one of the Desideratas presented by Pujol *et al.*, however, is an important criterion for the mechanism.

The second contribution is the set of design paradigms proposed for Multi-Analyst Differential Privacy (DP) query answering. These paradigms are illustrated in Figure 3.1. They are organized based on the Select Measure Reconstruct Paradigm.

- **Independent (a)**: In this common approach, each analyst receives a portion of the budget. The analysts can use their allocated budget independently to make their queries to the curator.
- **Workload Agnostic Mechanisms (b)**: An analyst can contribute their budget share to the collective mechanism. However, they cannot influence the queries that will be performed. The curator strategically selects the queries without considering input from the analysts.

- **Collect First Mechanisms (c)**: The curator aggregates all the queries that the analysts are interested in performing and has access to the budget allocated to each analyst. Based on these queries, the curator strategically prioritizes a subset of them.
- **Select First Mechanisms (d)**: In this paradigm, the analyst prioritizes the queries they want to perform before sending them to the curator. The curator then conducts a second round of strategic selection, prioritizing the queries based on the analyst's input.

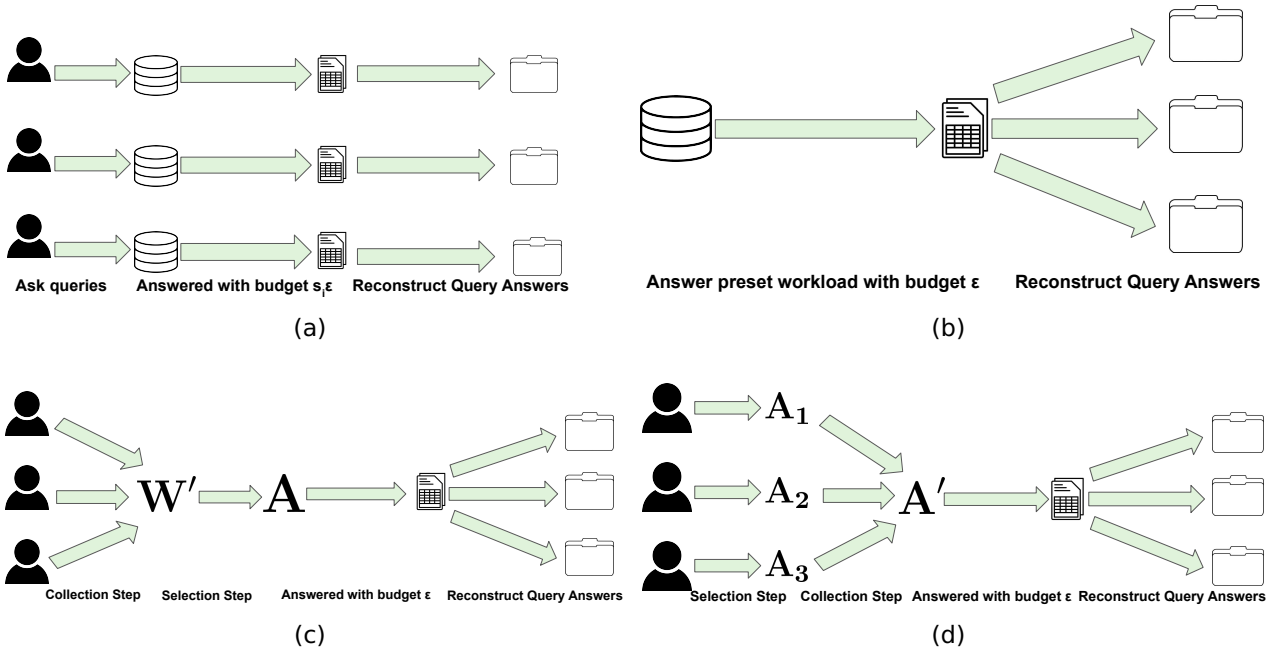


Figure 3.1 – Design Paradigms for Multi-Analyst DP Query Answering from Pujol *et al.* [40]

The authors classify a series of algorithms: Independent, Identity, Utilitarian, Weighted Utilitarian, and 0-Waterfilling, using the proposed Desiderata. This classification is presented in Table 3.1.

Desiderata/ Mechanism	**Sharing Incentive**	**Non-interference**	**Adaptivity**
Independent	x	x	x
Identity	x	x	
Utilitarian			x
Weighted Utilitarian	?		x
0-Waterfilling	x	x	x

Table 3.1 – Desiderata satisfied by algorithms

Finally, the authors propose the 0-Waterfilling mechanism, a "select first" approach that satisfies all three Desideratas. Additionally, it is an efficient algorithm. We will not consider the details here.

3.2 Budget Allocation in ML and Data Mining Scenarios

In this section, we provide a non-exhaustive overview of recent studies focusing on budget allocation strategies for DP in machine learning and data mining scenarios. Specifically, we concentrate on two key algorithms: k-cluster and Random Forest, where the implementation of budget allocation strategies is readily isolated and comprehensible. While we briefly mention a few other strategies applied to alternative algorithms, we do so with less detailed scrutiny.

3.2.1 k-Cluster

The k-Means [22] algorithm stands out as a cornerstone in unsupervised clustering tasks due to its widespread adoption and efficiency. Its primary goal is to segment a dataset into k clusters by introducing k centroids. Each centroid serves as a representative for its respective cluster, and every data point is allocated to the cluster whose centroid is nearest to it. The outcome of applying the algorithm with three clusters is depicted in Figure 3.2, showcasing its efficacy in pattern discovery and label assignment in scenarios where explicit labels are lacking. Given its popularity, a plethora of research explores the application of Differential Privacy (DP) in k-means scenarios, along with endeavors to optimize budget allocation within these contexts.

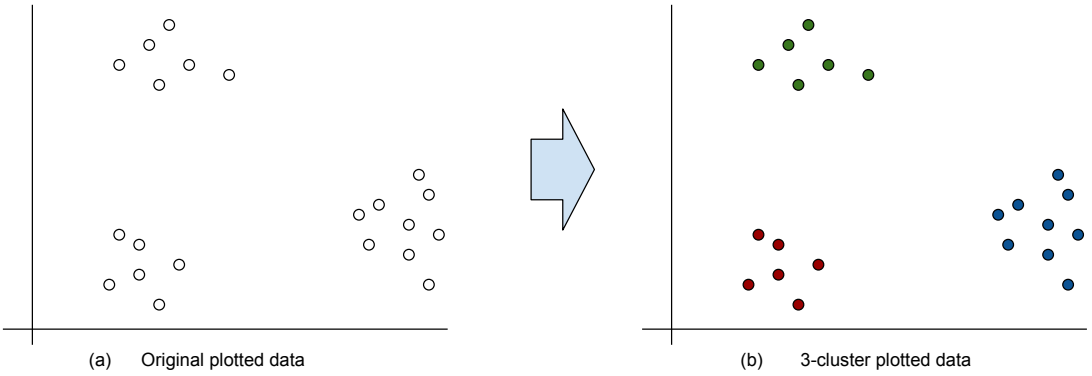


Figure 3.2 – Example of 3-Cluster application

The initial attempt to incorporate Differential Privacy (DP) into k-means clustering, known as DP k-means[3], encountered challenges in maintaining data utility due to privacy budget distribution issues. Addressing this concern requires a more efficient allocation of the privacy budget ϵ. In this regard, Li *et al.* introduced the GAPBAS algorithm [26], which employs a genetic algorithm[23] tailored to the DP k-means framework for optimal budget distribution. The distribution problem in DP k-means is NP-complete, under-

scoring the significance of leveraging heuristic-based algorithms like GAPBAS to navigate this complex scenario effectively.

While the GAPBA's study provided insights into setting initial centroids positions in k-means clustering, our focus lies in understanding the utilization of the genetic algorithm , as it closely aligns with our research objectives. The genetic algorithm aims to mimic natural selection in exploring potential solutions to a given problem. It comprises multiple steps, as Figure 3.3 depicts. The first step is encoding possible solutions into a string of bits and defining a heuristic function that evaluates the fitness or quality of each solution. Initially, the algorithm generates several random proposed solutions (Initial Population), each assigned a fitness score. The best solutions, those with lower scores, are retained, while the less optimal ones are discarded (selection). The retained solutions undergo a round of genetic operations to produce a new population, such as crossover with other surviving encoded solutions and random mutations. This process is repeated for a specified number of iterations, defined by the developer. Ultimately, only solutions closer to optimal solutions are expected to emerge, and the best one is selected.

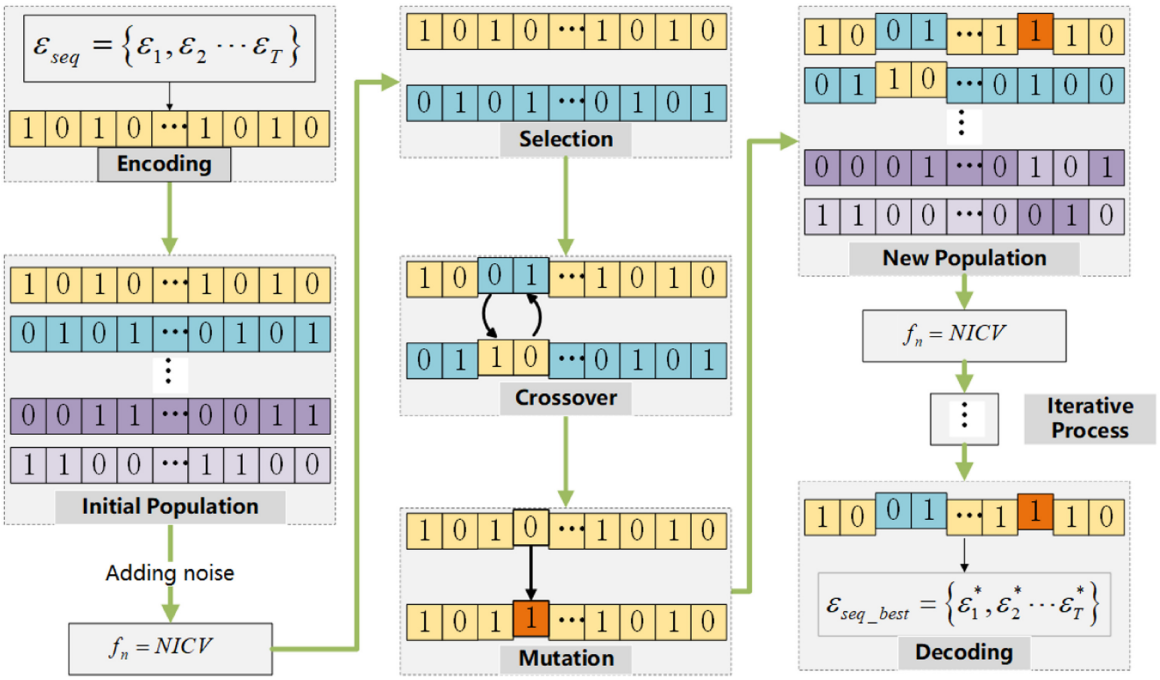


Figure 3.3 – Genetic Algorithm Process as presented in the work of Li *et al.*[26]

In this context, the problem lies in distributing a global privacy budget ϵ into T parts, where T represents the number of iterations in the k-means clustering algorithm, determined by the initial centroids' positions. Essentially, each iteration of the k-means algorithm, aimed at updating centroids, will be allocated a portion of the privacy budget ϵ. To encode these solutions, we create an array $\epsilon_{seq} = [\epsilon_0, \epsilon_1, \ldots, \epsilon_T]$, where each ϵ_i represents the budget assigned to the ith update of the centroids in the k-means algorithm. This array is then encoded in binary form.

The iterative process of the k-cluster and the genetic algorithms is different. Each possible solution in the genetic algorithm will undergo all the iterative processes of the k-cluster. In the end, a fitness score is generated for the potential solutions. The genetic algorithm will use the score to decide if the solutions will be used in the next iteration or discarded. The fitness score for GARPA is the Normalized Intra-cluster Variance (NICV), a standard k-cluster evaluation metric.

Another approach to DP k-Means involves using an arithmetic progression to manage the allocation of the privacy budget. This method, named APDPk-means, was proposed by Fan and Xu [17]. The privacy budget was evenly divided in the original DP k-means proposal by Blum et al. [3]. However, this uniform allocation presents a challenge. As the number of iterations in the k-means clustering algorithm increases to update the centroids, the overall noise also escalates, potentially compromising the quality of the k-means clustering. Fan and Xu found that earlier iterations exert a more significant impact on the resulting k-means model; thus, they should be allotted a larger share of the privacy budget to reduce noise during these critical stages. This rationale underpins their choice to employ a decreasing arithmetic progression, allocating more budget to initial iterations and progressively less to subsequent ones.

The proposal incorporates the notion of the minimum privacy budget ϵ_m introduced by Su et al. [42]. This value, ϵ_m, signifies a threshold beyond which smaller values of ϵ become overwhelmed by noise, failing to enhance centroids positions effectively. The allocation of the privacy budget unfolds as follows:

- Calculate the minimum privacy budget ϵ_m. Specify a maximum number of iterations for the k-means clustering algorithm, denoted as t_m, a value determined by the developer. Additionally, compute $t = \frac{\epsilon}{\epsilon_m}$. If $t > t_m$, the arithmetic progression proposed in the paper is employed; otherwise, a standard uniform distribution is utilized. This step ensures that the distribution of the privacy budget remains above ϵ_m; otherwise, uniform budget allocation is employed.

- For the arithmetic allocation progression, compute the sequence using ϵ_m as the first term and the total privacy budget ϵ as the sum of the progression. The difference between each term of the progression is calculated using the formula:

$$d = 2(\epsilon - \epsilon_m n)n(n-1)$$

  The progression is constructed based on the initial term ϵ_m, the difference between each term, and the number of terms t_m. In practice, it is applied in reverse, with ϵ_m as the last term and the term with the highest privacy budget as the first term.

- For the uniform distribution, allocate the privacy budget for each iteration as $\frac{\epsilon}{\epsilon_m}$.

3.2.2 Random Forest

Random Forest is a supervised machine learning algorithm commonly used for classification tasks that use an ensemble method [4]. It operates on the principle of ensemble learning, leveraging multiple decision trees, which are also supervised machine learning algorithms. Decision trees are constructed based on datasets with multiple dimensions. The non-leaf nodes in a decision tree represent decisions based on attribute values, while each leaf corresponds to a classification for an object. When classifying a new object, it traverses from the root to a leaf based on the decisions at each node, resulting in the assigned classification.

A Random Forest is built by randomly partitioning the dataset across its dimensions and constructing a decision tree for each partition, as illustrated in Figure 3.4. When classifying an object using the forest, it undergoes evaluation by each decision tree, with each tree providing a label for the object. The final assigned label is typically determined by majority voting among all decision trees in the forest.

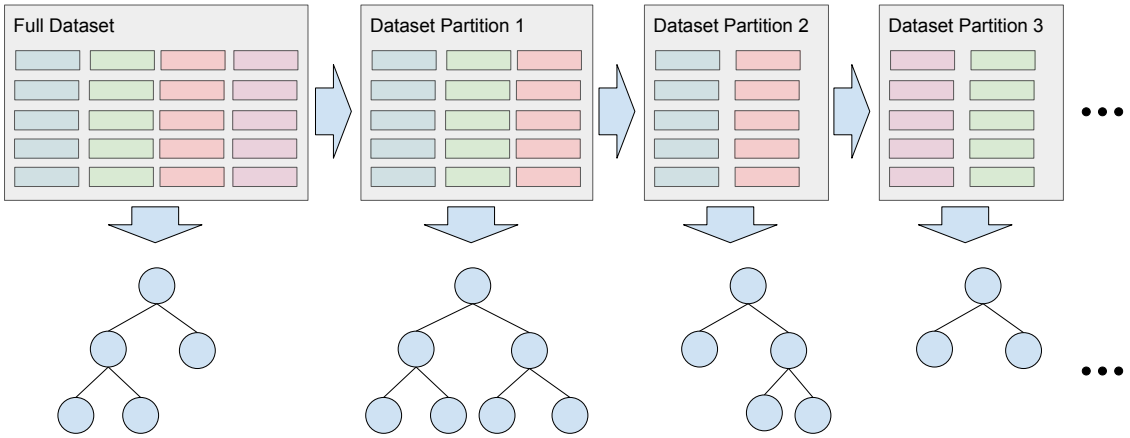


Figure 3.4 – Random Forest and partitions

Maintaining a balance between data utility and privacy becomes crucial when incorporating Differential Privacy (DP) into Random Forest algorithms. Effective budget allocation is critical to preserving a higher level of data utility. As outlined by Li et al. [25], one proposed approach utilizes out-of-bag estimation as a heuristic to determine the budget allocation for each decision tree and node within the Random Forest [39].

Out-of-bag estimation serves as a reliable validation method commonly utilized in evaluating Random Forest models. Its key advantage over cross-validation lies in its efficient use of data. Every data point contributes to model training while delivering results comparable to those obtained through cross-validation. This efficiency is rooted in bootstrap sampling, where, for each decision tree within the forest, the dataset is partitioned into two sets: the selected entries used for constructing that specific tree and the set of

unselected samples (out-of-bag). During validation, each data entry's label is assessed by aggregating votes from all decision trees in the model that included that entry out-of-bag. These assessments are then aggregated as correctly or incorrectly labeled to generate a final score based on all entries in the dataset.

The DP Out-of-bag estimation is defined using the formula:

$$B' = \frac{1}{2}\left(\frac{Y + n(\epsilon)}{Y_T} + \frac{N + n(\epsilon)}{N_T}\right)$$

In this equation, Y represents the total number of false positives in the out-of-bag classification of this tree, while N represents the number of false negatives. Y_T and N_T indicate the total misclassifications of false positives and false negatives, respectively. Additionally, $n(\epsilon)$ represents the noise added to ensure compliance with Differential Privacy constraints.

A weight can then be attributed to each decision tree using the formula:

$$W_T = \frac{1}{B'}$$

Each decision tree receives a privacy budget calculated as follows:

$$\epsilon_T = \frac{\epsilon}{T} * W_T$$

Where T is the total number of decision trees.

In Figure 3.5, the budget allocation by tree is depicted as $\frac{\epsilon}{T} \times W_T$, allocating from tree 1 to tree T. Each tree's budget is further distributed among its features (layers) in the figure. We won't explore the specifics of this secondary division.

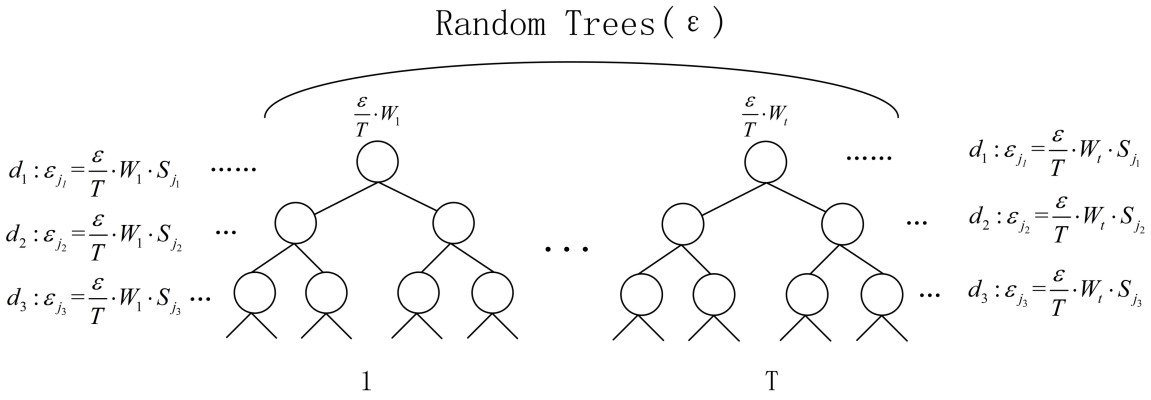


Figure 3.5 – Privacy Budget distribution in Random forest from Li *et al.*[25]

This approach strives to allocate more privacy to trees with more significant influence over the overall model, thereby reducing noise and preserving precision in these pivotal trees. Similarly, the concept extends to features, although the specific methodology for this allocation is not discussed. Certain features carry more weight in shaping

the model's decisions and are therefore allotted a more significant share of the privacy budget compared to others.

Another approach, similar to the one discussed earlier, is proposed by Hou *et al.* [20]. They introduce the DPRF (Differential Privacy Protection Random Forest) in their work. Although their research encompasses other collaborations in the realm of random forests, including hybrid decision tree algorithms, we will specifically investigate their proposal regarding privacy budget allocation.

Unlike the work of Li *et al.*, this proposal does not advocate for any deviation from the uniform distribution of the privacy budget among trees. Consequently, all decision trees in the forest receive an equal share of the total privacy budget. However, at the layer level, the budget is distributed unevenly.

The rationale behind this argument is that queries used to construct the top levels of the tree typically involve a more significant number of data entries. Consequently, the noise added to these queries less impacts data utility. Conversely, queries at the bottom nodes, closer to the leaf nodes, involve fewer data entries due to database partitioning. As a result, the noise added to these queries is more impactful. Based on this reasoning, a distribution is proposed where the bottom layers receive a more significant share of the privacy budget than the top layers.

The formula used to assign the privacy budget for an entire layer is as follows:

$$e_u = \frac{e}{w} \tag{9}$$

Here, e is the total privacy budget for that tree, and w is the weight for a specific layer.

The budget for each layer e_u will be divided among multiple queries used to construct the random forest. Each query will receive an amount of privacy budget equal to e_i:

$$e_i = e_u * (2 / (d_m - i + 1))$$

Where d_m is the tree maximum budget, i the layer, and $w_i = 2(d_m - i + 1)$. in Figure 3.6 a decision tree is presented including the budget allocation for each query in each level.

3.2.3 Others

The budget allocation problem remains a significant concern in the realm of DP, with numerous studies dedicated to addressing it. In this section, we'll provide an

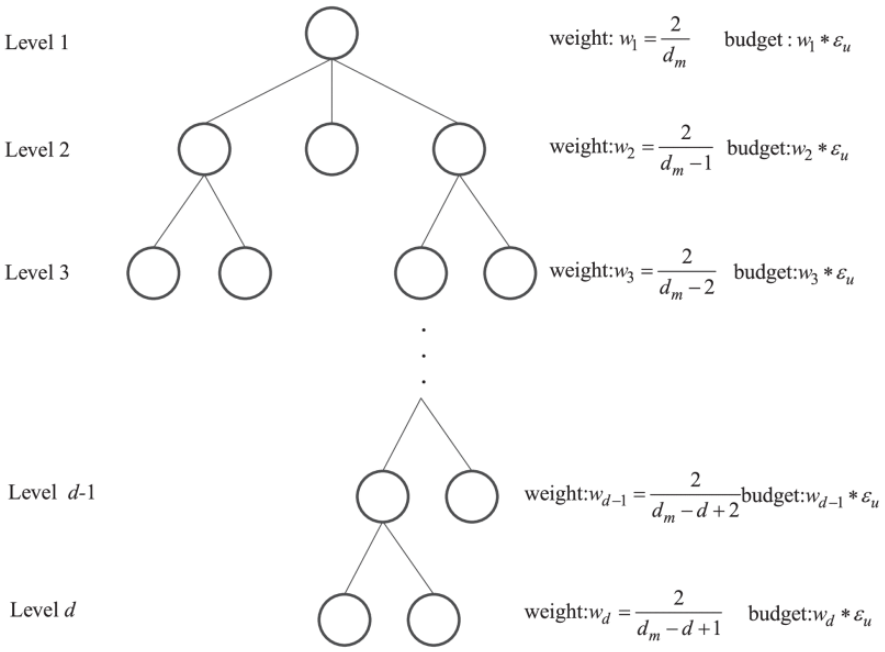


Figure 3.6 – Proposed budget distribution in a tree as proposed in Hou *et al.* [20]

overview of a few additional works, albeit in a less detailed manner compared to previous sections. Given the abundance of research in this area, it's impractical to investigate each one deeply. Moreover, our initial intention was not to provide an exhaustive review. Instead, we aim to cover highly relevant topics to the machine-learning community.

One significant topic is multi-party learning, where participants collaborate to construct a global model by sharing their locally trained model versions without disclosing their raw data. However, a crucial aspect of this approach is that it does not grant other participants direct access to the local data. Despite this precaution, there have been demonstrations indicating that privacy in multi-party learning can still be compromised [47]. Therefore, the use of DP in such scenarios holds promise and has garnered attention from academia.

A recent approach proposed by Pan and Feng [37] builds upon previous works on DP multi-party learning. Their method introduces the approximation of ρ-zero-concentrated DP and a dynamic privacy budget allocation strategy. This strategy entails injecting more noise during the initial stages of model construction and gradually reducing the amount of noise injected as the process progresses. Such innovations aim to enhance the data utility of DP multi-party learning frameworks.

Xie *et al.*[49] employ a comparable strategy aimed at enhancing DP within the Stochastic Gradient Descent (SGD) algorithm. Their approach mirrors the concept of injecting more noise in the initial steps of the algorithm and gradually reducing noise as the algorithm nears convergence. While their research focuses on applying this algorithm to backpropagation within a neural network, it's important to note that the utility of this

algorithm extends far beyond machine learning. Indeed, this DP-enhanced approach can find applications in other domains beyond the scope of traditional machine-learning tasks.

3.2.4 Final Remarks

Numerous studies in the field have shown that budget allocation is a crucial area of research in DP. However, the works discussed do not cover all research in this area. While some studies address more general DP, most focus on machine learning. In this context, we have dedicated our efforts to the budget allocation strategies for two specific machine learning algorithms: k-cluster and Random Forest.

The work presented in this thesis proposes a metric designed primarily to assist with the budget allocation problem. Consequently, this research is related to the works discussed here, but several key differences are worth highlighting. The most significant difference is that the proposed metric utilizes information provided by a developer regarding the future use of the queries answered by the mechanism, a novel approach that is not present in other works. Furthermore, the solution proposed here is specific to summary statistics, whereas the works discussed are either general applications or tailored for use in machine learning. Additionally, our research includes a study of how different differential privacy (DP) queries interact under basic mathematical operations, which differs from the objectives of the presented works.

To summarize this chapter, we present Table 3.2.4, which organizes the discussed works by paper reference, the application they target, whether they can be used in summary statistics, and if they are based on predictions of how DP queries will be used. The last two columns are crucial as they differentiate the presented works from our research. In the next chapter, we will demonstrate that our work is specifically designed for summary statistics and introduce the novel use of predictions for query usage within this context. Most reviewed works are classified as utilizing predictions for query usage because their algorithms adjust based on anticipated query applications in subsequent steps.

Also, the works discussed here provide a rich field of ideas that can be explored and incorporated into future research on the metric proposed in this thesis. The most notable ideas are:

- In the work of Bai *et al.* [1], an acceptable budget allocation threshold is proposed. This approach can be incorporated into this research with the same objective: to establish a minimum budget value that can be allocated to a query. Allocating values below this threshold would significantly impair data utility, making them ineffective.

Paper	Application	Can be used in summary statistics?	Based on prediction on how the DP queries will be used?
Bai *et al.* [1]	General	Yes	No
Pujol *et al.* [40]	multi-analyst scenarios	Yes, on multi-analyst scenario	Yes
Li *et al.* [26]	DP k-Means	No	Yes
Fan and Xu [17]	DP k-Means	No	Yes
Li *et al.* [25]	Random Forest	No	Yes
Hou *et al.* [20]	Random Forest	No	Yes
Pan and Feng [37]	multi-party learning	No	Yes
Xie *et al.* [49]	Stochastic gradient descent	No	Yes

Table 3.2 – List of Related Papers

- Additionally, in the work of Bai *et al.* [1] and Su *et al.*[42], the normalization of sensitivity is proposed, which can also be applied to facilitate the budget allocation process and the calculation of the metric in this research.
- The GAPBA study [26] employs a genetic algorithm to address the budget allocation problem effectively. Similarly, in our case, a genetic algorithm could be utilized in the same manner. Here, the fitness function would result from the proposed metric, with a good fit representing a budget allocation proposal that achieves a high metric value.
- Xie *et al.* [49] proposed a stochastic gradient descent (SGD) algorithm. Using a brute force strategy with our metric as the heuristic for better results would render the problem NP-complete. However, applying gradient descent with our metric can help find an optimal budget allocation in a much less computationally costly manner.

4. PROBLEM STATEMENT

In this chapter, we present the research problem, outlining its significance, contextual background, and our approach to addressing it. The chapter is structured as follows: first, we provide context for the problem by describing the environment in which it arises. Next, we articulate the issue or problem that our research endeavors to resolve and the opportunities it presents. Subsequently, we explore the potential impact of our research from various perspectives. Finally, we delineate our research objectives and the overall structure of our study.

4.1 Context

Differential Privacy (DP) has emerged as the leading approach for protecting individuals' privacy in datasets, offering more robust and transparent definitions compared to other methods [44]. Its significance has surged recently due to the increasing amount of data collected about individuals and the growing pressure from public opinion and legislative measures. A few of the most important legislation are:

- **General Data Protection Regulation (GDPR) (EU):** Enacted in 2018, GDPR governs the processing and handling of personal data of individuals within the EU, as well as EU citizens outside of the EU.
- **California Consumer Privacy Act (CCPA) (USA):** Implemented in 2018, CCPA is a privacy law applicable in the state of California. It grants residents rights regarding their personal information and imposes obligations on businesses handling private data.
- **Health Insurance Portability and Accountability Act (HIPAA) (USA):** Enacted in 1996, HIPAA protects sensitive patient data within the healthcare sector. It applies to entities such as hospitals, health plans, and healthcare providers.
- **Personal Information Protection and Electronic Documents Act (PIPEDA) (Canada):** Established in 2000, PIPEDA is a federal law regulating the collection and use of personal data in commercial activities.
- **Privacy Act (Australia):** Introduced in 1988 and amended in 2014, this legislation governs the handling of personal data by both government agencies and private organizations in Australia.
- **General Data Protection Law (LGPD) (Brazil):** Implemented in 2020, LGPD regulates the processing of personal data in Brazil.

- **Data Protection Act 2018 (UK):** The UK's Data Protection Act 2018 complements GDPR and outlines additional provisions related to data protection and processing within the UK.

For instance, a survey conducted by KPMG [1] reveals that 70% of business leaders increased the collection of consumer personal data over the past year. Moreover, it shows that 86% of the general population in the United States considers privacy a growing concern. The Key findings released by KPMG are in Figure 4.1. This data underscores the critical role of Differential Privacy in today's data-driven world.

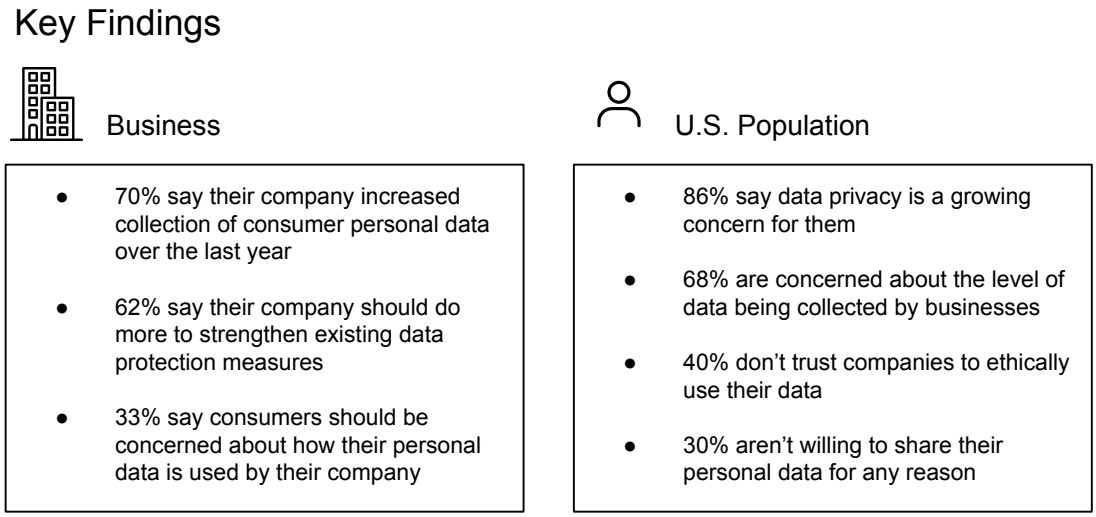


Figure 4.1 – KPMG Key Findings

As a result, many applications have begun to incorporate Differential Privacy (DP) as a means of safeguarding privacy. Major companies such as Microsoft, Apple, Google, and others have integrated DP into numerous applications. Moreover, DP principles are applied in areas where the intention to adhere to them may not be explicit. For instance, randomized response, a data collection method widely used in Social Sciences, aligns with the DP definition[7] [48]. In other applications, the adherence to DP is very natural, as in Federated Learning, a privacy-preserving method in Machine Learning [36].

However, as discussed in previous chapters, a significant hurdle to adopting Differential Privacy (DP) lies in its inherent nature. Compliance with DP requires modifications to the original data via the introduction of noise. This noise, an integral part of DP, serves to protect an individual's privacy. Yet, the introduction of this noise leads to a reduction in the utility of the modified data. This trade-off between privacy and utility is a pivotal area of research in Differential Privacy. Striking an incorrect balance can adversely affect the performance of applications.

[1] https://kpmg.com/us/en/articles/2023/bridging-the-trust-chasm.html

4.2 Issue and Opportunity

Research efforts aimed at enhancing data utility frequently concentrate on generating or processing queries. However, how data is utilized in the post-processing stage often receives less attention. This potential gap in the current research landscape underscores the need for more focus on optimizing data use during post-processing.

A specific aspect that can be exploited to enhance utility by fine-tuning the budget allocation problem involves considering how queries from the same DP application interact with each other. These queries often interact through mathematical operations.

Consider, for example, a database protected by DP containing information about vision health. An anonymized query retrieving the number of people with myopia can interact with another anonymized query retrieving the total number of people in the database to compute a ratio. This interaction will result in a varying amount of noise added, depending on how the budget ϵ is allocated to each query. An examination of how these interactions influence the added noise of the operation will be presented in subsequent chapters and constitutes a significant component of this thesis.

An opportunity has emerged from the interaction and the results of a preliminary study. A developer, aware of the behavior of interactions and anticipating how an analyst will utilize anonymized queries, especially when they interact with each other, can leverage these interactions to minimize the total amount of noise added, thereby enhancing data utility. Consequently, there is a need to develop tools that can seize this opportunity effectively.

4.2.1 Attack model

The idea of DP is to protect an individual's or group's privacy. However, different attack vectors are considered for various applications and approaches. Our proposal considers the attack model represented in Figure 4.2. In it, we have the individual who willingly shares his date in an application protected by DP to compose a dataset of multiple individuals. This DP Application is assumed to be private, where no data can leak. As part of the application there is the figure of the Curator, characteristic of DP scenarios. In this model, the curator is assumed to be trustworthy. The curator can freely interact with the dataset without any restrictions.

The Curator's and the application's objective is to release several summary statistics to the public. Since these statistics will be public, they need to be protected by DP to prevent an identification of an individual's presence in the dataset. Operations on the released summary statistics are the attack vector our proposal aims to protect. In this case,

the released summary statistics can be accessed by ordinary analysts with good intent on using the data and by a malicious analyst who intends to infringe on individuals' privacy. By using DP, our approach should be capable of frustrating any attempt by the malicious analyst.

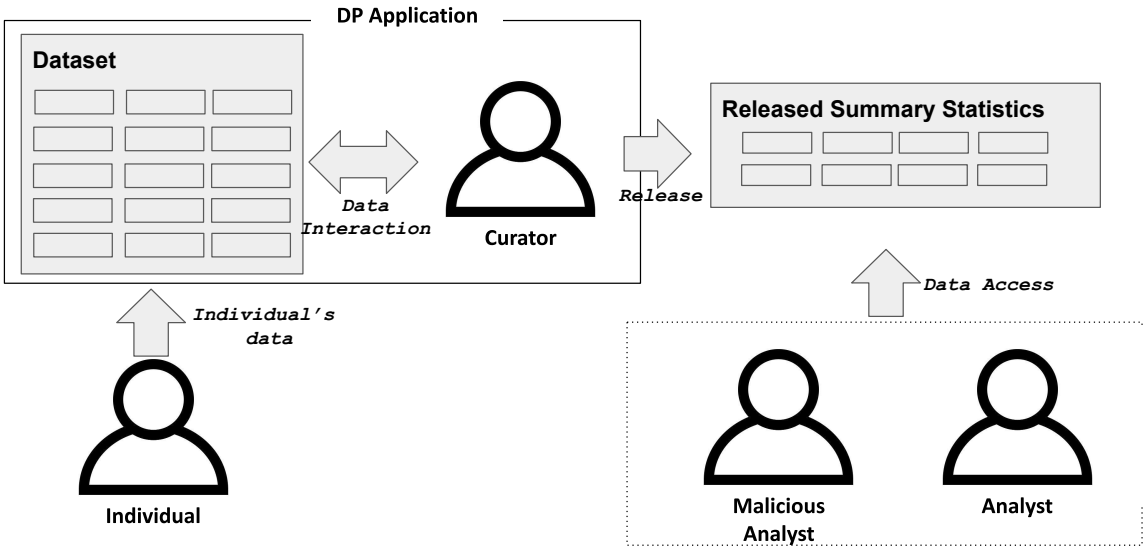


Figure 4.2 – Attack Model

Additionally, the literature presents various attack models where DP protection is applied to each interaction with the dataset. For instance, the contrasting works compared to ours of [25] *et al.* and [20] *et al.* demonstrate DP safeguarding each interaction with the dataset during the production of the machine learning model.

4.3 Relevance

The issue of budget allocation, governed by the parameter ϵ, is a pivotal aspect of Differential Privacy (DP) [2] [26] [25] [19]. This makes it a significant area of focus within the research community. The parameter ϵ not only regulates privacy but also influences data utility. Consequently, strategies are often customized for specific applications, including but not limited to machine learning techniques. This interplay between privacy and utility underscores the complexity and importance of optimal budget allocation in DP.

The research conducted in this thesis, which will be presented in the subsequent session, is situated within this primary area of differential privacy research. The proposed solutions are currently theoretical, and further work is required to make them practical in real-world scenarios. Presently, the focus is on specific scenarios of Global Differential Privacy and the release of summary statistics. Future discussions about these scenarios and potential improvements to this research will be presented in Section 6. However, the following benefits can be highlighted as positive impacts of our research:

- **Data Utility:** The primary benefit of this research is a proposal that, in specific scenarios, enhances data utility by reducing the noise generated from the interaction of various anonymized queries.
- **Privacy Protection:** Our proposal improves utility while maintaining the same level of privacy, as it does not alter the global ϵ value. This will be demonstrated in the forthcoming chapters.
- **Adaptability:** Our proposal is focused on summary statistics using Global Differential Privacy. It can be readily adapted to any similar applications. The only prerequisite is a knowledgeable analyst who understands the context in which the data will be used.

We want to emphasize the impact of our research on various stakeholders:

- **Organizations:** Often, the loss of data utility in DP scenarios is substantial enough to render several applications unfeasible and data incomprehensible. Improvements in data utility can mitigate this issue, making such applications feasible. Importantly, there is no increase in the risk of privacy leakage, a significant concern for organizations as it can lead to legal complications.
- **Individuals:** For individuals, the application of our research is transparent. There is no increase in the risk of privacy, which is likely the primary concern for individuals. As previously mentioned, in the context of organizations, individuals may also benefit from new private applications and data.
- **Academia:** Our research contributes significantly to academia. We propose a novel approach to improve data utility in summary statistics based on predictions of how the anonymized summary statistics will be used by the analyst in a DP scenario. This includes a comparative analysis of how anonymized queries behave under different mathematical operations. Therefore, we present a fresh avenue for research in DP.

4.4 Research Objectives and Structure

The central objective of this thesis is to address the following research question:

"*What is the impact of a newly proposed metric for Differential Privacy on the trade-off between privacy and data utility?*"

This research question leads to the following hypotheses:

H_0: The use of specific metrics for anonymizing datasets **does not** significantly improve data utility and privacy in practical applications. H_1: The use of specific metrics

for anonymizing datasets **can** significantly improve data utility and privacy in practical applications.

Enhancing the trade-off between privacy and data utility is a crucial topic in the field of Differential Privacy. Although this research is still in its early stages, initial results suggest that the proposed metric has the potential to positively impact this balance, particularly in the context of summary statistics and Global Differential Privacy.

Based on the research question, we have set and achieved the following objectives:

- Identify a gap in the current field of DP where introducing a new metric could be beneficial.
- Develop a new metric that can be used in Differential Privacy scenarios to potentially improve the privacy data-utility trade-off.
- Evaluate the proposed metric to ensure its positive impact on DP.

The research was organized into three parts, highly correlated to the research objectives. The next chapter will provide a detailed description of each part, including the adopted methodology.

- **Part One:** We demonstrate that in our target scenario, changes in noise addition occur based on how anonymized queries are used in mathematical operations after the anonymization process. This finding is pivotal for our research, as the proposed metric exploits this fact to improve data utility.
- **Part Two:** We present a metric that exploits how the anonymized query is used. This includes a mathematical definition. The metric depends on a new role in the DP scenario, the Developer, who is tasked with predicting how an analyst will use the anonymized queries. The accuracy of these predictions is crucial for the performance of the metric.
- **Part Three:** We evaluate the proposed metric in a theoretical scenario to confirm the possibility of a positive impact on a DP scenario. The performance highly depends on the Developer's ability to predict query usage. Thus, the objective of this part was to assert that it is possible to have a positive impact on data utility, depending on the Developer's performance.

5. CREATING A METRIC FOR ASSESSING DATA UTILITY IN SUMMARY STATISTICS UNDER DIFFERENTIAL PRIVACY

In this chapter, we discuss the core of our thesis, presenting our research and its results. Our work is organized into three main parts, as outlined in the previous chapter. We begin by examining the behavior of anonymized queries using DP when integrated with simple mathematical operations. This study reveals a potential avenue for achieving an improved privacy-utility trade-off. Next, we introduce a novel metric designed to measure the quality of budget allocation. This metric takes into account a forecast of data usage, which is a unique aspect and a significant contribution to our work. Finally, we assess the performance of the proposed metric, providing a comprehensive evaluation of its effectiveness. Each component contributes to our understanding of the subject and brings us closer to our research objectives. We hope that our findings will shed light on new possibilities and inspire further exploration in this field.

5.1 Part 1: Analysis of the impact of basic mathematical operations on queries created using DP

The primary aim of this study is to investigate the impact of implementing DP on the utility during the merge of outcomes from two distinct mechanisms into fundamental mathematical operations, with the privacy budget apportioned across varying ratios. The underlying hypothesis posits that diverse distributions of the privacy budget between interacting queries can either enhance or diminish the utility of the resultant data. This investigation sheds light on the behavior of noise introduced by differential privacy under the specified conditions and, more crucially, justifies the introduction of a metric designed to aid in allocating the privacy budget by considering the results presented here. The details of this metric will be elucidated in the subsequent study within this chapter.

5.1.1 Experiment Description

To assess the impact of noise in DP within our mathematical operation scenarios, we have designed an experiment detailed in Figure 5.1. We will examine each step individually. Our hypothesis posits that varying the distribution of the privacy budget may either augment or diminish the utility of the resultant data. We employ the methodology delineated in Section 2.6 to measure this data utility.

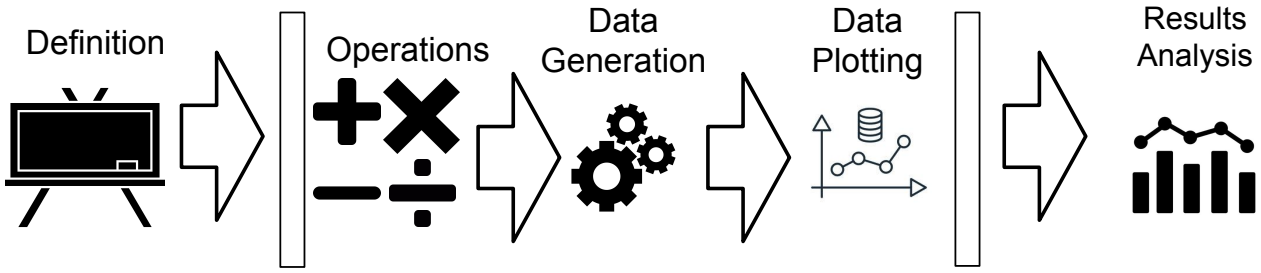


Figure 5.1 – Steps to the Execution of the Experiment

- **Definition:** The experiment entails the utilization of two mechanisms, namely **Mechanism A** and **Mechanism B**. The aim is to assess the utility achieved through operations incorporating both queries. To quantify and analyze the utility of the data, the method employed is the one outlined in Section 2.6. The operations involving both mechanisms will be the basic mathematical operations: Addition, Subtraction, Multiplication, and Division. All operations in our study will use **Mechanism A** as the first argument and **Mechanism B** as the second argument. Each operation is analyzed individually by systematically varying the values of variables associated with **Mechanism A** and **Mechanism B**.

  Both mechanisms encompass **intrinsic** and **external** variables.

  - **Intrinsic** variables are inherent to the query conducted within the database and the database itself. The intrinsic variables include **Result of Query A**, **Result of Query B**, **Sensitivity of Mechanism A**, **Sensitivity of Mechanism B**. Here, "value" denotes the outcome of the query conducted by the Curator before adding noise.
  - **External** variables are defined by the Differential Privacy (DP) scenario developer. The external variables consist of **Privacy budget allocated for Mechanism A**, and **Privacy budget allocated for Mechanism B**.

  DP scenarios generally operate within a specified total privacy budget. The defined total privacy budget is one ($total\ \epsilon = 1$), which aligns with a conservative yet robust privacy protection standard [34]. The external variables always respect the restriction in Equation 5.1.

$$BudgetofMechanismA + BudgetofMechanismB = total\ \epsilon \tag{5.1}$$

  In our experiment, variables from both intrinsic and external groups were adjusted one by one.

  The intrinsic variables **Result Query A** and **Result Query B** start at a base value of 10, then are increased logarithmically to 10, 100, 1000, and 10000. Similarly,

Sensitivity of Mechanism A and **Sensitivity of Mechanism B** begin at 1 and are increased logarithmically to 1, 10, 100, and 1000.

The external variable **Budget of Mechanism A** starts at 0.01, increasing by 0.01 increments until reaching 0.99. Due to the constraint in Equation 5.1, changing the **Budget of Mechanism A** external variable affects the **Budget of Mechanism B** variable.

Each combination of intrinsic and external variable changes will undergo 10,000 iterations. The utility measured for each combination will be obtained as the average of these 10,000 executions. The resulting data on utility will then serve as the basis for our analysis.

We determined this sample size for an infinite population, considering a Confidence Level of 99%, a Margin of Error of 1%, and a Standard Deviation of 50%. The calculated sample size was 9,604, which we rounded up to 10,000 for practical purposes.

- **Data Generation:** A script made in Python will be used to generate the data based on our definition. Important libraries used include Numpy version *1.21.5*, Seaborn version *0.12.2*, Pandas version *2.0.3*, and Matplot. The Numpy is particularly important because it generates the noise using the La Place distribution. There is a random component in the noise generation; for replication, the seed used was zero. The hardware does not impact the results.

- **Data Plotting:** The Seaborn and Matplotlib libraries are used for data visualization. The x-axis represents the **Privacy budget allocated for Mechanism A**, while the **Privacy budget allocated for Mechanism B** can be inferred from Equation 5.1. The y-axis denotes the utility variable. The aim is to observe how noise behaves as budget allocation changes. Multiple graphs are generated, each depicting different combinations of variable parameters. In total, there are 64 graphs, although not all will be presented here. However, they are accessible on the research's GitHub repository[1] along with the scripts used.

- **Results Analysis:** The analysis and findings are presented in the last part of the experiment. This step will be discussed in the following subsection.

5.1.2 Results

This subsection unveils the results obtained for each operation, discussing them individually within their respective subsections. To streamline data presentation, graphics

[1] https://github.com/conseg/TheImpactofDifferential Privacyondatautilityinfundamentalmathematicaloperations

are employed. Although a large number of graphics (68 in total) were generated due to the multiple variables, this section showcases just the most relevant ones for each mathematical operation. At the end, a broader discussion is presented.

In the preceding experiment description, we outlined the utilization of two mechanisms. In the ensuing discussions, we must discuss specific points of each mechanism, namely the query's original value and the noise added to it. In this regard, we are going to use the following equations similar to the ones presented in Section 2.6: For the "Mechanism A" $aA = qA + nA$; For "Mechanism B" $aB = qB + nB$. Where Capital "A" and "B" mean which query it is referring to, "a" means answer, "q" the original query value, and "n" noise.

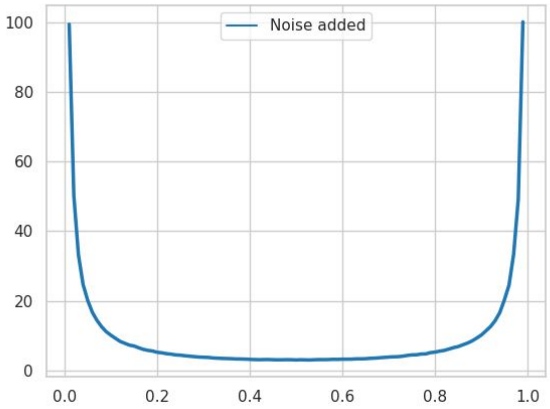
Addition

The addition operation takes the form $(qA + nA) + (qB + nB)$. Figure 5.2 A depicts the baseline cases. Observing the graph, the optimal distribution of the budget epsilon, which results in the least added noise, is when the budget is perfectly divided between Mechanism A and Mechanism B. Conversely, at the extremities of the graph, where one query benefits from more budget at the expense of the other, more added noise is generated. We speculate that the phenomenon is linked to how noise is generated. In the mechanism for noise generation, the Laplace distribution has a scale parameter **s**. The value of the parameter **s** is calculated using $s = \frac{sensitivity}{episilon}$ since at the borders. We are decreasing the budget epsilon of one of the queries with a $\lim_{episilon \to 0} = \infty$, where the tendency to infinity is generating a high amount of noise added. This behavior at the border is, in general, repeated in other operations.

Changes in the qA or qB do not impact the amount of added noise for addition operations. This lack of impact is evident in Figure 5.2 B, which seems identical to Figure 5.2 A.

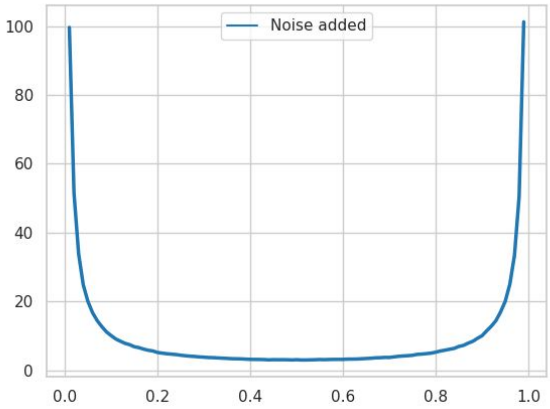
Sensitivity, on the other hand, significantly impacts the added noise, as illustrated in Figure 5.2 C. One notable observation is that increasing the sensitivity of one mechanism, Mechanism A in this graph, produces a proportional increase in the added noise. Another aspect is the change from the optimal point to divide the budget epsilon from 0.5 in the baseline to the right in the graph. To keep it optimal, the budget must prioritize the Mechanism with more sensitivity, Mechanism A in this case. However, it's crucial to note that excessively reducing the epsilon for Mechanism B may still lead to suboptimal outcomes. The value of noise added when the budget of B is near 0 still tends to infinity. It is less outstanding than the opposite when the budget for A tends to 0.

(A)



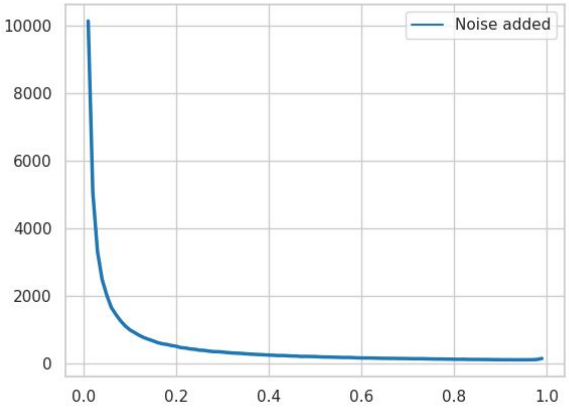
Sensitivity A: 1.0
Sensitivity B: 1.0
Result A: 10.0
Result B: 10.0

(B)



Sensitivity A: 1.0
Sensitivity B: 1.0
Result A: 10000.0
Result B: 10.0

(C)



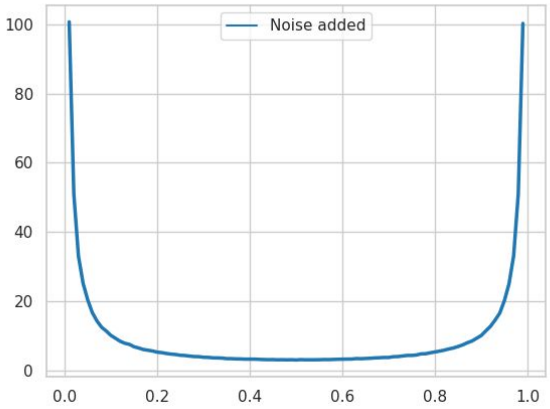
Sensitivity A: 100.0
Sensitivity B: 1.0
Result A: 10.0
Result B: 10.0

Figure 5.2 – Addition operation graphics

Subtraction

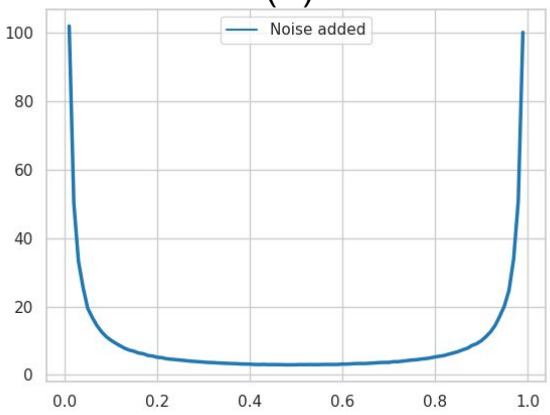
The subtraction operation is in the form $(qA + nA) - (qB + nB)$. This operation is similar to the addition, and we can take the same observations previously made. The noise increases at the borders of the graph, and we speculate that are for the same reason; Increases in the value of qA or qB do not change the amount of noise added; Changes in the sensitivity increase the amount of noise to the left of the graph in case of increases in Mechanism A, or to the right in case of Mechanism B. It also displaced the optimal point from the center to the opposite side which had an increase in the noise added.

(A)



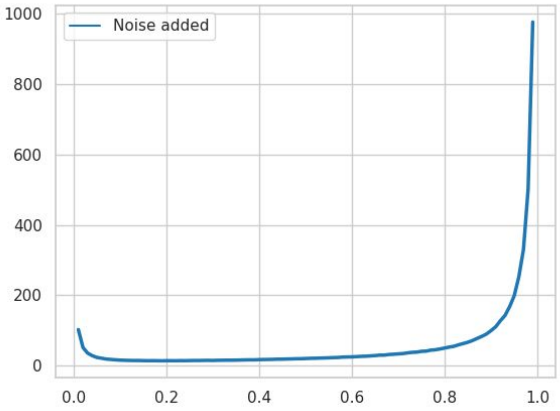
Sensitivity A: 1.0
Sensitivity B: 1.0
Result A: 10.0
Result B: 10.0

(B)



Sensitivity A: 1.0
Sensitivity B: 1.0
Result A: 10.0
Result B: 1000.0

(C)



Sensitivity A: 1.0
Sensitivity B: 10.0
Result A: 10.0
Result B: 10.0

Figure 5.3 – Subtraction operation graphics

Multiplication

The multiplication operation follows the format $(qA + nA) * (qB + nB)$. Similar to previous operations, at the edges of the graph, when one of the queries has a minimal budget epsilon allocation, the added noise substantially increases, as depicted in Figure 5.4 A. Note that the noise added values are much greater than in addition and subtraction.

Unlike previous operations, changes in qA or qB lead to modifications in the amount of noise added. Figure 5.4 B illustrates the added noise when qB is increased to 100. The increase of qB seems to increase the amount of added noise, mostly when

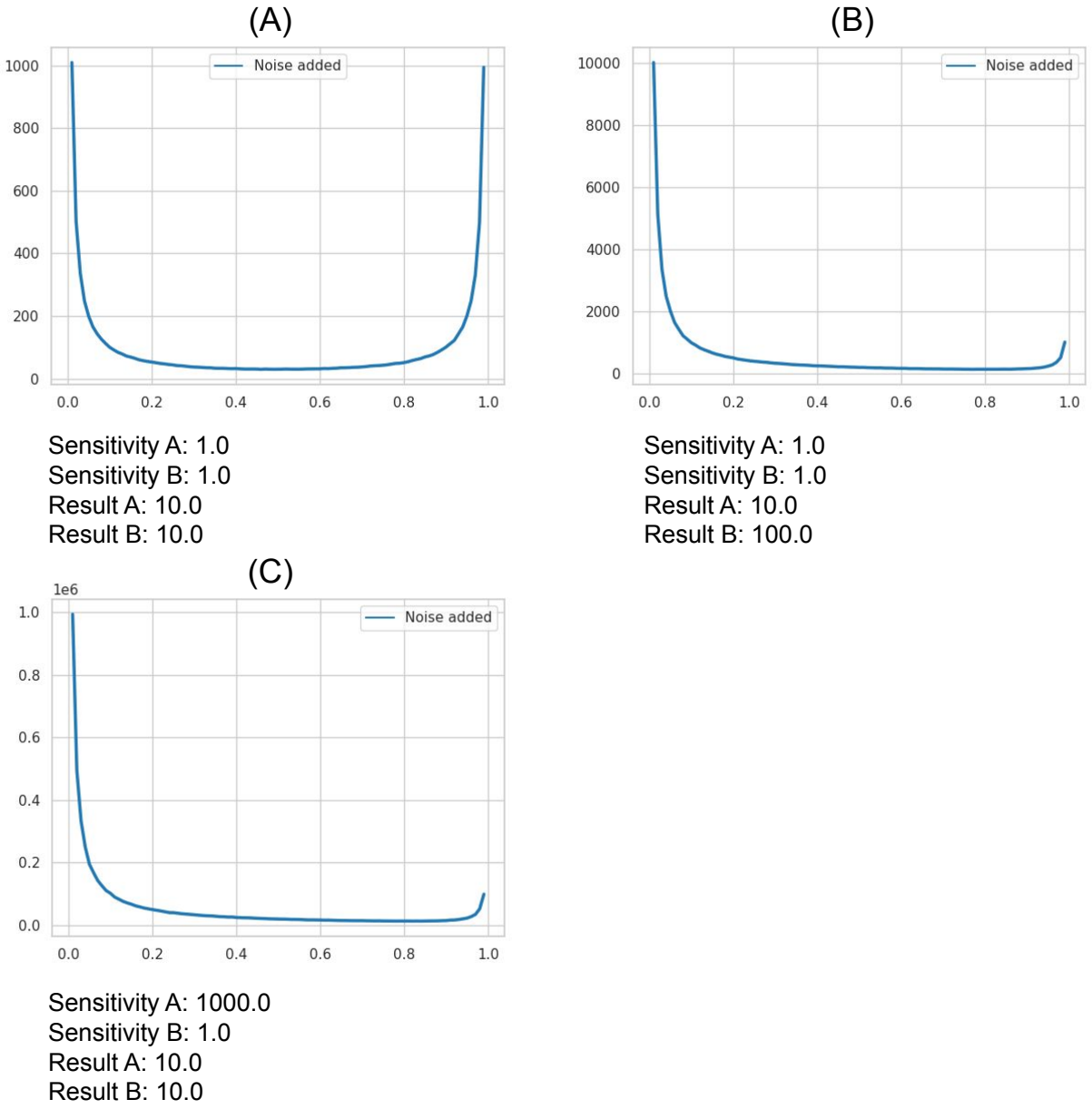


Figure 5.4 – Multiplication operation graphics

the budget allocated for Query A is smaller. The optimal distribution of the budget shifts to the right of the graph. The key conclusion is that an increase in the value of a query will result in the optimal budget distribution aligning closer to it when more budget is allocated to it, albeit not exclusively. This phenomenon can be understood by applying the distributive law to the original operation $(qA+nA)*(qB+nB) \Rightarrow qAqB+qAnB+nAqB+nAnB$. Remember that an increase of the budget epsilon for a query decreases the values of the noise in that query, nA and nB in our case, and vice-versa. For this scenario, the value of a Query interacts multiplicatively with the noise of the other query, $qAnB$ and $nAqB$. Increases in the value of a query increase the impact that the noise of the other query causes. Thus, decreasing the budget for a query and increasing the value for the other greatly impacts the added noise measurement.

Similar to previous cases, adjusting the sensitivity of one of the Queries increases the amount of noise added for that query result. Moreover, assigning a minimal portion of the budget to the query exacerbates the noise added. Examining the graph in Figure 5.4 C, the noise added seems to increase in a multiplicative manner. The optimal budget distribution point appears to shift towards the right, where the allocated budget for Query A is larger. However, maximizing the budget for Query A to the detriment of Query B will also increase the amount of noise added. This behavior is similar in the other operations.

5.1.3 Division

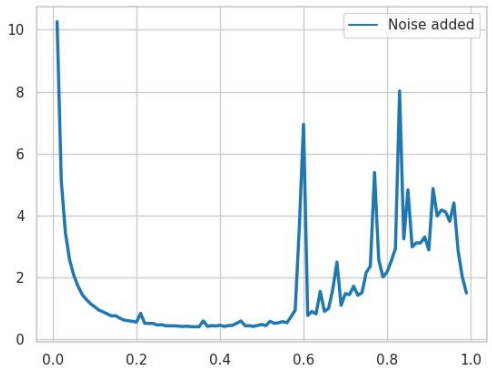
As represented by Equations 5.2, the division operation exhibits different behavior compared to previous operations. Examining the baseline case in Figure 5.5 A, while the left border shows an increase in the amount of noise added for this operation, the right side seems to add less noise. This contrasts with other operations, where the noise increases symmetrically at both borders. The reason appears to be rooted in the imbalance between the noise for the numerator ($qA + nA$) and the denominator ($qB + nB$). On the left side of the graph, the noise for the numerator is greater due to the smaller budget allocated for Query A, increasing nA. Conversely, the noise for the denominator is decreased due to the larger budget allocated for Query A, which reduces the value of nB. As a result, the numerator increases while the denominator decreases, leading to a larger overall value and, consequently, more added noise. The right side, however, does not exhibit the same effect. Increasing the denominator decreases the result, resulting in less added noise. This has a limit, as we can see in the graph where the noise increases again at the end of the graph on the right. The optimal point for budget distribution seems to be in the middle of the graph, where the budget is equally split.

$$\frac{(qA + nA)}{(qB + nB)} \tag{5.2}$$

Several spikes in the graph seem to occur randomly. This behavior can be attributed to the denominator. Depending on the noise nB generated, it can interact with the value qB, causing it to be near zero. With a denominator close to zero, the result of the division converges to infinity, significantly increasing the amount of noise added. Since it depends on the randomly generated noise, it exhibits unpredictable behavior and can occur anytime.

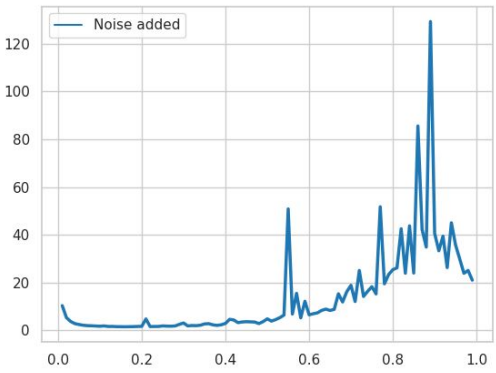
Unlike previous operations, changes in the qA and qB have different impacts on the amount of noise generated. Changes in the Value A (qA) can be observed in Figure 5.5 B. Focusing the budget on Mechanism B seems to maintain a low and predictable amount of added noise, consistent with observations from Figure 5.5 A. Increases in the budget for

(A)



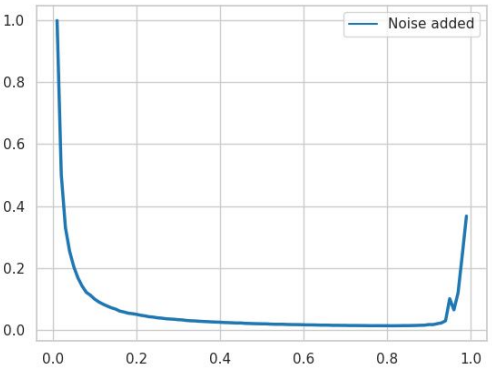
Sensitivity A: 1.0
Sensitivity B: 1.0
Result A: 10.0
Result B: 10.0

(B)



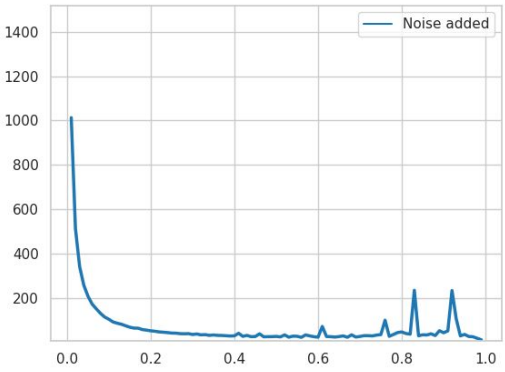
Sensitivity A: 1.0
Sensitivity B: 1.0
Result A: 100.0
Result B: 10.0

(C)



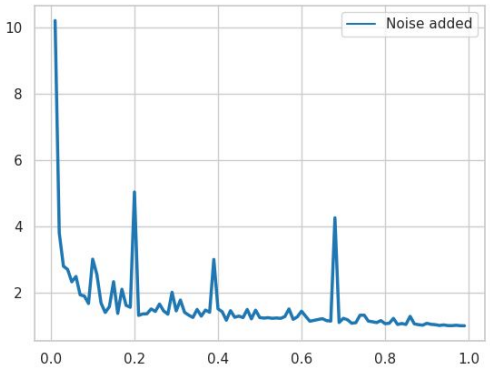
Sensitivity A: 1.0
Sensitivity B: 1.0
Result A: 10.0
Result B: 100.0

(D)



Sensitivity A: 1000.0
Sensitivity B: 1.0
Result A: 10.0
Result B: 10.0

(E)



Sensitivity A: 1.0
Sensitivity B: 100.0
Result A: 10.0
Result B: 10.0

Figure 5.5 – Division operation graphics

Mechanism A seem to raise the added noise and spike frequency in the graph. This can be justified by the fact that increasing qA also increases the amount of noise generated

when the denominator is close to zero. On the other hand, increases in Value B (qB) in Figure 5.5 C seem to decrease the noise added at all points of the graph compared to the baseline. The main cause can be linked to the increase in the value of the denominator, which, when interacting with the numerator, yields a smaller value. The spike frequency also reduces as the sensitivity for Mechanism B remains at 1 (SenB), and the randomly generated noise nB is insufficient to decrease the value of qB to near zero. The result is a much smoother graph.

Regarding changes to the sensitivity ($SenA$, and $SenB$). Analyzing Figure 5.5 D, modifications in $SenA$ seem to increase the amount of noise, mainly at the left of the graph when the budget allocated to Mechanism A is smaller. The natural cause is the increase in the magnitude of the noise nA, which augments the nominator while the denominator (Mechanism B) remains smaller due to no change in $SenB$, and an increase in the budget for $MechanismB$, which diminishes nB. Shifting the budget to $MechanismA$, the right side of the graph decreases the amount of noise added. It is theorized that the increase in funding for $MechanismA$ decreases the amount of noise in nA, thereby reducing the overall added noise. Spikes persist, caused by the denominator coming close to zero. The graph of Figure 5.5 D shows the case when $SenB$ is increased to 100. The amount of noise added is much smaller than the opposite case when $SenA$ is increased. Unlike other cases, the smaller amount of noise seems to be when the budget is maximum allocated to Mechanism B, decreasing the noise nB. The amount of spikes is very prominent in all the graphics, probably caused by a higher chance of the denominator Query B aB coming close to zero.

5.1.4 Discussion

One of the main objectives of this study is to determine whether budget distribution can impact the utility of post-anonymization operations that involve multiple mechanisms. In our context, utility refers to reduced **added noise**, as defined earlier. Our study yields positive results, focusing on a specific scenario where just two queries are used in basic mathematical operations.

The positive results bring the possibility of leveraging this behavior to enhance utility in practical applications. While further investigation is necessary, there may be specific scenarios where optimizing budget distribution can improve the utility of subsequent queries. In the following study, we will present a metric developed with this in mind [35].

However, this study can benefit from an expansion to encompass more general cases. Exploring complex operations involving multiple queries is one avenue that requires further attention. The possibility of adding a noise prediction based on the operation and the budget distribution without the need for repeated calculations to find the

optimal distribution, as done in our work, would be invaluable if feasible. The study works only in the canonical form of differential privacy; other forms, such as approximate differential privacy, would broaden the applicability of the results, bringing them closer to real-world scenarios where such forms are more prevalent. Moreover, real-world cases are also of interest.

Specifically, the behavior of some operations, like addition and subtraction, can also be used as an advantage by a developer. Division, on the other hand, exhibits more complex behavior. One aspect is the spikes. Addressing this aspect could reduce the likelihood of a denominator approaching zero, decreasing added noise. However, any such approach must be carefully evaluated to ensure it does not compromise privacy.

5.1.5 Final consideration of this study

In this study, we examine the impact of differential privacy on data utility when two distinct mechanisms interact in basic mathematical operations. Our evaluation, conducted through an experiment measuring **added noise** in each operation, provides insights into data utility. The findings reveal that variations in variables can strongly influence data utility, with each mathematical operation exhibiting different behaviors. Furthermore, we demonstrate that the budget allocation also impacts the data utility, indicating that developers should change the variables properly to optimize data utility and anticipate how the mechanisms will be used later.

As a future direction, this research could be extended to more complex cases of operations and real-world applications. Furthermore, additional research is needed to refine or devise budget allocation strategies to enhance data utility.

5.2 Part 2: A Metric proposal to improve data utility

In this section, we introduce a novel metric for budget allocation that quantifies the data utility of a specific budget distribution. This metric considers the budget assigned to each query released to the analyst as a summary statistic. It also incorporates predictions of potential future usage of the released queries. The predictions are introduced by a new role in DP scenarios, the developer. The inclusion of these predictions is an innovation in the field, accounting for scenarios where the interaction of two or more queries may cause their inherent noise to intermingle. Our proposed metric draws heavily on the findings from our previous section’s study of such interactions through basic mathematical operations.

To articulate the metric, we begin with a problem statement that outlines the scenarios for its application. Following this, we provide a mathematical definition of the metric. Subsequently, we briefly discuss the definition of metric used in this work. We conclude with some final remarks on the implications and potential applications of the metric.

5.2.1 Problem Statement

The **developer** is an entity that wants to release several summary statistics to the public using DP. In such situation, there will be a privacy parameter budget ϵ that is defined externally which must be respected by the **developer**. This budget needs to be divided among all the statistics that will be made available. The **budget allocation** between different statistics can be done in different ways. As already discussed, it is a pivotal part of our work and for the field. By the Sequential Composition property of DP, all ways of sharing the privacy budget keep the same privacy guarantee as presented in Section 2.2. When releasing the data, the **developer** intends to build a DP scenario, including its budget allocation that helps in providing the best possible quality of data to an **Analyst** that will use the statistics.

The **Analyst** is an entity that intends to consume the statistics that will be released to the public. It will use these statistics in equations to generate new insights about the data. The distinction between **Statistics**, a result from a query in the database with the DP noise added, and the **Equations** that use multiple statistics from a DP solution to create new insights from the data is essential. A third entity is the **Curator** that works as intended in DP, an intermediary that will add noise to queries before making them available to the analyst. Here, we will treat it as a non-interactive fashion DP solution; the queries are already defined and are released as **statistics** to the **Analyst**, which cannot create new queries.

Figure 5.6 summarises our problem. The developer will build (ii) the DP scenario. It will project which equations the analyst will create when the DP scenario (i) is made available. The Developer then creates one **budget allocation** for all statistics that will benefit these equations by considering the allocation that will cause the minimum amount of noise possible. This is a pivotal aspect of our work.

Since each statistic carries its own noise, the interaction between them caused by mathematical operations in equations changes the total amount of noise. This total noise will have a different size depending on the **budget allocation**. The developer wants to minimize it, which can be done by choosing the optimal **budget allocation**. In the next section, we present a metric to compare budget allocations that the developer can use to compare **budget allocations** to find the optimal solution that minimizes the noise.

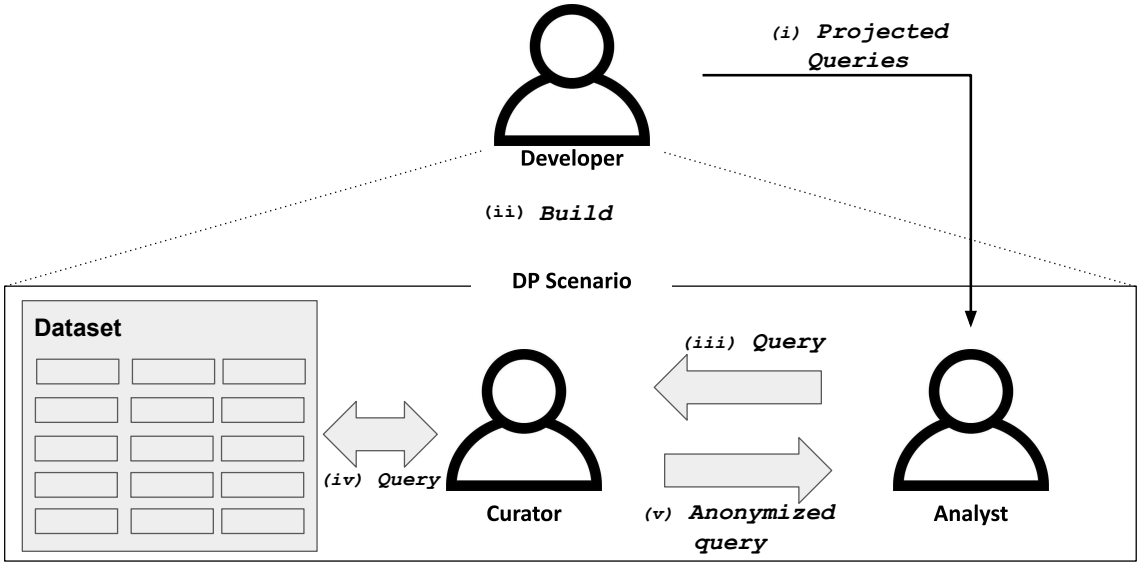


Figure 5.6 – Scenario

Having the optimal budget allocation, it can create the DP scenario (ii). After this, the solution works as a normal DP solution (iii) (iv) (v). The Analyst will receive the statistics that the Curator did as queries to the database with the added noise. Which the analyst will use in equations. This approach can potentially increase the utility of data without compromising privacy. However, it is highly dependent on the developer's ability to predict the equations that the analyst will use. It improves the utility of the data by prioritizing the predicted equations at the cost of the utility of non-predicted equations. Also, it is reliant on budget allocation, which is the parameter that the developer can fine-tune. In the next section, we propose a metric that allows comparing different **budget allocations** in order to optimise the selection.

5.3 A Metric to support the process of budget allocation

In this section, we propose a metric that can be used to compare different distributions of the privacy budget ϵ. Initially, an array (5.3) with a total of *nsta* statistics that will be released by the developer ranging from sta_1 to sta_{nsta} is defined.

$$Sta = [sta_1, sta_2, ..., sta_{nsta}] \tag{5.3}$$

A second array (5.4) *Sen* stores the **sensitivity** from sen_1 to sen_{nsta}. Each statistic in array *Sen* has an equivalent member in array *Sta* at the same position in the equivalent array. For example, the sensitivity for element sta_i is sen_i.

$$Sen = [sen_1, sen_2, ..., sen_{nsta}] \tag{5.4}$$

A third array that we will use is the budget allocation array (5.5) *Bud* with *nsta* elements. This array represents the privacy budget distribution, for each element in the original array *Sta* there is an element in *Bud* in the same position. This represents the privacy budget allocated for that specific statistic. For example, the privacy budget allocated for element sta_i is bud_i.

$$Bud = [bud_1, bud_2, ..., bud_{nsta}] \tag{5.5}$$

There are two limitations (5.6) (5.7) to the values in this array. The sum of all the elements in *Bud* must be equal to the privacy budget ϵ.

$$\sum_{i=1}^{nsta} bud_i = \epsilon \tag{5.6}$$

Also, all the elements in the array need to have a value greater than 0.

$$\forall i(bud_i > 0) \tag{5.7}$$

To help organize our data for further functions, we will consolidate all three arrays (*Sta*, *Sen*, and *Bud*) into one array of tuples (5.8) *Tup*. Each tuple *t* will aggregate the statistic, sensitivity, and budget $tup_i = (sta_i, sen_i, bud_i)$. We will also define a function (5.9) *os* that retrieves the value *sta* from a tuple.

$$Tup = [(sta_1, sen_1, bud_1)_1, (sta_2, sen_2, bud_2)_2, ..., (sta_{nsta}, sen_{nsta}, bud_{nsta})_{nsta}] \tag{5.8}$$

$$os((sta_i, sen_i, bud_i)_i) = sta_i \tag{5.9}$$

In the previous section, we described what are **Equations**. We will define them as a mathematical function $eq(Tup) \Rightarrow \mathbb{R}$. An equation uses multiple specific statistics to calculate its output value. The equation *eq* will receive all tuples with their statistics from array *Tup*, although it will use just a few specific ones.

The developer defines equations on a case-by-case basis. Thus, it is impossible to determine the operation beyond the function signature. However, we will show two examples of equations using lambda functions. Our example scenario has an array $Tup = [tup_1, tup_2, tup_3, tup_4]$. The developer will define two operations: The first one (5.10) eq_1

will receive the statistic value from tup_2, and tup_3 will sum up their values in the second one (5.11) eq_2, the value of tup_1, tup_2 will be added, and then divided by tup_4.

$$eq_1(Tup) = ((\lambda x, y.os(x) + os(y))tup_2)tup_3 \tag{5.10}$$

$$eq_2(Tup) = (((\lambda x, y, z.\frac{os(x) + os(y)}{os(z)})tup_1)tup_2)tup_4 \tag{5.11}$$

Finally, a fourth array (5.12) Eqs will hold tuples with two values $te_i = (eq_i, sen_i)$. The first value is the function of an equation defined by the developer. The second one is the sensitivity of that operation if it was a statistic directly retrieved from the database instead of an equation composed of multiple statistics. Further expanding, eq_1 previously described, is the sum of the statistics from tup_2, and tup_3. However, it is possible to get the result of this equation by directly querying the database, which would be the same as retrieving a new statistic. This alternative way of retrieving the value of eq_1 would have a sensitivity, the value of sen_1 is the sensitivity if instead of using eq_1 we would query the database for a new statistic with the same result as eq_1. For each equation defined by the developer a tuple in Eqs will be created. The size of this array depends on how many equations will be defined by the developer. We define the size as neq.

$$Eqs = [te_1, te_2, ...te_{neq}] \tag{5.12}$$

To quantify the utility of a single statistic, we will create a function $us(tup_i)$ that receives a single tuple from Tup. It will output a positive real number representing the decrease of utility, a higher number indicating less utility.

Similar to us, we will create another function to quantify the utility of an equation. This function $ue(te_i)$ receives a tuple from Eqs. It outputs a positive real number representing the decrease of utility,a higher number means less utility.

Finally, our metric (5.13) will receive the array Tup and Eqs as input. The output is a score representing the utility for a budget allocation Bud; a lower score means a better utility. The metric function is defined as follows.

$$Metric(Tup, Eqs) = \sum_{i=1}^{nsta} us(tup_i) + \sum_{i=1}^{neq} ue(te_i) \tag{5.13}$$

Array Sta and Eqs are based on the information that will be disclosed and equations that the developer predicts the analyst will use. This implies that it will not change after being defined. Although, array Bud represents a single instance of all possible divisions of the privacy budget ϵ to the statistics. The developer's objective is to find the combination of Bud that yields the lowest metric value, which means the highest utility.

One final consideration is why sensitivity is included in all tuples. The sensitivity can be used to relativize the values of the function *us* and *ue*. A counting query in the database would create a statistic with a sensitivity of one, which would result in a small noise in absolute values when used in a mechanism in DP. While other queries would create a statistic with higher values for sensitivity that could result in more considerable noise in absolute values. Both statistics are equally important, but the higher noise in absolute value would create a higher return in the function *us*. We plan to use the sensitivity in *us* and *ue* to balance all statistics and equations, giving them equal importance.

5.3.1 Definition of Metric

Metrics have definitions that are highly dependent on the context. In this discussion, we focus on two contexts where metrics are widely used: mathematical and software engineering definitions.

In mathematics, a metric is a well-established and robust concept. Its primary application is in metric spaces, where it is used to define the distance between elements of a set. To qualify as a mathematical metric, the function must satisfy four properties:

- **Positivity:** The distance d between two distinct elements a and b, where $a \neq b$, is always positive ($d > 0$).
- **Identity:** The distance between two elements a and b is 0 if and only if $a = b$.
- **Symmetry:** The distance from a to b is always the same as the distance from b to a.
- **Triangle Inequality:** The distance from a to c is less than or equal to the sum of the distances from a to b and from b to c.

Our proposed metric does not adhere to the properties of the mathematical definition, so it is not a metric in a mathematical sense. However, we adopt an approach that is more aligned with engineering standpoints.

In engineering, including software engineering, the use of metrics is a prevalent theme. Here, the definition is more closely related to measurement rather than distance. There are numerous metrics for each subfield; for instance, Cotroneo *et al.* [8] discuss metrics in software complexity, and Hatzivasilis *et al.*[18] address metrics in software dependability, security, and privacy. Therefore, we define our metric in the context of DP as **"A measure of the data utility that a specific budget allocation provides, relative to the queries that will be released and the predictions of use of these queries made by the developer"**.

5.3.2 Conclusion

This study presented a novel approach to improving the utility of DP scenarios by predicting equations that the analyst will do with the released statistics and benefiting those equations in the budget allocation. To support this approach, we proposed a new metric that can be used to measure the utility of a specific budget allocation. The approach is designed to be used with summary statistics.

The potential improvement in utility comes from the budget allocation. Specific allocations increase or decrease the amount of noise that a predicted equation will generate. A developer may choose the allocation with our proposed metric, considering all predicted equations. Since it works just on the budget allocation it does not affect the solution's privacy. We will evaluate this metric in our subsequent study.

5.4 Part 3: An evaluation of the proposed metric

In this study, we evaluate the proposed metric with the aim of demonstrating its effectiveness in achieving a better budget allocation compared to an equal distribution of the privacy budget. The dataset includes medical expenses for smokers and non-smokers from two regions: the Northwest and the Southeast. The data is summarized using summary statistics, and we suppose that the analyst would perform t-tests on the statistics.

We employed a brute-force method to generate all possible budget allocations. For each allocation, our metric was applied based on the expected absolute error derived from the statistics and equations of our scenario. This process is computationally intensive, which limited the scope of our experiment. Additionally, some concessions were made regarding the total privacy budget, resulting in a global budget of $\epsilon = 12.0$. Despite these limitations, our results were positive for the proposed metric, indicating its potential for broader applicability.

This section is organized as follows: Experiment Definition, Scenario Description, Results, and Conclusion.

5.4.1 Experiment Definition

This experiment aims to evaluate the proposed metric, specifically to demonstrate that it can identify a better budget allocation among various possible distributions. We strive to show that this metric can find the optimal solution. However, it can also be used to compare different budget allocations, although the latter is beyond the scope of

this study. To achieve our objective, we devised a specific scenario to generate data, which we compared against the benchmark of evenly distributing the global budget.

In the next subsection, we will describe the scenario in greater detail. The key points are that the scenario is hypothetical and based on a medical expenses dataset. Twelve statistics were selected to be released as summary statistics from this dataset. In our hypothetical scenario, the developer would predict that the summary statistics would be used to perform two different t-tests(Equations). The developer needs to find the best budget allocation that minimizes the amount of noise, considering the statistics to be released and the predicted usage.

To generate our results and test our metric, we created a script in Python (version 3.8), available on GitHub[2]. The script utilizes the Pandas library for data manipulation (version 2.0.3) and NumPy for random number generation in the Differential Privacy (DP) mechanism (version 1.24.4). The hardware used should have no impact on the results.

Figure 5.7 summarizes the process to find the optimal budget allocation according to our metric.

Statistics Definition (A): The statistics to be used in our scenario and the equations that utilize these statistics are defined.

Budget Allocation Sequences (B): Using a tree structure, multiple sequences representing possible budget allocations are created based on the global budget and the number of statistics.

Score Calculation (C): The score is calculated using a method similar to Part 5.1 for each sequence. This includes 1,000 executions of the statistics to calculate the absolute error. The equations are also executed 1,000 times to calculate the average absolute error.

The budget allocation sequence with the smallest metric score is considered to have the highest likelihood of generating the slightest noise. This sequence is presented as the result, with the sequence for equally distributing the budget included as a benchmark for comparison.

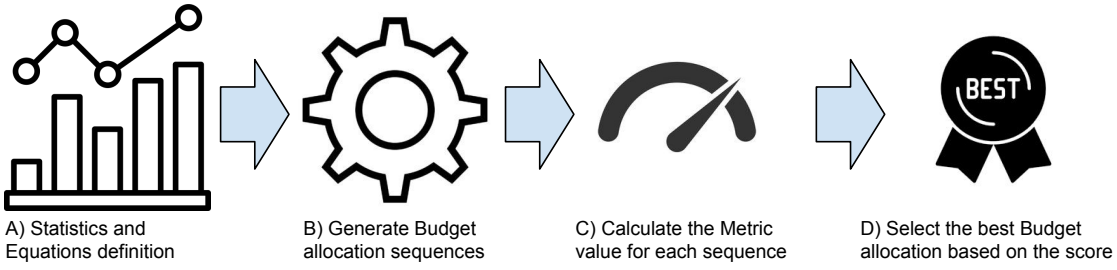


Figure 5.7 – Process to find the best score

[2]https://github.com/conseg/TheImpactofDifferential Privacyondatautilityinfundamentalmathematicaloperations

In our work, we encountered two main challenges that had to be addressed.

The first challenge was the high computational cost. The number of generated sequences with twelve statistics was 1,352,078, calculated using the combinatorics formula for distributing r identical objects into n distinct boxes: $\binom{(r+n-1}{r}$. Here, n is the number of statistics, and r is the global budget available for distribution. Additionally, for each statistic in each sequence, we had to calculate the average absolute error, requiring 1,000 executions per statistic. This limited the number of statistics and equations we could include in our study. Thus, we worked with only 12 statistics and 2 equations.

The second challenge involved stipulating the equations as two t-tests, which require calculating square roots. Depending on the noise, there was the possibility of encountering a square root of a negative number, which is undefined. To address this, we had two options:

- **Clipping:** This approach, sometimes used in Differential Privacy, involves clipping the value to a minimum or maximum depending on the query result.
- **Minimum Budget Allocation:** We could set a minimum value for the budget allocated to each statistic, ensuring that the amount of noise would not make the query result negative.

We adopted the second approach, resulting in a minimum budget allocation of 0.5 for each statistic.

We also set 0.5 as the granularity for budget distribution. Since the budget allocated for a statistic is a real number, creating all possible sequences for budget allocation would result in an infinite number of sequences. By defining a granularity of 0.5, we limited the number of sequences. The available global privacy budget of 12.0 and the chosen granularity constrained the number of possible sequences. Two examples of valid sequences are [0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 2.5, 4.5] and [0.5, 4.0, 0.5, 1.0, 0.5, 0.5, 1.0, 0.5, 1.0, 1.0, 0.5, 1.0].

5.4.2 Scenario

Our scenario is based on the Problem Statement outlined in Subsection 5.2.1. In this scenario, the developer of this DP scenario releases several statistics and attempts to predict how the analyst will use the data. The objective is to minimize the total noise from the statistics and the equations by selecting the best budget distribution. The main aspects of our scenario include the data source, the summary statistics and their predicted equations, and the data processing.

- **Data Source:**

  The data source is the Medical Cost Personal Datasets available on Kaggle[3]. This dataset, commonly used in machine learning education, contains synthetic data. Despite not being based on real data, this does not impact the findings of our study.

  The dataset consists of 1,338 entries, each representing an individual. The data for each person includes age, sex, body mass index (BMI), number of children, smoking status, region, and insurance charges. Our study focuses on three attributes: region (Northwest or Southeast), smoking status (smoker or non-smoker), and insurance charges.

- **Summary Statistics and predicted equation:**

  In our hypothetical scenario, the developer will release a series of summary statistics related to insurance charges for each combination of smoker status and region. Specifically, the released statistics will include the mean, count, and standard deviation (Std) for each group. This results in a total of twelve statistics:

  - Mean of Northwest smokers
  - Mean of Northwest non-smokers
  - Mean of Southeast smokers
  - Mean of Southeast non-smokers
  - Count of Northwest smokers
  - Count of Northwest non-smokers
  - Count of Southeast smokers
  - Count of Southeast non-smokers
  - Std of Northwest smokers
  - Std of Northwest non-smokers
  - Std of Southeast smokers
  - Std of Southeast non-smokers

  These values need to be protected using Differential Privacy (DP), so the challenge is to find the budget allocation that minimizes the noise in these statistics. However, analysts also need to consider the predicted usage of the data.

  In our scenario, the developer predicts the analyst will perform two t-tests on the dataset using the released statistics. Student's t-tests are statistical tests determining whether the difference between two samples is significant. For our case:

[3] https://www.kaggle.com/datasets/mirichoi0218/insurance?resource=download

  - The first t-test will check if there is a significant difference between the regions for smokers.
  - The second t-test will check if there is a significant difference between the regions for non-smokers.

  These two t-tests represent the equations in our metric's terminology and must be considered when searching for the best budget allocation. We aim to reduce the noise in these equations, which are influenced by the noise in the statistics since these statistics serve as inputs for the equations. The t-test formula for the test of two samples is as follows:

$$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{Std_1^2}{n_1} + \frac{Std_2^2}{n_2}}}$$

- **Data processing:**

  Before calculating the metric, we normalized the dataset. This step was necessary to ensure that statistics with higher DP sensitivity do not disproportionately influence the score. In the next section, we will present the results of our summary statistics both before and after normalization, including their sensitivities. For the t-tests, we will show the results without the inclusion of noise. We did not complete the t-test analysis in detail, as it is not relevant to the conclusion of our work.

5.4.3 Results

First, in Table 5.1, we present all the summary statistics the developer would release in our scenario. The table includes information about sensitivity; these values are not anonymized by DP. However, as described in the previous section, the normalized version of the dataset was used to calculate our metric's value. Table 5.2 presents the same statistics in their normalized form. Both tables provide an overall view of the dataset. A released version of these statistics would need to be anonymized using DP.

For the t-tests that we predicted that an analyst would make, the results are as follows:

- For comparing the mean of smokers between the Northwest and Southeast regions, $t = -2.433664$.
- For comparing the mean of non-smokers between the Northwest and Southeast regions, $t = 0.993141$.

	Smoker		non-Smoker	
	Northwest	Southeast	Northwest	Southeast
Mean	**30192.00**	**34844.99**	**8556.46**	**8032.21**
Mean Sensitivity	523.32	321.39	93.66	104.95
Count	**58**	**91**	**267**	**273**
Count Sensitivity	1	1	1	1
Std	**11413.82**	**11324.76**	**6128.55**	**6137.32**
Std Sensitivity	623.37	361.82	183.00	239.24

Table 5.1 – Table of Summary Statistics before data normalization

	Smoker		non-Smoker	
	Northwest	Southeast	Northwest	Southeast
Mean	**0.4640**	**0.5382**	**0.1186**	**0.1103**
Mean Sensitivity	0.0083	0.0083	0.0014	0.0016
Count	**58**	**91**	**267**	**273**
Count Sensitivity	1	1	1	1
Std	**0.1821**	**0.1807**	**0.0978**	**0.0979**
Std Sensitivity	0.0099	0.0057	0.0029	0.0038

Table 5.2 – Table of Summary Statistics after data normalization

After processing all 1,352,078 valid sequences for budget allocation in our scenario, we determined that the best distribution using the proposed metric is $[1.0, 0.5, 0.5, 0.5, 2.0, 1.5, 2.0, 2.0, 0.5, 0.5, 0.5, 0.5]$, with a score of 0.23013823722772533 (where a smaller value indicates better performance).

We use the following budget distribution strategies to compare the results and present their score using our metric.

- Equally splitting the budget among all statistics represented by the sequence $[1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0]$, which scored 0.3285110737126 in our metric.
- Using the Geometric series in a decreasing fashion, as described by Bai *et al.* [1]. The distribution is as follows $[1.54, 1.41, 1.3, 1.19, 1.09, 1.0, 0.92, 0.84, 0.77, 0.71, 0.65, 0.59]$, which scored 0.3340621832395.
- The Geometric series in an increasing fashion using the following distribution $[0.59, 0.65, 0.71, 0.77, 0.84, 0.92, 1.0, 1.09, 1.19, 1.3, 1.41, 1.54]$ scored 0.373786903904.
- Using the Taylor series in a decreasing fashion, also presented by Bai *et al.* [1]. The sequence is as follows $[0.08, 0.41, 1.02, 1.70, 2.12, 2.12, 1.76, 1.26, 0.79, 0.44, 0.22, 0.10]$, yielding a score of 0.452966176473.

- The Taylor series in an increasing fashion in the following sequence $[0.10, 0.22, 0.44, 0.79, 1.26, 1.76, 2.12, 2.12, 1.70, 1.02, 0.41, 0.08]$ scoring 0.4175178039182.

The lower score of our proposed sequence suggests that it should yield less noise overall than any other benchmark, considering the noise generated in both the statistics and the equations. In Figure 5.8, a box plot was created using the metric result for each valid sequence of our scenario. The box-plot graphic gives an idea of the distribution of the metric results.

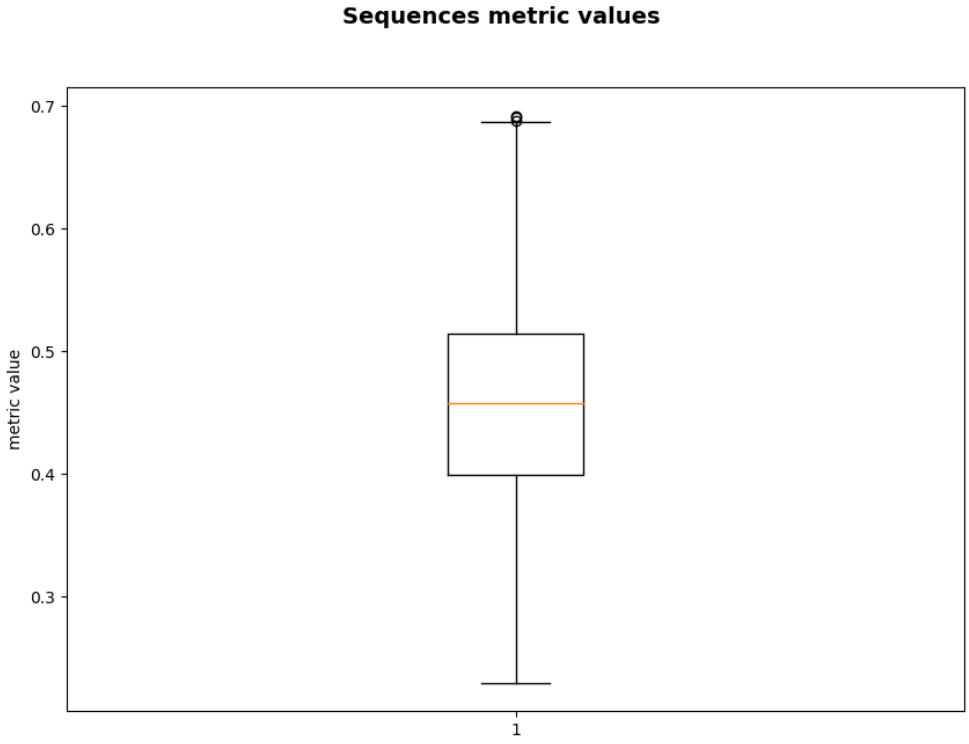


Figure 5.8 – Box-plot created using the metric values for each valid sequence generated

5.4.4 Conclusion

In the third part of this section, we implemented our proposed metric. We used a commonly used machine learning dataset and designed a hypothetical scenario in which multiple statistics would be released using Differential Privacy (DP). In our scenario, we assumed that the analyst would perform two t-tests on the data, which are the predicted equations that need to be considered when processing our metric.

Our method for calculating the metric is computationally intensive, limiting our experiment's scope. We processed 1,352,078 possible budget allocation distributions. From these, we identified the best distribution using our metric's score and compared it to

a benchmark distribution that equally splits the available global privacy budget among all statistics.

Our experiment demonstrates the utility of our metric in a simulated and controlled scenario. However, the high computational cost of calculating the metric can be extremely limiting for its practical application. Additionally, as described in the metric proposal, the results are highly dependent on the developer's ability to accurately predict how the analysts will use the statistics in a real scenario. Finally, the pure DP definition used is too strict to be applicable in real-world scenarios. However, we believe that the conclusions drawn here can be extrapolated to other situations by employing various adaptations of DP.

5.5 Final Remarks

In this chapter, we present the main body of this thesis, through which we address our research questions. We demonstrate that a new metric can be used to find the optimal budget distribution, albeit within a specific scenario. Optimizing budget distribution is a means to increase data utility while maintaining the same level of privacy. Our findings are based on three key parts:

- **Part One:** We demonstrated the existence of a gap that could be exploited to improve data utility based on how queries protected by Differential Privacy (DP) are used.
- **Part Two:** We presented a new metric that can be used to find the best budget distribution or compare two distributions. We proposed a scenario for using this metric based on summary statistics to be released using DP and the equations that utilize these statistics.
- **Part Three:** We applied the metric to a hypothetical scenario to demonstrate its use. Based on this metric, we were able to find the supposed best budget distribution. Thus, distribution presented better results than other benchmarks.

Given that data utility is one of the main challenges for DP, the work presented here is very important. Although our proposed method has several limitations in its current form, it also offers numerous avenues for further development. The high computational cost of processing the metric limited our study, and we confined our work to pure DP, a more rigorous but less practical version of DP.

Further improvements could involve reducing the computational burden, exploring more practical versions of DP, and refining the prediction models for data usage. These

steps will be crucial for enhancing the applicability and impact of our metric in real-world scenarios.

6. CONCLUSION

In this final chapter, we present our conclusion, summarizing the main points of this thesis. This chapter is divided into four subsections:

- Objectives: Here, we review the objectives of our work and discuss the primary contributions to the anonymization of datasets and differential privacy (DP).
- Research Done: This subsection summarizes the research conducted throughout the thesis.
- Answering the Research Question: We revisit our research questions and explain how our research addresses them.
- Future Work: In this final subsection, we discuss potential future research directions that can build on the findings of this work.

6.1 Results

We aimed to develop a new metric for optimizing budget distribution in DP scenarios. DP is a crucial method for dataset anonymization, a topic of growing interest. Budget distribution is a critical topic in the DP field as it impacts data utility without compromising privacy, thereby enhancing the privacy-utility trade-off. We reviewed several related works addressing the budget allocation problem to contextualize our research. However, these works primarily focused on different scenarios, such as machine learning algorithms, rather than summary statistics and did not consider predicting how the data would be used.

Our metric is specifically designed for the scenario of summary statistics. In our approach, we introduce the role of a developer who predicts how these statistics will be used and constructs the entire application. These predictions are termed equations in our solution because they involve multiple statistics to calculate. The budget allocation considers these equations while distributing the privacy budget among the statistics. Consequently, our solution highly depends on the developer's ability to predict data usage accurately.

During our work, we encountered several restrictions. Some are addressed as future work in Section 6.4. However, they are sufficient to impede utilization in real-world scenarios. One major issue is the computational requirements for processing our metric and finding the optimal budget allocation. Due to this limitation, we tested our metric in a constrained scenario with only twelve statistics and two equations. Additionally, our

metric is developed exclusively for pure Differential Privacy (DP), which poses a significant barrier to its application in real-world scenarios, as pure DP is a very restrictive definition that is often impractical for deployment. From a research perspective, our evaluation scenario is quite simplistic. While the positive results may extend to more complex scenarios, expanding and testing in more diverse contexts is crucial. Moreover, our approach assumes that the developer has perfect predictions of how the data will be used, which may not always be true. Addressing these issues in future research will be essential for the practical application of our metric.

6.2 Research done

Our research was conducted in three main parts presented in Chapter 5, each building upon the previous to achieve our overall objective of developing, and evaluating a new metric for optimizing budget distribution in DP scenarios. Here, we summarize the work.

6.2.1 Part One: Identifying the Gap

In this part, we conducted research demonstrating how the interaction between queries protected by DP impacts the noise. The resulting noise varies based on these interactions and, more importantly, on allocating the privacy budget to the queries. This variability provides a degree of control over the final noise, enabling improvements in data utility. Although our work focused on basic mathematical operations, we can infer that more complex interactions will also result in different noise levels depending on the interaction and budget allocation. This study is crucial to this thesis as it highlights a gap that can be exploited to enhance data utility without compromising privacy by accounting for such query results interactions.

6.2.2 Part Two: Developing the Metric

The second part of our research focused on developing a new metric designed to optimize budget distribution for summary statistics. We introduced the concept of a developer who predicts how the released statistics will be used, terming these predictions as equations. The metric was constructed to account for these equations while distributing the privacy budget among the statistics. We detailed the metric's theoretical framework and mathematical formulation, ensuring it aligns with the principles of DP.

6.2.3 Part Three: Experimental Validation

In the final part of our research, we applied the proposed metric to a hypothetical scenario using a well-known synthetic medical cost dataset. The objective of the experiment was to identify the optimal budget allocation, with the benchmark being an equal budget distribution, Geometric series, and Taylor series. The scenario included twelve statistics and two equations. Due to computational limitations, we experimented in a constrained environment, processing 1,352,078 possible budget allocations. The results demonstrated that our metric could effectively find an optimal budget distribution, significantly enhancing data utility.

6.3 Answering the Research Question and Hypothesis

To address our original research question, **”What is the impact of a newly proposed metric for Differential Privacy on the trade-off between privacy and data utility?”** outlined in Chapter 4, we can now provide a conclusive answer. In a specific scenario, we demonstrated that the proposed metric effectively identified a better budget distribution, enhancing data utility without compromising privacy guarantees. Therefore, a new metric can positively impact the privacy-utility trade-off, offering significant improvements.

This also allows us to validate the alternative hypothesis of our research (H_1). The hypothesis is ”The use of specific metrics for anonymizing datasets **can** significantly improve data utility and privacy in practical applications”. As described for the research question the impact of the new metric was positive, improving results when compared to the benchmark.

6.4 Future Work

Multiple paths for future exploration have emerged from the research conducted in this thesis. Here, we present the most important directions, organized along three axes: performance improvements, Differential Privacy definition enhancements, and real-world applications.

The primary constraint in our final experiment was the computational cost required to find the optimal budget allocation distribution. Calculating our metric for each statistic and equation required multiple executions, compounded by the thousands of possible budget distributions, necessitating a brute-force approach. To enhance performance,

both in metric calculation and limiting the number of sequences explored, we hypothesize that gradient descent - a commonly employed algorithm in machine learning for finding approximate optimal solutions — could be utilized. The cost function inherent in gradient descent could be the absolute error. Alternatively, an analytical approach, where a definitive formula is used to calculate the metric value without the need for multiple executions, might be explored. Preliminary investigations in this direction have not yet yielded success, using expected values to estimate noise in an equation based on a Laplace folded distribution [28].

Another constraint is the DP definition to which our metric is applied. Our work is based on Pure DP, a robust definition that tends to be too costly for data utility, making it impractical for most real-world applications. These applications often use weaker definitions, such as approximate DP, which, while offering less stringent privacy guarantees, maintain sufficient utility to be viable. Adapting our metric and scenario to these weaker DP definitions is necessary to extend our work's applicability to real-world scenarios. This research avenue is essential but should pose minimal challenges, as the adaptation process appears straightforward.

Finally, our research was conducted using only synthetic data in a hypothetical scenario. While this approach is realistic enough to suggest that the results should transfer to real scenarios, applying our metric to real-world data is an exciting research direction that could further validate and strengthen our findings. However, this avenue of development hinges on overcoming the limitations identified in the other two areas. To apply our research in real-world scenarios, we need to achieve performance improvements and adapt our metric to other DP definitions beyond Pure DP.

REFERENCES

[1] Bai, Y.; Yang, G.; Xiang, Y.; Wang, X. “Generalized and multiple-queries-oriented privacy budget strategies in differential privacy via convergent series”, *Security and Communication Networks*, vol. 2021, Dec. 2021, pp. 1–17.

[2] Bkakria, A.; Tasidou, A.; Cuppens-Boulahia, N.; Cuppens, F.; Bouattour, F.; Ben Fredj, F. “Optimal distribution of privacy budget in differential privacy”. In: Proceedings of the 2018 Risks and Security of Internet and Systems, Zemmari, A.; Mosbah, M.; Cuppens-Boulahia, N.; Cuppens, F. (Editors), 2019, pp. 222–236.

[3] Blum, A.; Dwork, C.; McSherry, F.; Nissim, K. “Practical privacy: the sulq framework”. In: Proceedings of the 24th ACM Symposium on Principles of Database Systems, 2005, pp. 128–138.

[4] Boateng, E. Y.; Otoo, J.; Abaye, D. A. “Basic tenets of classification algorithms k-nearest-neighbor, support vector machine, random forest and neural network: a review”, *Journal of Data Analysis and Information Processing*, vol. 8–4, 2020, pp. 341–357.

[5] Bun, M.; Steinke, T. “Concentrated differential privacy: Simplifications, extensions, and lower bounds”. In: Proceedings of the 14th International Conference of the Theory of Cryptography, Hirt, M.; Smith, A. (Editors), 2016, pp. 635–658.

[6] Chattamvelli, R.; Shanmugam, R. “Laplace Distribution”. Switzerland: Springer International Publishing, 2021, chap. 13, pp. 189–199.

[7] Chaudhuri, A.; Mukerjee, R. “Randomized Response: Theory and Techniques”. Routledge, 2020.

[8] Cotroneo, D.; Natella, R.; Pietrantuono, R. “Predicting aging-related bugs using software complexity metrics”, *Performance Evaluation*, vol. 70–3, 2013, pp. 163–178, special Issue on Software Aging and Rejuvenation.

[9] da Silva, M. P.; Nunes, H. C.; Neu, C. V.; Thomas, L. T.; Zorzo, A. F.; Morisset, C. “Impact of using a privacy model on smart buildings data for co2 prediction”. In: Proceedings of the 37th Conference Data and Applications Security and Privacy, Atluri, V.; Ferrara, A. L. (Editors), 2023, pp. 133–140.

[10] Dalenius, T. “Towards a methodology for statistical disclosure control”, *statistik Tidskrift*, vol. 15–429-444, 1977, pp. 1–2.

[11] Deconto, G. D.; Zorzo, A. F.; Dalalana, D. B.; Oliveira, E.; Lunardi, R. C. “pmachine learning for forensic occupancy detection in iot environments”. In: Proceedings of the

2024 Good Practices and New Perspectives in Information Systems and Technologies, Rocha, Á.; Adeli, H.; Dzemyda, G.; Moreira, F.; Poniszewska-Marańda, A. (Editors), 2024, pp. 102–114.

[12] Dong, J.; Roth, A.; Su, W. J. “Gaussian Differential Privacy”, *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 84–1, 02 2022, pp. 3–37.

[13] Du, R.; Ye, Q.; Fu, Y.; Hu, H.; Li, J.; Fang, C.; Shi, J. “Differential aggregation against general colluding attackers”. In: Proceedings of the IEEE 39th International Conference on Data Engineering, 2023, pp. 2180–2193.

[14] Dwork, C. “Differential privacy”. In: Proceedings of the 33rd international conference on Automata, Languages and Programming, 2006, pp. 1–12.

[15] Dwork, C. “Differential privacy: A survey of results”. In: Proceedings of the 5th International Conference conference on theory and applications of models of computation, 2008, pp. 1–19.

[16] Dwork, C.; McSherry, F.; Nissim, K.; Smith, A. “Calibrating noise to sensitivity in private data analysis”. In: Proceedings of the 3rd Theory of Cryptography Conference, Halevi, S.; Rabin, T. (Editors), 2006, pp. 265–284.

[17] Fan, Z.; Xu, X. “Apdpk-means: A new differential privacy clustering algorithm based on arithmetic progression privacy budget allocation”. In: Proceedings of the IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems, 2019, pp. 1737–1742.

[18] Hatzivasilis, G.; Papaefstathiou, I.; Manifavas, C. “Software security, privacy, and dependability: Metrics and measurement”, *IEEE Software*, vol. 33–4, 2016, pp. 46–54.

[19] Hemkumar, D.; Ravichandra, S.; Somayajulu, D. V. L. N.

“Impact of data correlation on privacy budget allocation in continuous publication of location statistics”, *Peer-to-Peer Networking and Applications*, vol. 14, mar 2021, pp. 1650–1655.

[20] Hou, J.; Li, Q.; Meng, S.; Ni, Z.; Chen, Y.; Liu, Y. “Dprf: A differential privacy protection random forest”, *IEEE Access*, vol. 7, 2019, pp. 130707–130720.

[21] Hsu, J.; Gaboardi, M.; Haeberlen, A.; Khanna, S.; Narayan, A.; Pierce, B. C.; Roth, A. “Differential privacy: An economic method for choosing epsilon”. In: Proceedings of the IEEE 27th Computer Security Foundations Symposium (CSF), 2014, pp. 398–410.

[22] Ikotun, A. M.; Ezugwu, A. E.; Abualigah, L.; Abuhaija, B.; Heming, J. “K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data”, *Information Sciences*, vol. 622, 2023, pp. 178–210.

[23] Katoch, S.; Chauhan, S. S.; Kumar, V. “A review on genetic algorithm: past, present, and future”, *Multimedia Tools and Applications*, vol. 80–5, Oct. 2020, pp. 8091–8126.

[24] Li, N.; Li, T.; Venkatasubramanian, S. “t-closeness: Privacy beyond k-anonymity and l-diversity”. In: Proceedings of the IEEE 23rd International Conference on Data Engineering, 2007, pp. 106–115.

[25] Li, X.; Qin, B.; Luo, Y.; Zheng, D. “A differential privacy budget allocation algorithm based on out-of-bag estimation in random forest”, *Mathematics*, vol. 10–22, 2022, pp. 4338.

[26] Li, Y.; Song, X.; Tu, Y.; Liu, M. “Gapbas: Genetic algorithm-based privacy budget allocation strategy in differential privacy k-means clustering algorithm”, *Computers Security*, vol. 139, 2024, pp. 103697.

[27] Liu, K.; Terzi, E. “Towards identity anonymization on graphs”. In: Proceedings of the 2008 ACM International Conference on Management of Data, 2008, pp. 93–106.

[28] Liu, Y.; Kozubowski, T. J. “A folded laplace distribution”, *Journal of Statistical Distributions and Applications*, vol. 2–1, Oct 2015, pp. 10.

[29] Machanavajjhala, A.; Gehrke, J.; Kifer, D.; Venkitasubramaniam, M. “L-diversity: privacy beyond k-anonymity”. In: Proceedings of the 2nd International Conference on Data Engineering, 2006, pp. 24–24.

[30] Mehner, L.; Voigt, S. N. v.; Tschorsch, F. “Towards explaining epsilon: A worst-case study of differential privacy risks”. In: Proceedings of the 2021 IEEE European Symposium on Security and Privacy Workshops, 2021, pp. 328–331.

[31] Mironov, I. “Rényi differential privacy”. In: Proceedings of the IEEE 30th computer security foundations symposium, 2017, pp. 263–275.

[32] Nanayakkara, P.; Smart, M. A.; Cummings, R.; Kaptchuk, G.; Redmiles, E. M. “What are the chances? explaining the epsilon parameter in differential privacy”. In: Proceedings of the 32nd USENIX Conference on Security Symposium, 2023.

[33] Narayanan, A.; Shmatikov, V. “Robust de-anonymization of large sparse datasets”. In: Proceedings of the 2008 IEEE Symposium on Security and Privacy, 2008, pp. 111–125.

[34] National Institute of Standards and Technology. “Differential privacy: Future work & open challenges”. Accessed: 2024-10-01, Source: https://www.nist.gov/blogs/cybersecurity-insights/differential-privacy-future-work-open-challenges.

[35] Nunes, H.; Silva, M.; Neu, C.; Zorzo, A. “A proposal to increase data utility on global differential privacy data based on data use predictions”. In: Proceedings of the 23th Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais, 2023, pp. 558–563.

[36] Ouadrhiri, A. E.; Abdelhadi, A. “Differential privacy for deep and federated learning: A survey”, *IEEE Access*, vol. 10, 2022, pp. 22359–22380.

[37] Pan, K.; Feng, K. “Differential privacy-enabled multi-party learning with dynamic privacy budget allocating strategy”, *Electronics*, vol. 12–3, 2023, pp. 658.

[38] Pankova, A.; Laud, P. “Interpreting epsilon of differential privacy in terms of advantage in guessing or approximating sensitive attributes”. In: Proceedings of the IEEE 35th Computer Security Foundations Symposium, 2022, pp. 96–111.

[39] Paul, A.; Mukherjee, D. P.; Das, P.; Gangopadhyay, A.; Chintha, A. R.; Kundu, S. “Improved random forest for classification”, *IEEE Transactions on Image Processing*, vol. 27–8, 2018, pp. 4012–4024.

[40] Pujol, D.; Wu, Y.; Fain, B.; Machanavajjhala, A. “Budget sharing for multi-analyst differential privacy”, *Proc. VLDB Endow.*, vol. 14–10, jun 2021, pp. 1805–1817.

[41] Rogers, R.; Roth, A.; Ullman, J.; Vadhan, S. “Privacy odometers and filters: pay-as-you-go composition”. In: Proceedings of the 30th International Conference on Neural Information Processing Systems, 2016, pp. 1929–1937.

[42] Su, D.; Cao, J.; Li, N.; Bertino, E.; Jin, H. “Differentially private k-means clustering”. In: Proceedings of the Sixth ACM Conference on Data and Application Security and Privacy, 2016, pp. 26–37.

[43] Sweeney, L. “Achieving k-anonymity privacy protection using generalization and suppression”, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10–05, 2002, pp. 571–588.

[44] Sweeney, L. “k-anonymity: A model for protecting privacy”, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10–05, 2002, pp. 557–570.

[45] Sweeney, L. “Weaving technology and policy together to maintain confidentiality”, *Journal of Law, Medicine 38; Ethics*, vol. 25–2–3, 2021, pp. 98–110.

[46] Tamersoy, A.; Loukides, G.; Nergiz, M. E.; Saygin, Y.; Malin, B. “Anonymization of longitudinal electronic medical records”, *IEEE Transactions on Information Technology in Biomedicine*, vol. 16–3, 2012, pp. 413–423.

[47] Wang, Z.; Song, M.; Zhang, Z.; Song, Y.; Wang, Q.; Qi, H. “Beyond inferring class representatives: User-level privacy leakage from federated learning”. In: Proceedings IEEE Conference on Computer Communications, 2019, pp. 2512–2520.

[48] Warner, S. L. “Randomized response: A survey technique for eliminating evasive answer bias”, *Journal of the American Statistical Association*, vol. 60–309, 1965, pp. 63–69.

[49] Xie, Y.; Li, P.; Wu, C.; Wu, Q. “Differential privacy stochastic gradient descent with adaptive privacy budget allocation”. In: Proceedings of the 2021 IEEE International Conference on Consumer Electronics and Computer Engineering, 2021, pp. 227–231.

[50] Zheleva, E.; Getoor, L. “Preserving the privacy of sensitive relationships in graph data”. In: Proceedings of the 1st ACM International Conference on Privacy, Security, and Trust in KDD, 2007, pp. 153–171.

[51] Øhrn, A.; Ohno-Machado, L. “Using boolean reasoning to anonymize databases”, *Artificial Intelligence in Medicine*, vol. 15–3, 1999, pp. 235–254.

Pontifícia Universidade Católica do Rio Grande do Sul
Pró-Reitoria de Pesquisa e Pós-Graduação
Av. Ipiranga, 6681 – Prédio 1 – Térreo
Porto Alegre – RS – Brasil
Fone: (51) 3320-3513
E-mail: propesq@pucrs.br
Site: www.pucrs.br