Differential Privacy and Anonymization: Defense Against Inference Attacks

Alper Şahin
Department of Computer Engineering
Koç University
Istanbul, Türkiye
alpersahin21@ku.edu.tr

İdil Görgülü
Department of Computer Engineering
Koç University
Istanbul, Türkiye
igorgulu21@ku.edu.tr

Pelin Önen
Department of Computer Engineering
Koç University
Istanbul, Türkiye
ponen20@ku.edu.tr

***Abstract*—The modern era places data above all other assets; consequently, every action taken is recorded somewhere as data. This brings out the problem of securely publishing data without violating one's privacy. Differential Privacy is a secure solution that ensures utility while preserving privacy over traditional methods like k-anonymity, l-diversity, and t-closeness. This paper first extensively defines the problem at hand. Then, it explains the traditional data privacy methods and their shortcomings. In addition, the logic behind differential privacy is mathematically defined and illustrated, highlighting how it differs from other solutions to gain better utility and privacy. Lastly, experimental results obtained using the differential privacy method are presented, followed by concluding remarks emphasizing that differential privacy is the pioneer method to achieve data privacy.**

I. Introduction

In today's data driven world, it is now possible to generate remarkable amounts of information in forms varying from photos and video clips to electronic documents and web logs. Though this growth in informative content accelerates innovation, it also increases deep concerns regarding the preservation of privacy. The widespread availability and improvements of contemporary technologies, particularly ones such as data mining tools, emphasize the imperative need for effective privacy-maintaining mechanisms [1]. Even data sets which are anonymized with traditional methods can be easily attacked in terms of privacy. Also, with the continuous developments in the field, a reliable privacy protection mechanism today may not be as safe and secure as it is today and can get breached in the future [1].

In order to solve such challenges, differential privacy has emerged as a systematic and mathematically rigorous paradigm that can guarantee suitable privacy guarantees while still enabling useful data analysis [2]. By introducing carefully quantified noise into data queries, differential privacy attempts to find an optimal trade-off between data utility and individual privacy, thus becoming a principled and robust privacy-enhancing framework. This methodology offers a quantifiable assurance that joining a statistical database does not cause any risk [3].

In this paper, we examine differential privacy as a structural approach to protect individual privacy in statistical databases. We start by discussing the motivation behind privacy-preserving data sharing and defining the problem. This is followed by a review of traditional anonymization methods such as k-anonymity, l-diversity, and t-closeness, as well as a discussion of their practical issues which limit their effectiveness. We then turn to differential privacy and introduce its core principles and how it addresses the shortcomings of other methods. In addition to the theoretical exposition, we implement a practical experiment that simulates a database with a fixed mean and applies both uniformly distributed and Gaussian noise to anonymize its values. In order to better understand how the scale of data affects the balance between privacy and utility, we repeat this process on datasets with varying sizes. We document the outcomes of this anonymization process, analyze them in the context of differential privacy, and assess their implications for both privacy preservation and practical usability. Finally, we summarize the key insights we obtained from both our practical application and our research. The paper aims to provide a clear understanding of differential privacy and its place within the concept of data protection.

II. Problem

Data cannot be fully anonymized without compromising its usefulness. In general, the more detailed and informative a dataset is, the more valuable it becomes for analysis. However, the richness that makes the data useful also makes it vulnerable. This has led to common practices such as removing personally identifiable information or selectively suppressing parts of the data in hopes of preserving individual privacy while retaining analytical value [2]. Individuals can generally be re-identified from combinations of what seem to be innocuous characteristics—like zip code, birthday, or gender. Such information can be employed in what are known as linkage attacks, where anonymized records are combined with public records in other data sets, which leads to effectively bypassing the privacy protection [2].

Datasets are most commonly accessed through queries. Although queries return aggregated information rather than

individual records and this may appear safe, it is possible to combine different queries in a way that reveals sensitive details about individuals. This becomes particularly troublesome in the event of differencing attacks, where an attacker compares answers of similar queries which return adjacent answers to infer private data. [4]

Another problem in this field is that traditional privacy-preserving techniques typically presume adversaries know only a limited set of information outside of the data set at hand, but attackers in practice will often have access to an extensive range of auxiliary information, including publicly released datasets, leaked records, and data gathered from social network websites. This kind of background knowledge can be used to link attributes that are harmless when taken alone but can be used to obtain sensitive information through increasing the probability of re-identification. Therefore, any privacy mechanism that fails to take into account such worst-case scenarios will not provide reliable protection in real world settings [5].

Inference attacks are also major sources of threat, as it makes access to private information reachable even to attackers who do not have any external information. Similarly to differencing attacks, inference attacks are also performed via examining aggregate query answers to infer sensitive information about individuals. This form of attack can be particularly stealthy, as it smoothly bypasses conventional anonymization methods but is still capable of privacy breaches through statistical reasoning. [6] Differential privacy is intended to counteract such attacks by introducing random noise to the outputs of queries.

III. Existing Solutions

Several anonymization techniques have been proposed over the years to ensure personal privacy in data sets without reducing their utility for analysis. These early techniques rely on modifying the data set in a way that reduces the possibility of re-identification. k-anonymity [7], l-diversity [8], and t-closeness [9] are the most prominent, each of which has increasingly stronger privacy guarantee arriving in response to the weaknesses of the last one. The common idea underlying such techniques is to divide the dataset into equivalence classes of records that share the same values for certain identifying attributes—also referred as quasi-identifiers. In doing so, each person's record becomes statistically unlinked from others in the same equivalence class. This section examines each of these approaches, detailing their mechanisms, advantages, and limitations.

A. k-Anonymity

The k-anonymity method is one of the early solutions that provide anonymity to the shared data to a level while internalizing generalization and suppression methods. k-anonymity provides data anonymity on the shared data table by ensuring at least k matching entries exist based on each characteristic combination that linking can be enforced on [7]. These entries are named as quasi-identifiers throughout the research in this field. Appearing at least k times means that when we compare the shared table with the actual table which is known as the private table (PT), at least k persons can be tied to each re-originated entry, which enhances anonymity.

Mainly, providing k-anonymity is possible through two methods. The first approach is generalizing data, which is narrowing down the value set for attributes by mapping to make attributes less informative. For example, if we are trying to generalize the ID number attribute, changing the last number to 0 would provide generalization while decreasing linkability. The most important thing to consider is minimal generalization, which is acquiring the generalized table in a way that no table provides the desired k-anonymity with less generalization on attributes—i.e., with less uniqueness [7]. This is important because increasing the generalization of data on the attribute level decreases the efficient usage of the data since it decreases informativeness.

The second approach is suppressing data, which means removing entries from the PT to disclose data and ensure k-anonymity. Suppressing is mainly used to decrease the generalization level, which is a costly process. To protect the data integrity, minimal required suppression should be considered when applying k-anonymity. That means after generalization, entries that appear less than k times—and those entries only—should be extracted from the table.

The best results are obtained when these two approaches are mixed to accomplish k-anonymity, since the data holder should consider data integrity and precision simultaneously [7]. That is why in real-life applications, there is a threshold level for suppression to keep the data complete at a certain level.

Despite all the advantages k-anonymity offers, there are major drawbacks that these approaches suffer from. First of all, it is almost impossible to know all the external information that can be used on the data to achieve backtracing; thus, it is impossible to determine what to hide completely [10]. Also, a shared data table can suffer from an unsorted matching attack, which is the sharing of entries in the same order as the PT [10]. Another problem is the complementary release attack, meaning new releases from the same PT should be based on prior releases to avoid linking using both datasets [10]. Finally, due to the temporal attack, data holders must be aware of the aliveness of the data entries; after any insertion, change, or deletion, they must regenerate the shared table based on prior releases to maintain k-anonymity without enabling linkage [10].

Another issue with k-anonymity is that its computational complexity is too high; it is found to be NP-hard when suppression is involved [11]. Moreover, the k-anonymity

model performs better for prosecutor re-identification, where the attacker knows specific individuals from the dataset and attempts to find their records. However, it performs poorly in journalist re-identification scenarios, where the attacker aims to identify someone without targeting a specific individual [12].

B. l-Diversity

Besides the problems discussed above, k-anonymity also falls short due to a lack of diversity in the sensitive attributes—the fields containing private information such as diseases, which can be seen in the shared table [8]. If all k entries that can be matched to a person have the same sensitive information, it is very easy to infer that this information belongs to that person. This is known as a homogeneity attack. k-anonymity also does not protect against background knowledge attacks, where the attacker possesses specific external information that makes linkage to an individual's sensitive data easier [8]. For example, even if there are k entries that could correspond to an individual, background knowledge might eliminate all but one, rendering the protection ineffective.

To address such issues, including adversary knowledge and lack of diversity, the l-diversity principle was proposed. l-diversity requires that the shared table contain l "well-represented" sensitive values for each equivalence class to prevent the aforementioned attacks [8]. This sets a threshold: for an adversary to deduce someone's sensitive value, they would need to possess $l - 1$ harmful pieces of background knowledge [8]. By ensuring diversity among sensitive attributes, l-diversity restricts background knowledge-based disclosures. Higher values of l increase protection.

Experimentally, the performance of k-anonymity and l-diversity is nearly the same in terms of runtime. However, in terms of utility, l-diversity can perform worse, as it trades off data utility for improved privacy [8]. Still, l-diversity has its limitations. For instance, when there is only one highly frequent sensitive value, or when two possible sensitive values differ drastically in importance, l-diversity fails to preserve utility without introducing significant information loss [9]. Furthermore, if the distribution of the sensitive attribute is skewed, then the likelihood of that value appearing within certain groups becomes much higher than in the overall population, exposing privacy risks [9].

Another concern is the similarity attack, where even though sensitive values differ as per the l-diversity condition, their semantic similarity may allow an attacker to approximate someone's sensitive value or group [9]. In such cases, an adversary can still infer damaging information despite the presence of diversity.

C. t-Closeness

To overcome the shortcomings of l-diversity such as its susceptibility to skewness and similarity attacks, the concept of t-closeness was introduced as a more robust privacy model. The main objective of t-closeness is to modify the update the dataset such that the distribution of sensitive attributes within each equivalence class is similar to their distribution in the entire original dataset [9]. More formally, an equivalence class is t-close if the distance between the distribution of a sensitive attribute in the class and its distribution over the entire table is at most a threshold t. A dataset is t-close if all its equivalence classes meet this condition. This approach assumes the distribution of sensitive attributes in general to be publicly known and limits the knowledge gain of the attacker to what can be concluded about specific individuals versus the population in general [9].

To measure the distance between distributions, t-closeness uses the Earth Mover's Distance (EMD), which is intuitively the minimum "work" required to transform one distribution into another. This metric is particularly helpful because, unlike simple metrics such as KL divergence or variational distance, it takes into account the semantic closeness of attribute values. That is, even when two distributions have different values, the privacy risk is lower if the values are close in meaning or magnitude semantically.

The usefulness of t-closeness comes from its ability to handle situations in which l-diversity cannot. For example, in skewness attacks, if the sensitive attribute is highly imbalanced (such as 99% negative test results and 1% positive), l-diversity can create equivalence classes with misleading 50/50 distributions, drastically increasing disclosure risk. Similarly, in similarity attacks, while the values in an equivalence class are distinct, they may be semantically proximate, allowing adversaries to infer sensitive information. t-closeness avoids these risks by maintaining the local distribution close to the global distribution in a way that outlier bias and semantic leakage are precluded.

In brief, t-closeness is a more refined and effective privacy model that transcends the limitations of current models by considering both distributional similarity and semantic proximity, thereby achieving a better trade-off between data utility and individual privacy.

IV. Differential Privacy

Differential Privacy (DP) is a mathematical approach to protecting individual data in statistical databases in such a manner that the existence or non-existence of an individual data point becomes approximately undetectable in the result of any analysis [2].Unlike other methods for privacy, DP provides formal guarantees without making any assumptions about the prior knowledge or computational power of an adversary [13].

A. Definition

Differential privacy is centered around the concept of adjacent datasets. Adjacent -also commonly referred to as neighboring- datasets differ only by a single entry. The formal

definition of differential privacy is as follows: A mechanism M is ϵ-differentially private if, for all pairs of neighboring datasets d and d' and all sets of possible outputs X, the following is true [14]:

$$\Pr[\mathcal{M}(d) \in X] \leq e^{\varepsilon} \cdot \Pr[\mathcal{M}(d') \in X] \tag{1}$$

Here, ϵ is a non-negative parameter known as the privacy budget or privacy loss parameter. The main purpose of this parameter is to control the level of privacy: smaller ϵ values imply stronger privacy guarantees but lower utility, whereas larger ϵ values permit more accurate outputs at the cost of weaker privacy.

This definition ensures that the presence or absence of any individual's data in the dataset changes the output distribution of the mechanism by at most a multiplicative factor of e^{ϵ}. Thus, no adversary —regardless of the amount of known external knowledge —can decide with certainty whether a particular person's data was included, from the output of $\mathcal{M}$ [2].

B. Interpretation and Guarantees

Differential privacy provides individual-level protection and aims to ensure "any single individual's data has a negligible effect on the result," and therefore, participation in the dataset cannot harm the individual [3].

Furthermore, it makes no assumptions about the adversary's background knowledge. Whether the attacker knows the entire dataset except for one record or has arbitrary external information, the privacy guarantee holds universally. This robustness is what distinguishes differential privacy from other models, which often fail in the presence of auxiliary information [13].

Another significant advantage of DP is composability. If multiple queries are performed on the same dataset, each with an individual privacy cost ϵ_i, the total privacy loss does not exceed $\sum_i \epsilon_i$. This property allows controlled, multi-step data analyses under strict privacy accounting.

C. Mechanisms to Achieve DP

There are several mechanisms designed to satisfy the formal definition of differential privacy, with the most well-known ones being the Laplace Mechanism, The Gaussian Mechanism and the Exponential Mechanism. The Laplace and Gaussian mechanisms add noise drawn from their respective probability distributions and are used mostly for numerical data, whereas the Exponential mechanism is used when the output is not numeric, but categorical by assigning probabilities to outputs based on a utility function [13]. In the differential privacy performed in this study, the Laplace Mechanism was employed.

The Laplace Mechanism adds noise drawn from a Laplace distribution centered at 0 with scale proportional to the global sensitivity of the query. The sensitivity Δf is the maximum amount that the query output can change due to a change in a single record [13]. More formally, the sensitivity of a mechanism $\mathcal{M}$ is given by the following equation:

$$S(\mathcal{M}) = \max_{d,d'} \|\mathcal{M}(d) - \mathcal{M}(d')\|_1 \tag{2}$$

Here, d and d' are any pair of adjacent datasets differing by one record, and $\|\cdot\|_1$ denotes the L_1 norm. The scale of the Laplace noise (b) is then calculated as $b = S(\mathcal{M})/\epsilon$. Once b is determined, the probability density function of the Laplace distribution from which the noise vector v is drawn is defined as [13]:

$$Pr[v] = \text{Lap}(x \mid \mu = 0, b) = \frac{1}{2b} e^{-\frac{\|v\|_1}{b}} \tag{3}$$

The detailed steps of how the Laplace mechanism was utilized in our implementation, including how sensitivity and noise were computed in practice, are discussed in the following section on Practical Implementation.

D. Practical Implementation

In order to test the proposed methods of anonymization empirically, we performed a set of experiments with Python, building on a set of scientific computing libraries. The implementation was intended to be modular, reproducible, and devoted to the intuitively showing the trade-offs between naive noise addition and formal Differential Privacy.

1) Data Generation

Synthetic datasets of sizes (1,000, 10,000 and 100,000 records) were generated to test the scalability of each method. `NumPy` library was applied to create data points sampled on a continuous uniform distribution within the range $[5.0, 45.0]$. This technique makes sure that the theoretical mean of the distribution underlying is 25.0. On every experimental run, the ground truth of error calculations was the *actual sample mean* of the generated dataset, i.e. the measured error is caused by the noise of anonymization only and not by any slight fluctuations of the data generation process.

2) Anonymization Mechanisms

The main query of our experiment was that of calculating the mean of the database. There were three techniques applied to anonymize the result of this query:

- **Naive Uniform Noise:** This is an intuitive method. To every single data point in the original dataset, a noise value was generated randomly based on a uniform distribution $U[-1, 1]$ with the help of `NumPy`. The mean was then computed using this newly, noisily disturbed dataset.
- **Naive Gaussian Noise:** Like in the uniform approach, we added noise to each data point. The noise was drawn from a Gaussian (Normal) distribution with a mean of 0 and standard deviation of 1 (i.e., $N(0, 1)$).
- **DP Laplace Mechanism:** This technique meets the formal definition of ε-differential privacy. Contrary to the naive methods, it calculates the actual mean of the initial

dataset. The aggregate result is then perturbed with a single noise value to preserve privacy. The noise is a random variable following a Laplace distribution with the scale determined by the privacy budget ($\varepsilon = 1.0$) and the global sensitivity of the query. For a mean query over a known range $[a, b]$, the sensitivity is specified as $S = (b - a)/n$ In our case, with a range of $[5, 45]$, the sensitivity becomes $40/n$.

3) Evaluation Metrics

The performance of both methods was evaluated based on two metrics: *utility* and *privacy*.

- **Utility:** The utility of both methods was measured using the **Mean Squared Error (MSE)**. It quantifies the squared difference between the true sample mean and the noisy mean output of an anonymization technique:

$$\text{MSE} = \frac{1}{k}\sum_{i=1}^{k}(\mu_i - \tilde{\mu}_i)^2 \tag{4}$$

  where μ_i is the true mean, $\tilde{\mu}_i$ is the noisy mean from run i, and $k = 50$ represents the number of experimental repetitions. These results were organized in a structured table using the `Pandas` library.
- **Privacy Resilience:** The strength of the privacy guarantee was assessed by simulating a **differencing attack** and computing the distinguishability between output distributions using the **Jensen-Shannon (JS) Divergence**. Two datasets were generated: D and a neighboring D' (with one record removed). Each anonymization method was applied thousands of times on both datasets. The JS Divergence between the resulting distributions was then calculated using the `SciPy` library. A high JS Divergence indicates easily distinguishable distributions (i.e., a privacy leak), while a value close to 0 indicates strong privacy preservation.

To provide clear and informative visualizations of the experimental results, all statistical graphics were created using the `Matplotlib` and `Seaborn` libraries.

V. Results and Analysis

This section presents the empirical results from the comparative experiments. The findings, summarized in Table I and Figures 1, 2, and 3, highlight the practical differences in utility and privacy between the tested methods.

1) Utility vs. Data Size

Table I and Figure 1 show the accuracy of each method in terms of utility of datasets of increasing size. The most important point is the *crossover* point in utility. The Laplace DP method is the least accurate at a small data size ($n =$ 1,000) because it has to introduce a lot of noise to ensure that privacy in this small group is satisfied. But when the data set increases to $n = 100{,}000$, the scenario turns over and the Laplace DP method turns out to be the most accurate by a significant margin.

TABLE I: Comparison of Noisy Mean and MSE Across Methods and Data Sizes

Data Size	Algorithm	Avg. Noisy Mean	MSE
1,000	Naive Uniform	24.913917	4.02×10^{-4}
1,000	Naive Gaussian	24.912416	1.14×10^{-3}
1,000	Laplace DP ($\varepsilon = 1.0$)	24.912867	3.22×10^{-3}
10,000	Naive Uniform	24.938313	3.25×10^{-5}
10,000	Naive Gaussian	24.937753	1.05×10^{-4}
10,000	Laplace DP ($\varepsilon = 1.0$)	24.937815	3.01×10^{-5}
100,000	Naive Uniform	24.936594	3.76×10^{-6}
100,000	Naive Gaussian	24.936551	9.02×10^{-6}
100,000	Laplace DP ($\varepsilon = 1.0$)	24.936561	2.53×10^{-7}

The naive approaches become more accurate because of the *Law of Large Numbers*, under which the zero-mean noise added to each record is canceled when averaged. This great utility, however, is an indication of a privacy failure: the noise becomes insignificant.

By contrast, the utility of DP increases, since noise of DP is scaled to sensitivity of the query ($40/n$). When n increases, sensitivity reduces, which implies that fewer noises are needed to assure the same mathematical level of privacy. This shows that DP scales gracefully and achieves substantial privacy with high utility on large datasets.

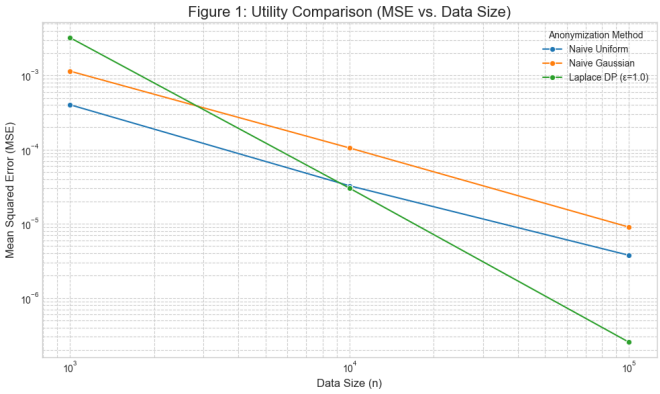

Fig. 1: Utility Comparison (MSE vs. Data Size)

2) The Privacy-Utility Trade-off

Figure 2 graphs the essence of the tuning knob of Differential Privacy: the privacy budget, ε. There is an inverse relationship evident in the graph. When ε is small (high privacy), a large amount of noise is needed and this results in a high MSE (low utility). As ε grows, the privacy guarantee is lightened, the MSE decreases rapidly and ultimately flattens out. This finding verifies that DP offers a formal quantifiable process that allows data curators to make a principled decision

regarding the extent of privacy required by them relative to the utility of the analysis they need—a highly advanced capability that naive approaches cannot achieve.

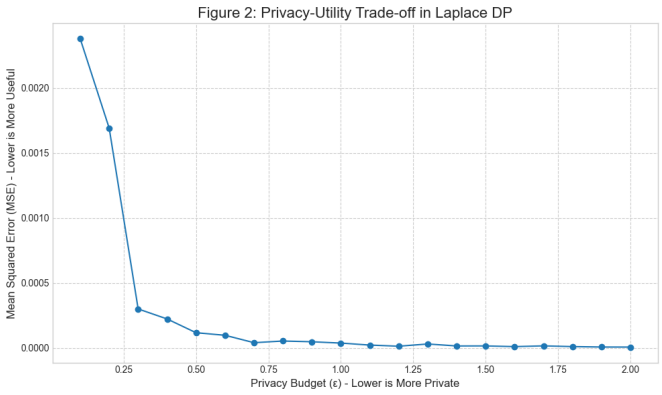

Fig. 2: Privacy-Utility Trade-off in Laplace DP

3) Resilience to Inference Attacks

Figure 3 gives a quantitative estimate of the resistance of each technique to a differencing attack, as measured by **JS Divergence** as a measure of distinguishability. A lesser score is an indication of stronger privacy protection. The outcomes are unambiguous: the Laplace DP approach has the lowest score of the JS Divergence (0.0251), which means that it is the most resilient. The naive approaches, especially Naive Uniform (0.1131), are much more susceptible, since their distributions of output are more distinguishable.

Empirically, this shows that the Laplace noise injected to the data is indeed appropriately scaled so as to render the outputs of adjacent datasets statistically indistinguishable, and thus successfully defeating this type of inference attack and safeguarding individual privacy as the theory would suggest.

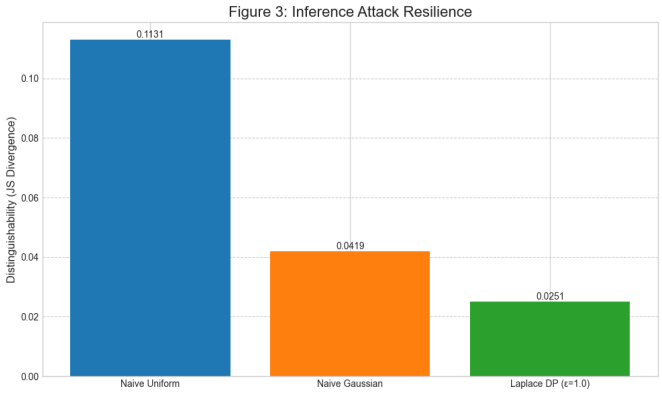

Fig. 3: Inference Attack Resilience

VI. Conclusion

Today, data are both incredibly valuable and surprisingly vulnerable. It is both a vital asset and an ever-growing privacy risk. This study nominates differential privacy as the favorable and analytically sound method to have analytical utility and individual privacy simultaneously. The paper chronologically compares traditional anonymization techniques while pointing out their weaknesses against attacks. Differential privacy ensures resilient privacy with efficient scalability, as the study demonstrates through theoretical explanations and practical applications. From our research and experiments, with the emphasis on the experimental results, we have learned that differential privacy is one step ahead of just being the improved version of traditional methods; instead, it is a safe and secure protection method in today's data-driven world. Balancing the utility and privacy was the hardest part of the concept but overall implementation was straightforward due to the proven algorithms. Learning about possible threats and attacks aimed at sensitive personal information was helpful in grasping the importance of data anonymity.

References

[1] S. Yu, "Big privacy: Challenges and opportunities of privacy study in the age of big data," *IEEE access*, vol. 4, pp. 2751–2763, 2016.

[2] C. Dwork, A. Roth *et al.*, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014.

[3] C. Dwork, "Differential privacy: A survey of results," in *International conference on theory and applications of models of computation*. Springer, 2008, pp. 1–19.

[4] N. Ashena, D. Dell'Aglio, and A. Bernstein, "Understanding ε for differential privacy in differencing attack scenarios," in *International Conference on Security and Privacy in Communication Systems*. Springer, 2021, pp. 187–206.

[5] A. Gadotti, L. Rocher, F. Houssiau, A.-M. Creţu, and Y.-A. de Montjoye, "Anonymization: The imperfect science of using data while preserving privacy," *Science Advances*, vol. 10, no. 29, p. eadn7053, 2024. [Online]. Available: https://www.science.org/doi/abs/10.1126/sciadv.adn7053

[6] C. Li, H. Shirani-Mehr, and X. Yang, "Protecting individual information against inference attacks in data publishing," in *Advances in Databases: Concepts, Systems and Applications: 12th International Conference on Database Systems for Advanced Applications, DASFAA 2007, Bangkok, Thailand, April 9-12, 2007. Proceedings 12*. Springer, 2007, pp. 422–433.

[7] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," 1998.

[8] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," *Acm transactions on knowledge discovery from data (tkdd)*, vol. 1, no. 1, pp. 3–es, 2007.

[9] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," in *2007 IEEE 23rd international conference on data engineering*. IEEE, 2006, pp. 106–115.

[10] L. Sweeney, "k-anonymity: A model for protecting privacy," *International journal of uncertainty, fuzziness and knowledge-based systems*, vol. 10, no. 05, pp. 557–570, 2002.

[11] A. Meyerson and R. Williams, "On the complexity of optimal k-anonymity," in *Proceedings of the twenty-third ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, 2004, pp. 223–228.

[12] K. El Emam and F. K. Dankar, "Protecting privacy using k-anonymity," *Journal of the American Medical Informatics Association*, vol. 15, no. 5, pp. 627–637, 2008.

[13] R. Danger, "Differential privacy: What is all the noise about?" 2022. [Online]. Available: https://arxiv.org/abs/2205.09453

[14] C. Dwork, "Differential privacy," in *International colloquium on automata, languages, and programming*. Springer, 2006, pp. 1–12.