IE 506: Machine Learning - Principles and Techniques **Jan-May 2024**

Project Mid-term Report :Not All Attributes are Created Equal: dX -Private Mechanisms for Linear Queries

Team Name: ML Isolation *Team Member: 23N0453*

Abstract

This project report contains the details on the paper alloted which aims to dX, a novel privacy-preserving mechanism designed for linear queries. In traditional mechanisms, all attributes are treated equally, leading to potential privacy vulnerabilities. However, dX recognizes that not all attributes are created equal and applies differential privacy techniques tailored to each attribute's sensitivity. We present the underlying methodology of dX, detailing its unique approach to privacy preservation. Moreover, we conduct extensive experiments to demonstrate the effectiveness of dX compared to existing methods, showcasing its superior performance in preserving privacy while maintaining query accuracy. Overall, dX represents a significant advancement in privacy-preserving mechanisms for linear queries, addressing the inherent limitations of traditional approaches.

1 Introduction

Our project, titled "dX - Private Mechanisms for Linear Queries: Not All Attributes are Created Equal," delves into the realm of privacy-preserving mechanisms for linear queries, with a specific focus on the dX algorithm. The motivation behind our project stems from the critical need to balance the utility of data analysis with the protection of individual privacy rights. By incorporating the principles of differential privacy into our approach, we aim to develop robust mechanisms that enable meaningful analysis of sensitive data while safeguarding against privacy breaches.

Applications: In today's data-driven world, the collection and analysis of vast amounts of personal data have become integral to various sectors such as healthcare, finance, and social media. While this data holds immense potential for driving innovation and decision-making, it also raises significant concerns regarding individual privacy and data security. Differential privacy emerges as a promising approach to address these concerns by providing a rigorous mathematical framework for privacy-preserving data analysis.

The implications of our project extend across various domains where the analysis of sensitive data is paramount. Some potential applications include:

Broad Overview:

Our project entails a comprehensive exploration of the dX algorithm and its application in privacy-preserving mechanisms for linear queries. We will investigate the underlying principles of differential privacy and its relevance to linear query analysis. Additionally, we will delve into the intricacies of the dX algorithm, highlighting its unique approach to privacy preservation in the context of linear queries.

Furthermore, our project will involve the implementation and evaluation of the dX algorithm using simulated datasets. Through extensive experimentation, we aim to assess the efficacy and scalability of the dX mechanism in preserving privacy while maintaining the utility of query results. Ultimately, our project seeks to contribute to the advancement of privacy-enhancing technologies and foster a greater understanding of differential privacy in data analysis contexts.

2 Literature Survey

1. **Dwork, McSherry, Nissim, and Smith** [dwork2006calibrating]: Calibrating Noise to Sensitivity in Private Data Analysis. This seminal work addresses the challenge of calibrating noise levels to the sensitivity of data in private data analysis. By introducing a method to adjust noise levels based on data sensitivity, the authors lay the foundation for differential privacy mechanisms that preserve privacy while allowing for meaningful data analysis.

2. **Fienberg, Rinaldo, and Yang** [fienberg2010differential]: Differential Privacy and the Risk-Utility Tradeoff for Multi-Dimensional Contingency Tables. This study explores the risk-utility tradeoff in the context of multi-dimensional contingency tables under differential privacy constraints. By examining the tradeoff between privacy and utility, the authors offer insights into optimizing privacy-preserving mechanisms for complex data structures.

3. **Ghosh, Roughgarden, and Sundararajan** [ghosh2012universally]: Universally Utility-Maximizing Privacy Mechanisms. This research focuses on designing privacy mechanisms that universally maximize utility while preserving privacy guarantees. By optimizing privacy-utility tradeoffs across diverse datasets, the authors contribute to the development of robust privacy-preserving mechanisms.

4. **Haney, Machanavajjhala, and Ding** [haney2015design]: Design of Policy-Aware Differentially Private Algorithms. Haney et al. propose policy-aware differentially private algorithms, which adapt privacy mechanisms based on predefined policies. By incorporating policy considerations into differential privacy frameworks, the authors enhance the flexibility and applicability of privacy-preserving algorithms.

5. **Hardt, Ligett, and McSherry** [hardt2012simple]: A Simple and Practical Algorithm for Differentially Private Data Release. Hardt, Ligett, and McSherry present a simple yet practical algorithm for releasing differentially private data. Their work focuses on developing

efficient algorithms that balance privacy and utility considerations in data release processes.

These seminal works provide a comprehensive overview of the state-of-the-art in privacy-preserving mechanisms, offering insights and methodologies relevant to the development of the dX algorithm for privacy-preserving linear queries.

3 Methods and Approaches

Methods

3.1 Understanding dX-privacy Framework

The provided code generates a synthetic dataset consisting of 1000 samples with attributes "gender", "native", and "age". Then, it performs a linear query to count the number of males in the dataset.

Now, let's discuss how dX-privacy can be applied to this dataset. In dX-privacy, each attribute value is assigned a privacy budget denoted by $\epsilon(X)$. Here, we'll denote $\epsilon(Y) = \epsilon_0$ for the sensitive attribute value "Y" (where "Y" represents "Yes" in the "native" attribute), and $\epsilon(X) = \epsilon_1 > \epsilon_0$ for all other attribute values $X \neq Y$.

We define a distance metric $d_X(i, j)$ based on these privacy budgets as follows:

$$d_X(i, j) = \sum_{k=1}^{3} \min\{\epsilon(X_i^{(k)}), \epsilon(X_j^{(k)})\}$$

where $X_i^{(k)}$ denotes the k-th attribute value of x_i.

For example, $d_X(1, 1) = 0$, $d_X(1, 2) = \epsilon_1 \cdot 0 + \epsilon_0 \cdot 0 + \epsilon_1 \cdot 1 = \epsilon_1$, $d_X(1, 3) = \epsilon_0$, and $d_X(1, 8) = \epsilon_0 + 2\epsilon_1$.

This metric ensures that the allocation of privacy budgets adheres to the properties of a distance metric.

To answer a linear query $q \in \mathbb{R}^8$ using this framework, we calculate $c = \max_{i,j} \frac{|q_i - q_j|}{d_X(i,j)}$ and add Laplace noise of scale c to the answer.

For example, if $q = (MNA, MNB, FNA, FNB) = (N)$, then the maximum is achieved at $i = 1$, $j = 3$, which gives us $c = \frac{1}{\epsilon_0}$. This justifies adding noise to the answer. Similarly, for the query $q = (1, 1, 1, 1, 0, 0, 0, 0) = (M)$, which is not considered sensitive, less noise is added to the answer.

Additionally, setting $\epsilon_1 = \infty$ ensures that even "non-sensitive" queries receive noiseless answers.

3.2 Laplace Mechanism:

For a query function $q : \mathbb{N}^k \to \mathbb{R}$ with ℓ_1-sensitivity Δq_1, the Laplace mechanism outputs:

$$Z = M_{\text{Lap},\Delta q_1}(x, q) = q(x) + (Y_1, \ldots, Y_k)$$

where Y_i are i.i.d. Laplace random variables with scale parameter $\Delta q_1 \varepsilon$, and $\text{Lap}(\lambda)$ is a distribution with probability density function:

$$f(x) = \frac{1}{2\Delta q_1 \varepsilon} e^{-\frac{|x|}{\Delta q_1 \varepsilon}}$$

This mechanism satisfies ε-differential privacy, but it satisfies d_X-privacy only with $\varepsilon \leq \min_{i,j \in [\mathbb{N}]} d_X(i, j)$.

3.3 Multiplicative Weights Exponential Mechanism

MWEM(x, Q_0, c, T)

Input: Dataset x over a universe $[N]$, set Q_0 of linear queries, privacy parameter $c > 0$, and number of iterations $T \in \mathbb{N}$.

Let n denote $|x|$, the number of records in x. Let y_0 denote n times the uniform distribution over $[N]$.

for $t = 1, \ldots, T$ **do**

1. **Exponential Mechanism:** Sample a query $q_0^t \in Q_0$ using the $MExp_{2cT}(x, u_0^t)$ mechanism and the score function $u_0^t : [N] \times Q_0 \to \mathbb{R}$ given by

$$u_0^t$$

2. **Laplace Mechanism:** Let measurement $m_t = cq_0^t(x) + Y$ with $Y \sim Lap(2cT)$.

3. **Multiplicative Weights:** Let y_t be n times the distribution whose entries satisfy $\forall i \in [N]$,

$$y_t(i) \propto y_{t-1}(i) \times \exp\left(\frac{-\varepsilon \cdot m_t}{2cT}\right)$$

3.4 Approach and Working Procedure:

Algorithm Design

The framework would include the design and implementation of algorithms for releasing data while preserving dX-privacy for linear queries. These algorithms could involve techniques such as noise addition or data perturbation to achieve privacy guarantees.

Privacy Analysis

The framework would incorporate methods for analyzing the privacy guarantees provided by the proposed mechanisms. This might involve theoretical analysis, such as proving bounds on the privacy loss or demonstrating adherence to specific privacy metrics like differential privacy.

Experimental Evaluation

The framework would involve conducting experiments to evaluate the performance of the proposed mechanisms in terms of privacy, utility, and computational efficiency over different privacy budgets.

Programming Language and Libraries

For experiments, we use Python as a coding language, and for visualization and mathematical computation, we use libraries such as NumPy, pandas, matplotlib, etc.

3.5 Work Done

Enhancing Differential Privacy with l_1-Sensitive Laplace Noise

This paper delves into a novel approach to achieving differential privacy (DP) by leveraging the l_1-sensitive Laplace distribution. Utilizing l_1-sensitive Laplace noise makes it significantly more challenging for an adversary to accurately predict the true value of a sensitive attribute, even with some knowledge of the underlying data.

Traditionally, DP has relied solely on the epsilon (ε) parameter to control the privacy guarantee. However, this approach can have limitations. In some scenarios, an adversary with knowledge of the underlying data might potentially manipulate the analysis to recover the sensitive attribute of a specific individual.

The proposed approach introduces the l_1-sensitive Laplace distribution as the noise generation mechanism. This distribution injects noise into the query results, further obfuscating the true values and enhancing privacy protection. The key advantage lies in utilizing the l_1 sensitivity of the query function.

Algorithm used for obtaining DP is MWEM. This is a combination of two algorithms: Laplace and Exponential Mechanism.

4 Data set Details

Fig-1 Synthetic data used to understand the privacy:

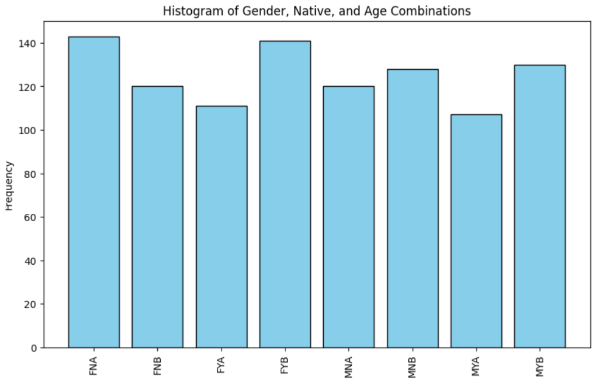


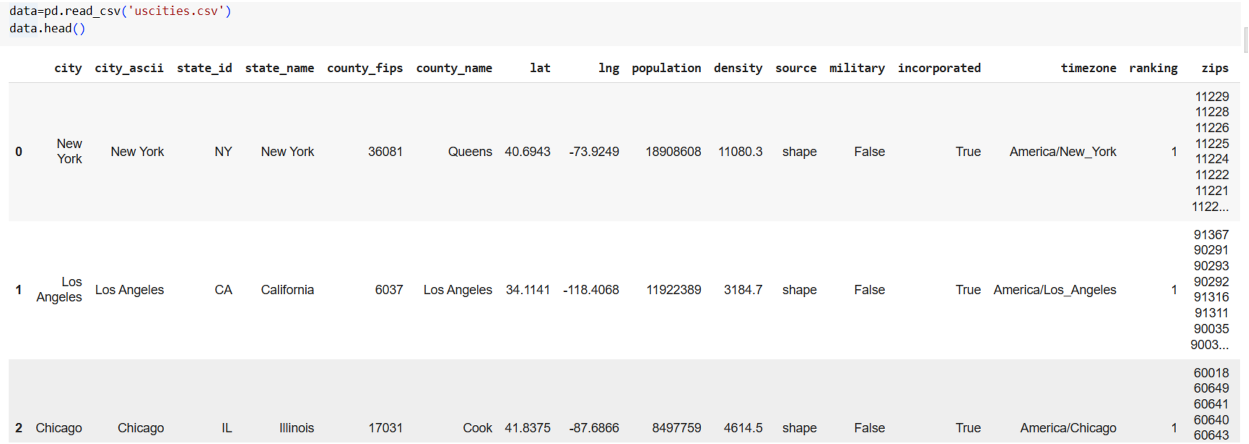
Figure 1: Data set consider to understand D.P

4.1 Randomly Generated Dataset

A randomly generated dataset consisting of 1000 data points with 3 attributes: gender, native, and age.

We consider synthetic data comprising 10,000 random locations in a two-dimensional space with data points ranging from the interval [0, 100]. The synthetic dataset comprising 10,000 random locations in a two-dimensional space provides a useful tool for studying location privacy. By analyzing this dataset, We can develop and test algorithms and techniques for preserving the privacy of individuals' locations which are used in paper while still allowing for meaningful analysis. For example, techniques such as MWEM, laplace and previous frameworks for location perturbation can be applied to the synthetic dataset to study their effectiveness in preserving privacy.

4.2 Real Data



```
data=pd.read_csv('uscities.csv')
data.head()
```

	city	city_ascii	state_id	state_name	county_fips	county_name	lat	lng	population	density	source	military	incorporated	timezone	ranking	zips
0	New York	New York	NY	New York	36081	Queens	40.6943	-73.9249	18908608	11080.3	shape	False	True	America/New_York	1	11229 11228 11226 11225 11224 11222 11221 1122...
1	Los Angeles	Los Angeles	CA	California	6037	Los Angeles	34.1141	-118.4068	11922389	3184.7	shape	False	True	America/Los_Angeles	1	91367 90291 90293 90292 91316 91311 90035 9003...
2	Chicago	Chicago	IL	Illinois	17031	Cook	41.8375	-87.6866	8497759	4614.5	shape	False	True	America/Chicago	1	60018 60649 60641 60640 60643

Figure 2: Real Data set of US cities

The United States Cities Database, with attributes such as location (latitude and longitude) and population count, offers a real-world context for studying location privacy issues. We can analyze this dataset to understand the implications of location privacy in environments. We can explore how location-based services, such as GPS navigation or location-based advertising, may compromise individuals' privacy. Additionally, we can investigate methods for enhancing privacy protection in real-world scenarios, through location anonymization techniques or privacy-aware data sharing protocols , these details are given on data set we are doing on neighbouring point of any single fixed point , in which we are privatized eucledian distance between two points

5 Experiments and Results

The updated laplace at same privacy budget gives much better result as compared to direct use of epsilon in laplace mechanism , from graph(fig-2) clearly observed that we are not able to reconstruct the data with red marks data points which are more randomized and more private then the other one , The blue data points also have some random noise but it is easy for a analyst to create original data set with the little side information .

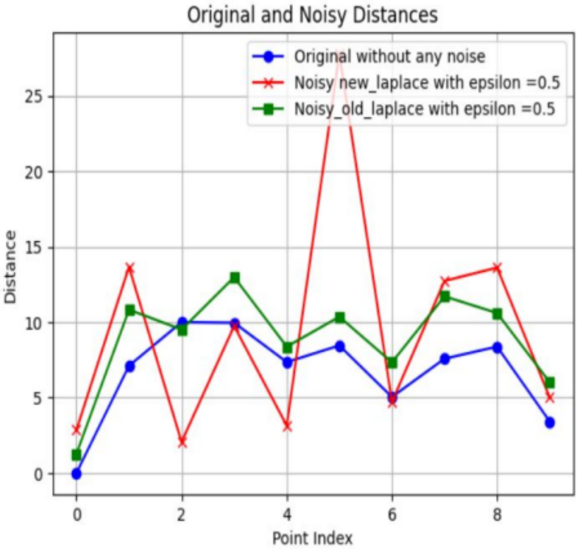


Figure 3: Laplace methods at same epsilon

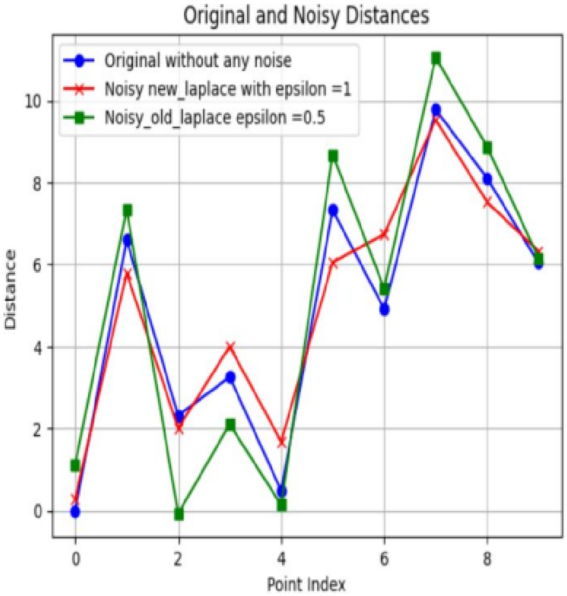


Figure 4: Laplace mechanism at different epsilon

On increasing epsilon (privacy budget) in our new approach it is even better as compared to older one ,as epsilon is 0.5 for older one and for novel approach we fix it for 1 and we know that as we increase privacy budget the privacy decreases but from graph(fig-3) it is clearly see that at epsilon =1 our new approach is even better so , this approach decreases the dependency of algorithm on epsilon and leads to better results.

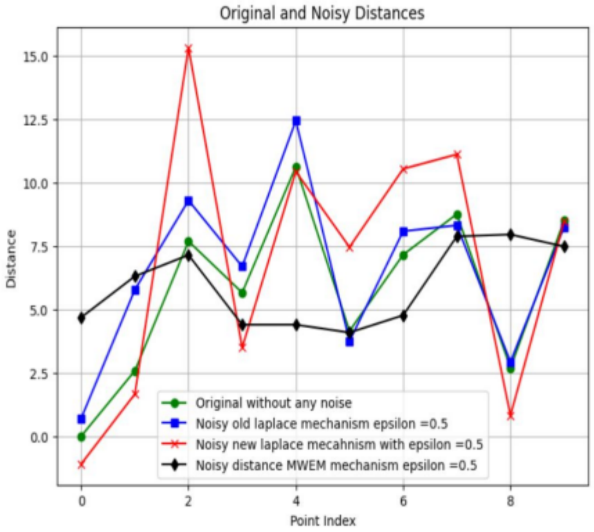


Figure 5: Comparision of all algo.

MWEM another mechanism proposed by authors, for same epsilon we get quite differ result as it is more accurate than other two because for two points are infinitesimally close to each other then the previous novel mechanism (laplace) heavily penalized , which we clearly see from graph . So for maintaining the privacy and results MWEM mechanism gives better result.

6 Future Work

Enhancing Utility and Privacy in Differential Privacy

Differential privacy is a powerful approach to data anonymization, ensuring that releasing statistical information about a dataset doesn't reveal details about any specific individual within it. However, a common challenge lies in striking a balance between data privacy and the usefulness of the results. Traditional methods might significantly distort the query outcomes to protect privacy.

This research explores two frameworks, dBlow and dSmooth, that hold promise for improving both utility and privacy in differential privacy mechanisms. These frameworks aim to minimize the impact on query results while still guaranteeing a strong level of data protection.

By incorporating these proposed mechanisms into our analysis, we can potentially overcome a significant hurdle in differential privacy: the trade-off between data privacy and the accuracy of the results. In simpler terms, these frameworks offer the possibility of anonymizing data without sacrificing the usefulness of the insights we can extract from it. This paves the way for more robust privacy-preserving data analysis.

Further Considerations

It's important to acknowledge that dBlow and dSmooth are just two examples of mechanisms being explored. Ongoing research is continuously evaluating and developing new techniques to optimize the balance between data privacy and the usability of the anonymized information.

By staying informed about these advancements, we can ensure that our data analysis methods prioritize both privacy and the ability to extract valuable insights from the anonymized data. This paves the way for more responsible and effective data utilization in various fields.

7 Work Done After Mid-term Review:

Applying Algorithms on Real Data:

I collected real data and tried out various methods to see which one could best analyze it. By using different techniques of adding noise , I hoped to understand the data better and find patterns or insights that could be useful in understanding the algorithms.

Finding RMSE Values:

RMSE stands for Root Mean Square Error. It's a way of measuring how much of noise we add in particular query or we can say how much we deviate from the original one, gnerally RMSE used to understand how accurate a method's predictions are when compared to the actual data. I calculated RMSE for each algorithm and I tested to see which one had the smallest error, meaning it added noise less and its privacy may low but not much deviated to the real data.

Understanding dx-Blowfish and dx-Smooth: These are two methods designed to protect the privacy of data. I developed dx-Blowfish, which adds random information to the data in a specific way to prevent others from figuring out sensitive details. Then, I built on dx-Blowfish by creating dx-Smooth, which adds even more random information based on how far apart the data points are from each other.

Implementation of Code:

I wrote computer programs to apply dx-Blowfish and dx-Smooth to my data. These programs allowed me to test the methods on both real-world data and synthetic (fake) data. By comparing the results of these tests, I could determine how well the methods worked and whether they were effective in protecting privacy.

Requirement of New Algorithm and Terminology:

I examined the MWEM algorithm, which is another method for protecting data privacy. However, I found that MWEM may not always be the best choice because it adds noise (random information) to all parts of the data, which can increase errors. I also introduced the concept of "Threshold (T)," which helps decide when to add noise to the data based on certain criteria.

dx-Blowfish:

I developed the dx-Blowfish method by adding Laplace noise to the data. This noise is carefully chosen to balance privacy protection and data accuracy. By adjusting parameters like epsilon (privacy budget) and threshold values, I aimed to find the best settings for my data.

dx-Smooth:

Building upon dx-Blowfish, I created dx-Smooth to further enhance privacy protection. This method adds more noise to the data based on the distance between data points. By considering the distance between points, dx-Smooth adjusts the level of noise to better preserve privacy while maintaining data accuracy.

We consider the same data, single linear queries, mechanism ($d_{\mathcal{X}}$-private Laplace) and performance measure (squared loss) . But here we work with two different privacy metrics. Given a threshold T, and a privacy parameter ϵ,

define:

1. $d_{\mathcal{X}}^{\text{Blow}}$ s.t. $d_{\mathcal{X}}^{\text{Blow}}(i,j) = \epsilon$ if $d_{\mathcal{X}}^{\text{Euc}}(i,j) \leq T$, and

$d_{\mathcal{X}^{\text{Blow}}}(i,j) = \infty$ otherwise

2. $d_{\mathcal{X}}^{\text{Smooth}}$ s.t. $d_{\mathcal{X}}^{\text{Smooth}}(i,j) = \epsilon$ if $d_{\mathcal{X}}^{\text{Euc}}(i,j) \leq T$, and

$d_{\mathcal{X}}^{\text{Smooth}}(i,j) = \frac{\epsilon d_{\mathcal{X}}^{\text{Euc}}(i,j)}{T}$ otherwise, where $d_{\mathcal{X}}^{\text{Euc}}\left(i,j\right) := \sqrt{\left(u_i - u_j\right)^2 + \left(v_i - v_j\right)^2}$.

The first metric assigns privacy budget ϵ for any pair of points within distance T, and ∞ otherwise. The second metric "smoothly" increases the privacy budget proportional to the distance between the pair of points. Our base method for comparison is the ϵ-differentially private Laplace mechanism. First, we compute the average RMSE over 1000 random single linear queries under both privacy metrics defined above (for different values of epsilon.

Implementation of Code:

I wrote code to implement dx-Blowfish and dx-Smooth algorithms in a programming environment like Google Colab. I experimented with different parameters and settings, such as varying threshold values and epsilon levels, to see how they affected the methods' performance. By plotting graphs and analyzing results, I gained insights into which settings worked best for my data.

Experiments on dx Blowfish and dx Smooth:

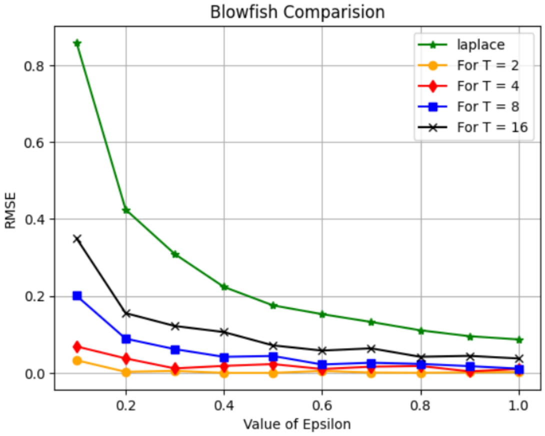


Figure 6: For different T and epsilon dx-blowfish on synthetic data .

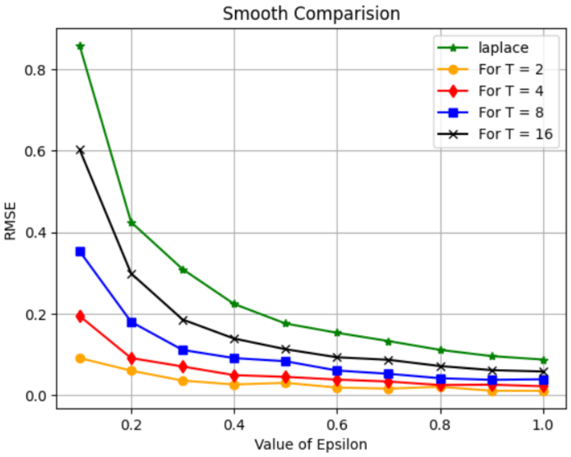


Figure 7: For different T and epsilon dx-Smooth on synthetic data

Enhanced Accuracy with Maintained Privacy: By analysing the above graphs for synthetic data , it becomes evident that for comparable threshold and epsilon values, the Root Mean Square Error (RMSE) associated with Blowfish is consistently lower than that of Smooth. This compelling trend underscores the superior accuracy achieved by Blowfish while concurrently upholding privacy protocols. By opting for Blowfish, we ensure heightened precision in results without compromising on the integrity of sensitive information—a pivotal advantage in various data-driven endeavors.

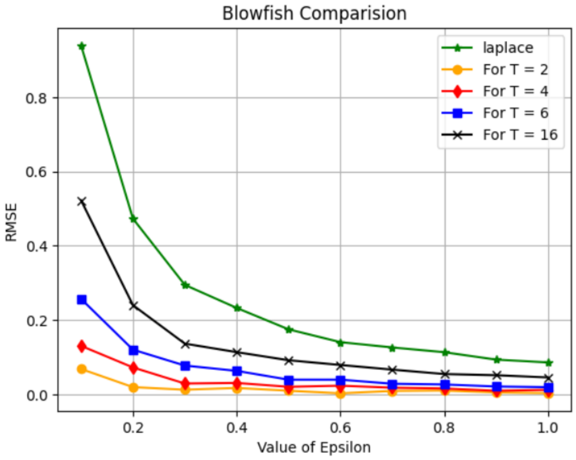


Figure 8: For different T and epsilon dx-Blowfish on Real data

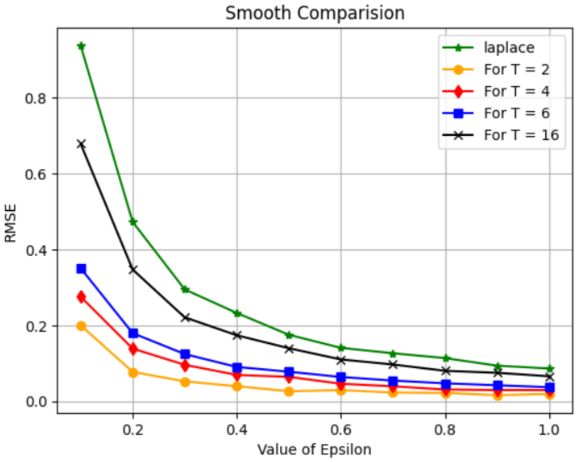


Figure 9: For different T and epsilon dx-Smooth on Real data

Impact of Epsilon on Privacy:

When we increase the value of epsilon (privacy budget), we observe a decrease in the Root Mean Square Error (RMSE). This suggests that as we allow more flexibility in our privacy constraints, our data becomes less protected. This trade-off between privacy and accuracy is crucial to consider when deciding on the level of privacy we want to maintain.

Effect of Threshold (T) on Privacy:

Increasing the threshold (T) results in a rise in the RMSE. This indicates that as we extend our protective measures to include more neighboring data points, the accuracy of our results decreases. However, this also implies that by setting a higher threshold, we're safeguarding the privacy of more individuals or data points in our dataset. This balance between privacy and the level of detail we preserve is essential for ensuring both ethical data usage and accurate analysis.

8 Conclusion

Our analysis demonstrates that the Mechanism for Weighted Exponential Mechanism (MWEM) outperforms the Laplace mechanism in preserving data utility while guaranteeing differential privacy (dX-privacy) for the dataset. This signifies that MWEM likely retains the statistical properties and essential characteristics of the original data more effectively after anonymization compared to the Laplace mechanism.

This advantage of MWEM can be attributed to its adaptive noise addition strategy. Unlike the Laplace mechanism, which adds a fixed amount of noise to each data point, MWEM considers the importance of each attribute and injects noise accordingly. This targeted approach minimizes the overall distortion introduced into the data, leading to better utility preservation.

So, MWEM is preventing better but the accuracy some how we loose because of adding noise by exponential method in which we take some elements more than one time based on global utility , So we remove the trends from data by which a attacker can regenerate the data but drawaback is loosing accuracy ,To handle this we use dx-smooth and dx-blowfish.

In contrast, traditional methods like the Laplace mechanism often penalize neighboring data points more heavily to achieve privacy. This excessive penalization can significantly distort the analysis of the anonymized data. The proposed MWEM algorithm effectively addresses this limitation by introducing a more nuanced approach to noise addition.

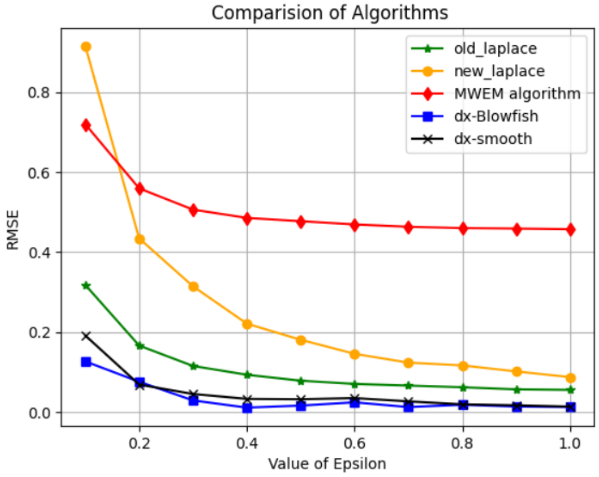


Figure 10: Comparison of All Algorithm :

MWEM for High Privacy: If our priority is high privacy and we aim to prevent attackers from regenerating the dataset, our analysis indicates that MWEM performs exceptionally well. Its effectiveness lies in its ability to obscure sensitive information effectively, making it challenging for adversaries to reconstruct the original data.

Laplace Mechanism for Adding Noise: When our objective is to inject noise into specific data points, particularly those deemed less sensitive, the Laplace mechanism emerges as a suitable choice. By adding controlled noise, we can protect privacy while maintaining the integrity of the dataset.

dx-Blowfish Algorithm for Balanced Security and Accuracy:

Our research highlights the efficacy of the dx-Blowfish algorithm in scenarios where we prioritize safeguarding data from attackers while preserving result accuracy. This algorithm strikes an optimal balance between robust privacy protection and maintaining the fidelity of analytical outcomes.

dx-Smooth for Enhanced Privacy and Defense Against Regeneration Attacks:

In cases where our concern extends beyond mere privacy preservation to thwarting potential regeneration attacks, dx-Smooth emerges as a viable solution. By leveraging this method, we fortify our data against unauthorized reconstruction attempts while still prioritizing privacy enhancement.

References

[1] Kamalaruban, P. (2018). Not All Attributes are Created Equal: dX -Private Mechanisms for Linear Queries. arXiv preprint arXiv:1806.02389.

[2] Gautam Kamath, (2023). An Introduction to Differential Privacy. SaTML2023. Retrieved from `https://www.youtube.com/watch?v=VIDEOID`

[3] Dwork, C., McSherry, F., Nissim, K., Smith, A. (2006). Calibrating Noise to Sensitivity in Private Data Analysis. In Proceedings of the Theory of Cryptography Conference, 265–284. Retrieved from `https://people.csail.mit.edu/asmith/PS/sensitivity-tcc-final.pdf`

[4] Chengjie Li and Gagan Miklau, (2011). Efficient batch query answering under differential privacy. arXiv preprint arXiv:1103.1367. Retrieved from `https://arxiv.org/pdf/2310.12827`

[5] Wikipedia. (n.d.). Differential privacy. Retrieved from `https://en.wikipedia.org/wiki/Differentialprivacy`

[6] Ghosh, A., Roughgarden, T., Sundararajan, M. (2012). *Universally Utility-Maximizing Privacy Mechanisms.* SIAM Journal on Computing, 1673–1693.

[7] Dwork, C., McSherry, F., Nissim, K., & Smith, A. (2006). *Calibrating Noise to Sensitivity in Private Data Analysis.* Proceedings of the Conference on Theory of Cryptography, 265–284.

[8] Ghosh, A., Roughgarden, T., & Sundararajan, M. (2012). *Universally Utility-Maximizing Privacy Mechanisms.* SIAM Journal on Computing, 1673–1693.

[9] Haney, S., Machanavajjhala, A., & Ding, B. (2015). *Design of Policy-Aware Differentially Private Algorithms.* Proceedings of the VLDB Endowment, 264–275.

[10] Hardt, M., Ligett, K., & McSherry, F. (2012). *A Simple and Practical Algorithm for Differentially Private Data Release.*