Differential Privacy with Dynamic Privacy Budget Allocation

Student: CHAN, Lo Yuet (1662438)

Supervisor: Dr. Grigorios Loukides

Module Number: 7CCSMDPJ

Date of Submission: 24-Aug-2017

1. Abstract

With the prevalence of social media platform and data capture solution, more and more sensitive personal data are available for different businesses. While the data can prove useful in business intelligence, research, review, and even urban planning tasks, there are growing concerns over privacy protection when handling sensitive data. Data leaks have been disastrous for both customers and the company. Hence, it becomes critical to strike a balance between privacy risk mitigation and publishing accurate but sensitive data. If the data will be published in histograms, the current state-of-art differential privacy algorithm for handling histograms, AHP, can be adopted. In this thesis, we have reviewed AHP and looked into its limitation. As AHP can only apply noises of equal scale to each bin, there may be a problem of data pattern loss due to big clusters at the lower threshold. To address the problem, we present an improved version of AHP which can apply different scale of Laplace noise to different bins such that privacy can be protected with better data pattern preservation. The novel algorithm DPA-AHP is capable of replicating and beating AHP when generating private but more accurate histograms while keeping the time complexity the same as the original AHP in various experiments.

2. Acknowledgement

I would like to thank my thesis supervisor Dr. Grigorios Loukides of the Department of Informatics at King's College London. Dr. Loukides has always been very helpful whenever I have encountered difficulties or questions about both research and thesis writing. While steering my thesis in the right direction and providing insights, Dr. Loukides allowed the research project to be my own work.

I would also like to thank Dr. Kevin Lano of the Department of Informatics at King's College London as the second reader of this thesis. I am gratefully indebted to Dr. Lano for his valuable comments on both the thesis and the thesis presentation.

3. Table of Contents

4. List of Figures and Tables

5. Nomenclature

AHP	Accurate Histogram Publication algorithm [1]
δ	Step parameter for DPA-AHP. The parameter needs to be a non-negative real number.
Differential Privacy	A technique to blur data into neighbour data such that it is unlikely to be traced backwards from the result.
DPA	Flexible Dynamic Privacy Budget Allocation mechanism.
DPA-AHP	Flexible Dynamic Privacy Budget Allocation mechanism on AHP.
ϵ	A real number greater than 0 which refers to the privacy budget.
$\boldsymbol{H}$	A histogram with bins $\{ \mathrm{h}_1, \mathrm{h}_2, \dots , \mathrm{h}_n \}$.
$Lap(b)$	Laplace distribution with mean 0, and scale b.
$Lap(\mu, b)$	Laplace distribution with mean μ, and scale b.
n	Number of bins in a histogram $\boldsymbol{H}$.
Privacy Budget	A parameter for specifying how strong the level of privacy is being enforced. By definition, a smaller budget will result in higher privacy protection.
SDPA	Simple Dynamic Privacy Budget Allocation mechanism.
SDPA-AHP	Simple Dynamic Privacy Budget Allocation mechanism on AHP.

6. Introduction

6.1. Research Domains

In the past few years, there is a bloom in social network applications like Facebook, Instagram, Snapchat, Foursqaures etc. which allow users to share their life moments and thoughts, as well as check-in to their current locations. The concept of IoT has pushed the situation further as every part of our life would be available on the internet. These has enabled analysts to mine intelligence from user life patterns [14], [15]. It is also made possible that social predictions, business strategies, brand decisions, advertisement campaigns and even strategy reviews can be drawn out of the analysed data [17], [19], [20], [21]. There are also examples where personal data on social networks can be used in urban planning, political campaigns, tourism development, and even detecting safety of medical products [22], [23], [24]. With personal data being more valuable in the process of making and reviewing decisions, it also opens up a market for selling and capturing personal data [21]. This has provided

enough incentives for people to sell more sensitive information, which would bring a win-win situation for both sides of the transactions [18].

While mining into personal data can bring billions of revenue [20], it has, at the same time, raised a lot of concerns over both privacy and security [16]. Such a vulnerability is not just theoretical. In fact, the U.S. Identity Theft Resource Center has published an estimate that about 130 million sensitive personal records are vulnerable to data breaches [21]. However, the story does not stop there. If a data breach happened, it will not only lead to privacy issue, but also a long-lasting catastrophic damage to the firm's reputation. For example, the colossal data breach of Home Depot in 2014 has cost the firm at least $179 million till now without including any legal fees. With the prevalence of social media platforms, the number would only grow bigger unless proper measures are adopted.

Before looking into possible solutions, we would need to understand the type of datasets first. For traditional relational databases, we would expect tables with a record in each row. Databases would grow more complex as there are more tables though. For example, high dimensional data like time series data and set-valued datasets may also be part

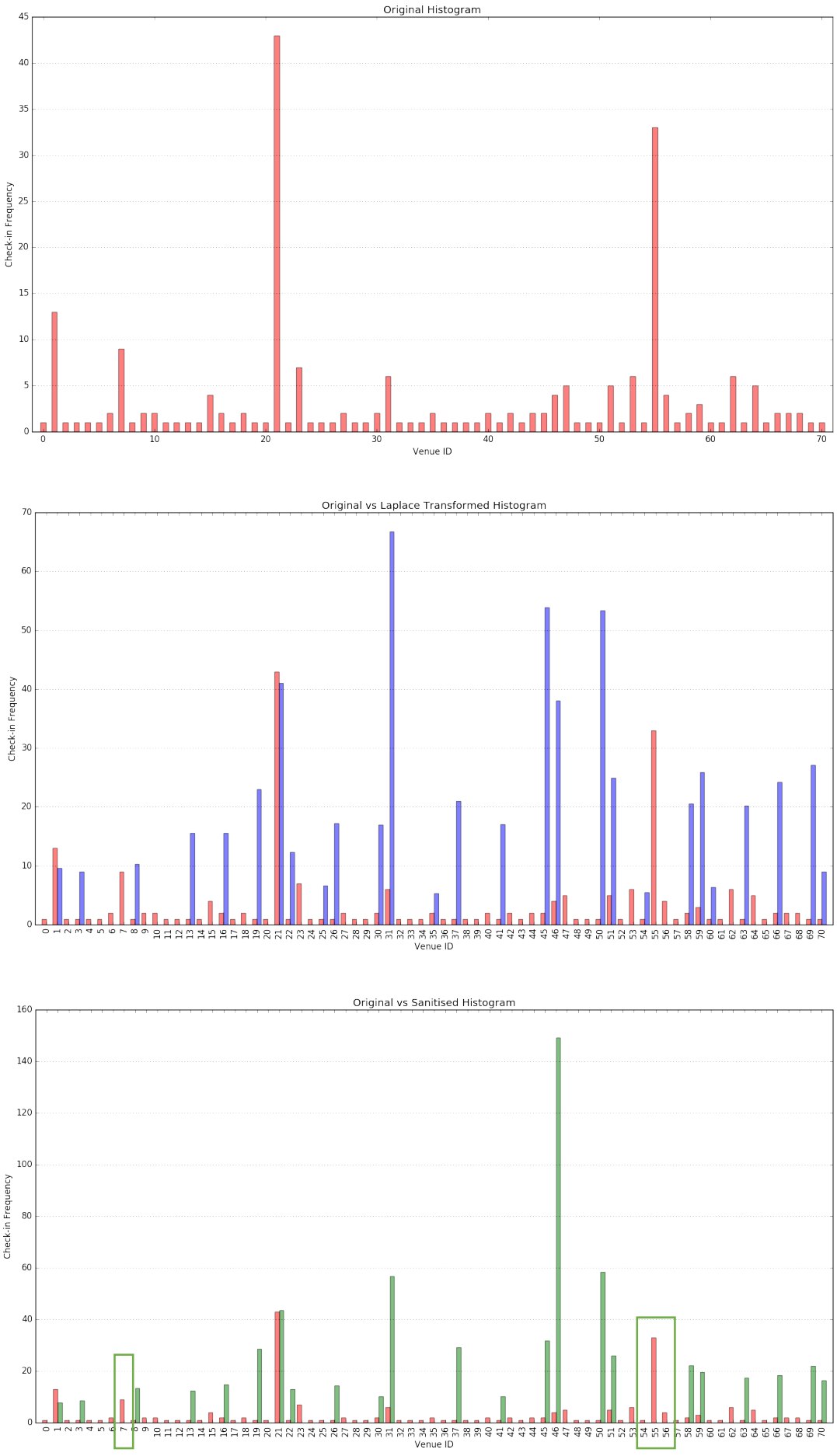


Figure 1 AHP Differential Privacy Mechanism

of the database. On top of all these, there are also dataset stored as graphs or maps.

Example 1. *To improve the hotel recommendation system, a travel metasearch engine collaborated with a marketing firm. In order to provide business intelligence insights, the marketing company has requested the booking records of all the venues in a certain city from all the users for the past three years. The table will thus contain anonymised user ID, anonymised venue ID, and an aggregated booking count during the requested time interval. Histograms will be plotted to look into data distribution, as well as to answer queries. As it is possible to backward-trace the corresponding users by conducting surveys, the travel metasearch engine would like to prevent the mining of any sensitive user data and to protect users from information leak. However, it is still of paramount importance to preserve essential pattern and trends such that accurate business intelligence can be generated.*

Example 2. *In order to review the brand strategy, a cosmetic company worked with an analytics firm. Anonymised datasets containing customer identifiers, customer age, customer gender, product identifier, product categories, and total buying frequency are provided for accurate analysis. Multiple one-dimensional histograms can then be drawn for data mining and query answering purposes. As certain products will have a much higher selling count, which provides a competitive advantage to the cosmetic company, the company would like to prevent any data mining that would harm its competitive advantages. A possible way for the company to do so is to apply noises to it such that the count provided to the analytics firm will be inaccurate but similar to the real ones. This would hopefully help protect the business secret while allow accurate and fruitful data analytics.*

In this thesis, we aim to look into a better solution to solve the dilemma between privacy protection and accurate analytics when publishing one-dimensional histograms as illustrated with the above examples with help of differential privacy. When handling histograms, there are two approaches to enforce higher level of privacy, namely approximation, and noise addition [1]. By adding noises, bins are blurred into neighbour bins such that it is highly unlikely that each bin can be backward recognised from the sanitised output. Laplace noise (following Laplace distribution) is one of the symmetric and thin-tailed type of noise that can be used for enforcing privacy. With the current state-of-art mechanism AHP, Laplace noise would be applied to every bin in the histogram to mask the original counts with the scale of noise being specified by an input parameter ϵ [1]. Such a parameter will determine how strong the privacy protection will be. AHP also attempt to minimise error introduced by clustering neighbour bins (bins with similar counts) after applying Laplace noise. The sanitised count is generated as a masked estimator for the original counts afterwards. Figure 1 shows a series of histograms throughout the process. In the histograms, red bars are representing the original counts, blue bars are representing the masked counts and green bars are representing the sanitised outputs. As we can observe, the sanitised histogram still inherit part of the data distribution of the original histogram while being vastly different from it.

6.2. Research Questions

While AHP can beat other competing algorithms like PHP and GS [1], AHP seems to have neglected the fact that by applying noises of equal scale to every bin, bins with smaller counts would be more likely to hit the lower threshold resulting in a big cluster around the lower threshold. It can be observed in Figure 1 that a number of the non-zero bins from the original histogram are clustered at the lower threshold, zero, after sanitisation. There are even peak bins that are being dropped after sanitisation (as shown in the red rectangles in Figure 1). This would lead to a loss of data pattern and distribution preservation.

In fact, real life datasets can contain data of any distributions. When applying noises onto these sets, we would need to be aware if a one-for-all mechanism can really achieve the global optimal results. Hence, in this project, we would focus on the following three research questions.

1. **Can AHP be adapted onto datasets with different distribution?**
   We will revise the current state-of-art algorithm to see if it is possible to tackle the problem from within. If it is unfortunately not possible, we will move on to the next research question.

2. **Is it possible to extend AHP or to develop a new mechanism to address the problem?**

As the current state-of-art algorithm failed to address the problem, we would then look into possible extensions or modifications on AHP to enhance performance and flexibility while addressing the problem of clustering around lower threshold.

3. **How much time would it cost to implement such a mechanism?**
   While it is important that the new mechanism can perform as well as AHP, it is also very important to minimise the run time such that it will not require too much of extra resources than AHP.

6.3. Summary of Findings

With the three research questions, we have looked into both AHP and all the related works. As AHP allocates differential privacy budget equally across all the bins in the original histogram, it is not possible for AHP to tune the distribution of privacy budget when encountering different datasets.

As a result, we have come up with two new $O(n)$ complex mechanisms on top of AHP, *Simple Dynamic Privacy Budget Allocation on AHP*(SDPA-AHP) and *Flexible Dynamic Privacy Budget Allocation on AHP*(DPA-AHP), where n is the number of bins in a histogram. SDPA-AHP would be a fixed algorithm that allocated descending amount of privacy budget onto bins based on their ranks across the histogram after sorting ascendingly. To allow more interactivity, the DPA-AHP will allow users to specify how would the privacy budget be allocated differently on different bins. The interactivity will be specified with an input parameter δ which accepts any value greater than zero.

More importantly, the mechanisms are proved to be differentially private which makes these mechanisms sufficient to enforce differential privacy. On top of that, the DPA-AHP mechanism extends the concept of AHP and SDPA-AHP, making both of them a special case under FDPA-AHP with $\delta = 0$ and $\delta = 1$ respectively.

After carrying out a series of testing and experiment, we have discovered that both KLD and MSE would decrease follow by an exponential increase as δ increases from 0 to 1 with different privacy budgets (0.01, 0.1, and 1). The optimal solution is usually located at the range between $\delta = 0.05$ and $\delta = 0.1$. As a result, $\delta = 0.05$ or $\delta = 0.075$ is suggested for obtaining a better result than AHP. The optimal δ would however be different for different dataset and different privacy budget.

While for time complexity, the Flexible Dynamic Privacy Budget Allocation align with the $O(n)$ time complexity of AHP, and hence would not require significant additional resources. From the experiments, the mechanism would take less than 20% of the total CPU time for sorting bin and allocating privacy budget dynamically.

6.4. Organisation

In the upcoming sections, we will look into preliminary knowledge for understanding the findings in this paper (Section 7). We have also included reviews of related work in the field of differential privacy (Section 8). Approaches and testing datasets will be discussed in Section 9 while test results and experiments concerning the novel mechanism will be analysed in Section 10. Our summarised conclusion will be included in Section 11.

7. Background

In this section, we will discuss the preliminary knowledge required to understand this thesis. We will look into histograms (Section 7.1), Laplace Noise (Section 7.2) and their definitions. With these knowledge, we will then discuss differential privacy, its definition and theorem in Section 7.3. Range Queries (Section 7.4) will also be included.

7.1. Histogram

Given an attribute θ with a set of values V in a one dimensional dataset D, we can aggregate the count (or the frequency), f for each value $v \in V$. With the set of θ and its corresponding count f, a histogram, H can be generated. A histogram H regarding the attribute θ can then be denoted as $\boldsymbol{H} = \{\mathrm{h}_1, \mathrm{h}_2, \dots, \mathrm{h}_n\}$ with h_i representing each bin i. Under most circumstances, bin i and bin j do not overlap for every $i \neq j$ (i.e. $range(\mathbf{H}_i) \cap range(\mathbf{H}_j) \neq \emptyset$). With such a histogram, we would be able to answer to different range queries.

7.2. Laplace Noise

In the field of differential privacy, Laplace noise is a very commonly used noise for masking the original data [1]. A Laplace noise will follow a Laplace distribution with parameter mean, μ and scale, b.

Definition 1. Denote a random variable X that follows a Laplace distribution with mean, μ and scale, b or $X \sim Lap(\mu, b)$. Its probability density function is defined as follows:

$$\Pr(X = x) = \frac{1}{2b}\exp\left(-\frac{|x-\mu|}{b}\right) \qquad for\ b > 0, x \in (-\infty, +\infty)$$

Such a probability density function will form a thin-tailed distribution that is symmetric on both sides of mean, μ.

7.3. Differential Privacy

In this article, we will look into algorithms generating output from databases while enforcing sufficient privacy such that released data. Assume there exists histograms $\mathbf{H}_1$ and $\mathbf{H}_2$ from database D_1 and D_2 which differs by exactly one record. $\mathbf{H}_1$ and $\mathbf{H}_2$ can then be referred as neighbours. Based on these assumptions, differential privacy can then be defined as follows:

Definition 2. [1] A randomised algorithm $\mathcal{A}$ is said to be ϵ-differentially private if for any two neighbour histograms $\mathbf{H}_1$, $\mathbf{H}_2$ and any subset of output value $S \subseteq Range(\mathcal{A})$,

$$\Pr(\mathcal{A}(\mathbf{H}_1) \in S) \leq \exp(\epsilon) \cdot \Pr(\mathcal{A}(\mathbf{H}_2) \in S)$$

where $\epsilon > 0$.

With the parameter ϵ, users can specify the level of privacy to be enforced. The smaller ϵ, the higher privacy protection will be applied. Normally, ϵ will be set as a small value (e.g. $\epsilon < 1$) such that removal of one individual data prior to running the algorithm will not significantly affect the output (i.e. the histogram).

It is often that a series of algorithms $\mathcal{A} = \{\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_N\}$ will be run to generate the output. The algorithms will each be ϵ_i-differentially private, and thus can come up with a total privacy budget of $\epsilon = \sum_{i=1}^{N} \epsilon_i$ that will be distributed across all the algorithms $\mathcal{A}_i$.

In the literatures, one of the most well-developed algorithms for enforcing differential privacy is the Laplace mechanism. This mechanism masks the original data with random noises generated from the Laplace distribution, $Lap(\sigma)$ (i.e. a Laplace distribution with mean 0 and scale σ). The scale is determined based on the concept of a sensitivity of a function f over the histogram.

Definition 3. Denote $f(H)$ as a function of the histogram H which generates a vector in $\mathbb{R}^d$. The Laplace mechanism will then give output of $f(I) + \boldsymbol{z}$, where $\boldsymbol{z}$ is a d-length vector and that each $z_i \sim Lap(\Delta f/\epsilon)$. The constant Δf is the sensitivity of f and is defined as the maximum difference in f between 2 neighbouring histograms H_1, H_2 (i.e. $\Delta f = max_{H_1,H_2} \|f(\mathrm{H}_1) - f(\mathrm{H}_2)\|_1$.

Theorem 1.[1] For any function $f : H \rightarrow \mathbb{R}^d$, the algorithm set $\mathcal{A}$ is said to be ϵ-differentially private if

$$\mathcal{A}(H) = f(\boldsymbol{H}) + [Lap_1(\Delta f/\epsilon), \dots, Lap_d(\Delta f/\epsilon)]$$

where each $Lap_i(\Delta f/\epsilon)$ are i.i.d. Laplace variables with scale $\sigma = \Delta f/\epsilon$.

As given any database, adding one new record will lead to an increase by exactly 1, the sensitivity constant $\Delta f = 1$. In other words, the mechanism can be simplified as adding random noises of $Lap(1/\epsilon)$, where ϵ is the privacy budget allocated to the Laplace mechanism.

7.4. Range Queries

When looking up data from histograms, range queries are often used for returning data within a user-specified interval of values.

Definition 4. Denote a range query function $Q(\boldsymbol{H}, i, j)$ with a histogram $\boldsymbol{H} = \{\mathrm{h}_1, \mathrm{h}_2, \dots, \mathrm{h}_n\}$ and i, j as the upper and lower boundary of the desired interval such that $i \neq j$.

$$Q(\boldsymbol{H}, i, j) = \begin{cases} \emptyset & if\ i, j < \min(\boldsymbol{H})\ or\ i, j > \max(\boldsymbol{H}) \\ Q(\boldsymbol{H}, \max(i, \min(\boldsymbol{H})), \min(j, \max(\boldsymbol{H}))) & otherwise \end{cases}$$

where $\max(\mathrm{a}, \mathrm{b})$ and $\min(\mathrm{a}, \mathrm{b})$ are the functions that returns the maximum and minimum value between a and b respectively; while $\max(\mathbf{H})$ and $\min(\mathbf{H})$ returns the maximum and the minimum value in the histogram $\mathbf{H}$.

8. Literature Review

There has been a number of new knowledge in the field of differential privacy recently. To be specific, this project is inspired by the work of Zhang et. al. [1] on AHP and its edge over other differential privacy algorithms that uses clustering. Despite being out of scope, research papers covering node differential privacy (Section 3.2), high dimension data (Section 3.3), differential privacy without external noise (Section 3.5) are also reviewed to obtain insights and a broader picture of the current development in differential privacy. Apart from that, work focusing on calibrating and benchmarking (Section 3.5) is also studied.

Table 1 shows the main algorithms reviewed in this section. The column Scope will specify whether the algorithm lies in the same scope as this thesis; while Dimension represents the dimension constraints on input datasets for the algorithm, for instance, PHP will require users to provide a one-dimension data while AHP does not have any constraints on it. The column Parameter shows the input parameters for the specific algorithm.

Algorithm	Scope	Dimension	Parameters
AHP	Within Scope	Multi-dimension	ρ, η
PHP	Within Scope	One-dimension	ρ
SF	Within Scope	One-dimension	ρ, k, F
MWEM	Out of Scope	Multi-dimension	
DualQuery	Out of Scope	Multi-dimension	

Table 1 Differential Privacy Algorithm Summary

8.1. Clustering Algorithms in Histogram

The principle behind Zhang et. al. [1] is to minimise Laplace error applied onto the dataset with help of AHP. To come up with the optimal solution, three candidate algorithms are put into comparisons.

The AHP suggested is still, however, questionable. Although the algorithm shows a clear edge over others like *PHP* [11], *GS* [12], *NoiseFirst* [13], *StructureFirst* [13], these comparisons are all made with experimental datasets of scale of 10^3 and 10^4. Whether the algorithm excels in datasets of other scales is still questionable.

Free parameters are also not fully discussed in the research. The work focuses on comparisons of algorithm over different privacy budget. However, the fact that privacy budget can be freely allocated at different stage is not fully exploited, but is preset and hard-coded instead makes the work extendable in the future.

8.2. Node Differential Privacy

In this domain, the general concept towards differential privacy is based on graph and networks. Once differential privacy is enforced, nodes and all its edges should not be distinguishable from other nodes and edges [4]. This domain is out of the scope of this project and is studied for gaining insights.

Day et. al. [4] focuses on graph data representation. In this domain, they have suggested two solutions, that has an edge over the current state-of-art in lower-degree nodes. These are all made possible with their novel graph projection mechanism.

Apart from the reminder on sensitivity, the way Day et. al. compared free tuning parameters can also be applicable on this project given some improvement. They have selected a few values for testing their new algorithm before finalising the preset value. This, however, has restricted the flexibility nature of the free tuning parameter, and thus missed that opportunity to optimise and generalise the algorithm with help of a function that help tune these attributes.

8.3. High Dimension Data

For datasets with higher dimensions, enforcing differential privacy will usually suffer from what is called ‘curse of high dimensionality’ resulting in either heavy computation power requirement, excessive error, or low privacy.

This domain is out of the scope of this project and is studied for gaining insights. The current state-of-art algorithm MWEM is reviewed to be a NP-hard computationally complex algorithm by Hardt et. al. [10].

Chen et. al. [5] proposed to solve the 'curse' with a top-down partitioning algorithm on a set-valued dataset. With the novel partition, they yielded a stronger privacy protection with the same amount of relative error. They also suggested that a non-interative data sanitisation can be achieved situationally if underlying data is carefully used. This which would allow users to directly interact with the sanitised dataset without needing a sanitising mechanism as a middleman to modify queries from users.

As their proof for non-interactive data sanitisation is fairly situational, there would still be a big gap of generalising such a sanitisation scheme. On the other hand, their utility is only based on overall amount of relative error. This would have allowed results with low relative error, while having a highly varied error slip through. Further work would be needed to improve the scheme such that error can be kept small and concentrated, and thus, for preserving accuracy of any queries.

In another research, Chen et. al. [3] focuses on solving the problem with a sampling-based inference which allows thicker spread of privacy budget. The solution depends on the conditional independence and the one dimensional association across attributes. With a higher privacy budget at each step, the accumulated noise in the sanitised dataset would be highly reduced.

As the solution depends on the important assumption of conditional independence, its effectiveness would be highly situational, and should be reconsidered before using it as a general approach for enforcing differential privacy.

Despite the presumption, this piece of work brings a new insight towards efficient spending of privacy budget. Sampling-based inference should be further studied so that it may be applied onto other algorithms to improve the overall accuracy.

Dimitrakakis et. al. [7] looked into sampling inference with posterior sampling in Bayesian network. With little previous differential privacy research in the field of Bayesian inference, this piece provided a building block and a proof for using posterior sampling for enforcing differential privacy.

This piece of work is mostly out of the domain of our project as it mainly focuses on theoretical use of Bayesian inference on high dimension datasets. It however shows that the field of differential privacy is evolving so quick that new scheme with different statistics skills are being looked into.

Gaboardi et. al. [8] proposed an algorithm, named DualQuery, with worst-case complexity in exponential relation against dataset dimension. Despite the exponential increase, they have shown that their algorithm is computationally more concise than the state-of-art by applying the algorithm on a synthetic dataset with more than 500,000 attributes.

Their algorithm, however, only shows edge over in high-dimension queries. In other words, MWEM still remains to be the state-of-art in lower-dimensions queries, while Gaboardi et. al. did not provide the boundary for switching from MWEM to DualQuery for saving computational power.

8.3.1 Time-series Data in Histogram

In the domain of time-series data, Chen et. al. [2] considered it as an infinite stream of data. Their work focuses on generating differentially private histograms continuously. A sampling based algorithm with a retroactive grouping mechanism is adopted to enhance computational, and spatial efficiency, and thus incurring a smaller delay when publishing histograms.

As timestamps are not the major focus of this project, the work from Chen et. al. is mainly out of scope. However, the use of a Bernoulli sampling scheme may be applicable on this project for getting a more accurate differentially private result while suffering from reduced Laplace noise.

8.4. Differential Privacy without External Noise

While most of the works are focusing on enforcing differential privacy with help of introducing noises into the dataset, Duan [9] worked on answering sum queries without any external noises. Duan pointed out that state-of-art differential privacy techniques is flawed for aggregate queries due to the zero-mean symmetric nature of Laplace distribution. He then argued that if the dataset is sufficiently large, aggregate queries would be private enough itself given that sum of multivariate Gaussian distributions converges when n is large as stated in central limit theorem.

Although aggregate queries are not the main focus of this project, Duan's work has opened a big gap in the field of differential privacy as a possible flaw has been discovered. The work also provides an insight of whether the original data is private enough once the dataset is large enough. If the dataset is, indeed, private enough, publishing an unsanitised anonymised would ensure the most accurate result for various data mining tasks.

8.5. Benchmark and Metrics

While all the above works are focusing on varies algorithms optimising in different scenarios, there are actually limited work on benchmarking algorithms across the table. Hay et. al. [6] suggested DPBench by considering performance of different data-dependent and data-independent algorithms under different scales and sizes.

The work from Hay et. al. adopted a very objective comparison scheme between different algorithms. It also addressed that as the algorithm will provide random noises, it would be biased if comparisons are drawn based on the shape of noise graph across different scale and attributes among different different algorithms. Such standard should also be upheld in this research should a similar comparison be adopted.

9. Approach

In this section, we will talk about the research questions addressed above in detail. We have looked into and coded AHP to address the first research question (Section 9.1). We have further suggested two novel mechanisms for addressing the second research question about dynamic privacy budget allocation (Section 9.2). In order to evaluate the novel mechanisms, we will be using two evaluation metrics (Section 9.3) namely KLD and MSE and six different datasets (Section 9.3) from which three of them will be real, and the rest will be synthetic.

9.1. AHP and Its Limitations

With AHP, Laplace noise with the same scale are applied to every bin [1]. That is, it is not possible to dynamically allocate privacy for each bin. The masked count will then go through the threshold function. This would restrict all the cluster to be in the band of valid value.

However, as stated in earlier sections, this will result in a big cluster at the lower threshold. In order to solve this problem, the Laplace noise applied to each bin would need to be relative to its count such that $\Pr\left(h_i + Lap(1/\epsilon_i) < lower\ threshold\right)$ and $\Pr\left(h_i + Lap(1/\epsilon_i) > upper\ threshold\right)$ can be minimised or reduced.

The optimal way is to allocate dynamic privacy budget such that it would always be related to count of each bin. This however is not sufficient to prove ϵ-differentially privacy. In order to align with ϵ-differentially privacy, a possible way would be to discretise the privacy budget ϵ into decimal points (e.g. 0.001). Using weak composition to obtain all the possible combinations of numbers that sums up to the total privacy budget ϵ, an optimal solution can then be found. This would however only provide an estimate of the optimal solution as we are handling a discretised solution set. While a more accurate solution can be come up by discretising the privacy budget into smaller sections (e.g. 0.00001), it would also make the algorithm an exponentially complex problem.

9.2. Dynamic Privacy Budget Allocation

9.2.1. Simple Dynamic Privacy Budget Allocation on AHP (SDPA-AHP)

To tackle the problems mentioned above, we have developed a novel dynamic privacy budget allocation function which depends on only the sorted rank of the bins instead of the counts of each bin. Such a function would allocate privacy budget linearly according to the sorted rank of a bin. The function will be applied on top of AHP to form a new mechanism SDPA-AHP.

Definition 5. When applying dynamic privacy on histogram $\mathbf{H}$, a dynamic privacy budget allocation function is defined as the following:

$$f(i,n) = \frac{n-i}{n \times \frac{n+1}{2}} \qquad for\ 0 \le i \le n-1$$

where n is the number of bins, i as the index of the bin.

To illustrate how the function would allocate budget, we assume an example with 5 bins and a budget $\epsilon = 0.05$ to be spent on Laplace noise masking. With AHP, each bin would get an $\epsilon_i = 0.05$ for $0 \le i \le 4$. This would result in an average $\bar{\epsilon} = \epsilon = 0.05$. With the novel function, we will obtain the following $f(i,n)$

$$f(i,n) = \begin{cases} \frac{2\times5}{5\times6} = 0.3333 & for\ i = 0 \\ \frac{2\times4}{5\times6} = 0.2667 & for\ i = 1 \\ \frac{2\times3}{5\times6} = 0.2 & for\ i = 2 \\ \frac{2\times2}{5\times6} = 0.1333 & for\ i = 3 \\ \frac{2\times1}{5\times6} = 0.0667 & for\ i = 4 \end{cases}$$

We would then calculate $\epsilon_i = f(i,n) \times n \times \epsilon$ such that we can obtain $\bar{\epsilon} = \epsilon = 0.05$. As an result, we will obtain a series of ϵ_i as shown in the Figure 1. Laplace noise with scale $1/\epsilon_i$ will then be applied onto each bin.

As a result, we can conclude an algorithm of the following:

1. $f = \emptyset$
2. $f[0] = \frac{n-i}{n\times\frac{n+1}{2}}$
3. $i = 1$
4. while $i < n$ do
5. $\quad f[i] = f[i-1] - \frac{1}{n\times\frac{n+1}{2}}$

Figure 2 shows the privacy budget allocated to each bin under AHP and SDPA-AHP. Figure 3 shows the scale of Laplace Noise added to each bin under AHP and SDPA-AHP. From the two figures, it is shown that the Simple Dynamic Privacy Budget Allocation is able to apply noises of smaller scales to bins with smaller counts, while applying noises with larger scale to bins with higher counts as a compensation by applying different privacy budget to different bins.

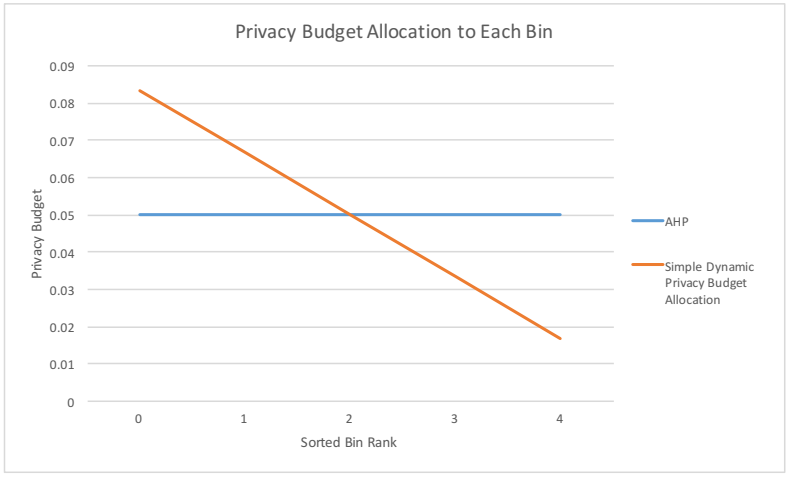


Figure 2 Simple Privacy Budget Allocation 1

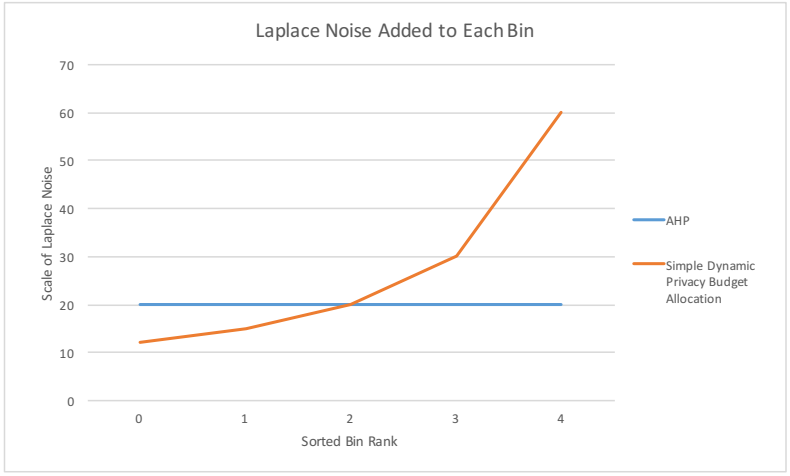


Figure 3 Simple Privacy Budget Allocation 2

Theorem 2. The Simple Dynamic Privacy Budget Allocation mechanism is ϵ-differentially private.

Proof.

Assumes we have two neighbour histograms $\mathbf{H}_1, \mathbf{H}_2$. $\Pr(\mathcal{A}(\mathbf{H}_1) = \alpha)$ can be calculated as follows

$$\begin{aligned} \Pr(\mathcal{A}(\boldsymbol{H}_1) = \alpha) &= \prod_{i=0}^{n-1} \Pr\left(h_{1,i} + Lap\left(\frac{1}{\epsilon_i}\right) = \alpha_i\right) \quad \text{(Laplace noise of different scale applied to different bins)} \\ &= \prod_{i=0}^{n-1} \frac{\epsilon_i}{2} \exp\left(-\left|\alpha_i - h_{1,i}\right|\epsilon_i\right) \\ &= \prod_{i=0}^{n-1} \frac{1}{2} f(i,n) \times n \times \epsilon \times \exp\left(-\left|\alpha_i - h_{1,i}\right|\epsilon_i\right) \\ &= \left(\frac{n\epsilon}{2}\right)^n \exp\left(\sum_{i=0}^{n-1} -\left|\alpha_i - h_{1,i}\right|\epsilon_i\right) \times \prod_{i=0}^{n-1} f(i,n) \\ &= \left(\frac{n\epsilon}{2}\right)^n \exp\left(n \times \epsilon \times \sum_{i=0}^{n-1} -\left|\alpha_i - h_{1,i}\right| f(i,n)\right) \times \prod_{i=0}^{n-1} f(i,n) \end{aligned}$$

Repeat the process, and we will obtain the following for $\mathbf{H}_2$

$$\Pr(\mathcal{A}(\boldsymbol{H}_2) = \alpha) = \prod_{i=0}^{n-1} \Pr\left(h_{2,i} + Lap\left(\frac{1}{\epsilon_i}\right) = \alpha_i\right) \quad \text{(Laplace noise of different scale is applied to different bins)}$$
$$= \prod_{i=0}^{n-1} \frac{\epsilon_i}{2} \exp\left(-|\alpha_i - h_{2,i}|\epsilon_i\right)$$
$$= \prod_{i=0}^{n-1} \frac{1}{2} f(i,n) \times n \times \epsilon \times \exp\left(-|\alpha_i - h_{1,i}|\epsilon_i\right)$$
$$= \left(\frac{n\epsilon}{2}\right)^n \exp\left(\sum_{i=0}^{n-1} -|\alpha_i - h_{2,i}|\epsilon_i\right) \times \prod_{i=0}^{n-1} f(i,n)$$
$$= \left(\frac{n\epsilon}{2}\right)^n \exp\left(n \times \epsilon \times \sum_{i=0}^{n-1} -|\alpha_i - h_{2,i}|f(i,n)\right) \times \prod_{i=0}^{n-1} f(i,n)$$

$$\frac{\Pr(\mathcal{A}(\boldsymbol{H}_1)=\alpha)}{\Pr(\mathcal{A}(\boldsymbol{H}_2)=\alpha)} = \exp\left(\epsilon \times \sum_{i=0}^{n-1} -|\alpha_i - h_{1,i}|f(i,n)\right) / \exp\left(\epsilon \times \sum_{i=0}^{n-1} -|\alpha_i - h_{2,i}|f(i,n)\right)$$

$$= \exp\left(\epsilon \times \sum_{i=0}^{n-1} f(i,n)\left(|\alpha_i - h_{2,i}| - |\alpha_i - h_{1,i}|\right)\right)$$

From the definition of neighbour histograms, we know that there will be one bin with count difference of exactly one. Hence, we have

$$\frac{\Pr(\mathcal{A}(\boldsymbol{H}_1)=\alpha)}{\Pr(\mathcal{A}(\boldsymbol{H}_2)=\alpha)} \leq \exp\left(\epsilon \times \sum_{i=0}^{n-1} f(i,n)\left(|h_{2,i} - h_{1,i}|\right)\right)$$
$$= \exp\left(\epsilon \times \left[f(0,n)\left(|\alpha_0 - h_{2,0}| - |\alpha_0 - h_{1,0}|\right) + f(1,n)\left(|\alpha_1 - h_{2,1}| - |\alpha_1 - h_{1,1}|\right) + \cdots + f(n-1,n)\left(|\alpha_{n-1} - h_{2,n-1}| - |\alpha_{n-1} - h_{1,n-1}|\right)\right]\right)$$

As $v(0,n,\delta)$ will always be larger than $v(i,n,\delta)$ with $0 < i < n$, we have

$$\frac{\Pr(\mathcal{A}(\boldsymbol{H}_1)=\alpha)}{\Pr(\mathcal{A}(\boldsymbol{H}_2)=\alpha)} \leq \exp\left(\epsilon \times f(0,n)\right)$$
$$\leq \exp(\epsilon) \qquad \text{as } f(0,n) > 0 \text{ for all } 0 \leq i < n.$$

And thus, from Definition 1, we have proved that this mechanism is of ϵ-differentially private.

9.2.2. Flexible Dynamic Privacy Budget Allocation on AHP (DPA-AHP)

In the previous section, Simple Dynamic Privacy Budget Allocation is introduced. However, from the graph, we can observe that reducing scale of noise at bins with smaller counts would lead to an exponential increase in scale for the bins with higher count. This would make the function applicable only when the histogram of the dataset shows the same trend after sorting.

In order to make the function more general and flexible, we have come up with the *Flexible Dynamic Privacy Budget Allocation (DPA)* which users can specify how diverse the privacy budget should be allocated. This would be specified with a variable δ.

Definition 6. When applying flexible dynamic privacy budget allocation on histogram H, the dynamic privacy budget allocation function consist of a weight function $v(i,n,\delta)$ and the allocation function $f(i,n,\delta)$:

$$v(i,n,\delta) = \left(\left\lfloor\frac{n}{2}\right\rfloor + \frac{n-2i-1}{2} \times \delta\right) \qquad for\ 0 \leq i \leq n-1$$

$$f(i,n,\delta) = \frac{v(i,n,\delta)}{\sum_{i=0}^{n-1} v(i,n,\delta)} \qquad for\ 0 \leq i \leq n-1$$

where n is the number of bins, i as the index of the bin, and δ (step) as decrement of the portion of differential privacy budget allocated at the bin i.

To illustrate how the function would allocate budget, we again assume an example with 5 bins and a budget $\epsilon = 0.05$ to be spent on Laplace noise masking. With AHP, each bin would get an $\epsilon_i = 0.05$ for $0 \leq i \leq 4$. This would result in an average $\bar{\epsilon} = \epsilon = 0.05$. With the novel function, we will obtain the following $f(i, n, \delta)$

$$f(i,n,0) = \begin{cases} 0.2 & for\ i = 0 \\ 0.2 & for\ i = 1 \\ 0.2 & for\ i = 2 \\ 0.2 & for\ i = 3 \\ 0.2 & for\ i = 4 \end{cases} \quad f(i,n,0.5) = \begin{cases} 0.2667 & for\ i = 0 \\ 0.2333 & for\ i = 1 \\ 0.2000 & for\ i = 2 \\ 0.1667 & for\ i = 3 \\ 0.1333 & for\ i = 4 \end{cases} \quad f(i,n,1) = \begin{cases} 0.3333 & for\ i = 0 \\ 0.2667 & for\ i = 1 \\ 0.2000 & for\ i = 2 \\ 0.1333 & for\ i = 3 \\ 0.0667 & for\ i = 4 \end{cases}$$

We would then calculate $\epsilon_i = f(i, n, \delta) \times n \times \epsilon$ such that we can obtain $\bar{\epsilon} = \epsilon = 0.05$. As an result, we will obtain a series of ϵ_i as shown in the Figure 3. Laplace noise with scale $1/\epsilon_i$ will then be applied onto each bin.

As a result, we can conclude an algorithm of the following:

1. $f = \emptyset$
2. $v = \emptyset$
3. $v[0] = \left(\left\lceil \frac{n}{2} \right\rceil + \frac{n-1}{2} \times \delta\right)$
4. $i = 1$
5. while $i < n$ do
6. $\quad v[i] = v[i-1] - \delta$
7. $sum = \sum_i v[i]$
8. $i = 0$
9. while $i < n$ do
10. $\quad f[i] = v[i]/sum$

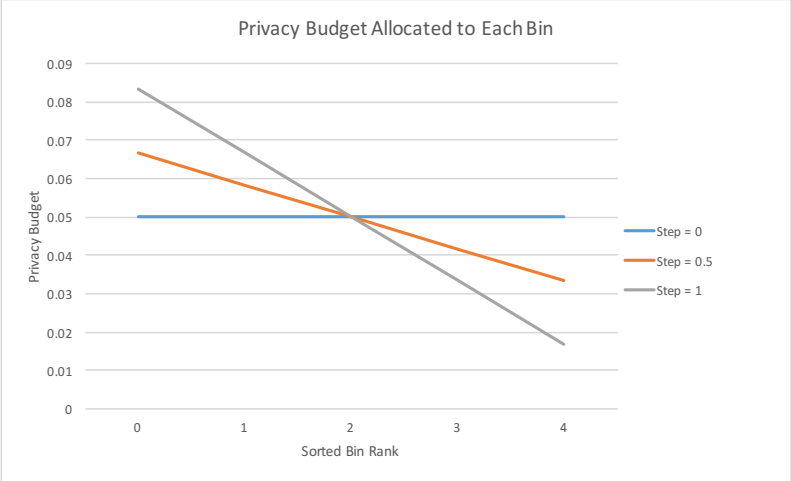


Figure 4 Flexible Dynamic Privacy Budge Allocation 1

As an example, Figure 4 shows the privacy budget allocated to each bin in a sorted histogram with 5 bins. As from Figure 4, we can observe that AHP and the SDPA-AHP are essentially special cases of the DPA-AHP where step is 0 or 1 respectively. With the introduction of δ, we can now adjust the privacy budget to suit sorted histograms with different distribution. Figure 5 shows the scale of Laplace noise added to each bin of a 5-bin sorted histogram under different δ. It can be observed that as δ decreases, the scale of noise across bins will be more linear than exponential.

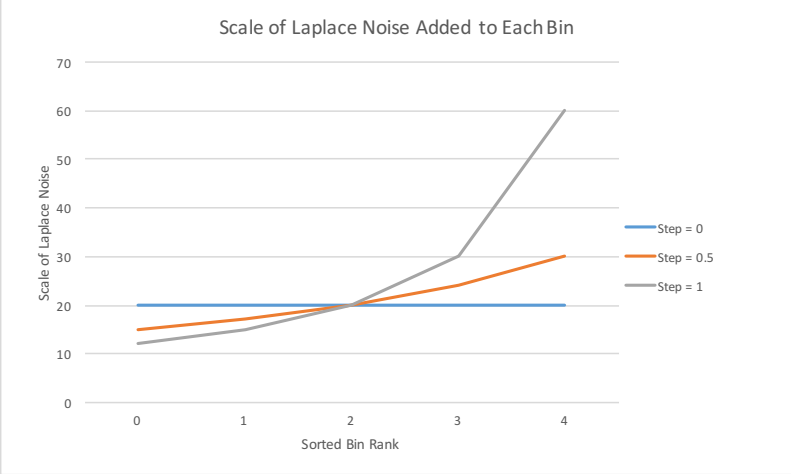


Figure 5 Flexible Dynamic Privacy Budge Allocation 2

Now that the function can replicate the performance of AHP ($\delta = 0$), and provide the flexibility of dynamically allocating privacy budget, it is also very important to prove that adopting such a function will not violate the principle of ϵ-differentially privacy.

Theorem 3. The Flexible Dynamic Privacy Budget Allocation mechanism is ϵ-differentially private.

Proof.

Assumes we have two neighbour histograms $\mathbf{H}_1$, $\mathbf{H}_2$. $\Pr(\mathcal{A}(\mathbf{H}_1) = \alpha)$ can be calculated as follows

$$\begin{aligned}
\Pr(\mathcal{A}(\boldsymbol{H}_1) = \alpha) &= \prod_{i=0}^{n-1} \Pr\left(h_{1,i} + Lap\left(\frac{1}{\epsilon_i}\right) = \alpha_i\right) \quad \text{(Laplace noise of different scale applied to different bins)} \\
&= \prod_{i=0}^{n-1} \frac{\epsilon_i}{2} \exp\left(-|\alpha_i - h_{1,i}|\epsilon_i\right) \\
&= \prod_{i=0}^{n-1} \frac{1}{2} \frac{v(i,n,\delta) \times n \times \epsilon}{\sum_{i=0}^{n-1} v(i,n,\delta)} \exp\left(-|\alpha_i - h_{1,i}|\epsilon_i\right) \\
&= \left(\frac{n\epsilon}{2}\right)^n \exp\left(\sum_{i=0}^{n-1} -|\alpha_i - h_{1,i}|\epsilon_i\right) \times \prod_{i=0}^{n-1} f(i,n,\delta) \\
&= \left(\frac{n\epsilon}{2}\right)^n \exp\left(\frac{\epsilon}{\sum_{i=0}^{n-1} v(i,n,\delta)} \sum_{i=0}^{n-1} -|\alpha_i - h_{1,i}|v(i,n,\delta)\right) \times \prod_{i=0}^{n-1} f(i,n,\delta)
\end{aligned}$$

Repeat the process, and we will obtain the following for $\mathbf{H}_2$

$$\begin{aligned}
\Pr(\mathcal{A}(\boldsymbol{H}_2) = \alpha) &= \prod_{i=0}^{n-1} \Pr\left(h_{2,i} + Lap\left(\frac{1}{\epsilon_i}\right) = \alpha_i\right) \quad \text{(Laplace noise of different scale is applied to different bins)} \\
&= \prod_{i=0}^{n-1} \frac{\epsilon_i}{2} \exp\left(-|\alpha_i - h_{2,i}|\epsilon_i\right) \\
&= \prod_{i=0}^{n-1} \frac{1}{2} \frac{v(i,n,\delta) \times n \times \epsilon}{\sum_{i=0}^{n-1} v(i,n,\delta)} \exp\left(-|\alpha_i - h_{1,i}|\epsilon_i\right) \\
&= \left(\frac{n\epsilon}{2}\right)^n \exp\left(\sum_{i=0}^{n-1} -|\alpha_i - h_{2,i}|\epsilon_i\right) \times \prod_{i=0}^{n-1} f(i,n,\delta) \\
&= \left(\frac{n\epsilon}{2}\right)^n \exp\left(\frac{\epsilon}{\sum_{i=0}^{n-1} v(i,n,\delta)} \sum_{i=0}^{n-1} -|\alpha_i - h_{2,i}|v(i,n,\delta)\right) \times \prod_{i=0}^{n-1} f(i,n,\delta)
\end{aligned}$$

$$\begin{aligned}
\frac{\Pr(\mathcal{A}(\boldsymbol{H}_1)=\alpha)}{\Pr(\mathcal{A}(\boldsymbol{H}_2)=\alpha)} &= \exp\left(\frac{\epsilon}{\sum_{i=0}^{n-1} v(i,n,\delta)} \sum_{i=0}^{n-1} -|\alpha_i - h_{1,i}|v(i,n,\delta)\right) / \exp\left(\frac{\epsilon}{\sum_{i=0}^{n-1} v(i,n,\delta)} \sum_{i=0}^{n-1} -|\alpha_i - h_{2,i}|v(i,n,\delta)\right) \\
&= \exp\left(\frac{\epsilon}{\sum_{i=0}^{n-1} v(i,n,\delta)} \sum_{i=0}^{n-1} v(i,n,\delta)\left(|\alpha_i - h_{2,i}| - |\alpha_i - h_{1,i}|\right)\right)
\end{aligned}$$

From the definition of neighbour histograms, we know that there will be one bin with count difference of exactly one. Hence, we have

$$\begin{aligned}
\frac{\Pr(\mathcal{A}(\boldsymbol{H}_1)=\alpha)}{\Pr(\mathcal{A}(\boldsymbol{H}_2)=\alpha)} &\leq \exp\left(\frac{\epsilon}{\sum_{i=0}^{n-1} v(i,n,\delta)} \sum_{i=0}^{n-1} v(i,n,\delta)\left(|h_{2,i} - h_{1,i}|\right)\right) \\
&= \exp\left(\frac{\epsilon}{\sum_{i=0}^{n-1} v(i,n,\delta)} \left[v(0,n,\delta)\left(|\alpha_0 - h_{2,0}| - |\alpha_0 - h_{1,0}|\right) + v(1,n,\delta)\left(|\alpha_1 - h_{2,1}| - |\alpha_1 - h_{1,1}|\right) + \right.\right. \\
&\left.\left. \cdots + v(n-1,n,\delta)\left(|\alpha_{n-1} - h_{2,n-1}| - |\alpha_{n-1} - h_{1,n-1}|\right)\right]\right)
\end{aligned}$$

As $v(0,n,\delta)$ will always be larger than $v(i,n,\delta)$ with $0 < i < n$, we have

$$\begin{aligned}
\frac{\Pr(\mathcal{A}(\boldsymbol{H}_1)=\alpha)}{\Pr(\mathcal{A}(\boldsymbol{H}_2)=\alpha)} &\leq \exp\left(\frac{\epsilon}{\sum_{i=0}^{n-1} v(i,n,\delta)} v(0,n,\delta)\right) \\
&\leq \exp(\epsilon) \qquad \text{as } v(i,n,\delta) > 0 \text{ for all } 0 \leq i < n.
\end{aligned}$$

And thus, from Definition 1, we have proved that this mechanism is of ϵ-differentially private.

9.3. Evaluation Metrics

For evaluation, we would use Kullback-Leibler divergence (KLD) to measure the difference in data distribution between the original and the sanitised histograms. We would also use Mean Squared Error (MSE) to evaluate accuracy of range queries.

Definition 7. Denote histogram $\boldsymbol{H}$ and its sanitised histogram $\widetilde{\boldsymbol{H}}$. The Kullback-Leibler divergence (KLD) can be calculated as follows:

$$KLD(\boldsymbol{H}, \widetilde{\boldsymbol{H}}) = \sum_{i=1}^{n} H_i \ln \frac{H_i}{\widetilde{H_i}}$$

where H_i represents the proportion of bin h_i is histogram $\boldsymbol{H}$. The formula follows the convention that $0 \ln 0 = 0$.

If $\boldsymbol{H}$ and $\widetilde{\boldsymbol{H}}$ follow the same distribution, we have:

$$KLD\left(\boldsymbol{H}, \widetilde{\boldsymbol{H}}\right) = 0$$

Definition 8.[1] Denote histogram $\boldsymbol{H}$ and its sanitised histogram $\widetilde{\boldsymbol{H}}$. The Mean Squared Error (MSE) of a set of range queries $\boldsymbol{Q} = \{Q_1, Q_2, \ldots, Q_m\}$ can be calculated as follows:

$$MSE(\boldsymbol{H}, \widetilde{\boldsymbol{H}}, \boldsymbol{Q}) = \frac{\sum_{i=1}^{m} \left(Q_i(\boldsymbol{H}) - Q_i\left(\widetilde{\boldsymbol{H}}\right)\right)^2}{m}$$

where $Q_i(\boldsymbol{H})$ and $Q_i\left(\widetilde{\boldsymbol{H}}\right)$ return the answers of Q_i on $\boldsymbol{H}$ and $\widetilde{\boldsymbol{H}}$ respectively.

9.4. Datasets

The TSMC_NYC dataset is available on http://www-public.it-sudparis.eu/~zhang_da/pub/dataset_tsmc2014.zip. This dataset includes check-in data of restaurant venues in New York City from Foursqare from April 2012 to February 2013. There are 227428 check-ins in this dataset. Only User ID and Venue ID in dataset_TSMC2014_NYC.txt are used in this paper.

The TSMC_TKY dataset is available on http://www-public.it-sudparis.eu/~zhang_da/pub/dataset_tsmc2014.zip. This dataset includes check-in data of restaurant venues in Tokyo from Foursqare from April 2012 to February 2013. There are 537703 check-ins in this dataset. Only User ID and Venue ID in dataset_TSMC2014_TKY.txt are used in this paper.

The UBICOMP dataset is available on http://www-public.it-sudparis.eu/~zhang_da/pub/dataset_ubicomp2013.zip. This dataset includes check-in data of restaurant venues in Tokyo from Foursqare from October 2011 to February 2012. There are 3112 users, 3298 venues with 27149 check-ins in this dataset. Only User ID and Venue ID in dataset_ubicomp2013_checkins.txt are used in this paper.

The CONPOLBLOGS dataset is available on http://www-personal.umich.edu/~mejn/. This set represent a graph of weblogs on politics with edges (u; u0) as hyperlinks from node u to node u0.

The LOGNORMAL dataset is a synthetic dataset which contains 65 synthetic users. Their counts are generated by a lognormal distribution with parameter as a random value from 0 to 5 exclusively.

The EXPONENTIAL dataset is a synthetic dataset which contains 51 synthetic users. Their counts are generated by an exponential distribution with parameter as a random value from 0 to 10 exclusively.

10. Results

In this section, only DPA-AHP will be compared with AHP as the SDPA-AHP is just a special case of the more the flexible one with $\delta = 1$. To conform with AHP, we will look into two aspects when measuring utility and errors, namely KLD for data distribution, and MSE for range queries. On top of that, we would also look into the run time complexity of the novel mechanism.

All the code is written in Python 3, and the testing environment is on a macOS Sierra terminal with 2.2 GHz Intel Core i7 CPU, and 16 GB 1600 MHz DDR3 RAM.

Characteristics of the testing datasets, analysis of KLD metrics, MSE metrics, and run time will be included in Section 5.1, 5.2, 5.3 and 5.4 respectively.

10.1. Testing Datasets

In the followings sections, we will apply the test onto 6 different datasets, each with a different shape of distribution.

Dataset Name		Rows	Histograms	Average Count in Histogram	Variance in Counts	Range of Count
TSMC_NYC	Real	227428	1083	209.998	35486.037	[100, 2697]
TSMC_TKY	Real	57303	2293	250.198	49072.056	[100, 2991]
UBICOMP	Real	27149	2060	13.179	296.540	[1, 208]
CONPOLBLOGS	Real	14409	1	14408	N/A	[14408, 14408]
LOGNORMAL	Synthetic	3544	66	53.697	953.090	[2, 99]
EXPONENTIAL	Synthetic	25929	51	498.635	65447.501	[59, 996]

Table 2 Characteristics of Testing Datasets

10.2. Data Distribution

In this section, we attempt to test the ability of the Flexible Dynamic Privacy Allocation to preserve data distribution of different testing datasets with different δ. Table 3 to 5 shows the KLD of the testing datasets under different privacy budgets (0.01, 0.1, and 1) and different δ (0, 0.025, 0.05, 0.075, 0.1, and 0.15). As from the definition of the Flexible Dynamic Privacy Allocation, the algorithm will be exactly AHP when $\delta = 0$. We can observe that the new mechanism provides an improvement of around 10% when δ is between 0.05 to 0.1 depend on both ϵ and the dataset. Based on the experiment data, a $\delta = 0.075$ is suggested for improving the preservation of data distribution while enforcing differential privacy.

Dataset	$\delta = 0$ (AHP)	$\delta = 0.025$	$\delta = 0.05$	$\delta = 0.075$	$\delta = 0.1$	$\delta = 0.15$
TSMC_NYC	8.625	8.484	7.934	8.588	8.620	8.606
TSMC_TKY	9.252	9.082	8.861	8.346	8.768	9.009
UBICOMP	0.711	0.666	0.643	0.604	0.703	0.737
CONPOLBLOGS	7.787	7.630	7.565	6.582	7.184	7.528
LOGNORMAL	5.012	4.895	4.526	4.454	4.829	4.843
EXPONENTIAL	11.783	11.597	11.449	11.364	10.666	11.356

Table 3 KLD results for $\epsilon = 0.01$

Dataset	$\delta = 0$ (AHP)	$\delta = 0.025$	$\delta = 0.05$	$\delta = 0.075$	$\delta = 0.1$	$\delta = 0.15$
TSMC_NYC	8.602	8.567	7.937	7.879	8.490	8.654
TSMC_TKY	8.813	8.719	8.464	8.034	8.511	8.594
UBICOMP	0.698	0.681	0.622	0.708	0.734	0.747
CONPOLBLOGS	6.429	6.303	6.201	6.304	6.429	6.604
LOGNORMAL	5.038	4.973	4.573	4.671	4.841	4.883
EXPONENTIAL	7.504	7.325	6.928	7.208	7.293	7.292

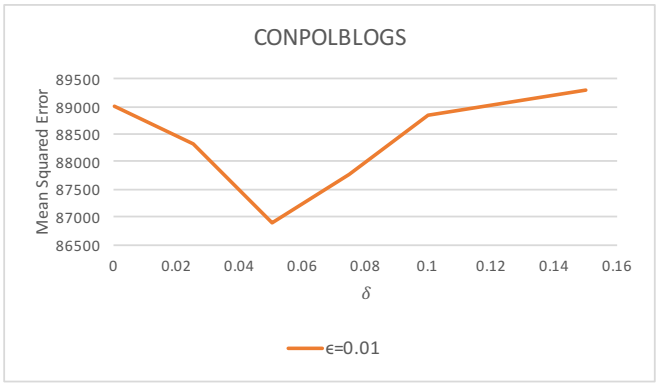
Table 4 KLD results for $\epsilon = 0.1$

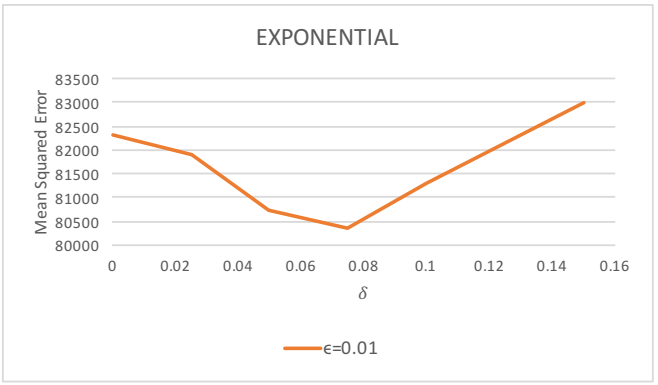
Dataset	$\delta = 0$ (AHP)	$\delta = 0.025$	$\delta = 0.05$	$\delta = 0.075$	$\delta = 0.1$	$\delta = 0.15$
TSMC_NYC	2.782	2.694	2.560	2.582	2.689	2.757
TSMC_TKY	2.831	2.808	2.655	2.658	2.705	2.794
UBICOMP	1.046	1.046	1.029	0.958	0.978	1.01
CONPOLBLOGS	1.220	1.186	1.196	1.221	1.221	1.270
LOGNORMAL	2.994	2.940	2.740	2.617	2.677	2.898
EXPONENTIAL	0.904	0.837	0.826	0.822	0.842	0.851

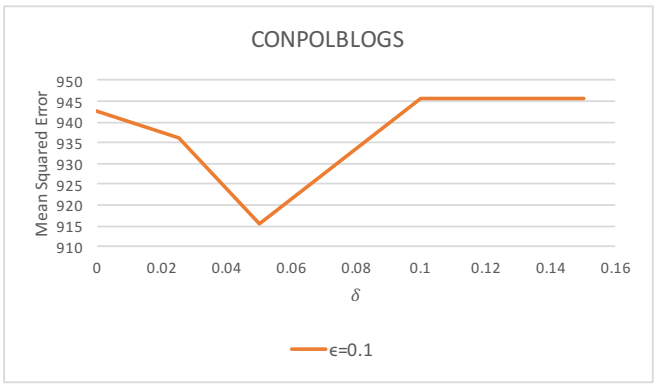
Table 5 KLD results for $\epsilon = 1$

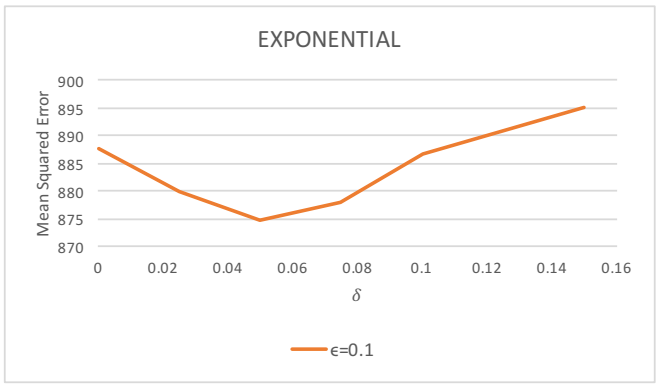
10.3. Range Queries

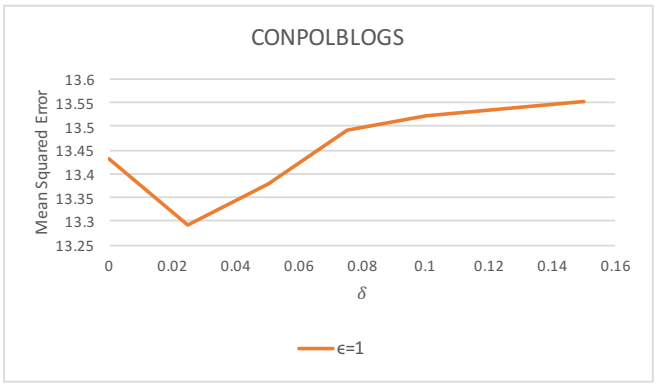
For range queries, we apply the mechanism onto the testing datasets for 30 times and use the average MSE for comparisons. Figure 6 to 10 show the trend of MSE for different testing datasets under DPA-AHP with various δ and various ϵ. In the experiments, we have taken $\epsilon \in \{0.01, 0.1, 1\}$ and $\delta \in \{0.025, 0.05, 0.075, 0.1, 0.15\}$. From the series of plots, we can identify that MSE reduces as ϵ increases from 0.01 to 1 which aligns with the differential privacy policy as smaller budget implies larger scale of Laplace noise being applied. It can also be observed that MSE decreases as δ increases towards the optimal, then increases exponentially as δ increases. As for the testing datasets, we can observe that the optimal δ is usually around 0.05 and 0.075. Hence, we suggest $\delta = 0.05$ or $\delta = 0.075$ to be adopted to obtain the best result.











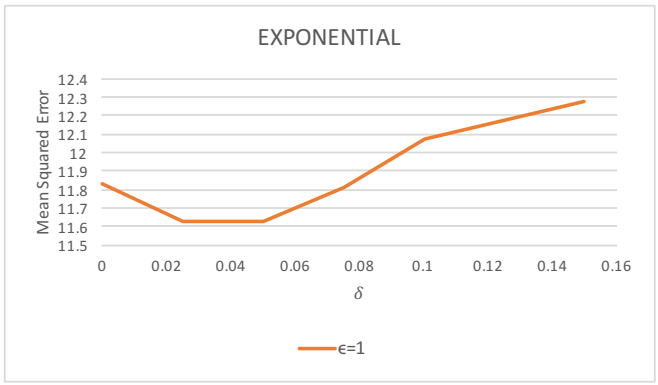
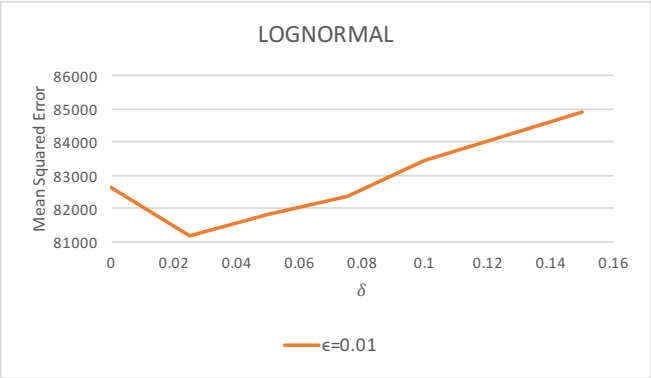
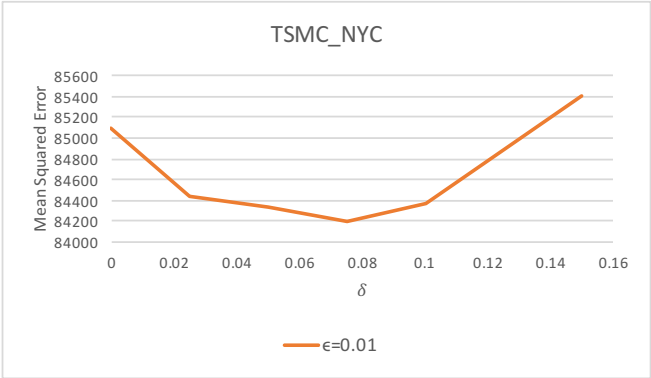
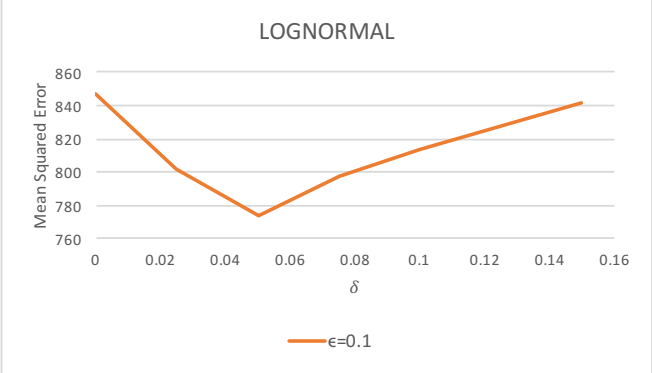


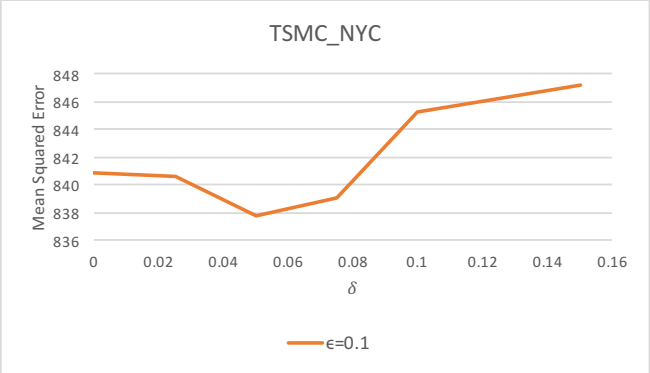
Figure 6 Mean Squared Error for CONPOLBLOGS under Different Delta

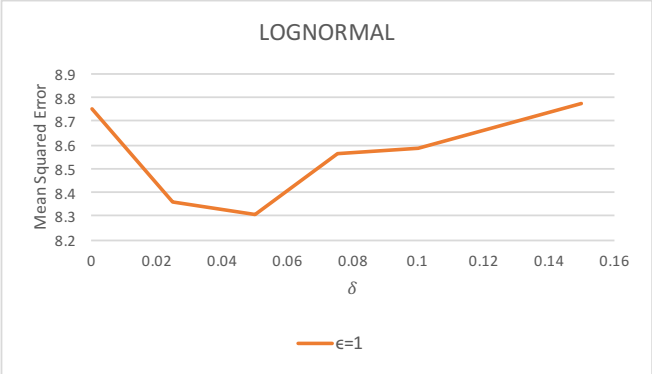
Figure 7 Mean Squared Error for EXPONENTIAL under Different Delta











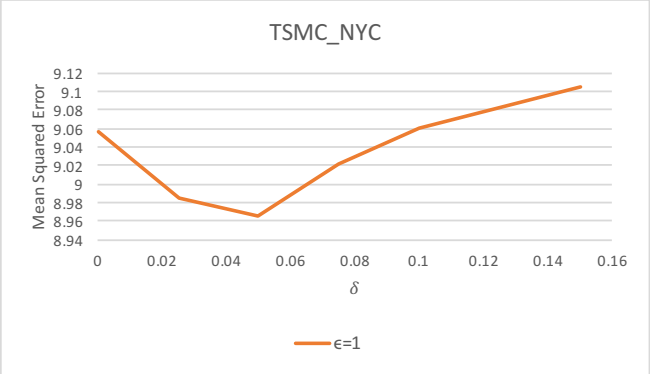
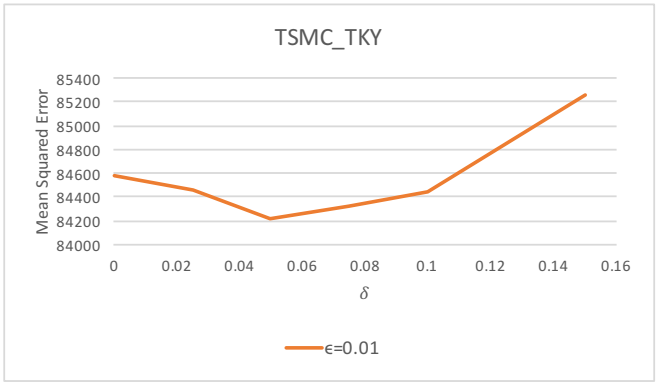
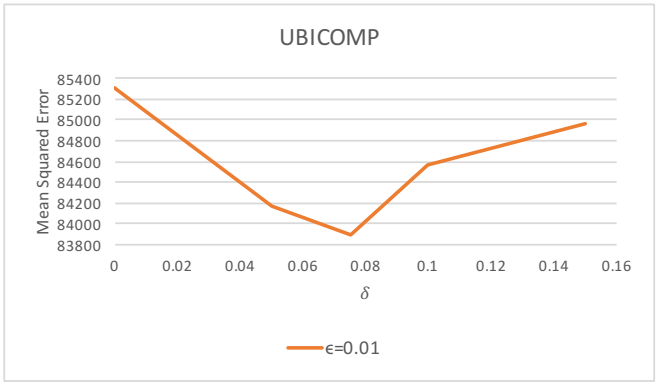
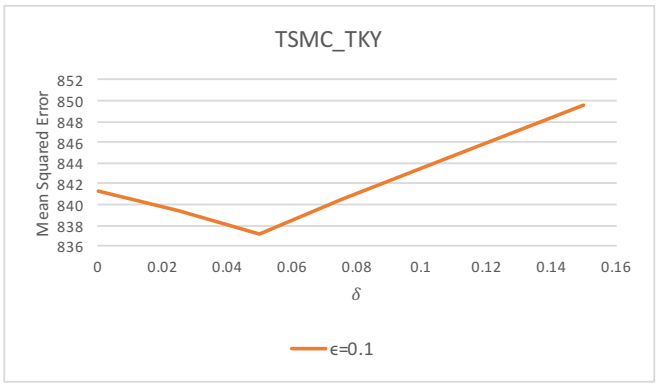


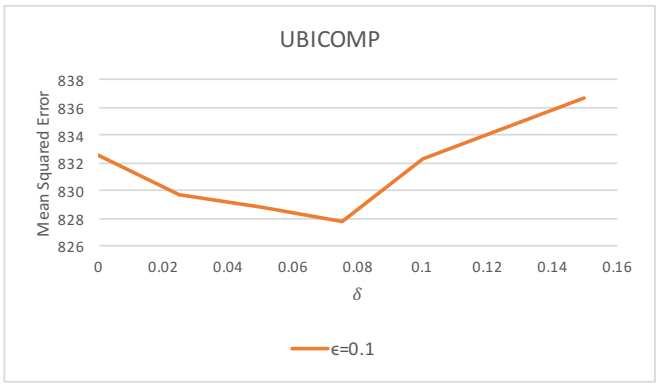
Figure 8 Mean Squared Error for LOGNORMAL under Different Delta

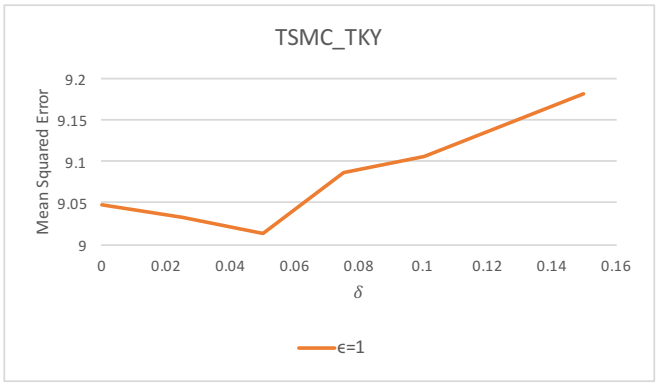
Figure 9 Mean Squared Error for TSMC_NYC under Different Delta











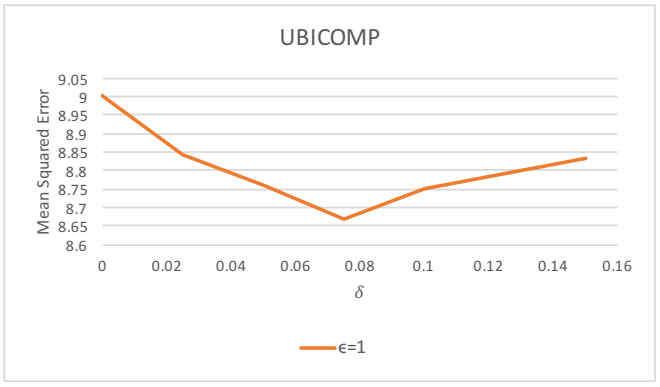


Figure 10 Mean Squared Error for TSMC_TKY under Different Delta

Figure 11 Mean Squared Error for TSMC_TKY under Different Delta

10.4. Time Spent on DPA in DPA-AHP

For time complexity, we apply the mechanism onto the datasets for 30 times and use the average portion of CPU time spent on allocating privacy budget for comparisons. Figure 11 shows the result we have obtained with six different datasets. It can be observed the algorithm responsible for the DPA mechanism take up less than 20% of the total CPU time when generating sanitised histograms with DPA-AHP. On top of that, the mechanism also conforms with the the $O(n)$ time complexity of AHP where n refers to the number of bins.

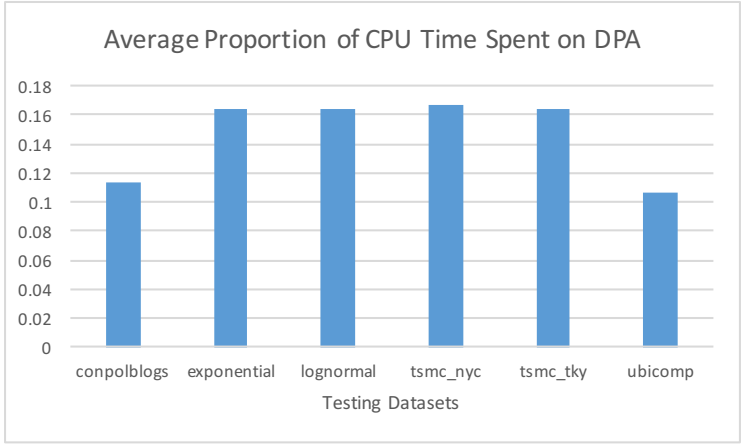


Figure 12 Proportion of Time Spent on DPA in DPA-AHP

11. Conclusion

In this thesis, we have looked into the field of enforcing differential privacy in histograms. We have reviewed the current state-of-art algorithm AHP and discovered a potential problem of having a big cluster at the lower threshold as a result of applying equal amount of privacy budget to every bin. To address this issue, we have suggested two novel differential privacy budget allocation mechanism to apply on AHP to form SDPA-AHP and DPA-AHP. In fact, both AHP and SDPA-AHP are special cases under DPA-AHP with $\delta = 0$ and $\delta = 1$ respectively.

From the experiments, the new DPA-AHP has provided a 10% improvement on data distribution preservation based on the KLD from the six testing datasets. We can also observe a better MSE when δ is between 0 and 0.1. The optimal solutions for both metrics are pointing at $\delta = 0.05$ and $\delta = 0.075$. Hence, the two values are suggested for obtaining a better result than AHP. The mechanism will be taking around 20% of the total CPU running time on the testing terminal and will align with the $O(n)$ time complexity of AHP where n refers to the number of bins in the histogram. Most importantly, the mechanism has been proved to be a ϵ-differentially private algorithm.

11.1. Future Research

In this paper, we have found the Flexible Dynamic Privacy Budget Allocation mechanism. We have also found the general trend when adopting this mechanism. Based on all the testing and experiments, we suggest $\delta = 0.05$ or $\delta = 0.075$ to be adopted. In order to extend this research, utility function for finding the optimal δ can be looked into so that users can use the best δ instead of the suggest level of 0.05 or 0.075 here in this thesis.

Another possible direction would be to look into allocating privacy budget in a none linear manner, for instance, privacy budget will decrement exponentially as the sorted rank increases. This would make the algorithm more generic and complete. With these features, it would be possible that the algorithm can generate sanitised histogram based on the input dataset without needing users to specify how privacy budget should be allocated to each bin.

11.2. Lesson Learnt

It has been a very back and forth process when deciding the direction of the thesis. Originally, the thesis is going to be around threshold function. When looking into the threshold function, the problem of big clusters around lower threshold is discovered. If the project was to be started again, I would probably suggest myself to look into broader areas before deciding to look into threshold function as it has cost us a few months to get back on the correct track.

Another aspect to be improved is on the testing and experiments period of the project. As the acquired datasets requires days to be completely processed by the algorithm due to sheer size, it has seriously dragged the progress on evaluating and improving the mechanism. A better way of doing the testing and experiments will be to sample from the huge datasets such that an objective picture can be obtained while minimising the total running time. Another viable strategy would be to run the initial experiments on smaller datasets first.

12. Bibliography

1. Xiaojian Zhang, Rui Chen, Jianliang Xu, Xiaofeng Meng, Yingtao Xie. "Towards Accurate Histogram Publication under Differential Privacy." *2014 SIAM International Conference on Data Mining*, 2014: 9.
2. Rui Chen, Yilin Shen, Hongxia Jin. "Private Analysis of Infinite Data Streams via Retroactive Grouping." *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 2015: 1061-1070.
3. Rui Chen, Qian Xiao, Yu Zhang, Jianliang Xu. "Differentially Private High-Dimensional Data Publication via Sampling-Based Inference." *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015: 129-138.
4. Weiyen Day, Ninghui Li, Min Lyu. "Publishing Graph Degree Distribution with Node Differential Privacy." *Proceedings of the 2016 International Conference on Management of Data*, 2016: 123-138.
5. Rui Chen, Benjamin C. M. Fung, Li Xiong. "Publishing Set-Valued Data via Differential Privacy." *Proceedings of the 37th International Conference on Very Large Data Bases*, 2011: 1087.
6. Michael Hay, Ashwin Machanavajjhala, Gerome Miklau, Yan Chen, Dan Zhang. "Principled Evaluation of Differentially Private Algorithms using DPBench." *Proceedings of the 2016 International Conference on Management of Data*, 2016: 139-154.
7. Christos Dimitrakakis, Blaine Nelson, Zuhe Zhang, Aikaterini Mitrokotsa, Benjamin I. P. Rubinstein. "Differential Privacy for Bayesian Inference through Posterior Sampling." *Journal of Machine Learning Research* 18, no. 11 (2017): 1-39.
8. Marco Gaboardi, Emilio Jesu´s Gallego Arias, Justin Hsu, Aaron Roth, Zhiwei Steven Wu. "Dual Query: Practical Private Query Release for High Dimensional Data." *Proceedings of the 31st International Conference on Machine Learning*, 2014: 1170-1178.
9. Duan, Yitao. "Differential Privacy for Sum Queries without External Noise." *Proceedings of the ACM Conference on Information and Knowledge Management*, 2009: 1517-1520.
10. Moritz Hardt, Katrina Ligett, Frank Mcsherry. "A Simple and Practical Algorithm for Differentially Private Data Release." *Advances in Neural Information Processing Systems 25*, 2012.
11. G. Acs, C. Castelluccia, R. Chen. "Differentially private histogram publishing through lossy compression." *Proceedings of ICDM*, 2012: 1-10.
12. Papadopoulos, G. Kellaris and S. "Practical differential privacy via grouping and smoothing." *Proceedings of VLDB Endow* 6, no. 5, 2013: 301-312.
13. J. Xu, Z. Zhang, X. Xiao, and G. Yu. "Differentially private histogram publicaiton." *Proceedings of ICDE*, 2012: 32-43.
14. Y. Ye, Y. Zheng, Y. Chen, J. Feng, and X. Xie. “Mining Individual Life Pattern Based on Location History.” *Mobile Data Management: Systems, Services and Middleware*, 2009.
15. F. Chovatiya, P. Prajapati, J. Vasa, and J. Patel. “A Research Direction on Data Mining with IOT.” *Information and Communication Technology for Intelligent Systems*, 2017.
16. J. Zhou, Z. Cao, X. Dong, A. V. Vasilakos. “Security and Privacy for Cloud-Based IoT: Challenges.” *IEEE Communications Magazine*, 2017: 26-33.
17. A. Pegoraro, O. Scott, and L. M. Burch. “Strategic Use of Facebook to Build Brand Awareness: A Case Study of Two National Sport Organisations.” *International Journal of Public Administration in the Digital Age*, 2017: 69-87.
18. V. Benndorf, and H.T. Normann. “The Willingness to Sell Personal Data.” *The Scandinavian Journal of Economics*, 2017.
19. D. Markovikj, S. Gievska, M. Kosinski, and D. Stillwell. “Mining Facebook Data for Predictive Personality Modelling.” *Proceedings of the 7th International AAAI Conference on Weblogs and Social Media*, 2013: 23-26.
20. J. G. Cabañas, Á. Cuevas, and R. Cuevas. “FDVT: Data Valuation Tool for Facebook Users.” *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 2017: 3799-3809.
21. K. D. Martin, A. Borah, and R. W. Palmatier. “Data Privacy: Effects on Customer and Firm Performance.” *Journal of Marketing*, 2017.
22. A. Serna, J. K. Gerrikagoitia, U. Bernabé, and T. Ruiz. “Sustainability Analysis on Urban Mobility Based on Social Media Content.” *Transportation Research Procedia*, no. 24, 2017: 1-8.
23. J. A. Gottfried. “The Changing Nature of Political Debate Consumption: Social Media, Multitasking, and Knowledge Acquisition.” *Political Communication*, 2016: 172-199.

24. C. E. Pierce, K. Bouri, C. Pamer, S. Proestel, H. W. Rodriguez, H. V. Le, C. C. Freifeld, J. S. Brownstein, M. Walderhaug, I. R. Edwards, and N. Dasgupta. “Evaluation of Facebook and Twitter Monitoring to Detect Safety Signals for Medical Products: An Analysis of Recent FDA Safety Alerts.” *Drug Safety*, no. 40, 2017: 317-331.