Information Theoretic-Based Privacy Protection on Data Publishing and Biometric Authentication

Chengfang Fang

(B.Comp. (Hons.), NUS)

A THESIS SUBMITTED

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

IN DEPARTMENT OF COMPUTER SCIENCE

NATIONAL UNIVERSITY OF SINGAPORE

2013

Declaration

I hereby declare that the thesis is my original work and it has been written by me in its entirety.

I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

Chengfang Fang

30 October 2013

Contents

Summary

We are interested in providing privacy protection for applications that involve sensitive personal data. In particular, we focus on controlling information leakages in two scenarios: data publishing and biometric authentication. In both scenarios, we seek privacy protection techniques that are based on information theoretic analysis, which provide unconditional guarantee on the amount of information leakage. The amount of leakage can be quantified by the increment in the probability that an adversary correctly determines the data.

We first look at scenarios where we want to publish datasets that contain useful but sensitive statistical information for public usage. To publish such information while preserving the privacy of individual contributors is technically challenging. The notion of differential privacy provides a privacy assurance regardless of the background information held by the adversaries. Many existing algorithms publish aggregated information of the dataset, which requires the publisher to have a-prior knowledge on the usage of the data. We propose a method that directly publish (a noisy version of) the whole dataset, to cater for the scenarios where the data can be used for different purposes. We show that the proposed method

can achieve high accuracy w.r.t. some common aggregate algorithms under their corresponding measurements, for example range query and order statistics.

To further improve the accuracy, several relaxations have been proposed to relax the definition on how the privacy assurance should be measured. We propose an alternative direction of relaxation, where we attempt to stay within the original measurement framework, but with a narrowed definition of datasets-neighbourhood. We consider two types of datasets: spatial datasets where the restriction is based on spatial distance among the contributors, and dynamically changing datasets, where the restriction is based on the duration an entity has contributed to the dataset. We proposed a few constructions that exploit the relaxed notion, and show that the utility can be significantly improved.

Different from data publishing, the challenge of privacy protection in biometric authentication scenario arises from the fuzziness of the biometric secrets, in the sense that there will be inevitable noises present in biometric samples. To handle such noises, a well-known framework *secure sketch* (DRS04) was proposed by Dodis et al. Secure sketch can restore the enrolled biometric sample, from a "close" sample and some additional helper information computed from the enrolled sample. The framework also provides tools to quantify the information leakage of the biometric secret from the helper information. However, the original notion of secure sketch may not be directly applicable in practise. Our goal is to extend and improve the constructions under various scenarios motivated by real-

life applications.

We consider an asymmetric setting, whereby multiple biometric samples are acquired during enrollment phase, but only a single sample is required during verification. From the multiple samples, auxiliary information such as variances or weights of features can be extracted to improve accuracy. However, the secure sketch framework assumes a symmetric setting and thus does not provide protection to the identity dependent auxiliary information. We show that, a straightforward extension of the existing framework will lead to privacy leakage. Instead, we give two schemes that "mix" the auxiliary information with the secure sketch, and show that by doing so, the schemes offer better privacy protection.

We also consider a multi-factor authentication setting, whereby where multiple secrets with different roles, importance and limitations are used together. We propose a mixing approach of combining the multiple secrets instead of simply handling the secrets independently. We show that, by appropriate mixing, entropy loss on more important secrets (e.g., biometrics) can be "diverted" to less important ones (e.g., password or PIN), thus providing more protection to the former.

List of Figures

List of Tables

Acknowledgments

I have been in National University of Singapore for ten years since my bridging courses that prepare me for the undergraduate study. During my ten-year stay at NUS, I am always grateful to her supports for her students, which make our academic lives enjoyable and fulfilling.

Perhaps the most wonderful thing I had in NUS is that I met my supervisor, Chang Ee-Chien in my last year of undergraduate study. I have constantly been inspired, encouraged and amazed by his intelligence, knowledge and energy. Following his advices and guiding, I have survived from the Final Year Project of my undergraduate, through the Ph.D. research.

Many people have contributed to this thesis. I thank Dr. Li Qiming, Dr. Lu Liming and Dr. Xu Jia for their helps and discussions. It has been a fruitful experience and pleasant journey working with them. I have also received a lot from my fellow students, namely, Zhuang Chunwang, Dong Xinshu, Dai Ting, Li Xiaolei, Zhang Mingwei, Patil Kailas, Bodhisatta Barman Roy and Sai Sathyanarayan. We are proud of the discussion group we have, from which we harvest all sorts of great research ideas.

Lastly, but most importantly, I owe my parents and my wife for their selfless supports. They have taught me everything I need to face the toughness, setbacks, and doubts. They have always been believing in me, and they are always there when I need them.

Chapter 1

Introduction

This work focuses on controlling privacy leakage in applications that involve sensitive personal information. In particular, we study two types of applications, namely data publishing and robust authentication.

We first look at publishing applications which aim to release datasets that contain useful statistical information. To publish such information while preserving the privacy of individual contributors is technically challenging. Earlier approaches such as k-anonymity (Swe02), ℓ-diversity (MKGV07), achieve indistinguishability of individuals by generalizing similar entities in the dataset. However, there are concerns of attacks that identify individuals by inferring useful information from the published data together with background knowledge that the publishers might be unaware of. In contrast, the notion of differential privacy (Dwo06) provides a strong form of assurance that takes into accounts of such inference attacks.

Most studies on differential privacy focus on publishing statistical values, for instance, k-means (BDMN05), private coreset (FFKN09), and

median of the database (NRS07). Publishing specific statistics or data-mining results is meaningful if the publisher knows what the public specifically wants. However, there are situations where the publishers want to give the public greater flexibility in analyzing and exploring the data, for example, using different visualization techniques. In such scenarios, it is desired to *"publish data, not the data mining result"* (FWCY10).

We propose a method that, instead of publishing the aggregate information, directly publishes the noisy data. The main observation of our approach is that sorting, as a function that takes in a set of real numbers from the unit interval and outputs the sorted sequence, interestingly has *sensitivity* one (Theorem 1), which is independent of the number of points to be output. Hence, the mechanism that first sorts, and then adds independent Laplace noise can have high accuracy while preserving differential privacy. From the published data, one can use isotonic regression to significantly reduce the noise. To further reduce noise, before adding the Laplace noise, consecutive elements in the sorted data can be grouped and each point is replaced by the average of its group.

There are scenarios where publishing specific statistics are required. In some of the applications, the assurance provided by differential privacy comes with a cost of high noise, which leads to low utility of the published data. To address this limitation, several relaxations have been proposed. Many relaxations capture alternative notions of "indistinguishability", in particular, on how the probabilities on the two neighbouring datasets are compared. For example, (ϵ, δ)-differential privacy (DKM$^+$06) relaxes the

bound with an additive factor δ, and (ϵ, τ)-probabilistic differential privacy (MKA+08) allows the bound to be violated with a probability τ.

We propose an alternative direction of relaxing the privacy requirement, which attempt to stay within the original framework while adopting a narrowed definition of neighbourhood, so that known results and properties still applied. The proposed relaxation takes into account of the underlying distance of the entities, and "redistributes" the indistinguishability assurance with emphasis on individuals that are close to each other. Such redistribution is similar to the original framework, which stresses on datasets that are closer-by under set-difference.

Although the idea is simple, for some applications, the challenge lies on how to exploit the relaxation to achieve higher utility. We consider two types of datasets, spatial datasets and dynamic datasets, and show that the noise level can be further reduced by constructions that exploit the δ-neighbourhood, and the utility can be significantly improved.

In the second part of the thesis, we look into protections on biometric data. Biometric data are potentially useful in building secure and easy-to-use security systems. A biometric authentication system enrolls users by scanning their biometric data (e.g. fingerprints). To authenticate a user, the system compares his newly scanned biometric data with the enrolled data. Since the biometric data are tightly bound to identities, they cannot be easily forgotten or lost. However, these features can also make user credentials based on biometric measures hard to revoke, since once the biometric data of a user is compromised, it would be very difficult

to replace it, if possible at all. As such, protecting the enrolled biometric data is extremely important to guarantee the privacy of the users, and it is important that the biometric data is not stored in the system.

A key challenge in protecting biometric data as user credentials is that they are fuzzy, in the sense that it is not possible to obtain exactly the same data in two measurements. This renders traditional cryptographic techniques used to protect passwords and keys inapplicable: these techniques give completely different outputs even when there is only a small difference in the inputs. Thus, the problem of interest here is how can we allow the authentication process to be carried out without storing the enrolled biometric data in the system.

Secure sketches (DRS04) are proposed, in conjunction with other cryptographic techniques, to extend classical cryptographic techniques to fuzzy secrets, including biometric data. The key idea is that, given a secret d, we can compute some auxiliary data S, which is called a *sketch*. The sketch S will be able to correct errors from d', a noisy version of d, and recover the original data d that was enrolled. From there, typical cryptographic schemes such as one-way hash functions can then be applied on d.

However, the secure sketch construction is designed for symmetric setting: only one sample is acquired during both enrollment and verification. To improve the performance, many applications (JRP04; UPPJ04; KGK$^+$07) adopt an asymmetric setting: during enrollment phase, multiple samples are obtained, whereby an average sample and auxiliary informa-

tion such as variances or weights of features are derived; whereas during verification, only one sample is acquired. The auxiliary information is identity-dependent but it is not protected in the symmetric secure sketch scheme. Li et al. (LGC08) observed that by using the auxiliary information in the asymmetric setting, the "key strength" could be enhanced, but there could be higher leakage on privacy.

We propose and formulate asymmetric secure sketch, whereby we give constructions that can protect such auxiliary information by "mixing" it into the sketch. We extend the notation of entropy loss (DRS04) and give a formulation on information loss for secure sketch under asymmetric setting. Our analysis shows that while our schemes maintain similar bounds of information loss compared to straightforward extensions, but they offer better privacy protection by limiting the leakage on auxiliary information.

In addition, biometric data are often employed together with other types of secrets as in a multi-factor setting, or in a multimodal setting where there are multiple sources of biometric data, partly due to the fact that human biometrics is usually of limited entropy. A straightforward method of combining the secrets independently treats each secret equally, thus may not be able to address the different roles and importance of the secrets.

We propose and analyze a cascaded mixing approach, which uses the less important secret to protect the sketch of the more important secret. We show that, under certain conditions, cascaded mixing can "divert" the information leakage of the latter towards the less important secrets. We

also provide counter-examples to demonstrate that, when the conditions are not met, there are scenarios where mixing function is unable to further protect the more important secret and in some cases it will leak more information overall. We give an intuitive explanation on the examples and based on our analysis, we provide guidelines in constructing sketches for multiple secrets.

Thesis Organization and Contributions

1. Chapter 1 is the introductory chapter.

2. Chapter 3 gives a brief survey on the related works.

3. Chapter 2 provides the background materials.

4. In Chapter 4, we propose a low-dimensional pointset publishing method that, instead of answering one particular task, can be exploited to answer different queries. Our experiments show that it can achieve high accuracy w.r.t. to some other measurements, for example range query and order statistics.

5. In Chapter 5, we propose further improve the accuracy by adopting a narrowed definition of neighbourhood which takes into account of the underlying distance of the entities. We consider two types of datasets, spatial datasets and dynamic datasets, and show that the noise level can be further reduced by constructions that exploit the narrowed neighbourhood. We give a few scenarios where δ-neighbourhood would be more appropriate, and we believe the notion provides a

good trade-off for better utility.

6. In Chapter 6, we consider biometric authentication with asymmetric setting, where in the enrollment phase, multiple biometric samples are obtained, whereas in verification, only one sample is acquired. We pointed out that, sketches that reveal auxiliary information could leak important information leading to sketch distinguishability. We propose two schemes to reduce the linkages among sketches, which offer better privacy protection by limiting the linkages among sketches.

7. In Chapter 7 we consider biometric authentication under multiple secrets setting, where the secrets differ in importance. We propose "mixing" the secrets and we show that by appropriate mixing, entropy loss on more important secrets (e.g., biometrics) can be "diverted" to less important ones (e.g., password or PIN), thus providing more protection to the former.

Chapter 2

Background

This chapter gives the background materials. We first look at the data publishing, where we want to publish information on a collection of sensitive data. We then describe biometric authentication, where we want to authenticate a user from his sensitive biometric data. We give a brief remark on the relations of both scenarios.

2.1 Data Publishing and Differential Privacy

We consider a *data curator*, who has a dataset $D = \{d_1, \ldots, d_n\}$ of private information collected from a group of *data owners*, wants to publish some information of D using a *mechanism*. Let us denote the mechanism as $\mathcal{P}$ and the published data as $S = \mathcal{P}(D)$. An *analyst*, from the published data and some background knowledge, attempts to infer some information pertaining to the "privacy" of a data owner.

2.1.1 Differential Privacy

As described, we consider mechanisms that provide differential privacy to the data owners. We treat a dataset D as a multi-set (i.e. a set with possibly repeating elements) of elements in $\mathbf{D}$. A probabilistic publishing mechanism $\mathcal{P}$ is differentially private if the published data is sufficiently noisy, so that it is difficult to distinguish the membership of an entity in a group. More specifically, a mechanism $\mathcal{P}$ on D is ϵ-differentially private if the following bound holds for any $R \subseteq \text{range}(\mathcal{P})$:

$$Pr(\mathcal{P}(D_1) \in R) \leq \exp(\epsilon) \cdot Pr(\mathcal{P}(D_2) \in R), \tag{2.1}$$

for any two *neighbouring datasets* D_1 and D_2, i.e. datasets that differ on at most one entry.

There are two interpretations of the term "differ on at most one entry". One interpretation is that $D_1 = D_2 - \{x\}$, or $D_2 = D_1 - \{x\}$, for some x in the data space $\mathbf{D}$. This is known as *unbounded neighbourhood* (Dwo06). Another interpretation of this is that D_2 can be obtained from D_1 by replacing one element, i.e. $D_1 = \{x\} \cup D_2 \setminus \{y\}$ for some $x, y \in \mathbf{D}$. Differential privacy with this definition of neighborhood is known as the *bounded differential privacy* (DMNS06; KM11). We focus on the second definition but we show that some of the result can be easily extend under the first definition.

2.1.2 Sensitivity and Laplace Mechanism

It is shown (DMNS06) that given a function $f : \mathbf{D} \to \mathbb{R}^k$ for some $k \geq 1$, the probabilistic mechanism $\mathcal{A}$ that outputs:

$$f(D) + (Lap(\triangle_f/\epsilon))^k,$$

achieves ϵ-differential privacy, where $(Lap(\triangle_f/\epsilon))^k$ is a vector of k independently and randomly chosen values from the Laplace distribution, and $\triangle_f$ is the *sensitivity* of the function f. The sensitivity of f is defined as the least upper bound on the ℓ_1 difference of all possible neighbours:

$$\triangle_f := \sup \|f(D_1) - f(D_2)\|_1,$$

where the supremum is taken over pairs of neighbours D_1 and D_2. Here, $Lap(b)$ denotes the zero mean distribution with variance $2b^2$, and a probability density function:

$$\ell(x) = \frac{1}{2b} e^{-|x|/b}.$$

2.2 Biometric Authentication and Secure Sketch

Similar to the data publishing process, in biometric authentication applications, we consider a *user* who wants to get authenticated from a *system*. In enrollment phase, the user presents his biometric data d to the system, and in the verification phase, the user can get authenticated if he can provide d', a biometric data that is "close" to d. To facilitate the closeness comparison between d and d', the system need to store some information

S on d. The privacy requirement is that such stored helper information cannot leak much information about d.

2.2.1 Min-Entropy and Entropy Loss

Before we introduce secure sketch, let us first give the formulation for information leakage. One measurement of the information is the entropy of the secret d. That is, from the adversary point of view, before obtaining S, the value of d might follow some distribution. With S, the analyst might improve his knowledge over d, and thus obtain a new distribution for d. From the distribution, we can compute the uncertainty as the *entropy* of d. Thus, the notion of entropy loss, i.e. the difference between the entropy after obtaining S and the entropy before, can be used to measure the protection. There are a few types of entropy, each relates to a different model of attacker. The most commonly used Shannon entropy (Sha01) provides an absolute limit of the average length on the best possible lossless encoding (or compression) of a sequence of i.i.d. random variables. That is, it captures the expected number of predicate queries an analyst needs, in order to get the value of d_i.

Another popular notion of entropy is the min-entropy, defined as the logarithm of the probability of the most likely value of d_i. The min-entropy captures the probability of the best guess of the analyst of the value of d_i, which is guessing the value with the highest probability. Thus it describes the maximum likelihood of correctly guessing the secret without additional information, thus it gives a bound on the security of the system.

Formally, the *min-entropy* $\mathbf{H}_\infty(A)$ of a discrete random variable A is $\mathbf{H}_\infty(A) = -\log(\max_a \Pr[A = a])$. For two discrete random variables A and B, the *average min-entropy* of A given B is defined as $\widetilde{\mathbf{H}}_\infty(A|B) = -\log(\mathbb{E}_{b \leftarrow B}[2^{-\mathbf{H}_\infty(A|B=b)}])$

The *entropy loss* of A given B is defined as the difference between the min-entropy of A and the average min-entropy of A given B. In other words, the entropy loss $\mathcal{L}(A, B) = \mathbf{H}_\infty(A) - \widetilde{\mathbf{H}}_\infty(A|B)$. Note that for any n-bit string B, it holds that $\widetilde{\mathbf{H}}_\infty(A|B) \geq \mathbf{H}_\infty(A) - n$, which means we can bound $\mathcal{L}(A, B)$ from above by n regardless of the distributions of A and B.

2.2.2 Secure Sketch

Our constructions are based on the secure sketch scheme proposed by Dodis et al. (DRS04). A secure sketch scheme should consist of two algorithms: An encoder $\mathsf{Enc} : \mathcal{M} \to \{0,1\}^*$, which computes a sketch S on a given fuzzy secret $d \in \mathcal{M}$, and a decoder $\mathsf{Dec} : \mathcal{M} \times \{0,1\}^* \to \mathcal{M}$, which outputs a point in $\mathcal{M}$ given S and d', where $\mathcal{M}$ is the space of the biometric. The correctness of secure sketch scheme will require $\mathsf{Dec}(S, d') = d$ if the distance of d and d' is less than some threshold t, with respect to an underlying distance function.

Let R be the randomness invested by the encoder Enc during the computation of the sketch S, it is shown (DRS04) that when R is recoverable from d and S and L_S is the size of the sketch, then we have

$$\mathbf{H}_\infty(d) - \widetilde{\mathbf{H}}_\infty(d|S) \leq L_S - \mathbf{H}_\infty(R) \tag{2.2}$$

In other words, the amount of information leaked from the sketch is bound-

ed from above, by the size of the sketch subtracted by the entropy of recoverable randomness invested during sketch construction, $\mathbf{H}_\infty(R)$, which is just the length of R if it is uniform. Furthermore, this upper bound is independent of d, hence this is a worst case bound and it holds for any distribution of d.

The inequality (2.2) is useful in deriving a bound on the entropy loss, since typically the size of S and $\mathbf{H}_\infty(R)$ can be easily obtained regardless of the distribution of d. This approach is useful in many scenarios where it is difficult to model the distribution of d, for example, when d represents the features of a fingerprint.

2.3 Remarks

Interestingly, the frameworks of both scenarios are similar, in the sense that we want to reveal some information of a sensitive data from users for the utility of applications, but we also want to control the leakage of sensitive information. In both scenarios, we aim to provide unconditional privacy guarantee by information theoretic techniques. Such guarantees are assured by bounding the increment in the probability of the adversary's best guess. In data publishing, we try to maximize the utility of the published data, while meeting a privacy requirement; whereas in the biometric authentication, we need to support the operations while try to minimize the information leakage.

Chapter 3

Related Works

3.1 Data Publishing

We first consider the data publishing setting: each *data owner* provide his private information d_i to the *data curator*. The data curator wants to publish information on $D = \{d_1 \dots d_n\}$, without compromising the privacy of individual data owner. There are extensive works on privacy-preserving data publishing. We refer the readers to the surveys by Fung et al. (FW-CY10) and Rathgeb et al. (RU11) for a comprehensive overview on various notions, for example, k-anonymity (Swe02), ℓ-diversity (MKGV07), and differential privacy (Dwo06). Let us briefly describe some of the most relevant works here.

3.1.1 k-Anonymity

When the data d_i contains list of attributes, one privacy concern is that individuals might be recognized from some of the attributes, and thus

information about the data owner might be leaked. The notion of k-anonymity (Swe02) addresses such linkage by forcing indistinguishability of every individual, by the attributes that might be in $\tilde{D}$, from at least $k-1$ other individuals. The strength of the protection is thus measured by the parameter k. However, in addition to the parameter k, Machanavajjhala et al. (MKGV07) show that the analyst might still learn information about the data owner, if the k individuals also sharing the same sensitive information. Therefore, they pose another requirement, that the sensitive information of the individuals sharing the same linkable information has to be ℓ-diverse: every group of individuals sharing the same linkable attributes, should have at least ℓ different unlinkable attributes. Addressing the same problem, Li et al. (LLV07) proposed a notion of t-closeness, which requires that the distribution of the linkable attributes in every group to be close to the distribution of the linkable attributes in the overall dataset with a threshold t.

The notion of k-anonymity and its variants are widely involved in the context of protecting location privacy(BWJ05; GL04), preserving privacy in communication protocol(XY04; YF04) data mining techniques(Agg05; FWY05) and many others.

3.1.2 Differential Privacy

There is another line of privacy protection is known as differential privacy. Its goal is to ensure that that distributions of any output released about the dataset are close, whether or not any particular individual d_i

is included. As outlined in the surveys (FWCY10), there are many successful constructions on a wide range of data analysis tasks including k-means (BDMN05), private coreset (FFKN09), order statistics (NRS07) and histograms (LHR+10; BCD+07; XWG10; HRMS10).

Among which, the histogram of a dataset contains rich information that can be harvested by subsequent analysis of multiple purposes. Exploiting the *parallel composition* property of differential privacy, we can treat non-overlapping bins independently and thus achieving high accuracy. There are a number of research efforts (LHR+10; BCD+07) investigating the dependencies of frequencies counts of fixed overlapping bins, where parallel composition cannot be directly applied. Such overlapping bins are interesting as different domain partition could lead to different accuracy and utility. For instance, Xiao et al. (XWG10) proposes publishing wavelet coefficients of an equi-width histogram, which can be viewed as publishing a series of equi-width histograms with different bin-widths, and is able to provide higher accuracy in answering range queries compare to a single equi-width histogram.

Hay et al. (HRMS10) proposed a method that employs isotonic regression to boost accuracy, but in a way different from our mechanism. They consider publishing *unattributed histogram*, which is the (unordered) multi-set of the frequencies of a histogram. As the frequencies are unattributed (i.e. order of appearance is irrelevant), they proposed publishing the sorted frequencies and later employing isotonic regression to improve accuracy.

Machanavajjhala et al. (MKA+08) proposed a 2D dataset publishing method that can handle the sparse data in 2D equi-width histogram. To mitigate the sparse data, their method shrinks the sparse blocks by examining publicly available data such as a previously release of similar data. They demonstrate this idea on the commuting patterns of the population of the United States, which is a real-life sparse 2D map in large domain.

3.2 Biometric Authentication

We now briefly describe the existing works on secure sketch, a tool introduced to handle the fuzziness in biometric secrets in authentication process.

3.2.1 Secure Sketches

The fuzzy commitment (JW99) and the fuzzy vault (JS06) schemes are among the first error-tolerant cryptographic techniques. The fuzzy commitment employs the error correcting codes to handle errors in Hamming distance: it randomly picks a codeword in the set of codes and subtract it from a biometric sample that can be represented as bit string of same length. During verification, the newly obtained biometric sample is then added back to it and thus the error can be corrected by mapping to the nearest codeword. The fuzzy vault scheme handles fuzzy data represented as set of elements by encoding the elements as points on a randomly generated polynomial of lower degree with random points not on the polynomial. During verification, given a set of small enough set difference, we can locate enough points on the polynomial and thus reconstruct it.

The security of the schemes rely on the number of codewords or possible polynomials, and they do not give a guarantee on how much information is revealed by the sketches, especially when the distribution of the biometric samples is unknown. More recently, Dodis et al. (DRS04) give a general framework of secure sketches, where the security is measured by the entropy loss of the secret given the sketch in min-entropy. The framework provides a bound on the entropy loss, and the bound applies to any distribution of biometric samples with high enough entropy. They also give specific schemes that meet theoretical bounds for Hamming distance, set difference and edit distance respectively.

Another distance measure, point-set difference, motivated from a popular representation for fingerprint features, is investigated in a number of studies (CKL03; CL06; CST06). Different approaches (LT03; TG04; TAK$^+$05) focus on information leakage defined using Shannon entropy on continuous data with known distributions.

There are also a number of investigations on the limitations of secure sketches under different security models. Boyen (Boy04) studies the vulnerability that when the adversary obtains enough sketches constructed from the same secret, he could infer the secret by solving linear system. This concern is more severe when the error correcting code involved is biased: the value 0 is more likely to appear than the value 1. Boyen et al. (BDK$^+$05) further study the security of secure sketch schemes under more general attacker models, and techniques to achieve mutual authentication are proposed.

This security model is further extended and studied by Simoens et al. (STP09), which focuses more on privacy issues. Kholmatov et al. (KY08) and Hong et al. (HJK+08) demonstrate such limitations by giving correlation attacks on known schemes.

3.2.2 Multiple Secrets with Biometrics

The idea of using a secret to protect other secrets is not new. Souter et al. (SRS+99) propose integrating biometric patterns and encryption keys by hiding the cryptographic keys in the enrollment template via a secret bit-replacement algorithm. Some other methods use password protected smartcards to store user templates (Ada00; SR01). Ho et al. (HA03) propose a dual-factor scheme where a user needs to read out a one-time password generated from a token, and both the password and the voice features are used for authentication. Sutcu et al. (SLM07) study secure sketch for face features and give an example of how the sketch scheme can be used together with a smartcard to achieve better security.

Using only passwords as an additional factor is more challenging than using smartcards, since the entropy of typical user chosen passwords is relatively low (MT79; FH07; Kle90). Monrose (MRW99) presents an authentication system based on Shamir's secret sharing scheme to harden keystroke patterns with passwords. Nandakuma et al. (NNJ07) propose a scheme for hardening a fingerprint minutiae-based fuzzy vault using passwords, so as to prevent cross-matching attacks.

3.2.3 Asymmetric Biometric Authentication

To improve the performance in terms of relative operating characteristic (ROC), many applications (JRP04; UPPJ04; KGK+07) adopt an asymmetric setting. During enrollment phase, multiple samples are obtained, whereby an average sample and auxiliary information such as variances or weights of features are derived. During verification, only one sample is acquired. The derived auxiliary information can be helpful in improving ROC. For example, it could indicate that a particular feature point is relatively inconsistent and should not be considered, and thus reducing the false reject rate. Note that the auxiliary information is identity-dependent in the sense that different identity would have different auxiliary information. Li et al. (LGC08) observed that by using the auxiliary information in the asymmetric setting, the "key strength" could be enhanced due to the improvement of ROC, but there could be higher leakage on privacy.

Current known works, for example, the schemes given by Li et al. (LGC08) and by Kelkboom (KGK+07), store the auxiliary information in clear. Li et al. (LGC08) employ a scheme that carefully groups the feature points to minimize the differences of variance among the groups. The derived grouping is treated as auxiliary information and is published in clear. The scheme proposed by Kelkboom et al. (KGK+07) computes the means and variances of the features from the multiple enrolled face images, and selects the k features with least variances. The selection indices are also published in clear. The revealed auxiliary information could potentially leak important identity information as an adversary could distinguish

whether a few sketches are of from the same identity by comparing the auxiliary information. Such leakage is similar to the sketch distinguishability in the typical symmetric setting (STP09). Therefore, it is desired to have a sketch construction that can protect the auxiliary information as well.

Chapter 4

Pointsets Publishing with Differential Privacy

In this chapter and Chapter 5, we consider the data publishing problem with differential privacy.

In this chapter, we consider D as low-dimensional pointset, and propose a data publishing algorithm that, instead of publishing aggregated values such as k-means (BDMN05), private coreset (FFKN09), or median of the database (NRS07), it publishes the pointset data itself. Such data publishing can be later exploited in different scenarios where the data serve multiple purposes, in which cases it is more desired to *"publish data, not the data mining result"* (FWCY10).

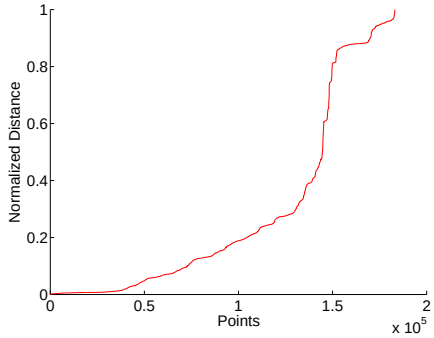
4.1 Pointset Publishing Setting

We treat the data D as a multi-set (i.e. a set with possibly repeating elements) of low-dimensional points in a normalized domain. That is, we

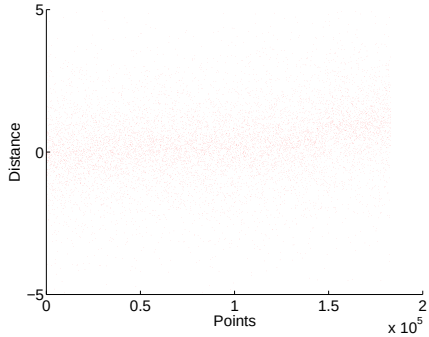
consider $D = \{d_1 \ldots, d_n\}$, where $d_i \in [0,1]^k$ for some small k. We want to publish statistical information on D for queries with different purposes.

One way to retain rich information that can be harvested by subsequent analysis is to publish a histogram of the dataset D. In the context of differential privacy, *parallel composition* can be exploited to treat non-overlapping bins independently and thus achieving high accuracy. There are a number of research efforts (LHR+10; BCD+07) investigating the dependencies of frequencies counts of fixed overlapping bins, where parallel composition cannot be directly applied. Such overlapping bins are interesting as different domain partition could lead to different accuracy and utility. For instance, Xiao et al. (XWG10) proposed publishing wavelet coefficients of an equi-width histogram, which can be viewed as publishing a series of equi-width histograms with different bin-widths, and is able to provide higher accuracy in answering range queries compare to a single equi-width histogram.

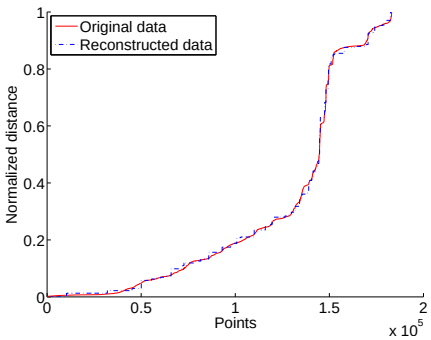
It is generally well accepted that equi-depth histogram and V-optimal histogram provide more useful statistical information compare to equi-width histogram (PSC84; PHIS96), especially for multidimensional data. These histograms are adaptive in the sense that the domain partitions are derived from the data such that denser regions will have smaller bin-widths and the sparser regions will have larger bin-widths, as illustrated in Fig. 4.7(b). Since the bin-widths are derived from the dataset, they leak information about the original dataset. There are relatively few works that consider adaptive histogram in the context of differential privacy.



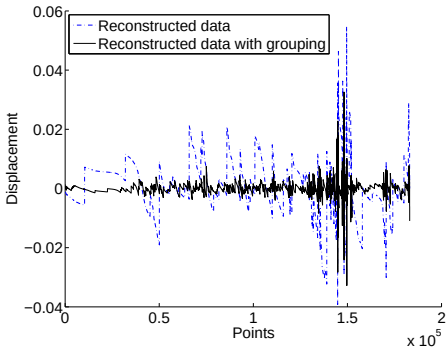
(a) Sorted 1D points.



(b) The sorted points with Laplace noise added. To avoid clogging, only 10% of the points (randomly chosen) are plotted.



(c) Reconstructed with isotonic regression.



(d) The differences of the reconstructed points from the original.

Figure 4.1: Illustration of pointset publishing.

One exception is the work by Xiao et al. (XXY10). Their method consists of two steps where firstly synthetic data are generated from the differentially private equi-width histogram. After that, a k-d tree (which can be viewed as an adaptive histogram) is generated from the synthetic data, and the noisy counts are then released with the partition. Machanavajjhala et al. (MKA+08) proposed a mechanism that publishes 2D histograms with varying bin-widths, where the bin-widths are determined from a previously released similar data. The histograms generated are not adaptive in the sense that the partitions do not depend on the data to be published.

In this chapter, instead of publishing the noisy frequency counts in equi-width bins, we propose a method that directly publishes the noisy da-

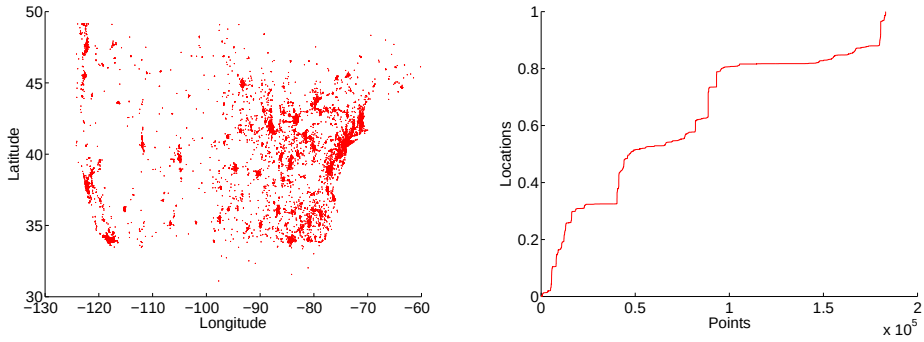
ta, which in turn leads to an adaptive histogram. To illustrate, let us first consider a dataset consisting of a set of real numbers from the unit interval, for example, the normalized distance of Twitter users' locations (web) to New York City (Fig. 4.1(a)). We observe that sorting, as a function that takes in a set of real numbers from the unit interval and outputs the sorted sequence, interestingly has sensitivity one (Theorem 1). Hence, the mechanism that first sorts, and then adds independent Laplace noise of LAP($1/\epsilon$) to each element achieves ϵ-differential privacy. Fig. 4.1(b) shows the noisy output data after the Laplace noise has been added to the sorted sequence. Although seemingly noisy, there are dependencies to be exploited because the original sequence is sorted. By using isotonic regression, the noise can be significantly reduced (Fig. 4.1(c)). To further reduce noise, before adding the Laplace noise, consecutive elements in the sorted data can be grouped and each point is replaced by the average of its group. Fig. 4.1(d) shows the difference of the original and the reconstructed points with and without grouping.

To extend the proposed method to higher dimension data, for example, location data of 183,072 Twitter users in North America as shown in Fig. 4.2(a), we employ *locality-preserving mapping* to map the multidimensional data to one-dimension (Fig. 4.2(b)), such that any two close points in the one-dimension domain are mapped from two close multidimensional points. After that, the publisher can apply the proposed method on the 1D points, and publish the reverse mapped multidimensional points.

One desired feature of our scheme is its simplicity: there is only one

parameter, the group size, to be determined. The group size affects the accuracy in three ways: (1) its effect on the *generalization error*, which is introduced due to averaging; (2) its effect on the level of Laplace noise to be added by the differentially private mechanism; and (3) its effect on the number of constraints in the isotonic regression. Based on our error model, the optimal parameter can be estimated without knowledge of the dataset distribution. In contrast, many existing methods have many parameters whose optimal values are difficult to be determined differentially privately. For instance, although the equi-width histogram has only one parameter, i.e. the bin-width, its value significantly affects the accuracy, and it is not clear how to differentially privately obtain a good choice of the bin width.

As mentioned, we measure the utility of the published spatial dataset with Earth mover's distance(EMD). We show that publishing pointset under this measurement may still attain high accuracy w.r.t. other measurements. We conduct empirical studies to compare against a few related known methods: equi-width histogram, wavelet-based method (XWG10) and smooth sensitivity based median-finding (NRS07). The experiment results show that our method outperforms the wavelet-based method w.r.t. accuracy of range-query, even for ranges with large sizes. It is also comparable to the smooth sensitivity based method in publishing median.



(a) Locations of Twitter users. To avoid clogging, only 10% of the points (randomly chosen) are plotted.

(b) Sorted 1D images of the data.

Figure 4.2: Twitter location data and their 1D images of a locality-preserving mapping.

4.2 Background

4.2.1 Isotonic Regression

Given a sequence of n real numbers $a_1, \ldots, a_n$, the problem of finding the least-square fit $x_1, \ldots, x_n$ subjected to the constraints $x_i \leq x_j$ for all $i < j \leq n$ is known as the isotonic regression. Formally, we want to find the $x_1, \ldots, x_n$ that minimizes

$$\sum_{i=1}^{n} (x_i - a_i)^2, \qquad \text{subjected to } x_i \leq x_j \text{ for all } 1 \leq i < j \leq n.$$

The unique solution can be efficiently found using pool-adjacent-violators algorithms in $O(n)$ time (GW84). When minimizing w.r.t. ℓ-1 norm, there is also an efficient $O(n \log n)$ algorithm (Sto00). There are many variants of isotonic regression, for example, variants with a smoothness component in the objective function (WL08; Mey08).

Isotonic regression has been used to improve a differentially private query result. Hay et al. (HRMS10) proposed a method that employs isotonic regression to boost accuracy, but in a way different from our mech-

anism. They consider publishing *unattributed histogram*, which is the (unordered) multi-set of the frequencies of a histogram. As the frequencies are unattributed (i.e. order of appearance is irrelevant), they proposed publishing the sorted frequencies and later employing isotonic regression to improve accuracy.

4.2.2 Locality-Preserving Mapping

A locality-preserving mapping $T : [0,1]^d \rightarrow [0,1]$ maps d-dimensional points to the unit interval, while preserving locality. For the proposed method, we seek a mapping that, if the mapped points $T(x)$, $T(y)$ are "close", then x and y are "close" in the d-dimensional space. More specifically, there is some constant c s.t. for any x, y in the domain of the mapping T,

$$\|x - y\|_2 \leq c \cdot (\|T(x) - T(y)\|)^{1/d}. \tag{4.1}$$

The well-known Hilbert curve (GL96) is a locality-preserving mapping. It is shown that for any 2D points x, y in the domain of T, $\|x-y\|_2 \leq 3\sqrt{|T(x) - T(y)|}$. Niedermeier et al. (NRS97) showed that with careful construction, the bound can be improved to $2\sqrt{|T(x) - T(y)|}$ for 2D points and $3.25\sqrt[3]{\|T(x) - T(y)\|}$ for 3D points. In our construction, for simplicity, we use Hilbert curve in our experiments.

Note that it is challenging in preserving locality "in the other direction", that is, any two "close" points in the d-dimensional domain are mapped to "close" points in the one-dimensional range (MD86). Fortunately, in our problem, such property is not required.

4.2.3 Datasets

We conduct experiments on two datasets: locations of Twitter users (web) (herein called the Twitter location dataset) and the dataset collected by Kaluža et al. (KMD+10) (herein called Kaluža's dataset). The Twitter location dataset contains over 1 million Twitter users' data from the period of March 2006 to March 2010, among which around 200,000 tuples are labeled with location (represented in latitude and longitude) and most of the tuples are in the North American continent, concentrating in regions around the state of New York and California. Fig. 4.2(a) shows the cropped region covering most of the North American continent. The cropped region contains 183,072 tuples. The Kaluža's dataset contains 164,860 tuples collected from tags that continuously record the location information of 5 individuals. While some of the tuples consist of many attributes, in our experiments, only the 2D location data are being used.

4.3 Proposed Approach

Before receiving the data, the publisher has to make a few design choices. The publisher needs to decide on a locality-preserving mapping T, and the strategy (which is represented as a lookup table) of determining the group size from the privacy requirement ϵ and the size of dataset n. Now, given the dataset D of size n, and the privacy requirement ϵ, the publisher carries out the following:

A1. The publisher maps each point in D to a real number in the unit

interval $[0, 1]$ using T, and lookups the group size based on n and ϵ. Let $T(D)$ be the set of transformed points. For clarity in exposition, let us assume that k divides n.

A2. The publisher sorts the mapped points, divides the sorted sequence into groups of k consecutive elements, and then for each group, determines its average over the k elements. Let the averages be $S = \langle s_1, \ldots, s_{n/k} \rangle$.

A3. The publisher releases $\widetilde{S} = S + (\mathrm{Lap}(\epsilon^{-1})/k)^{(n/k)}$ and the group size k.

A public user may extract information from the published data as follow:

B1. The user performs isotonic regression on $\widetilde{S}$ and obtains $\mathrm{IR}(\widetilde{S})$, and then replaces each element $\widetilde{s}_i$ in $\mathrm{IR}(\widetilde{S})$ with k points of value $\widetilde{s}_i$. Let P be the set of resulting points.

B2. The user maps the data point back to the original domain, that is, computes $\widetilde{D} = T^{-1}(P)$. Let us call $\widetilde{D}$ the reconstructed data.

Note that the public user is not confined to performing step B1 and B2. The user may, for example, incorporates some background knowledge to enhance accuracy. To relieve the public from computing step B1 and B2, the regression and the inverse mapping can be carried out by the publisher on behalf of the users. Nevertheless, the raw data $\widetilde{S}$ should be (although it is not necessary) published alongside the reconstructed data for further statistical analysis.

4.4 Security Analysis

In this section, we show that the proposed mechanism (Step A1 to A3) achieves differential privacy. The following theorem shows that sorting, as a function, interestingly has sensitivity 1. Note that a straightforward analysis that treats each element independently could lead to a bound of n, which is too large to be useful.

Theorem 1 *Let $S_n(D)$ be a function that on input D, which is a multi-set containing n real numbers from the unit interval $[0, p]$, outputs the sorted sequence of elements in D. The sensitivity of S_n w.r.t. the bounded differential privacy is p.*

Proof Let D_1 and D_2 be any two neighboring datasets. Let $\langle x_1, x_2 \ldots x_i \ldots x_n \rangle$ be $S_n(D_1)$, i.e. the sorted sequence of D_1. WLOG, let us assume that an element x_i is replaced by a larger value A to give D_2, for some $1 \leq i \leq n-1$ and $x_i < A$. Let j to be largest index s.t. $x_j < A \leq p$. Hence, the sorted sequence of D_2 is:

$$x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_j, A, x_{j+1}, \ldots, x_n.$$

The L_1 difference due to the replacement is,

$$\begin{aligned}
& \|S_n(D_1) - S_n(D_2)\|_1 \\
= \; & |x_{i+1} - x_i| + |x_{i+2} - x_{i+1}| + \ldots + |x_j - x_{j-1}| + |A - x_j| \\
= \; & (x_{i+1} - x_i) + (x_{i+2} - x_{i+1}) + \ldots + (x_j - x_{j-1}) + (A - x_j) \\
= \; & A - x_i \leq p.
\end{aligned}$$

We can easily find an instance of D_1 and D_2 where the difference $A - x_i = p$. Hence, the sensitivity is p.

Thus, when the points are mapped to $[0, 1]$, the sensitivity S_n is 1. Therefore, the mechanism $S_n(D) + Lap(1/\epsilon)^n$ enjoys ϵ-differential privacy. Also note that the value of n is fixed. Hence, the size of D is not a secret and is made known to the public.

The following corollary shows that grouping (in Step A2) has no effect on the sensitivity.

Corollary 2 *Consider a partition $H = \{h_1, h_2 \ldots h_m\}$ of the indices $\{1, 2, \ldots, n\}$. Let $S_H(D)$ be the function that, on input D, which is a multi-set containing n real numbers from the unit interval $[0, p]$, outputs a sequence of m numbers:*

$$y_i = \sum_{j \in h_i} x_j,$$

for $1 \leq i \leq m$ where $\langle x_1, x_2, \ldots, x_n \rangle$ is the sorted sequence of D. The sensitivity of S_H is p.

Proof Again Let D_1 and D_2 be any two neighboring datasets. Let $\langle x_1, x_2 \ldots x_i \ldots x_n \rangle$ be $S_n(D_1)$, i.e. the sorted sequence of D_1, and $\langle y_1, \ldots, y_m \rangle$ be $S_H(D_1)$. WLOG, Consider when an element x_i is replaced by a larger value A to give D_2 and let j to be largest index s.t. $x_j < A$. Hence, the sorted sequence of D_2 is:

$$x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_j, A, x_{j+1}, \ldots, x_n.$$

Let $\langle y'_1, \ldots, y'_m \rangle$ be $S_H(D_2)$. Thus, we have $y'_i \geq y_i$ for all i, and the L_1 difference due to the replacement is,

$$\begin{aligned}
& \|S_H(D_1) - S_H(D_2)\|_1 \\
= \; & (y'_1 - y_1) + (y'_2 - y_2) \ldots + (y'_m - y_m) \\
= \; & (x_{i+1} - x_i) + (x_{i+2} - x_{i+1}) + \ldots + (x_j - x_{j-1}) + (A - x_j) \\
= \; & A - x_i \leq p.
\end{aligned}$$

Again, we can easily find an instance of D_1 and D_2 where the difference $A - x_i = p$. Hence, the sensitivity is p.

Note that the grouping in step A2 is a special partition with equal-sized h_i's, whereas Corollary 2 gives a more general result where H can be any partition. From Corollary 2, the proposed mechanism achieves ϵ-differential privacy.

4.5 Analysis and Parameter Determination

The main goal of this section is to analyze the effect of the privacy requirement ϵ, dataset size n and the group size k on the error in the reconstructed data, which in turn provides a strategy in choosing the parameter k given n and ϵ.

Intuitively, when n and ϵ are fixed, the choice of parameter k affects the accuracy in following three ways: (1) a larger k decreases the number of constraints in isotonic regression, which leads to lower noise reduction; (2) a larger k reduces the effect of the Laplace noise; and (3) a larger k introduces higher generalization error due to averaging.

Our analysis consists of the following parts: We first describe our utility function in Section 4.5.1. In Section 4.5.2, we consider the case where $k = 1$ and empirically show that the expected error of a typical dataset can be well approximated by the expected error on a synthetic equally-spaced dataset. Let us call this error $Err_{n,\epsilon}$. Next in Section 4.5.3, we investigate and estimate the generalization error due to the averaging and show that with a reasonable assumption on the dataset distribution, the expected error can be approximated by $\frac{k}{4n}$. Let us call this error $Gen_{n,k}$. Finally, in Section 4.5.4, we consider the general case of $k \geq 1$ and give an approximation of the expected error in terms of $Err_{n,\epsilon}$ and $Gen_{n,k}$.

4.5.1 Earth Mover's Distance

To measure the utility of a published spatial dataset, one commonly compares the distance of the published data S to the original sensitive data D. Some existing works measure the accuracy of a histogram by its distance, such as L_2 distance or KL divergence, to a reference equi-width histogram. One limitation of this measurement is that the reference histogram can be arbitrary and thus arguably ill-defined. If the reference bin-width is too small, each bin will contain either one or no point, which leads to significantly large distance from a seemingly accurate histogram. On the other hand, if its bin-width is too large, the reference histogram would be over generalized. We choose to measure the utility of the published dataset by the earth mover's distance (EMD) (RGT97), which measures the distance of the published data and original points, where the "reference" is the orig-

inal points and thus well-defined. The EMD between two pointsets of equal size is defined to be the minimum cost of bipartite matching between the two sets, where the cost of an edge linking two points is the cost of moving one point to the other. Hence, EMD can be viewed as the minimum cost of transforming one pointset to the other. Different variants of EMD differ on how the cost is defined. In this thesis, we adopt the typical definition that defines the cost as the Euclidean distance between the two points.

In one-dimensional space, the EMD between two sets D and $\widetilde{D}$ is simply the L_1 norm of the differences between the two respective sorted sequences, i.e. $\|S_n(D) - S_n(\widetilde{D})\|_1$, which can be efficiently computed. Recall that $S_n(D)$ outputs the sorted sequence of elements in D. In other words,

$$\mathrm{EMD}(D, \widetilde{D}) = \sum_{i=1}^{n} |p_i - \widetilde{p}_i|, \tag{4.2}$$

where p_i's and $\widetilde{p}_i$'s are the sorted sequence of D and $\widetilde{D}$ respectively. Note that this definition assumes D and $\widetilde{D}$ have the same number of points.

Given a dataset D and the published dataset $\widetilde{D}$ of a mechanism $\mathcal{M}$ where $|D| = |\widetilde{D}| = n$, let us define the *normalized error* as $\frac{1}{n}\mathrm{EMD}(D, \widetilde{D})$ and denote $Err_{\mathcal{M},\mathcal{D}}$ the expected normalized error,

$$Err_{\mathcal{M},D} = Exp\left[\ \frac{1}{n}\ \mathrm{EMD}(D, \widetilde{D})\ \right], \tag{4.3}$$

where the expectation is taken over the randomness in the mechanism.

Our mechanism publishes $\widetilde{D}$ based on two parameters: the privacy requirement ϵ and the group size k. Therefore, let us write $Err_{\epsilon,k,D}$ for the expected normalized error of the dataset published in Step B2.

4.5.2 Effects on Isotonic Regression

Let us consider the expected normalized error when $k = 1$, in other words, we first consider the mechanism without grouping. In such case, the reconstructed dataset is $\mathrm{IR}(S_n(D) + \mathrm{Lap}(\epsilon^{-1})^n)$. Thus, the expected normalized error is

$$Err_{\epsilon,1,D} = Exp\left[\frac{1}{n}\ \mathrm{EMD}(D, \mathrm{IR}(S_n(D) + \mathrm{Lap}(\epsilon^{-1}))^n)\ \right].$$

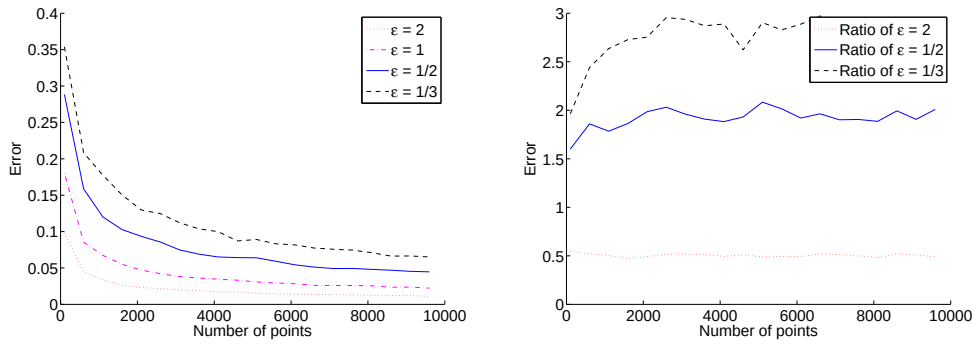
To estimate the above expected error, we compute the expected normalized error on a few datasets of varying size n: (1) Multi-sets containing elements with the same value 0.5 (herein called repeating single-value dataset), (2) sets containing equally-spaced numbers $(i/(n-1))$ for $i = 0, \ldots, n-1$ (herein call equally-spaced dataset), (3) sets containing n randomly chosen elements from the Twitter location data (web), and (4) sets containing n randomly chosen elements from the Kaluža's data (KMD+10).

Fig. 4.4(a) shows the expected error $Err_{1,1,D}$ for the four datasets with different n. Each sample in the graph is the average over 500 runs. Observe that the error on equally-spaced data well approximates the errors on the two real-life dataset (Twitter location dataset and Kaluža's dataset). Hence, we take the error on the equally-spaced dataset as an approximation of the errors on other datasets. For abbreviation, let $Err_{\epsilon,n}$ denote the expected error $Err_{\epsilon,1,D}$ where D is the equally-spaced dataset with n points. Based on experiences on other datasets, we suspect that the expected error depends on the difference of the minimum and the maximum element in D, and the repeating single-value dataset is the extreme case whose error

could be served as a lower bound as shown in Fig. 4.4(a).

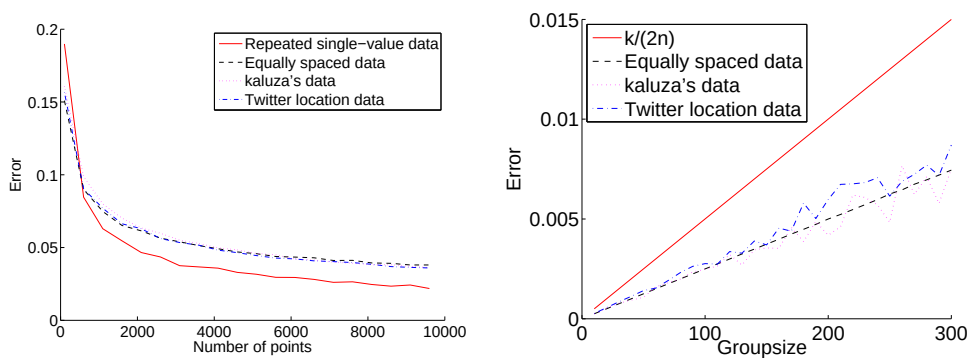
Fig. 4.3(a) shows the expected error $Err_{\epsilon,1,D}$ for dataset on equally-spaced points for different ϵ and n, and Fig. 4.3(b) shows the ratios of error for different ϵ to $Err_{1,n}$. The results agree with the intuition that when ϵ is increased by a factor of c, the error would approximately decrease by factor of c, that is,

$$Err_{\epsilon,1,D} \approx \frac{1}{c} Err_{c\epsilon,1,D}. \tag{4.4}$$



(a) The normalized error $Err_{\epsilon,n}$. (b) The ratio of $Err_{\epsilon,n}$ to $Err_{1,n}$.

Figure 4.3: The normalized error for different security parameter.



(a) Expected normalized error $Err_{1,1,D}$. (b) Normalized generalization error $Gen_{D,k}$.

Figure 4.4: The expected normalized error and normalized generalization error.

4.5.3 Effect on Generalization Noise

When $k > 1$, the grouping introduces a *generalization error*, which is incurred when all elements in a group are represented by their mean. Before giving formal description of generalization error, let us introduce some notations.

Given a sequence $D = \langle x_1, \ldots, x_n \rangle$ of n numbers, and a parameter k, where k divides n, let us call the following function *downsampling*:

$$\downarrow_k (D) = \langle s_1, \ldots, s_{(n/k)} \rangle,$$

where each s_i is the average of $x_{k(i-1)+1}, \ldots, x_{ik}$. Given a sequence $D' = \langle s'_1, \ldots, s'_m \rangle$ and k, let us call the following function *upsampling*,

$$\uparrow_k (D') = \langle x'_1, \ldots, x'_{mk} \rangle,$$

where $x'_i = s'_{\lfloor (i-1)/k \rfloor + 1}$ for each i.

The *normalized generalization error* is defined as,

$$Gen_{D,k} = \frac{1}{n} \| D - \uparrow_k (\downarrow_k (D)) \|_1.$$

It is easy to see that, for any k and D of size n, the normalized generalization error is at most $k/(2n)$. However, this bound is often an overestimate. Fig. 4.4(b) shows the generalization error of different group size a dataset containing $10,000$ equally-spaced values, a dataset containing $10,000$ numbers randomly drawn from the transformed Kaluža's dataset, and a dataset of $10,000$ numbers randomly drawn from the transformed Twitter location data.

Observe that, empirically, the generalization error can be well approximated by $\frac{k}{4n}$. To see that such approximation holds for a typical

dataset, consider the following partition of the unit interval: $0 = p_0 < p_1 < p_2, \ldots, p_{(n/k)-1} < p_{n/k} = 1$. Let us consider a sorted sequence S of elements in dataset D, where the $jk+1, jk+2, \ldots (j+1)k$-th elements in S are uniformly independent and identically distributed over $[p_j, p_{j+1})$ for $j = 0, 1, \ldots, (n/k)-1$. We can verify that the expected generalization error $Gen_{D,k} \approx \frac{k}{4n}$ with simulations. Hence, we approximate the generalization error by $\frac{k}{4n}$ and denote it as $Gen_{n,k}$.

4.5.4 Determining the group size k

Now, let us combine the components and build an error model of how k affects the accuracy. First, grouping reduces the number of constraints by a factor of k. As suggested by Fig. 4.4(a), when the number of constraints decreases, the error reduction from isotonic regression decreases. On the other hand, recall that the regression is performed on the published values divided by k (see the role of k in Step A3). This essentially reduces the level of Laplace noise by a factor of k. Hence, the accuracy attained by grouping k elements is "equivalent" to the accuracy attained without grouping but with the privacy parameter ϵ increased by a factor of k. These two components can be estimated in terms of $Err_{\epsilon,n}$ as follow:

$$Err_{\epsilon,k,D} \approx \frac{1}{k} Err_{\epsilon,n/k}.$$

For general k, the reconstructed dataset is

$$\widetilde{D} = \uparrow_k (\mathrm{IR}(\widetilde{S})),$$

where $\widetilde{S}$ is an instance of $\downarrow_k (S_n(D)) + \text{Lap}(1)^{n/k}$. Now, we have,

$$\begin{aligned}
&\text{EMD}\ (D, \widetilde{D}) = \|S_n(D) - \uparrow_k (\text{IR}(\widetilde{S}))\|_1 \\
&= \|S_n(D) - \uparrow_k (\downarrow_k (S_n(D)) + \uparrow_k (\downarrow (S_n(D))) - \uparrow_k (\text{IR}(\widetilde{S}))\|_1 \\
&\leq n \cdot Gen_{D,k} + \| \uparrow_k (\downarrow_k (S_n(D))) - \uparrow_k (\text{IR}(\widetilde{S}))\|_1 \\
&= n \cdot Gen_{D,k} + k \cdot \| \downarrow_k (S_n(D)) - \text{IR}(\widetilde{S})\|_1 \\
&= n \cdot Gen_{D,k} + k \cdot \text{EMD}(\downarrow_k (S_n(D)), \text{IR}(\widetilde{S})). \qquad (4.5)
\end{aligned}$$

Note that the first term $n \cdot Gen_{D,k}$ is a constant independent of the random choices made by the mechanism. Also note that the second term is the EMD between the down-sampled dataset and its reconstructed copy obtained using group size 1. Thus, by taking expectation over randomness of the mechanism, we have

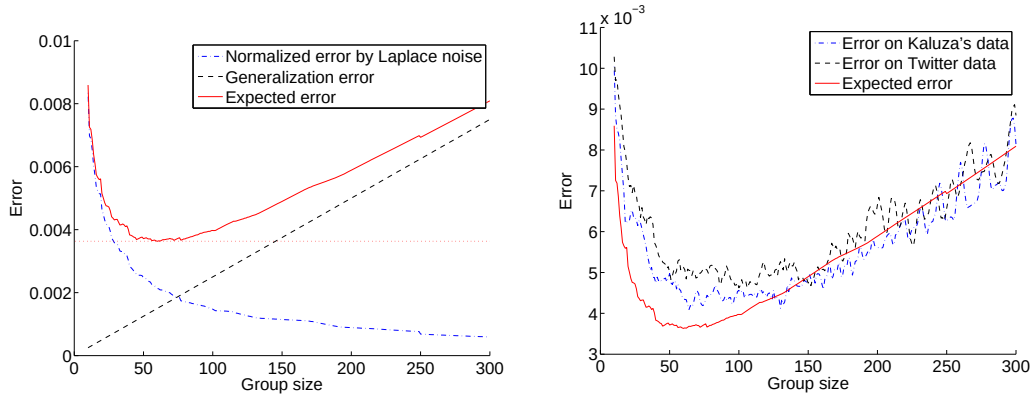
$$Err_{\epsilon,k,D} \leq Gen_{D,k} + \frac{1}{k} Err_{\epsilon,1,\downarrow_k(D)}. \qquad (4.6)$$

In other words, the expected normalized error is bounded by the sum of normalized generalization error, and the normalized error incurred by the Laplace noise. Fig. 4.5(a) shows the three values versus different group size k for equally-spaced data of size 10,000. The minimum of the expected normalized error suggests the optimal group size k.

Fig. 4.5(b) illustrates the expected errors for different k on the Twitter location data with 10,000 points. The red dotted line is $Err_{\epsilon,k,D}$ whereas the blue solid line is the sum in the right-hand-side of the inequality (4.6). Note that the differences between the two graphs are small. We have conducted experiments on other datasets and observed similar small differences. Hence, we take the sum as an approximation to the expected

normalized error,

$$Err_{\epsilon,k,D} \approx Gen_{n,k} + \frac{1}{k} Err_{\epsilon,n/k}. \tag{4.7}$$



(a) The expected error.

(b) Comparison with the actual error.

Figure 4.5: The expected error and comparison with actual error.

Now, we are ready to find the optimal k given ϵ and n. From Fig. 4.4(a) and Fig. 4.4(b) and the approximation given in equation (4.7), we can determine the best group size k when given the size of the database n and the security requirement ϵ. From the parameter ϵ, we can obtain the value $\frac{1}{k}Err_{n/k,e}$ for different k. From the database's size n, we can determine $Gen_{n,k}$ which is $\frac{k}{4n}$. Thus, we can approximate the normalized error $Err_{k,D}$ with equation (4.7) as illustrated in Fig. 4.5(a). Using the same approach, the best group size given different n and ϵ can be calculated and is presented in table 4.1.

4.6 Comparisons

In this section, we compare the performance of the proposed mechanism with three known mechanisms w.r.t. different utility functions. We first

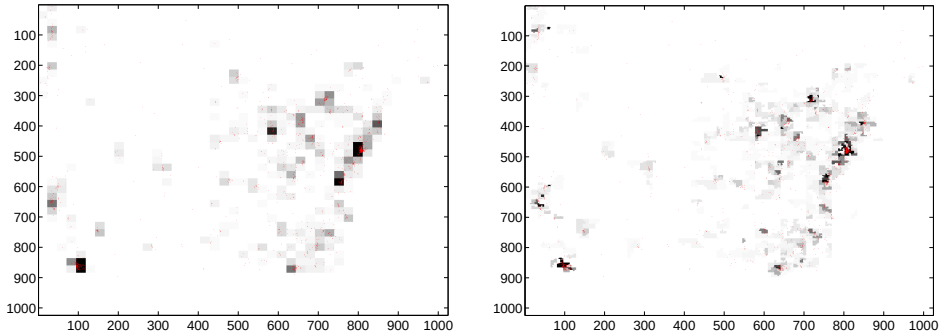
Table 4.1: The best group size k given n and ϵ

	$\epsilon = 0.5$	$\epsilon = 1$	$\epsilon = 2$	$\epsilon = 3$
n= 2,000	44	29	20	12
n= 5,000	59	37	27	18
n= 10,000	79	51	36	27
n= 20,000	121	83	61	41
n= 100,000	234	150	98	73
n= 180,000	300	177	110	94

compare the mechanism that outputs equi-width histograms. Next, we investigate the wavelet-based mechanism proposed by Xiao et al. (XWG10) and measure accuracy of range queries. Lastly, we consider the problem of estimating median, and compare with a mechanism based on smooth sensitivity proposed by Nissim et al. (NRS07). We do not conduct experiments to compare with the k-d tree method (XXY10) because it is designed for high dimensional data and it is not clear how to apply it to low dimension effectively. For comparison purposes, we empirically choose the best parameters for the known mechanisms, although this apriori information is not available to the publisher. We remark that the parameter k of our proposed mechanism is chosen from Table 4.1.

4.6.1 Equi-width Histogram

We want to compare the performance of our method with the equi-width histogram method. Fig. 4.6(a) shows a differentially private equi-width histogram. To visualize the reconstructed points of our method as a histogram, we construct the bins in the following way: let B be the set of *distinct*-points in D, and we construct the Voronoi diagram of B. The cells in the Voronoi diagram are taken to be the bins of a histogram as depicted



(a) Equi-width method. (b) Proposed method.

Figure 4.6: Visualization of the density functions.

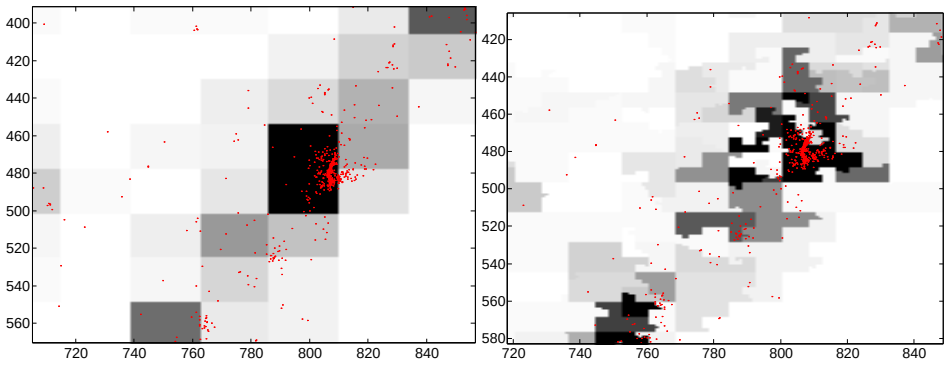
in Fig. 4.6(b).

To facilitate comparison, we treat the histograms as estimations of the underlying probability density function f, and use the statistical distance between density functions as a measure of utility. The value of $f(x)$ can be estimated by the ratio of the number of samples, over the width of the bin where x belongs to, with some normalizing constant factor.

In this section, we qualify the mechanism's utility by the distance between the two density functions: one that is derived from the original dataset, and the other that is derived from the mechanism's output.

Fig. 4.6(a) and 4.6(b) show the estimated density function from the Twitter's location dataset, by equi-width histogram mechanism and by our mechanism. For comparisons, 1% of the original points are plotted on top of the two reconstructed density functions. Fig. 4.7(a) and 4.7(b) show the zoom-in view of the dense region around New York City. Observe that the density function produced by our mechanism has "variable-sized" cells and thus is able to adaptively capture the fine details.

The statistical difference, measured with ℓ_1-norm and ℓ_2-norm, be-



(a) Zoom in view of Fig. 4.6(a). (b) Zoom in view of Fig. 4.6(b).

Figure 4.7: A more detailed view of the density functions.

tween the two estimated density functions derived from the original and the mechanism's output are shown in Table 2. We remark that it is not easy to determine the optimal bin-width for the equi-width histogram prior to publishing. Fig. 4.8 shows that the optimal bin-width differs significantly for three different datasets. For comparison purposes, we empirically choose the best parameters to the advantage of the compared algorithms, although such parameters could be dependent on the dataset.

4.6.2 Range Query

We consider the scenario where a dataset is to be published, and subsequently used to answer a series of range queries, where each range query asks for the total number of points within a query range. Publishing an equi-width histogram would not attain high accuracy if the size of the query ranges varies drastically. Intuitively, wavelet-based techniques (XWG10) are natural solutions to address such multi-scales queries. However, there are many parameters, including the bin-widths at various scales and the amounts of privacy budget they consume, to be determined prior to pub-

lishing.

To apply the proposed method in this scenario, given a query, we obtain the number of points within the range from the estimated density function (as described in Section 4.6.1) by accumulating the probability over the query region and then multiplying by the total number of points.

We compare the range query results of the wavelet-based mechanism, the equi-width histogram mechanism and our mechanism on the 1D Twitter data, and on the 2D Twitter location dataset. To incorporate the knowledge of the database's size n, the total number of points is adjusted to n for the histogram mechanism and the DC component of the wavelet transform is set to be exactly n for the wavelet mechanism. For each range query, the absolute difference between the true answer and the answer derived from the mechanism's output is taken as the error. We compare the results over different query range sizes and for each query range. For each range size s, 1,000 randomly chosen queries of size s are asked, and the corresponding errors are recorded. More precisely, the center of a 1D query range of size s is chosen uniformly at random in the continuous interval $[\frac{s}{2}, 1 - \frac{s}{2}]$, whereas the center of a 2D query range of size s is chosen uniformly at random in the region $[\frac{s}{2}, 1 - \frac{s}{2}] \times [\frac{s}{2}, 1 - \frac{s}{2}]$.

	equi-width	proposed method
ℓ_1-norm	1.23	1.13
ℓ_2-norm	0.25	0.20

Table 4.2: Statistical differences of the two methods.

To determine the parameters for the two compared mechanisms, we conduct experiments on a few selected values and choose the values to the

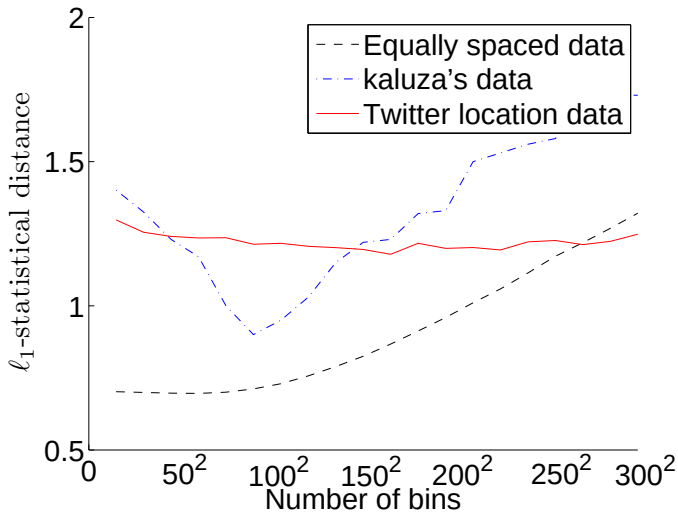
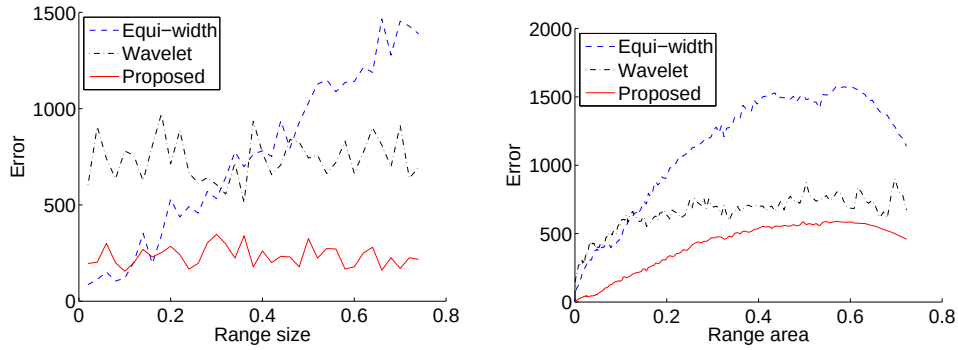


Figure 4.8: Optimal bin-width.

advantage of the compared mechanisms. For the equi-width histogram, the only parameter is the number of bins (n_1). For the wavelet-based mechanism, the parameter we considered is the number of bins (n_2) of the histogram whereby wavelet transformation is performed on, that is, the number of bins in the "finest" histogram. From our experiments, we choose $n_1 = 1000$ and $n_2 = 1024$ for the 1D data, and $n_1 = 40 \times 40$ and $n_2 = 512 \times 512$ for the 2D data. The parameter k for our mechanism is looked up from Table 4.1. The choice of group size k according to Table 4.1 is 177 ($n = 180,000, \epsilon = 1$). The average errors of the range query is shown in Fig. 4.9(a) and 4.9(b).

Observe that our proposed method is less sensitive to the query range in the 1D case as expected because the accuracy of our range query results depend only on the boundary points, as opposed to the equi-width histogram method where errors are induced by each bins within the range. The wavelet-based mechanism outperforms the equi-width histogram mechanism in larger size range queries, but performs badly for small range due



(a) 1D range query.

(b) 2D range query.

Figure 4.9: Comparison of range query performance.

to the accumulation of noise.

4.6.3 Median

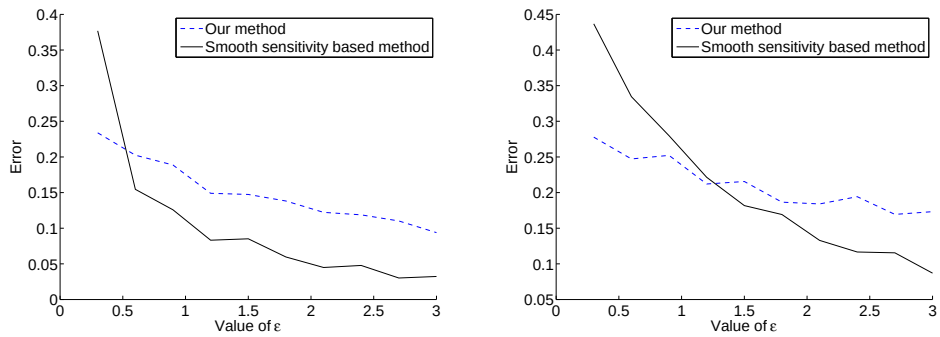
The median is an important statistic, and a differentially private median finding process can be useful in many constructions, such as in pointset spatial decomposition (CPS+12). However, finding the median accurately in a differentially private manner is challenging due to the high "global sensitivity": there are two datasets that differ by one element but having a completely different median. Nevertheless, for many instances, their "local sensitivity" are small. Nissim et al. (NRS07) showed that in general, by adding noise proportional to the "smooth sensitivity" of the database instance, instead of the global sensitivity, can also ensure differential privacy. They also gave an $\Theta(n^2)$ algorithm that find the smooth sensitivity w.r.t. median.

Our mechanism outputs the sorted sequence differentially privately, and thus naturally gives the median. Compare to the smooth sensitivity-based mechanism, our mechanism provides more information in the sense

that it outputs the whole sorted sequence. Furthermore, our mechanism can be efficiently carried out in $O(n \log n)$ time.

We conduct experiments on synthetic datasets of size 129 to compare the accuracy of both mechanisms. The experiments are conducted for different local sensitivity and different ϵ values. To construct a dataset with a particular local sensitivity, 66 random numbers are generated with the exponential distribution and then scaled to the unit interval. The dataset contains the 66 random numbers and 63 ones. Fig. 4.10(a) and 4.10(b) shows the noise level with different ϵ on datasets that has a local sensitivity of 0.1 and 0.3.

When the local sensitivity of the median is high, our mechanism tends to provide a better result. In addition, our mechanism performs well under higher requirement of security: when the ϵ is smaller, the accuracy of our mechanism decreases slower than the smooth sensitivity-based method.



(a) Local sensitivity of 0.1.

(b) Local sensitivity of 0.3.

Figure 4.10: The error of median versus different ϵ from two datasets.

4.7 Summary

In this chapter, we propose a mechanism that is very simple from the publisher's point of view. The publisher just has to sort the points, group consecutive values, add Laplace noise and publish the noisy data. There is also minimal tuning to be carried out by the publisher. The main design decision is the choice of the group size k, which can be determined using our proposed noise models, and the locality-preserving mapping for which the classic Hilbert curve suffices to attain high accuracy. Through empirical studies, we have shown that the published raw data contain rich information for the public to harvest, and provide high accuracy even for usages like median-finding and range-searching that our mechanism is not initially designed for.

Chapter 5

Data Publishing with Relaxed Neighbourhood

In this chapter, we will consider data publishing with relaxed differential privacy. The assurance provided by differential privacy comes with a cost of high noise, which leads to low utility of the published data. To address this limitation, several relaxations have been proposed. Many relaxations (DKM+06; MKA+08) capture alternative notions of "indistinguishability", i.e. how the probabilities on the two neighbouring datasets are compared by the utility function $\mathcal{U}$. We attempt to stay within the original framework while relaxing the privacy requirement by adopting a narrowed definition of neighbourhood, so that known results and properties still applied. That is, we consider a narrowed $\tilde{D}$.

5.1 Relaxed Neighbourhood Setting

Under the original neighbourhood (Dwo06; DMNS06) (let us call it the standard neighbourhood), two neighbouring datasets D_1 and D_2 differ by one entity, in that sense that $D_1 = D_2 - \{d_1\}$, or $D_1 = D_2 - \{d_1\} + \{d_1'\}$ for some d_1, d_1', in other words, D_2 differs from D_1 by either adding a new entity d_1 or replacing an entity d_2 by d_3. We propose considering a narrowed form of neighbourhood: instead of having arbitrary entity x and z, they have to meet some conditions. The new x must near to some "sources" and the replacement z must near to y within a threshold δ. Such neighbourhood naturally arise from spatial datasets, for example locations of Twister users (web) where the distance between two entities is the geographical distance between them. We called this narrowed variant *δ-neighbourhood*, where δ is the threshold.

There are a few ways to view the assurance provided by the proposed neighbourhood. First, note that if the domain (where the entities of the datasets are drawn from) is connected and bounded under the underlying metric, then a mechanism that is differentially private under δ-neighbourhood is also differentially private under the standard neighbourhood. However, the guaranteed bound (as in inequality (2.1)) is weaker when the entities are farther apart. Hence, the δ-neighbourhood essentially "redistributes" the indistinguishability assurance with emphasis on individuals that are close to each other, in a way similar to the original framework which stresses on datasets that are closer-by under set-difference.

Viewing from another perspective, one can treat this relaxation as

an added constraint on the datasets, so that not all datasets are valid. For example, locations of government service vehicles that are restricted in their bounded regions. When there is such an implicit constraint on the dataset, the two notions of neighbourhood are equivalent. Illustrating examples will be discussed in Section 5.3 and 5.5.

The δ-neighbourhood can also be adopted for dynamic datasets where entities are added and removed over time. One example is the scenario considered by Dwork et al. (DNPR10), where aggregated information on users' health conditions in a region or building (say airport) are to be monitored over time. Under the standard neighbourhood, due to the fixed budget, it is impossible to publish the dataset repeatedly with high utility. However, there are scenarios where entities do not stay in the dataset for long and thus, intuitively, the effect of information published earlier would diminish over time, and hence we should be able to continuously publish with high utility. We can define a δ-neighbourhood that captures the observation, so as to achieve *sustainable privacy* on dynamic datasets.

Existing differential private mechanisms designed for the standard neighbourhood naturally are also differentially private under the δ-neighbourhood. Some mechanisms, for example, smooth sensitivity based median publishing (NRS07), can be easily modified to achieve higher utility. For some applications, it is not clear how to exploit the relaxation to achieve higher utility. For publishing of histogram on spatial dataset, we propose a few constructions and show that the utility can be significantly improved. Whereas for dynamic dataset, we investigate how the budget is to be allo-

cated in an offline and online setting.

5.2 Formulations

5.2.1 δ-Neighbourhood

We assume that there is a distance function $d : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ on the domain that captures the distance between a pair of entities. We also assume that there is a set of *sources* $S \subseteq \mathcal{M}$. With this distance function and sources, for a threshold δ, we say that two datasets D_1, D_2 are δ-neighbours if, and only if the following holds:

1. there exists x_1 and $x_2 \in \mathcal{M}$, such that $d(x_1, x_2) \leq \delta$, and $D_1 - \{x_1\} = D_2 - \{x_2\}$, or

2. there exists a x_3 and $s \in S$ s.t. $d(x_3, s) \leq \delta$ and $D_1 - \{x_3\} = D_2$.

In other words, either D_1 can be obtained from D_2 by replacing an entity x_2 with a nearby entity x_1, or by adding a new entity x_3 emerged near a source s. Note that if S is empty, then the size of the size of D_1 and D_2 must be the same.

Given two datasets $D_1, D_2 \in \mathbf{D}$, we say that D_1 and D_2 are *connected* if there exists a finite sequence of $E_0, E_1, E_2, \ldots, E_m$ with $E_0 = D_1$ and $E_m = D_2$ s.t. for any i, the consecutive E_i and E_{i+1} are δ-neighbours, and call the smallest such m the distant between D_1 and D_2. If any two datasets in $\mathbf{D}$ are connected, we say that $\mathbf{D}$ is connected, and called the least upper bound on the distance, if it exists, the *diameter* of $\mathbf{D}$.

5.2.2 Differential Privacy under δ-Neighbourhood

Now, we say that a mechanism $\mathcal{A}$ is ϵ-differential privacy under δ-neighbourhood if for all $R \subseteq \text{range}(\mathcal{A})$ and any pair of δ-neighbours (D_1, D_2), we have:

$$Pr(\mathcal{A}(D_1) \in R) \leq \exp(\epsilon) \cdot Pr(\mathcal{A}(D_2) \in R). \tag{5.1}$$

Similar to the definitions with standard neighbourhood, we can define the sensitivity of a function $f\colon \mathbf{D} \to \mathbb{R}$ with respect to the δ-neighbourhood, which is

$$\sup \|f(D_1) - f(D_2)\|_1,$$

where the supremum is taken over all pairs (D_1, D_2) of δ-neighbours.

5.2.3 Properties

Since δ-neighbours are also neighbours under the standard neighbourhood, thus a ϵ-differentially private mechanism under standard neighbourhood is also ϵ-differential private mechanism under δ-neighbourhood. The converse also holds but with a weaker bound, as stated in the following lemma:

Lemma 3 *If a mechanism $\mathcal{A}$ is ϵ-differential private under the δ-neighbourhood and the diameter of $\mathbf{D}$ is d, then it is $(d\epsilon)$-differential private under the standard neighbourhood.*

Proof If a mechanism $\mathcal{A}$ is ϵ-differential private under δ-neighbourhood, then for any pair of neighbouring datasets $D_1, D_2 \in \mathbf{D}$ under standard neighbourhood, we can find a sequence $E_0 = D_1, E_2, \ldots, E_m = D_2$ s.t. E_i

and E_{i+1} are δ-neighbours for $i = 0 \ldots m-1$ for some $m \leq d$. Therefore, for any $R \subseteq Range(\mathcal{A})$, we have:

$$Pr(\mathcal{A}(E_{i-1}) \in R) \leq \exp(\epsilon) \cdot Pr(\mathcal{A}(E_i) \in R), \text{ for } i = 1 \ldots m.$$

Combining the inequalities, we have:

$$Pr(\mathcal{A}(E_0) \in R) \leq \exp(m\epsilon) \cdot Pr(\mathcal{A}(E_m) \in R).$$

Since $E_0 = D_1$ and $E_m = D_2$, therefore the mechanism $\mathcal{A}$ is $(d\epsilon)$-differential private.

Sequential composition: The composition of two differentially private mechanisms is also differentially private. It is easy to show that this property also holds under δ-neighbourhood: given a sequence of k mechanisms, $\mathcal{A}_1, \mathcal{A}_2, \ldots \mathcal{A}_k$, where $\mathcal{A}_i$ is ϵ_i differentially private under δ-neighbourhood, then the mechanism

$$\mathcal{A}^*(D) = \mathcal{A}_1(D, \mathcal{A}_2(D, \ldots))$$

is $(\sum_{i=1}^{k} \epsilon_i)$-differentially private under δ-neighbourhood.

5.3 Construction for Spatial Datasets

The δ-neighbourhood can be naturally defined on spatial points, say $\mathcal{M} = [0,1]^k$ for some $k \geq 1$. We are only interested in low dimensions, for example $k = 1$ or 2, since it is generally very difficult to achieve high utility for high dimensional data. The underlying distance function $d(\cdot, \cdot)$ can be the Euclidean distances and the sources can be the boundary of $\mathcal{M}$,

which implies that entities enter through the boundary, or simply none, corresponding to the bounded differential privacy. Let us investigate a few scenarios where the proposed notion is meaningful.

5.3.1 Example 1

Consider a situation where the dataset is constrained, in the sense that not all multisets of entities from $\mathcal{M}$ is in $\mathbf{D}$ (recall that $\mathbf{D}$ is the set of all possible datasets). Let us call the multisets that are not in $\mathbf{D}$ *invalid datasets*. If those invalid datasets are captured by the restriction on δ-neighbourhood, then essentially the two assurances, either under standard neighbourhood or δ-neighbourhood, are equivalent. For example, consider a D containing the locations of a cab sampled at periodic intervals, say at time $1, 2, \ldots, n$, and is represented as a set of tuples where each tuple (t, x) indicates that the cab is at location x on time t. Suppose D is to be published by a mechanism $\mathcal{A}$ that is ϵ-differentially private under the standard neighbourhood, then for any possible output r, any D, (t, x) and (t, y), we have

$$Pr(\mathcal{A}(D + \{(t, x)\}) = r) \leq exp(\epsilon) Pr(\mathcal{A}(D + \{(t, y)\}) = r).$$

Recall that the above bound is required to hold only for datasets $D_1 = D + \{(t, x)\}$ and $D_2 = D + \{(t, y)\}$ in $\mathbf{D}$. Since each t is associated with a unique tuple, we only need to consider replacing (t, x) in D_1 by another tuple with the same time (t, y).

We can take a step further. Due to speed limit of the cabs (which is public knowledge), some datasets are invalid. For example, if D_1 is a

valid dataset, a location y that is far from x will lead to an invalid dataset. Since the bound is not required to hold for the invalid datasets, thus, with an appropriate metric and a sufficiently large δ, the assurance provided under δ-neighbourhood is equivalent to the assurance provided under the standard neighbourhood.

5.3.2 Example 2

Consider a dataset D that contains locations of entities belonging to a particular group G, and is published ϵ-differential privately under the standard neighbourhood. An analyst wants to combine some background information with the published data to infer whether a particular entity, say Bob, is in that group G. From the background information, the analyst obtained a set K of possible entities in the group and their corresponding locations, and Bob's information is also in the list K. If D is published in clear, Alice could check where Bob's location is in D and inferred with high confidence of Bob's membership in G. However, D is published with differential privacy under standard neighbourhood. Hence, the published data does not distinguish Bob's location with any other entity's in K, and thus from Bob's perspective, his privacy is preserved.

Now, Bob might be contended with a weaker assurance that, the published data does not distinguish him from any entity in K who are located 50 km near him. This is reasonable as Bob is quite sure that there are entities similar to him w.r.t the background knowledge within 50 km (that is, there exists an entity in the set K who is within 50 km). This

assurance is captured by the δ-neighbourhood with $\delta = 50$ km.

Consider another more resourceful analyst who has more accurate background information on region near Bob. With respect to this background information, the nearest entities similar to Bob is 100 km away. In this case, even if D is published by a ϵ-differential private mechanism under 50 km-neighbourhood, Bob's privacy is still protected but with a weaker assurance similar to a 2ϵ-differential privacy under the standard neighbourhood.

5.3.3 Example 3

Let us revisit the scenario in Example 2 and consider another data contributor Alice. Alice is an outlier in her region, and the background information leads to a set of possible entities who are located far, say at 500 km, from Alice. Although the data are published under 50 km-neighbourhood, there is still protection on Alice's privacy, but with a weaker (e.g. equivalent to 10ϵ differentially private) assurance.

Hence, we can also view the δ-neighbourhood as a redistribution of "protection" that provides more protection to entities who are "typical" in their proximity, but lesser protection to entities who are outlier in their proximity.

5.4 Publishing Spatial Dataset: Range Query

In this section, we consider publishing the histogram of a spatial dataset differential privately, so that subsequent range queries can be accurately

answered. Although there are many known methods on publishing histograms, it is not clear how the restriction imposed by δ-neighbourhood can be exploited. We consider approaches that publish a linearly transformed histogram. The sensitivity incurred by transformations under δ-neighbourhood contains an interesting combinatoric structure that is not present in the standard neighbourhood, which can be exploited to improve accuracy. We also extend Theorem 1 of the pointset publishing method described above in Chapter 4 to capitalize the bounded neighbourhood. We show that the extension achieves high utility but it is not clear how to generalize the method to 2 and higher dimensions.

5.4.1 Illustrating Example

Let us demonstrate how to capitalize the notion of δ-neighbourhood with the following simple example in 1D. Consider a dataset containing (possibly with repetitions) 4 possible values: $\{\frac{1}{4}, \frac{2}{4}, \frac{3}{4}, 1\}$. Let c_i be the number of points with value $i/4$. Table 5.1 gives an 1-differentially private mechanism under the standard neighbourhood that publishes the counts (c_1, c_2, c_3, c_4). This mechanism is also 1-differentially private under 0.25-neighbourhood.

Now let us publish the counts as shown in Table 5.2, where a linear transformation is applied before adding noise. Our main observation is that, the sensitivity of publishing the values (a_1, a_2, a_3, a_4) is 1 with respect to the 0.25-neighbourhood: a change of a single entity by a distance of 0.25 affects only a_i for some i. Hence, a Laplace noise of $Lap(1)$ is sufficient to guarantee 1-differential privacy under 0.25-neighbourhood. However,

under the standard neighbourhood, an entity changing from value $\frac{1}{4}$ to 1 will decrease each a_1, a_2, a_3 by 1, leading to a sensitivity of 3.

By publishing the a_i''s in Table 5.2, we can answer range queries with higher accuracy through linear combination of the a_i's. For example, when a query asks for the frequency counts in the range $[0.4, 0.6]$, reporting the value c_2' leads to an unbiased estimator with variance 8, which is the variance of $Lap(2)$. On the other hand, from Table 5.2, it can be estimated by $a_2' - a_1'$ giving an unbiased estimator with a smaller variance of 4, which is the variance of the sum of two independent Laplace noises, $Lap(1) + Lap(1)$. Such difference is more significant for larger query range. The comparisons are shown in Table 5.3: row i of the table shows the noise variances when the query range covers exactly i number of the counts c_i's.

Table 5.1: Publishing c_i's directly.

Actual Values	Published values
c_1	$c_1' = c_1 + Lap(2)$
c_2	$c_2' = c_2 + Lap(2)$
c_3	$c_3' = c_3 + Lap(2)$
c_4	$c_4' = c_4 + Lap(2)$

Table 5.2: Publishing a linearly transformed histogram.

Actual values	Published values
$a_1 = c_1$	$a_1' = a_1 + Lap(1)$
$a_2 = c_1 + c_2$	$a_2' = a_2 + Lap(1)$
$a_3 = c_1 + c_2 + c_3$	$a_3' = a_3 + Lap(1)$
$a_4 = c_1 + c_2 + c_3 + c_4$	$a_4' = a_4 + Lap(1)$

By exhaustive checking, it can be verified that, in terms of minimizing the total variance of all possible range queries, i.e. the weighted sum

Table 5.3: Variance of the estimator for different range size.

Number of c_i's covered	Number of possible range queries	Derived from Table 5.1	Derived from Table 5.2
1	4	8	4
2	3	16	4
3	2	24	4
4	1	32	2

of the variance in the rightmost column with the weights in the second column in Table 5.3, the construction in Table 5.2 is optimal among all linear combinations of c_1, c_2, c_3 and c_4 where the coefficients are binary, i.e. either 0 or 1.

However, note that the above methods estimate the query results using linear combinations of the published values. One could enforce the constraints that all c_i's are non-negative, leading to a non-linear estimator. Although this may create bias, it could lower the variance of the estimation.

5.4.2 Generalization of Illustrating Example

The method shown in Table 5.1 corresponds to the direct method of adding noise to the frequency counts of an equi-width histogram, whereas Table 5.2 corresponds to a method that applies a linear transformation before adding noise. Li et al. (LHR+10) studied such general form of publishing under the standard neighbourhood. In this section, we extend it to δ-neighbourhood. As illustrated in the example, the key difference of our method is the lower sensitivity incurred under δ-neighbourhood.

Formally, a histogram $\mathcal{H}_B(D)$ for a partition of the domain $B = \{b_1, \ldots, b_k\}$ on D gives a column vector of frequency counts $\mathbf{c} = (c_1, \ldots, c_k)^t$

where $c_i = |D \cap b_i|$. We call each set in the partition B a bin. In particular, an equi-width histogram corresponds to a partition whose bins are of the same size. Since all the bins do not overlap, the effect of replacing an entity in D affects frequency counts in at most two bins, and thus the sensitivity of $\mathcal{H}_B(\cdot)$ is 2 under the standard neighbourhood. Hence the mechanism of publishing $\mathbf{c} + Lap(2/\epsilon)^k$ is ϵ-differential private under the standard neighbourhood.

We consider queries whose answers are linear combination of counts in $\mathbf{c}$, and can be expressed as $\mathbf{qc}$ where $\mathbf{q}$ is a row vector. For example, a range query can be a summation of counts in selected bins. For a sequence of m queries, let us express it as an m by k matrix $\mathbf{Q}$ and hence the answer to the queries are the coefficients in $\mathbf{Qc}$. As proposed by Li et al., to answer the query $\mathbf{Q}$, one may employ a *strategy* $\mathbf{A}$, which is represented as a k by n matrix, and publish

$$\widetilde{\mathbf{c}} = \mathbf{Ac} + Lap(\triangle_{\mathbf{A}}/\epsilon)^n,$$

where $\triangle_{\mathbf{A}}$ is the sensitivity of the function that on input D, returns $\mathbf{Ac}$. From the published $\widetilde{\mathbf{c}}$, we want to estimate the query results. It can be shown (Sil75) that the following estimate is unbiased:

$$\mathbf{A}^+\widetilde{\mathbf{c}},$$

where $\mathbf{A}^+ = (\mathbf{A}^t\mathbf{A})^{-1}\mathbf{A}^t$ is the pseudo-inverse of $\mathbf{A}$, and the variance of the estimator is

$$(\triangle_{\mathbf{A}})^2 trace(\mathbf{Q}(\mathbf{A}^t\mathbf{A})^{-1}\mathbf{Q}^t). \tag{5.2}$$

Now, given $\mathbf{Q}$, we want to find the $\mathbf{A}$ s.t. the variance is minimized. In the

illustrating example, $\mathbf{Q}$ is a 10 by 4 matrix where each row corresponds to a range queries, and

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \tag{5.3}$$

As discussed by Li et al. (LHR+10), solving the optimization problem in general is difficult for standard neighbourhood, partly due to the fact that the sensitivity $\triangle_{\mathbf{A}}$ as a function of $\mathbf{A}$, is non-differentiable. Likewise it is difficult for δ-neighbourhood. Nevertheless, the formulation provide a guideline in determining a good strategy.

5.4.3 Sensitivity of A

The sensitivity of $\mathbf{A}$ under δ-neighbourhood leads to an interesting combinatoric structure that is not present in the standard neighbourhood. Under the standard neighbourhood, the sensitivity of $\mathbf{A}$ is

$$\max_{i,j \in \mathbb{Z}_n} \{ \ \|\mathbf{a}_i - \mathbf{a}_j\|_1 \ \}, \tag{5.4}$$

where each $\mathbf{a}_i$'s is a column vector in $\mathbf{A}$, that is, $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n]$. To understand the above expression, note that $\|\mathbf{a}_i - \mathbf{a}_j\|_1$ is the sum of L_1 difference when an entity change between bin i and bin j. Since the sensitivity is the least upper bound on all possible pairs of neighbouring datasets, we have the above expression.

Under the δ-neighbourhood, the sensitivity of $\mathbf{A}$ is

$$\max_{(i,j)\in N} \{ \ \|\mathbf{a}_i - \mathbf{a}_j\|_1 \ \},$$

where N is a set induced from the requirement on δ-neighbourhood,

$$N = \{(i,j) \mid \exists\ x \in b_i, y \in b_j,\ s.t.\ d(x,y) \leq \delta\}.$$

Compare to the expression in (5.4), the maximum is taken over a smaller set N and thus could be smaller.

For the matrix $\mathbf{A}$ in the illustrating example, we have $N = \{(1,2),(2,3),(3,4)\}$ under 0.25-neighbourhood, and thus the sensitivity of $\mathbf{A}$ is 1; whereas the sensitivity under the standard neighbourhood is 3, as $\|\mathbf{a}_1 - \mathbf{a}_4\|_1 = 3$.

Graphical representation: We can capture the relationship between the sensitivity of $\mathbf{A}$ and N using a graph when the entries in $\mathbf{A}$ are binary, i.e. either 0 or 1. Let us treat each bin as a vertex in the graph. Hence there are k vertices $v_1, v_2, \ldots, v_k$. There is an edge between two vertices v_i and v_j iff $(i,j) \in N$.

For a matrix $\mathbf{A}$, since the entries are binary, each row corresponds to a subset of bins. Hence, $\mathbf{A}$ can be viewed as a collection of sets $\{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m\}$ where each set in $\mathbf{A}$ is a set of bins. For an edge (v_i, v_j), we say it is being *cut* by a set $\mathbf{a}$ iff

$$(v_i \in \mathbf{a} \wedge v_i \notin \mathbf{a}) \ \vee \ (v_i \notin \mathbf{a} \wedge v_i \in \mathbf{a}).$$

For each edge e, let us call the number of sets in $\mathbf{A}$ that cut the edge e the number of cuts on e, denoted as $C(e)$. Now, the sensitivity of $\mathbf{A}$ is the maximum number of cuts over all edges, i.e. $\max_e C(e)$.

Note that given a particular $\mathbf{A}$ of sensitivity 1, it may be possible to insert a set into $\mathbf{A}$ without increasing its sensitivity. That is, it may be possible to find a subset that only cuts edges that have not been cut by subsets in $\mathbf{A}$. Since sensitivity is not increased, it would not hurt to add this set into $\mathbf{A}$ which in turn publishes this extra information[1]. This observation leads to a simple greedy algorithm that improves a strategy: simply add rows to $\mathbf{A}$ until it is not possible to do so without increasing the sensitivity.

For instance, consider a 2D histogram with bins $\{b_{i,j} \mid i, j \in \mathbb{Z}_n\}$, with neighbourhood $N = \{((i,j),(i',j')) \mid |i-i'| + |j-j'| \leq 1\}$ as shown in Figure 5.1 where each bin is a bullet(blue), and the neighbours are connected by a dotted(red) line. Consider the set $\mathbf{A}$ that contains $\mathbf{a}_{i,j} = \{b_{2i-1,2j-1},\ b_{2i-1,2j},\ b_{2i,2j-1},\ b_{2i,2j}\}$, for $i, j = 1, 2, \ldots \frac{n}{2}$, that is, each $a_{i,j}$ is a dash(black) square that contains four blue vertices. Note that the dash(black) squares do not "cut" all the neighbouring edges, and therefore, if we adds $\mathbf{a}'_{i,j} = \{b_{2i,2j},\ b_{2i,2j+1},\ b_{2i+1,2j},\ b_{2i+1,2j+1}\}$ to $\mathbf{A}$, for $i, j = 1, 2, \ldots \frac{n}{2} - 1$, (i.e. the solid(blue) squares containing 4 vertices each), the sensitivity remains the same.

5.4.4 Evaluation

1D range query

The earlier example described in Section 5.4.1 can be generalized to publish linear transformation of histograms with n bins. The transformation $\mathbf{A}$ is a

[1] One may see this from expression (5.2), where adding a row to $\mathbf{A}$ without increasing $\triangle_{\mathbf{A}}$ will not increase the variance.

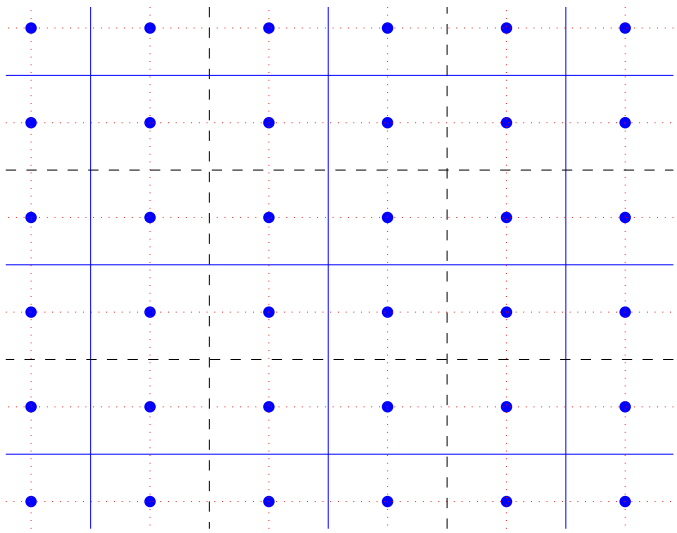


Figure 5.1: Demonstration of adding a' to $\mathbf{A}$ without increasing sensitivity.

lower triangular matrix of size $n \times n$ and the entries on and below diagonal are 1. Essentially, row i of $\mathbf{A}$ cumulates the counts for bin 1 to bin i. Let us call this strategy C_n. The answer to a range query that covers bin i to j can be obtained by subtracting the j-th row and $(i-1)$-th row. We are interested in how accurate C_n performs in answering 1D range queries, i.e. in answering the set of all range queries, $\mathbf{Q}$.

Li et al. (LHR+10) consider the maximum error and total error of three strategies: H_n which queries a series of equi-width histograms (HRM-S10), Y_n which is a Haar wavelet transformation matrix (XWG10) and the identity matrix I_n. Figure 5.2 shows H_4, Y_4 I_4 and C_4. The maximum errors refers to the maximum variance among all row vectors of $\mathbf{Q}$, and total errors refers to the sum of the variance. The errors of H_n, Y_n and I_n are as shown in Table 5.4. The constructions do not exploit δ-neighbourhood, and the errors of H_n, Y_n and I_n are the same under either standard neigh-

bourhood or δ-neighbourhood.

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 1 & -1 \\ 1 & -1 & 0 & 0 \\ 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$$H_4 \qquad Y_4 \qquad I_4 \qquad C_4$$

Figure 5.2: Strategy H_4, Y_4, I_4 and C_4.

C_n benefits from δ-neighbourhood, in the sense that the sensitivity $\triangle_{C_n}$ is lower for smaller δ. The corresponding maximum error and total error of $C_{n,\delta}$ is also shown in Table 5.4. When $\delta = n$, it performs similar to identity matrix, but when δ is small, we can reduce the errors by exploiting the δ-neighbours.

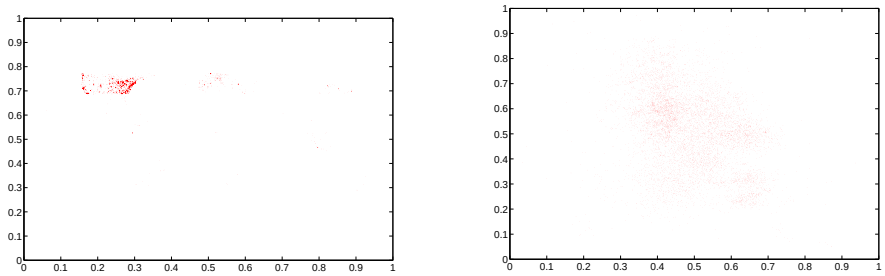
Table 5.4: Max and total errors.

	H_n	Y_n	I_n	$C_{n,\delta}$
max error	$\Theta(\frac{log^3 n}{\epsilon^2})$	$\Theta(\frac{log^3 n}{\epsilon^2})$	$\Theta(\frac{n}{\epsilon^2})$	$\Theta(\frac{\delta}{\epsilon^2})$
total error	$\Theta(\frac{n^2 log^3 n}{\epsilon^2})$	$\Theta(\frac{n^2 log^3 n}{\epsilon^2})$	$\Theta(\frac{n^3}{\epsilon^2})$	$\Theta(\frac{n^2\delta}{\epsilon^2})$

2D range query

We consider mechanisms that answer 2D range queries with fixed range size on a datasets D where the domain is $[0,1)^2$. A 2D range query of size s asks for the number of points in the region $[x-\frac{s}{2}, x+\frac{s}{2}) \times [y-\frac{s}{2}, y+\frac{s}{2})$. We derive an algorithm as described in Section 5.4.3, we compare our algorithm with the equi-width histogram.

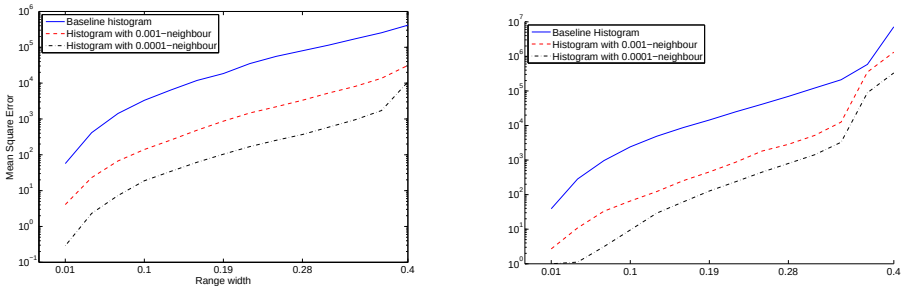
An equi-width histogram in 2D correspond to the partition $B =$



(a) Randomly selected 5% of points from Dataset 1.

(b) Randomly selected 5% of points from Dataset 2.

Figure 5.3: The 2D location datasets.



(a) Dataset 1.

(b) Dataset 2.

Figure 5.4: The mean square error of range queries in linear-logarithmic scale.

$\{b_{1,1}, b_{1,2}, \ldots, b_{k,k}\}$, where each bin $b_{i,j}$ is a square region $[\frac{i-1}{k}, \frac{i}{k}) \times [\frac{j-1}{k}, \frac{j}{k})$. Let $\widetilde{c}_x$ be the published frequency counts in bin b_x.

Given a range query q, we estimate the answer to q as:

$$\sum_{b_x \in B} \left(\frac{|b_x \cap q|}{|b_x|} \; c_x \right). \tag{5.5}$$

where $|b_x|$ is the area of b_x. Note that if the query partially intersect with a bin, that bin contributes proportionally to the answer.

Our strategy is constructed as illustrated by Figure 5.1, where a series of equi-width histograms is to be published. Each histogram is shifted by an offset δ from the previous histogram in the series. Specifically, let $B_0, B_1 \ldots B_{m-1}$ be the partitions correspond to the histograms, where $m = \lceil \frac{1}{k\delta} \rceil$ and B_x is a partition $\{b^x_{1,1}, b^x_{1,2}, \ldots, b^x_{k+1,k+1}\}$ with each $b^x_{i,j}$ is a square

region $[\frac{i-2}{k} + x\delta, \frac{i-1}{k} + x\delta) \times [\frac{j-2}{k} + \delta, y + \frac{j-1}{k} + x\delta)$.

To answer a range query q, the estimation in equation (5.5) is used. Note that a Laplace noise of $Lap(4)$ is sufficient to guarantee ϵ-differential privacy under δ-neighbourhood.

We conduct experiments on two 2D datasets. Dataset 1 contains the locations of Twitter users in the world (web) The dataset contains over 193,841 Twitter users' data from the period of March 2006 to March 2010. Dataset 1 (KMD$^+$10) contains 164,860 tuples collected from tags that continuously record the location information of 5 individuals. We normalize the data points to the space $\mathcal{M} = [0, 1]^2$, and Figure 5.3(a) and Figure 5.3(b) shows 5% of the points randomly selected from the respective datasets.

We consider two cases where $\delta = 0.001$ and $\delta = 0.0001$, which translate to a bound of 40 and 4 kilometers for dataset 1 respectively.

For comparison purpose, we empirically choose the optimal bin width for each query range, as shown in Table 5.5. Figure 5.4 shows the details of the experiment result.

Table 5.5: Query range and corresponding best bin-width for the Dataset 1.

	Query range	Best k	Mean Square Error
Q1	0.001	0.001	1.6991
Q2	0.01	0.01	39.146
Q3	0.1	0.025	2411.7
Q3	0.2	0.025	14434
Q4	0.4	0.025	716068

5.5 Construction for Dynamic Datasets

We now investigate publishing of dynamic datasets. We consider situations where information on entities are collected periodically over time, say at discrete time $1, 2 \ldots$ Occasionally, statistics are to be published. Intuitively, with limited budget, it is impossible to continuously publish meaningful information indefinitely, in fact, Dwork et al. (DNPR10) shown a negative result under a setting that captures this intuition. However, in some scenarios, the entities are not required to contribute at all collection times, and is likely to leave within a short period. With such restriction, it should be now possible to continuously publish with low noise indefinitely, as effect of information contributed earlier would diminish in time.

5.5.1 Publishing Dynamic Datasets

Formally, let a sequence $x_1, x_2, \ldots$ be the data contributed by an entity, where each $x_i \in \mathbf{U} + \{\perp\}$ is the data contributed at time i, with $\mathbf{U}$ being the domain of the contributed data, and $\perp$ being a special symbol indicating that the entity is not contributing at that time. Let us call a sequence containing only the symbol $\perp$ a *null sequence*. A dataset D is a set of the aforementioned sequences. We assume that every entity in D has contributed a data in $\mathbf{U}$ at some time, and thus D does not contain null sequence. The prefix of a sequence x contains data contributed by the entity up to time n, denoted $x_{[1..n]}$, where n is the length of the prefix. Let us denote $D_{[1..n]}$ the set of such prefixes in D that are not null sequence. In addition, denote D_n the set of all n-th elements of the sequences in D

that is not $\perp$, that is, D_n contains all data contributed at time n.

At certain time, say time t, some information on D_t is to be published. WLOG, we assume that information is published at every time, and let $\mathcal{A}_i$ be the publishing mechanism employed at time i. Hence, the data published are $\mathcal{A}_1(D_1), \mathcal{A}_2(D_2), \ldots$. Combining all the data published on and before time n, we can treat the whole process of applying mechanisms $\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_n$ as a single mechanism $\mathcal{A}_n^*$ that operates on $D_{[1..n]}$.

5.5.2 δ-Neighbour on Dynamic Dataset

Given two datasets, D and D', under the standard neighbourhood, they are neighbours if, and only if they differ by one entity. That is, there is a sequence x and y s.t. $D + \{x\} = D'$, or $D + \{x\} = D' + \{y\}$. This is essentially the same notion of neighbourhood for *user-level privacy* studied by Dwork et al. (DNP$^+$10; DNPR10).

For two sequences $\mathbf{x} = \langle x_1, x_2, \ldots \rangle$ and $\mathbf{y} = \langle y_1, y_2, \ldots \rangle$, let us define $d(\mathbf{x}, \mathbf{y})$ to be the value $i_s - i_t$ where i_s is the smallest index s.t. $x_{i_s} \neq y_{i_s}$ and i_t is the largest index s.t. $x_{i_t} \neq y_{i_t}$. That is, it is the length of the smallest consecutive subsequence that contains all the differences. We take the null sequence as the source. Hence, D and D' are δ-neighbourhood if, and only if $D + \{\mathbf{x}\} = D'$, or $D + \{\mathbf{y}\} = D' + \{\mathbf{z}\}$, for some $\mathbf{y}, \mathbf{z}$ s.t. $d(\mathbf{y}, \mathbf{z}) \leq \delta$, or some $\mathbf{x}$ s.t. $d(\mathbf{x}, \hat{\perp})$ where $\hat{\perp}$ denotes the null sequence.

5.5.3 Example 1

One situation where publishing dynamic dataset can benefit from δ-neighbourhood is when dataset is constrained, in the sense that the sensitive information of entities only last for a short period. Consider a regional flu response organization may want to continuously collect daily information on the health conditions of visitors, and release the information occasionally. Alice, wants to infer whether Bob has been to the region based on the released information. If the publishing mechanism $\mathcal{A}$ is ϵ-differential privacy, then Alice's inference is bounded by:

$$Pr(\mathcal{A}(D_0 + \{\mathbf{x}\}) \in R) \leq exp(2\epsilon)Pr(\mathcal{A}(D_0) \in R),$$

where $\mathbf{x}$ is the health information of Bob. If all visitors must leave within 14 days, then $\mathbf{x}$ must near the source, i.e. $d(\mathbf{x}, \perp) < 14$ days, otherwise the dataset is invalid. Hence, under this constraint on the datasets, the guarantee under the stand neighbourhood and δ-neighbourhood are equivalent.

5.5.4 Example 2

Let us revisit Example 1. Suppose the authority allows some visitors to stay for a longer period, say 28 days, even if the dataset is published under 14-neighbourhood, there is still protection. If Bob indeed stayed for 28 days, the bound is relaxed to $exp(2\epsilon)$. Hence, similar to the spatial datasets, the protection is being redistributed with more protection to entities with shorter stay.

5.6 Sustainable Differential Privacy

If each mechanism A_i is ϵ-differentially private under either notions of neighbourhood, then the mechanism A_n^* is $(n\epsilon)$-differentially private under the respective neighbourhood. However, for δ-neighbourhood, we should able to "reuse" the budget spent on much earlier published data. This observation is formulated in the following theorem:

Theorem 4 *Let D be a dynamic dataset with the mechanism $\mathcal{A}_n^*$, $\mathcal{A}_1$, $\mathcal{A}_2$, $\ldots \mathcal{A}_n$ as defined above in Section 5.5. If the mechanism $\mathcal{A}_i$ is ϵ_i-differentially private under the standard neighbourhood for each $i \in \{1, \ldots, n\}$, and*

$$\sum_{i=1}^{\delta} \epsilon_{k+i} \leq \epsilon, \quad \textit{for } k \in \{0, 1, \ldots, (n-\delta)\},$$

then $\mathcal{A}_n^*$ is ϵ-differentially private under δ-neighbourhood.

Proof Consider two datasets D and D', where $D' + \{\mathbf{y}\} = D + \{\mathbf{x}\}$ and $d(\mathbf{x}, \mathbf{y}) \leq \delta$. Let i_s be the smallest index at which $\mathbf{x}$ and $\mathbf{y}$ differ. Consider an output $\mathbf{a} = \langle a_1, a_2 \ldots a_n \rangle$ of $\mathcal{A}_n^*(D)$, we have the probability that $\mathcal{A}_n^*$ gives the same output on dataset D' as:

$$\begin{aligned}
& Pr(\mathcal{A}_n^*(D') = r) = \prod_{i=1}^{n} Pr(\mathcal{A}_i(D_i') = a_i) \\
\leq \; & \left(\prod_{i=i_s}^{i_s+\delta-1} exp(\epsilon_i) \right) \cdot \left(\prod_{i=1}^{n} Pr(\mathcal{A}_i(D_i) = a_i) \right) \\
= \; & exp\left(\sum_{i=i_s}^{i_s+\delta-1} \epsilon_i \right) \cdot Pr(\mathcal{A}_n^*(D) = r) \\
\leq \; & exp(\epsilon) Pr(\mathcal{A}_n^*(D) = r)
\end{aligned}$$

Similarly argument holds for any pair D and D' where $D' = D + \{x\}$ and x is near the source. Therefore, $\mathcal{A}_n^*$ is ϵ-differentially private under δ-neighbourhood.

5.6.1 Allocation of Budget

The privacy requirement ϵ is also called *privacy budget* as it can be divided and allocated to a few mechanisms, and yet the composition of these mechanisms still meet the ϵ requirements. With respect to a mechanism, different budget leads to different level of error introduced by the mechanisms. We assume that there is real valued function $\mathcal{E}rr(\cdot)$ associated with a mechanism, that gives the error of the mechanism in term of the budget. For instance, the error of the Laplace mechanism is often taken as the expected mean square error, that is, $\frac{1}{\epsilon^{-2}}$.

Theorem 4 states the condition on ϵ_i's for $\mathcal{A}_n^*$ to be differentially private. Note that there are many ways to allocate the ϵ_i's and yet the condition is meet, and different allocations give different total error. In this section, we focus on finding a good budget allocations with an objective of minimizing the total error subjected to the condition in given by Theorem 4.

Let $\mathcal{E}rr_i(\cdot)$ to be error function of the mechanism A_i. In particular, we are interested in error function of the form: $\mathcal{E}rr_i(\epsilon) = w_i\epsilon^{-2}$ where w_i is some non-negative weight indicating the level of significance of the published data.

We study two settings: the offline setting where all w_i's are known,

and online setting where the value of w_i is only known at time i and the budget ϵ_i has to be committed before time $i+1$.

5.6.2 Offline Allocation

The offline allocation problem is as follow:

Problem 1 Offline Budget Allocation

Given:	$\delta \in \mathbb{Z}_n, \mathbf{w} = \langle w_1 \dots w_n \rangle \in \mathbb{R}^n_{\geq 0}$
Find:	$\langle \epsilon_1, \epsilon_2, \dots, \epsilon_n \rangle$
Minimize:	$\sum_{i=1}^{n} \mathcal{E}rr_i(\epsilon_i) = \sum_{i=1}^{n} w_i \epsilon_i^{-2}$
Subject to:	$\sum_{i=1}^{\delta} \epsilon_{k+i} \leq \epsilon,$ for $k = 1, 2, \dots, (n-\delta)$.

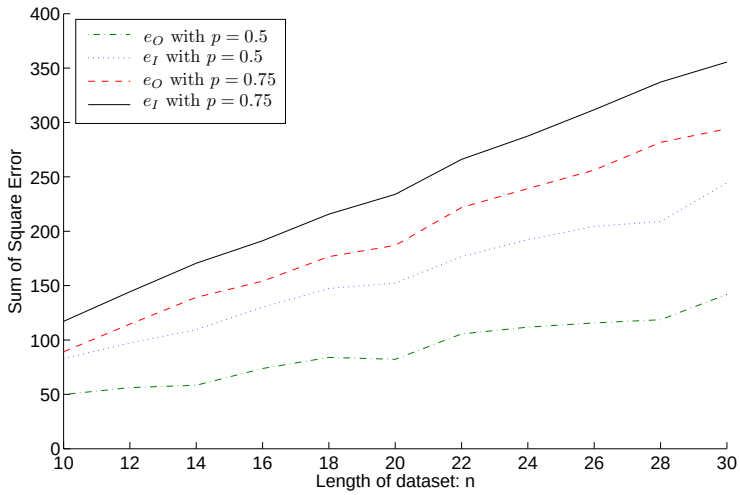


Figure 5.5: Improvement of offline version for $\delta = 4$.

Note that we allow $w_i = 0$ for some i, which indicates that no data are published at time i.

The above is a convex optimization problem whose solution can be found using existing optimization solvers, for example, a SDPT3 solver (TT-T99; TTT03). The allocation $\mathbf{e}_I = \langle \frac{\epsilon}{\delta}, \dots, \frac{\epsilon}{\delta} \rangle$ is in the feasible region of the

problem, and will be a good initial solution for the solvers. Let us denote the solution of Problem 1 $\mathbf{e}_O$.

Figure 5.5 shows the comparison of errors between $\mathbf{e}_I$ and the optimal budget allocation $\mathbf{e}_O$, where $\mathbf{w}$ is a binary vector and each $w_i \in \{0, 1\}$ is independently randomly chosen to be 1 with probability $p = 0.5$ and $p = 0.75$, respectively.

5.6.3 Online Allocation

In the online allocation problem, only $w_1 \dots w_i$ is available at time i. We consider the scenarios where $\mathbf{W}$ as a random variable that follows some distribution known to the analyst. We give an online algorithm as follows: given the committed budget allocation $\epsilon_1 \dots \epsilon_{i-1}$ and the observed weight vector $w_1 \dots w_i$ at time i, the analyst find the ϵ_i that minimizes the expected total error w.r.t. the distribution of $\mathbf{W}$. That is, consider Problem 2 as follow:

Problem 2 Constrained Offline Allocation

Given:	$\delta \in \mathbb{Z}_n, \mathbf{e}' = \langle \epsilon'_1, \epsilon'_2, \dots, \epsilon'_m \rangle \in \mathbb{R}^m_{\geq 0}$,
	$\mathbf{w} = \langle w_1 \dots w_n \rangle \in \mathbb{R}^n_{\geq 0}$
Find:	$\langle \epsilon_1, \epsilon_2, \dots, \epsilon_n \rangle$
Minimize:	$\sum_{i=1}^{n} \mathcal{E}rr_i(\epsilon_i) = \sum_{i=1}^{n} w_i \epsilon_i^{-2}$
Subject to:	$\sum_{i=1}^{\delta} \epsilon_{k+i} \leq \epsilon,$ for $k = 1, 2, \dots, (n - \delta)$;
	$\epsilon_k = \epsilon'_k,$ for $k = 1, 2, \dots, m$.

Let $E(\mathbf{e}', \mathbf{w})$ be the sum of error $\sum_{i=1}^{n} \mathcal{E}rr_i(\epsilon_i)$ of the solution of

Problem 2, then the goal of the analyst is to find ϵ_i that minimizes:

$$\sum_{\mathbf{w}}(Pr(\mathbf{W} = \mathbf{w})E(\epsilon_1 \ldots \epsilon_{i-1}, \epsilon_i, \mathbf{w}). \tag{5.6}$$

The online allocator repeat the step at each time i, and let us denote the output solution as $\mathbf{e}_X$.

5.6.4 Evaluations

We evaluate the performance of the online algorithm, comparing to the offline optimal solution and $\mathbf{e}_I$. We consider $\epsilon = 1$, and $\delta = 4$ or 7. For each setting, we repeat the experiment for 1,000 times and record the average error of the three solutions.

We consider the following approximation of the online allocator: at time i, 1,000 $\mathbf{w}$ with prefix $w_1 \ldots, w_i$ are sampled from the distribution of $\mathbf{W}$, and Equation (5.6) is computed for $\epsilon_i = 0.01 \ldots 1$ on the 1,000 sampled $\mathbf{w}$. Then the ϵ_i with the smallest error is taken as the allocated ϵ_i.

We consider a $\mathbf{w}$ where each $w_i \in \{0, 1\}$ is taken to be 1 with probability $p = 0.5$. Figure 5.6 shows the errors of $\mathbf{e}_O$, $\mathbf{e}_X$ and $\mathbf{e}_I$ for $\delta = 4$, and Figure 5.7 shows errors when $\delta = 7$.

Figure 5.8 consider a $\mathbf{w}$ where each $w_i \in \{0, 1\}$ is taken to be 1 with probability $p = 0.75$, and Figure 5.9 consider a $\mathbf{w}$ where each $w_i \in \{0, 1, 2\}$ is taken to be $0, 1$ and 2 with equal probability.

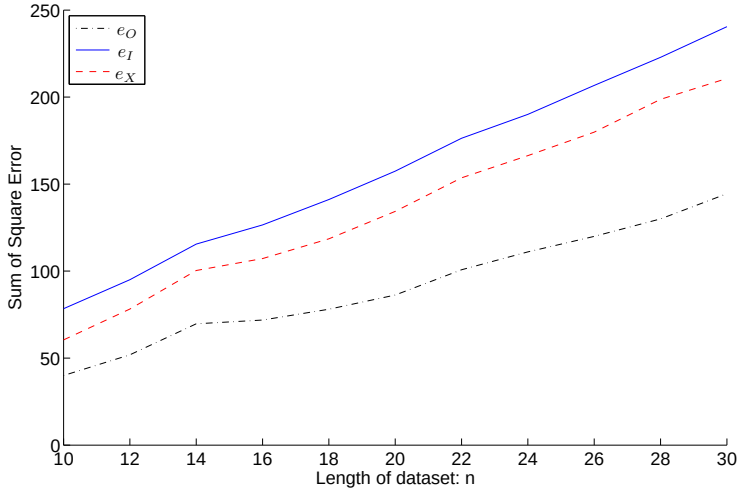


Figure 5.6: Comparison of offline and online algorithms for $\delta = 4$, $p = 0.5$.

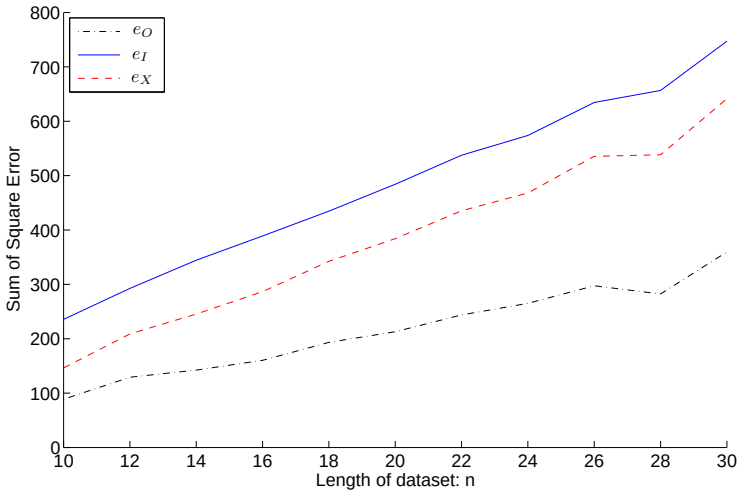


Figure 5.7: Comparison of offline and online algorithms for $\delta = 7$, $p = 0.5$.

5.7 Other Publishing Mechanisms

For some mechanisms, it is easier to apply the notion of δ-neighbourhood. In this Section we analyze their performance under δ-neighbourhood.

5.7.1 Publishing Sorted 1D Points

In Chapter 4 we proposed a method of publishing low-dimensional sorted points. We show that, the sensitivity of publishing n points in the domain

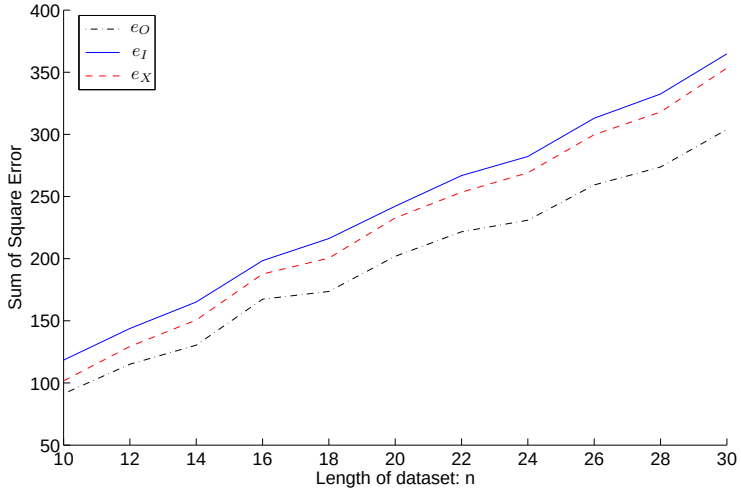


Figure 5.8: Comparison of offline and online algorithms for $\delta = 4$, $p = 0.75$.

of $[0, m]$ is m, independent to the value of n. Now let us consider the sensitivity of the method under δ-neighbourhood.

Recall that in Theorem 1, we show that the sensitivity of the published pointset is bounded by $A - x_i$, where A is the value of the replaced point. Under δ-neighbourhood, the value of$A - x_i$ is reduced to δm and therefore the Laplace noise required to achieve ϵ-differential privacy is reduced from $Lap(m/\epsilon)$ to $Lap(m\delta/\epsilon)$. Thus, there is significant improvement when applying the publishing method as it is. Figure 5.10 shows the improvement for expected mean square error for range query with different size.

Although the error is significantly decreased, it is not clear how to generalize the construction to higher dimensions. The method of using locality preserving transformation would not help since here we are required to preserve locality in the "difficult" direction.

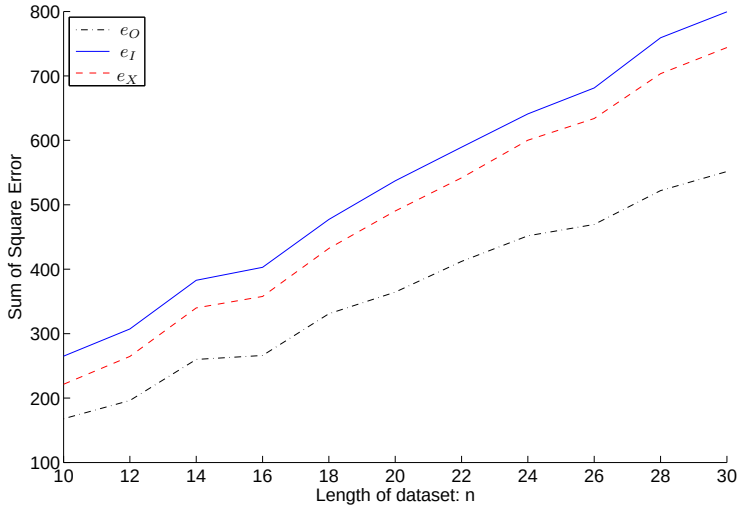


Figure 5.9: Comparison of offline and online algorithms for $\delta = 4$, and w_i is uniformly randomly taken to be $0, 1$ or 2.

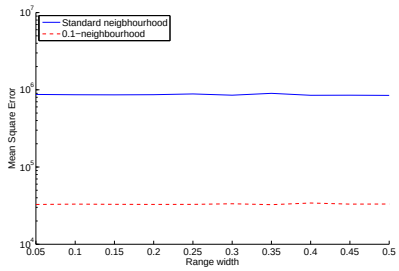


Figure 5.10: The comparison of range query error over 10,000 runs.

5.7.2 Publishing Median

Sometimes only aggregate information of a dataset, e.g. the median of the pointset, is required. To publish the median of a set of 1D points in $[0, m]$, a noise of $Lap(m/\epsilon)$ is required, although for most database instances, the "local sensitivity" is low, i.e. changing any element in that particular database instance will not significantly change the value of the median. Nissim et al. (NRS07) proposed a method that adds noise proportional to the "smooth sensitivity" (a smooth bound of the local sensitivity) of a database instance. He showed that this mechanism has high accuracy when the smooth sensitivity is low.

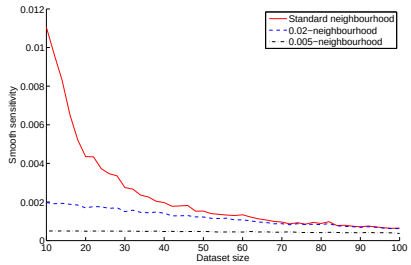


Figure 5.11: Noise required to publish the median with different neighbourhood.

The δ-neighbourhood can further reduce the noise requirement when "local sensitivity" can be still large. With δ-neighbourhood, we can reduce the global sensitivity, and thus bound the smooth sensitivity for some worst case scenarios. Figure 5.11 shows the experiment result on a synthesized dataset with random numbers generated under the exponential distribution and then scaled to $[0, 1]$. For each size of the dataset, we repeat the process 300 times and the average smooth sensitivity is recorded under different neighbourhood definitions.

5.8 Summary

In this chapter, we proposed to relax differential privacy by adopting a narrowed definition of neighbourhood which takes into account of the underlying distance of the entities. Although the idea is simple, for some applications, it is not clear how to exploit the relaxation to achieve higher utility. We consider two types of datasets, spatial datasets and dynamic datasets, and show that the noise level can be further reduced by constructions that exploits the δ-neighbourhood. We give a few scenarios where δ-neighbourhood would be more appropriate, and we believe the notion

provides a good trade-off for better utility.

Chapter 6

Secure Sketches with Asymmetric Setting

In this chapter and Chapter 7, we consider the biometric authentication problem.

In this chapter, we extend the secure sketch constructions to handle the asymmetric setting: in the enrollment phase, multiple biometric samples are obtained, whereas in verification, only one sample is acquired. This is a commonly deployed setting that can improve the authentication accuracy without increasing the process time in the verification phase. In this setting, d contains more information than d'. Therefore, the formulation of secure sketch under the asymmetric setting is different from the symmetric setting. Let us first define the formulation, then analyze the security and privacy impacts of different constructions with two biometric representations.

6.1 Asymmetric Setting

We consider the biometric data d submitted by a user consists of two component b, v, where $b \in \mathcal{M}$ is information of the biometric data, and v is the *auxiliary information*, which assists in improving the authentication accuracy. For example, b is the average value of the features, and v indicates the indices of the "consistent" features, similar to the version considered by Park et al. (PPJ08) and Moon et al. (MYCC04).

6.1.1 Extension of Secure Sketch

In the asymmetric secure sketch, the biometric data d submitted for enrollment is different from d' (recall that d' is the data submitted for verification, as described in Chapter 2). Let $\mathbf{D}$ be the space of d, and $\mathcal{M}$ be the space of d', and we can extend the definition of secure sketch as follow:

Asymmetric secure sketch. An asymmetric secure sketch scheme contains two algorithm $\mathsf{Enc}, \mathsf{Dec}$, where $\mathsf{Enc} : \mathbf{D} \to \{0,1\}^*$ is an encoder and $\mathsf{Dec} : \mathcal{M} \times \{0,1\}^* \to \mathbf{D}$ is a decoder such that $\mathsf{Dec}(d', \mathsf{Enc}(d)) = d$ if the distance of d and d' is less than some threshold t w.r.t. an underlying distance function.

In this case, d can be reconstructed during verification and thus can be used as the consistent secret.

6.1.2 Entropy Loss from Sketches

The security of a sketch scheme relies on how much useful information is leaked from the sketch. Recall the entropy bound for sketch in symmetric setting given by equation (2.2).

In the asymmetric setting, we take the following as the information leakage,

$$\mathbf{H}_\infty(d) - \widetilde{\mathbf{H}}_\infty(d|S) \tag{6.1}$$

It can be shown that a bound similar to (2.2) holds:

$$\mathbf{H}_\infty(d) - \widetilde{\mathbf{H}}_\infty(d|S) \leq L_S - \mathbf{H}_\infty(R) \tag{6.2}$$

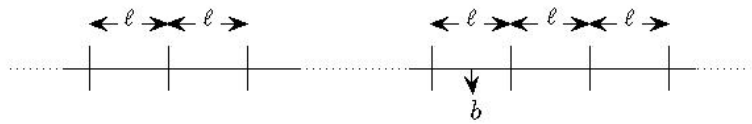
where $S = \mathsf{Enc}(D)$ is the asymmetric sketch and R is the recoverable randomness invested in asymmetric sketch construction.

To further illustrate how this bound is different from the secure sketch under the symmetric setting, let us first describe the constructions for a concrete underlying matrix for the utility measurement.

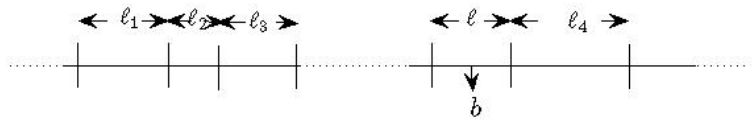
6.2 Construction for Euclidean Distance

In this section, we give a construction for Euclidean distance. We first consider one-dimensional data where during enrollment of d, multiple integer samples in $[0, n)$ are acquired. Two integers (b, v) are then extracted from the multiple scans, where the average sample $b \in [0, n)$ is the mean of the

multiple samples and the auxiliary information $v \in [1, q]$ is a threshold. During the verification, a sample d' in $[0, n)$ is acquired. d' is consider to be from same identity who enrolls d if $|d' - b| < v$, i.e. the close relation $C = \{(d, d') \mid \ |d' - b| < v\}$. The choice on the value of v decides the performance: a larger v gives lower false reject rate but lowers the key strength.



(a) Sketch with same length $\ell = 2v - 1$



(b) Sketch with different lengths

Figure 6.1: Two sketch schemes over a simple 1D case.

Let us first describe a straightforward scheme s_1 as follow: (1) Enc_1 on input $d = (b, v)$ outputs the set $S = \{c \mid \ c \in [0, n)$ and $c \equiv b(\mod 2v - 1)\}$; (2) Dec_1 on d' finds the point b in S that is closest to d', and outputs (b, v). It is easy to verify that for all $d, d' \in C$, $\mathsf{Dec}_1(d', \mathsf{Enc}_1(d)) = d$. Essentially, scheme s_1 divides $[0, n)$ into intervals of length $\ell = 2v - 1$ where b is at the center of one of the intervals as shown in Figure 6.1(a). Since the length is given in clear, the auxiliary information v is revealed.

Now, we describe our proposed scheme s_2. The main idea of our construction is to partition the domain $[0, n)$ into non-uniform intervals, in which $(b - v, b + v)$ is one of the intervals, as shown in Figure 6.1(b). Given $d = (b, v)$, the encoder Enc_2 of our proposed scheme s_2 constructs

the sketch S in following steps:

1. Let G be the set $\{b - v, b + v\}$;

2. While $\min(G) > 0$ add $\min(G) - 2r$ into S, where r is randomly draw from $[1, q]$;

3. While $\max(G) < n$ add $\max(G) + 2r$ into S, where r is randomly draw from $[1, q]$;

4. Sort G in ascending order and let the sorted list be $\langle g_1, \ldots, g_k \rangle$, note that g_1 is negative and $g_k > n$;

5. return $S = \langle g_1, g_2 - g_1, g_3 - g_2, \ldots, g_k - g_{k-1} \rangle$.

The Dec_2 algorithm on d' and S, reconstructs the set $G = \langle g_1, g_2, \ldots, g_k \rangle$, and finds the first i such that $g_i > d'$ (note that $i > 1$ since $g_1 < 0$), then returns $((g_{i-1} + g_i)/2, (g_i - g_{i-1})/2)$.

The correctness (i.e. $\mathsf{Dec}_2(d', \mathsf{Enc}_2(d)) = d$) of the scheme can be verified as follow: if d' and d are from the same identity, i.e. (d, d') is in C, then $b - v < d' < b + v$ and since there is no element in G falls in the interval $(b - v, b + v)$. Thus, for Dec_2, we have $g_i = b + v$, $g_{i-1} = b - v$, which will give us $\mathsf{Dec}_2(d', \mathsf{Enc}_2(d)) = d$ as required.

6.2.1 Analysis of Entropy Loss

The following analysis gives a bound on the entropy loss and comparison on privacy for scheme s_1 and s_2. Recall that such bound holds for any distribution of X.

Lemma 5 *The entropy loss of the sketch produced by s_2 is at most* $1 + 2\log q$.

Proof Let k be the number of elements in G, thus, the sketch contains k elements. Each of the $g_{i+1} - g_i$ is an even number in $[2, 2q]$ and thus can be describe with $\log q$ bits, and g_1 is in $(-2q, 0]$ and thus can be described with $\log 2q$ bits. While Dec_2 reconstructing G from d' and S, the randomness used in generating the $k-2$ intervals can be recovered. By equation (6.2), the entropy loss is at most $\log 2q + (k-1)\log q - (k-2)\log q = 1 + 2\log q$.

For the scheme s_1, since the number of bits required to describe the sketch is $|v| + |b \mod 2v|$, and there is no randomness involved, the entropy loss is bounded by $|v| + |b \mod 2v|$. Note that v is in range $[1, q]$, $|v| \leq \log q$, and $|b \mod 2v| \leq \log 2q$, thus, the entropy loss of scheme s_1 is bounded by $1 + 2\log q$.

While the entropy loss over the secret d are the same for schemes s_1 and s_2, scheme s_1 reveals the auxiliary information v in clear, whereas scheme s_2 hides the auxiliary information by mixing it with other secrets in the template. In this section we will analyze the impact of such difference. Let us assume that there are a threshold t and a small number ϵ such that for two biometric data $d_1 = (b_1, v_1)$ and $d_2 = (b_2, v_2)$ obtained from the same identity but during two enrollments, we will have $Pr(|v_1 - v_2| > t) < \epsilon$, and for two biometric data $d_1 = (b_1, v_1)$ and $d_2 = (b_2, v_2)$ obtained from two different identities, $Pr(|v_1 - v_3| < t) < \epsilon$.

Consider an adversary who wants to determine whether two sketches were generated using the same biometric (but with different noise), as

considered by Simoens et al.(STP09).

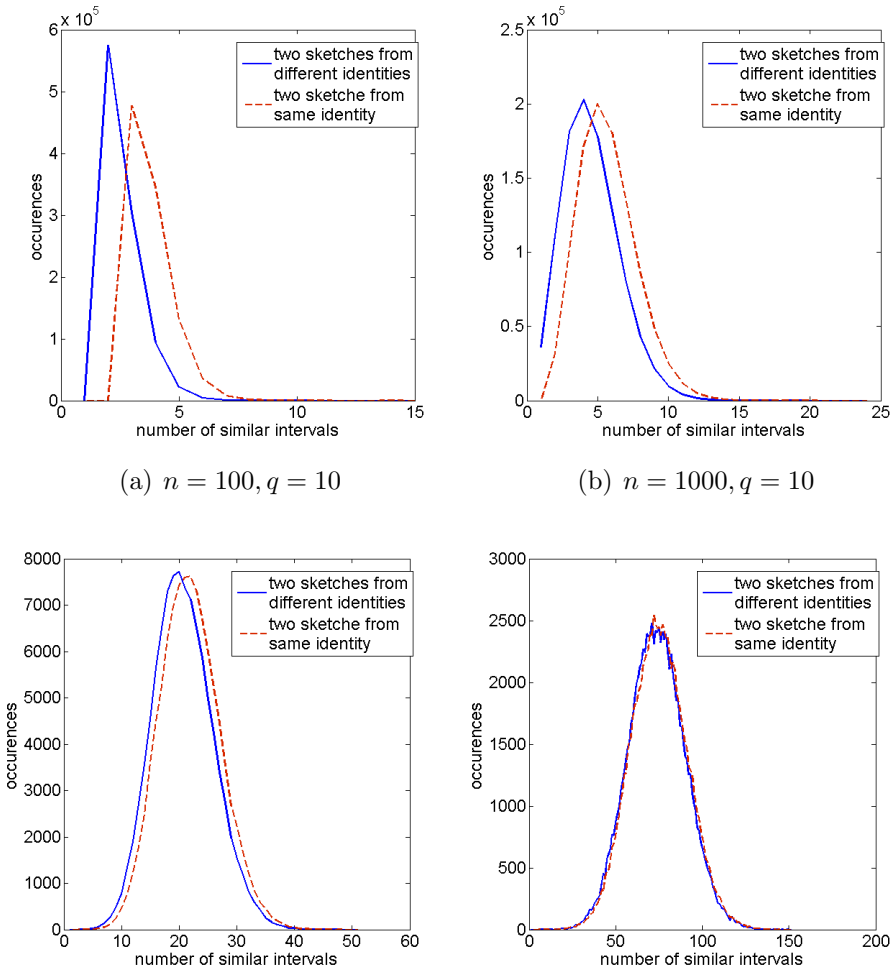
For scheme s_1, there is an effective adversary $\mathcal{A}_{s_1}$ in guessing whether d_1 and d_2 are from the same identity: it outputs "yes" if and only if $|v_0 - v_1| < t$. In this case, the probability $Pr(a' = a \mid a = 0) \geq 1 - \epsilon$ and $Pr(a' = a \mid a = 1) \geq 1 - \frac{2t-1}{q} - \epsilon$.

For scheme s_2, One possible strategy of $\mathcal{A}_{s_2}$ is to count the number of "similar intervals" between S_1 and S_2, and output "yes" if the count is larger than some threshold, otherwise outputs "no". Two intervals (c_0, c_1), (c'_0, c'_1) are similar if $|(c_1 - c_0) - (c'_1 - c'_0)| < 2t$ and $|(c_1 + c_0)/2 - (c'_1 + c'_0)/2| < t$, i.e. the lengths and centers of the two intervals are within the threshold t.

The intuition of the above strategy is that, when $a = 0$, the count is expected to be larger. However, when n is large and q is small, the domain $[0, n)$ is divided into many intervals and this will reduce the effectiveness of the strategy of $\mathcal{A}_{s_2}$. Thus, the attack will depends not only on the parameter q, t but also on n.

Figure 6.2 shows our experiment on how the parameters will affect the privacy protection: we implement the scheme s_2 and for different values of n and q with $t = 1$ and $\epsilon = 0.001$, we randomly generated 10^6 biometrics d_1, construct $\mathsf{Enc}(D \cup d_1)$, $\mathsf{Enc}(D \cup d_2)$ with different randomness then count the number of similar intervals, where d_2 is a noisy version of d_1 as described above. The histogram of the counts is shown by the red dotted line in the figure. We then randomly generated 10^6 pairs of d_1, d_2, construct S_1 and S_2 and count the number of similar intervals, where d_1 and d_2 are

two different biometric templates as described above. The histogram of the counts is shown by the blue solid line in the figure.



(a) $n = 100, q = 10$ (b) $n = 1000, q = 10$

(c) $n = 1000, q = 5$ (d) $n = 10000, q = 5$

Figure 6.2: The histogram of number of intervals for different n and q.

For example, when $n = 1000, q = 5$ as shown in Figure 6.2(d), the best guess without additional information for $\mathcal{A}_{s_2}$ is to output "no" when there are no more than 21 similar intervals. In that case, approximately he can guess with probability $p_{s_2} < (0.52 + 0.58)/2 = 0.55$, whereas $\mathcal{A}_{s_1}$ can guess with $p_{s_1} > 0.9$. When n gets larger and q gets smaller, the p_{s_2} will get closer to $\frac{1}{2}$.

6.3 Construction for Set Difference

Another commonly used distance metric is the set difference, and we will give a construction under the asymmetric setting of fuzzy vault scheme (JW99)[1] to handle the set difference, where a biometric sample can be represented as a set of elements in a space $\mathbb{Z}_p$. Under asymmetric setting, multiple sets are enrolled and two sets can be extracted: a set $d = \{x_0, x_1, \ldots, x_{m-1}\}$ where $x_i \in \mathbb{Z}_p$ of the elements appeared, and a set V denoting the importance, derived by the consistency, of each element.

Let us first describe the fuzzy vault scheme(JW99):

1. Randomly pick a polynomial F of degree $m - 2t - 1$ in field $\mathbb{Z}_p$;
2. Construct a set $(1, y_1), (2, y_2), \ldots, (p, y_p)$ in this way:. For each i, if $i \in d$, then y_i is chosen to be $F(i)$, otherwise, randomly picks an element from $\mathbb{Z}_p - \{F(i)\}$ to be y_i.
3. output $S = \{(1, y_1), (2, y_2), \ldots, (p, y_p)\}$.

Given a d', the reconstruction process attempts to find the polynomial F using the sampled points $\{(i, y_i) | i \in d'\}$, and then reconstruct d. When there is enough common points in d and d', the polynomial F can be reconstructed. Information on d are hidden as an adversary does not know which samples in the sketch S are from d. The samples in S are call the "chaff points".

This scheme can be considered as a special case where all the elements are equally important. The main idea of our construction is to extend

[1]There are many enhanced scheme over the fuzzy vault scheme by Juels et al., our technique can also be applied to the enhanced schemes in similar way.

the above scheme by mapping the more consistent elements to more points on the polynomial F so that they will contribute more roots in verification.

6.3.1 The Asymmetric Setting

In the asymmetric setting, we consider biometric representation which two pieces of information $d = (b, v)$ are extracted during enrollment, where $d = \{x_0, x_1, \ldots, x_{m-1}\}$ is a vector of m elements with $x_i \in \mathbb{Z}_p$, and $v = \{(x_0, v_0), (x_1, v_1), \ldots, (x_{m-1}, v_{m-1})\}$ is the corresponding weight of each elements, with each $v_i \in \mathbb{Z}_q$. A biometric template $d' = \{x'_0, x'_1, \ldots, x'_{k-1}\}$ is close to d if the sum of the weights of the common elements is larger than a threshold t, i.e. $\sum_{v \in W} v > t$ where $W = \{v | \exists x, (x, v) \in V, x \in (d \cap d')\}$.

The main idea of our construction is to extend the above scheme by associating the more important elements to more points to the polynomial F so that they will contribute more roots in verification. Let $H(x, y) = (x + qy)$ be a function on $\mathbb{Z}_q \times \mathbb{Z}_p \to \mathbb{Z}_{pq}$. Given x_i with weight v_i, we will first compute the set $S_i = \{H(0, x_i), H(1, x_i), \ldots, H(q-1, x_i)\}$. From S_i, we randomly pick v_i elements without replacement and let S'_i be the set of the v_i elements picked. Instead of adding $(x_i, F(x_i))$ to S, we add $(H(j, x_i), F(H(j, x_i)))$ to R for $H(j, x_i) \in S_i$, we add $(H(j, x_i), y_{j,i})$ to R for $H(j, x_i) \in (S_i - S'_i)$ where $y_{m,i}$ is randomly chosen from $\mathbb{Z}_p - \{F(H(j, x_i))\}$. To prevent adversary from finding out the X from the chaff points, we need to create chaff points associate with the q values in the similar way. Specifically, the construction procedure of fuzzy vault in asymmetric setting is as follow:

1. Randomly pick a polynomial F of degree $g-2t'-1$ in field $\mathbb{Z}_{pq}$, where $g = \sum_{v \in V}(v)$ and $t' = (g-t)$;

2. Starts with a set $G = X$ and an empty set S;

3. For $i = 0$ to $m-1$, compute $S_i = \{H(0, x_i), H(1, x_i), \ldots, H(q-1, x_i)\}$, and uniformly randomly pick S'_i from set $\{X | X \in \mathbf{P}(S_i), |X| = v_i\}$, where $\mathbf{P}(S_i)$ is the powerset of S_i and $|X|$ is the size of set X. Add $(H(j, x_i), F(H(j, x_i)))$ to S for $H(j, x_i) \in S_i$ and add $(H(j, x_i), y_{j,x_i})$ to S for $H(j, x_i) \in (S_i - S'_i)$ where y_{j,x_i} is randomly chosen from $\mathbb{Z}_p - \{F(H(j, x_i))\}$.

4. For $i = m$ to r, randomly pick $x_i \notin G$, add x_i to G, compute $S_i = \{H(0, x_i),\ H(1, x_i), \ldots, H(q-1, x_i)\}$ and add (x_i, y_i) to S, $(H(j, x_i), y_{j,x_i})$ to S for $H(j, x_i) \in (S_i)$ where y_{j,x_i} is randomly chosen from $\mathbb{Z}_p - \{F(H(j, x_i))\}$.

5. Output S.

During verification, given a $d' = \{x'_0, x'_1, \ldots, x'_{k-1}\}$, Dec first computes the set S' of $\{H(j, x'_i) | x'_i \in d', j \in [0, q-1]$, then attempts to find the polynomial F of degree $g - 2t' - 1$ with points in the set. If such F is found, the original d can be reconstructed.

6.3.2 Security Analysis

Now let us bound the entropy loss of sketch constructed by the above scheme. The recoverable randomness involved is the coefficients of the polynomial F, as well as the generated $y_{j,i}$. Thus the amount of randomness

is $(g-2t'-1)\cdot\log p+(qp-g)\cdot\log(p-1)$. By setting the parameter $r=p$, we can omit the $H(j,x_i)$ and have a compact description of the sketch. Hence, the size of sketch is $pq\cdot\log p$ and the entropy loss can be bounded as follow:

$$\begin{aligned}
& \mathbf{H}_\infty(d)-\widetilde{\mathbf{H}}_\infty(d|S) \\
= \;& pq\log p-(g-2t'-1)\log p-(qp-g)\log(p-1) \\
= \;& pq\log\frac{p}{p-1}+g\log\frac{p}{p-1}+(2t'+1)\log p \\
\leq \;& q\log e+g\log\frac{p}{p-1}+(2t'+1)\log p
\end{aligned}$$

When q is small, and p is large, the bound is similar to symmetric case. However, when q is large, i.e. when the auxiliary information has high entropy, and the amount of information leak can be high.

In the work by Juels et al. (JW99), the security strength is given by the number of spurious polynomials, i.e. polynomials that have degree $m-2t-1$ and m roots in the sketches. For the symmetric scheme described above, with probability $1-\mu$, there exists at least $\frac{\mu}{3}p^{(m-2t-1)-m}(\frac{r}{m})^m$ spurious polynomials.

Similarly, in the asymmetric scheme, with probability $1-\mu$, there will be at least $\frac{\mu}{3}p^{(g-2t'-1)-g}(\frac{qr}{g-2t'-1})^{g-2t'-1}$ polynomials with degree $g-2t'-1$ and g roots. Let us call these polynomials in asymmetric setting the spurious polynomials. However, the analysis of the spurious polynomials is not sufficient for asymmetric setting as the likelihood of a spurious polynomial to be F depends on the distribution of the roots. Let us call a spurious polynomial a *candidate polynomial* if the number of distinct C_i's that contains the roots of the polynomial is less than a threshold a.

The probability that a random spurious polynomial is a candidate polynomial can be view as a variance of the birthday attack analysis. For example, the probability of the case when $q = 2$ (i.e. the consistent elements are twice important as the inconsistent) is as follow:

$$\frac{1}{\binom{2r}{g}} \sum_{x=g-a}^{g/2} \left(2^{g-x} \cdot \binom{r}{g-x} \binom{g-x}{x} \right).$$

For $r = p = 10^4, t = 2, m = 22$ there is 9.7629×10^{33} spurious polynomials with probability $1 - 1/10^4$ in symmetric setting; and with $g = 35$ and $a = 32$, (i.e. the sum of weights is 35, and polynomials with weight higher than 32 are candidate polynomials). There is in total 2.6996×10^{47} spurious polynomials with probability $1-1/10^4$. Note that the reason it has more spurious polynomials than symmetric setting is because each element contributes two (chaff) points. Therefore, approximately 2.4113×10^{-5} of the spurious polynomials are candidate polynomials, which is 6.5095×10^{46}.

6.4 Summary

In this chapter, we demonstrate sketches constructions that reveal auxiliary information will leak important information which can be utilized in distinguishing sketches from different identities. To reduce the linkages among sketches, we proposed two schemes. The first scheme handles Euclidean distance and it outputs sketches with non-uniform sized intervals, while the second scheme handles set-differences whereby the more consistent elements are assigned with more points in the underlying polynomial. Our schemes and analysis demonstrate that, by mixing the auxiliary informa-

tion within the biometric data appropriately, although there is no reduction in overall identity information lost (measured by entropy loss), the linkage of sketches can be reduced.

Chapter 7

Secure Sketches with Additional Secrets

In this chapter, we will consider extension of secure sketches under the multiple secrets setting, where the secrets differ in role and importance. In particular, we consider cases when the data d consists of a secret k that is less important, together with a secret b, that is fuzzy and important to the authentication system.

One may simply generate a sketch for each secret independently and concatenate them, but this does not address the fact that the secrets are of different importance. we propose a mixing approach, that in the multi-secret setting can 'divert" the information loss of more important secrets to less important ones , and thus providing more protection to the former.

7.1 Multi-Factor Setting

We consider the private data d submitted by the data owner consists of two component b, k, and the biometric secret b is fuzzy and important (e.g. the iris data or fingerprint data), whereas the secret k is less important, but could be either fuzzy (e.g. soft biometrics such as keystrokes) or consistent (e.g. password). We study the secure sketch schemes under this setting.

A straightforward extension of sketch construction to two secrets is to simply apply two sketch schemes, for the two secrets b and k independently. For example, when k is a fuzzy secret, we can have the final sketch for the two secrets is the concatenation of the sketches $S_1 = \mathsf{Enc}_1(b)$ and $S_2 = \mathsf{Enc}_2(k)$.That is, the sketch $S = S_1 \mid S_2$, where $\mid$ represents concatenation. Furthermore, the final key $K = K_1 \mid K_2$ can be the concatenation of the keys, where K_1 and K_2 are the keys extracted from b and k respectively. The key K can thus be used in standard cryptographic applications as a credential to authenticate the data owner. Similarly, the straightforward extension simply omit S_2 when k is a consistent secret.

Suppose the entropy loss of the first secret given the sketch is at most $\mathcal{L}_1$, and that of the second secret is at most $\mathcal{L}_2$, then it is clear that the overall entropy loss is at most $\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2$, since the secrets are independent.

As we have mentioned, this straightforward approach is not able to differentiate secrets with different characteristics, and give equal protection to both secrets.

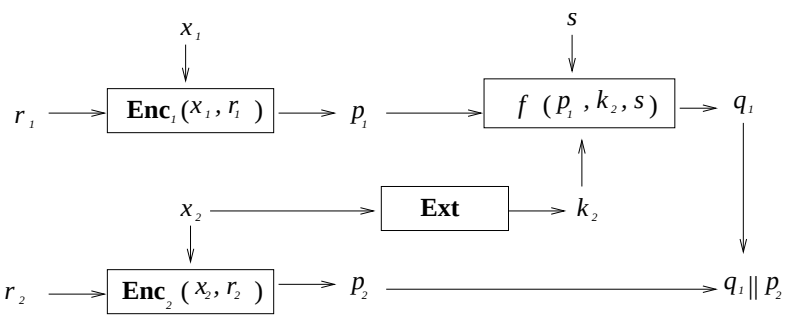


Figure 7.1: Construction of cascaded mixing approach.

7.1.1 Extension: A Cascaded Mixing Approach

Instead of treating the two secrets independently, it may be desirable to combine different types of secrets to achieve additional security goals. Here we give an alternative sketch construction. Figure 7.1 illustrates our proposed method.

To provide protection to the data $d = (b, k)$, we first compute sketches S_1 and S_2 as in the concatenating approach, and extract keys K_1 and K_2 respectively. After that, we encrypt S_1 using K_2 as the key, that is, we compute $Q = f(S_1, K_2, R_f)$, where f is a deterministic function and R_f is an auxiliary random string. The final sketch S output by the cascaded mixing approach is $Q \mid S_2$.

Let us call f the *mixing function* which serves as an encryption with K_2 as the key. As the leftover entropy of K_2 given S_2 could be low, we should not rely on the computational difficulty in inverting f to protect S_1. Thus, it is important to analyze how much information about the two secrets X_1 and X_2 is revealed.

Let us consider the mixing function $f : \mathcal{M}_{S_1} \times \mathcal{M}_{K_2} \times \mathcal{M}_{R_f} \to \mathcal{M}_Q$ and random variables Q, S_1, K_2 and R_f such that $Q = f(S_1, K_2, R_f)$. We

require f to have certain properties. First, as an encryption function, f must be invertible.

Invertibility We say that a mixing function f is invertible if there is a function g such that for all $S_1 \in \mathcal{M}_{S_1}$, $K_2 \in \mathcal{M}_{K_2}$ and $R_f \in \mathcal{M}_{R_f}$, $g(f(S_1, K_2, R_f), K_2) = S_1$.

In addition, in our analysis we consider mixing functions with the following properties on recoverability of the randomness invested.

Recoverable Randomness For a mixing function f, the randomness R_f is called recoverable if $S_1 \in \mathcal{M}_{S_1}$, $K_2 \in \mathcal{M}_{K_2}$ and $R_f, R'_f \in \mathcal{M}_{R_f}$, we have $f(S_1, K_2, R_f) = f(S_1, K_2, R'_f)$ implies $R_f = R'_f$.

β-Recoverable Key For a mixing function f, the key K_2 is called β-recoverable if for any $Q \in \mathcal{M}_Q$, the size of support for K_2 given Q is at most 2^β, i.e., we should have the following inequality:

$$|\{K_2 \in \mathcal{M}_{K_2} \mid \exists S_1 \in \mathcal{M}_{S_1}, K_2 \in \mathcal{M}_{K_2}, R_f \in \mathcal{M}_{R_f}, s.t. f(S, K, R_f) = Q\}| \leq 2^\beta$$

It is easy to construct mixing function achieving both invertability and recoverability. For example, we can obtain one from a block cipher $f(S_1, K_2, R_f) = R_f \mid E_{K_2}(S_1 \mid R_f)$. Note that the recoverability properties are not necessary for the recovery of the secrets, but will become handy in the security analysis.

When a user presents b' and k' that are close to b and k respectively, k is first reconstructed using k' and S_2, and a key K_2 is extracted from k, which in turn is used to retrieve S_1 if f is invertible. After that, b is

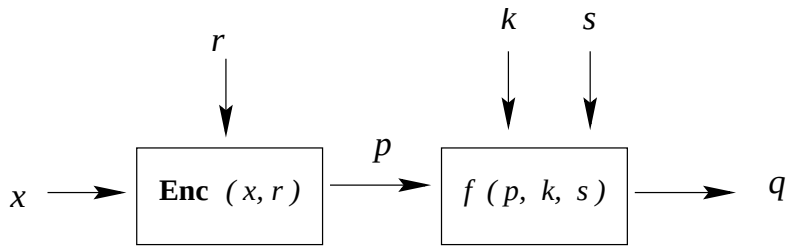


Figure 7.2: Process of Enc: computation of mixed sketch.

reconstructed using b' and S_1. An extractor can be further applied on $b \mid k$ to extract a key.

Intuitively, compared to the approach that treats the two secrets independently, this alternative approach gives more protection to the first secret b, since it would require the attacker to guess k using S_2 first, only when the attacker is successful can the attacker gain information on b from S_1 by computing S_1 from Q and k. However, it may leak more information on k as the attacker may exploit his knowledge on b to guess k.

7.2 Analysis

We now study the case of two secrets and a scheme that follows the cascaded sketch construction (Section 7.1.1). Let $b \in \mathcal{M}_b$ be a fuzzy secret (say, a fingerprint), and let $k \in \mathcal{M}_k$ be an independent secret key that is not fuzzy. Consider a sketch scheme with encoder Enc_1, and let the sketch $S_1 = \mathsf{Enc}_1(b, R)$ with randomness R. Figure 7.2 illustrates the process.

It is clear that when the key k is uniform and no shorter than the sketch, we can easily hide the sketch S_1 completely (e.g., by using the key as a one-time pad), and as such, the analyst cannot get any further information on S_1 from the final output S (note that it is not necessary

to store helper data for the consistent secret k, therefore we have $S = Q$). However, in practical scenarios (e.g., user chosen PIN/password as the key), k can be shorter than S_1, and the analysis of security may become challenging. In fact, we will show that, for shorter k, mixing is not always a better strategy than the straightforward method of treating the secrets independently. We will also show the conditions under which mixing is desirable.

7.2.1 Security of the Cascaded Mixing Approach

Analysis of overall remaining entropy $\widetilde{\mathbf{H}}_\infty(b, k \mid S)$.

First, let us investigate the remaining entropy when we treat (b, k) as a single secret, i.e. the remaining entropy $\widetilde{\mathbf{H}}_\infty(b, k \mid S)$.

Lemma 6 *Given random variables b, k, R, S and mixing function f as described above, We have*

$$\widetilde{\mathbf{H}}_\infty(b, k|S) \geq \mathbf{H}_\infty(b) + \mathbf{H}_\infty(k) + \mathbf{H}_\infty(R) - L_S.$$

Proof Since R is recoverable, we can consider Enc_1 and f together as the encoding algorithm for the final sketch S, R and R_f together as the recoverable randomness, and the inequality (2.2) in Chapter 2 applies. Note that $|S| = |S_1| + |R_f|$, and we have

$$\begin{aligned}\widetilde{\mathbf{H}}_\infty(b, k \mid S) &\geq \mathbf{H}_\infty(b, k) + \mathbf{H}_\infty(R) + \mathbf{H}_\infty(R_f) - L_S \\ &= \mathbf{H}_\infty(b) + \mathbf{H}_\infty(k) + \mathbf{H}_\infty(R) - L_S.\end{aligned}$$

Hence the lemma holds as claimed.

Lemma 6 gives a lower bound of the remaining entropy of b and k. In general, if both secrets are fuzzy, we can similar obtain the bound:

$$\widetilde{\mathbf{H}}_\infty(b, k \mid S) \geq \mathbf{H}_\infty(b) + \mathbf{H}_\infty(k) + \mathbf{H}_\infty(R_1) + \mathbf{H}_\infty(R_2) - L_{S_1} - L_{S_2}.$$

where R_1, R_2, are the randomness invested in constructing the sketch S_1, S_2 for the two respective secrets. Note that this bound is the same when we use the straightforward concatenation approach.

Analysis of individual secret $\widetilde{\mathbf{H}}_\infty(x \mid S)$ and $\widetilde{\mathbf{H}}_\infty(k \mid S)$.

Now, let us look at the remaining entropy of individual secret, i.e. $\widetilde{\mathbf{H}}_\infty(x \mid S)$ and $\widetilde{\mathbf{H}}_\infty(k \mid S)$.

If the sketch is not uniformly distributed, then given the mixed s, it is possible that $(k \mid S = s)$ is not uniform. That is, S will leak some information about k. Indeed, an adversary, given s, may enumerate all possible k's and the correspond sketch S to determine the most likely k. Nevertheless, leakage of k is acceptable as long as it can provide more protection to b. Next theorem gives a lower bound on the remaining entropy of b given the mixed sketch S.

Theorem 7 *Given three independent random variables b, k and R distributed over $\mathcal{M}_b$, $\mathcal{M}_k$ and $\mathcal{M}_{L_R}$ respectively and a sketch scheme with encoder Enc_1, Let S_1 be the sketch of b, i.e., $S_1 = \mathsf{Enc}_1(b, R)$, where R is recoverable, and let $f : \mathcal{M}_{S_1} \times \mathcal{M}_k \times \mathcal{M}_{R_f} \to \mathcal{M}_S$ be an mixing function and $S = f(S_1, k, R_f)$, where R_f is a L_{R_f} bits of recoverable randomness. If*

f is invertible and the key k is L_{R_f}-recoverable. Then

$$\widetilde{\mathbf{H}}_\infty(b \mid S) \geq \mathbf{H}_\infty(b) + \mathbf{H}_\infty(k) - L_S. \tag{7.1}$$

Proof First, let $\mathcal{K}_{b,S} \subset \{0,1\}^{L_k}$ be the set of secret k with which there exist R and R_f such that S can be computed from b, R, k and R_f. That is,

$$\mathcal{K}_{b,S} = \{k \in \mathcal{M}_k \mid \exists R, R_f, f(\mathsf{Enc}(b,R), k, R_f) = S\}.$$

Since the key of the mixing function f is L_{R_f}-recoverable, it is clear that the cardinality $|\mathcal{K}_{b,S}|$ is no more than the number of all possible R's multiplied by $2^{L_{R_f}}$. note that $L_{R_f} = L_S - L_{S_1}$. That is, $|\mathcal{K}_{b,S}| \leq 2^{L_R + L_{R_f}}$ for any b and S. Now, consider

$$\begin{aligned}
A &= 2^{-\widetilde{\mathbf{H}}_\infty(b \mid S) - L_R - L_{R_f}} \\
&= \sum_s \Pr[S = s] \max_x \Pr[b = x \mid S = s] 2^{-L_R - L_{R_f}} \\
&= \sum_s \max_x \Pr[b = x, S = s] 2^{-L_R - L_{R_f}}.
\end{aligned}$$

On the other hand, we have

$$B = 2^{-\widetilde{\mathbf{H}}_\infty(b,k \mid S)} = \sum_s \max_{x,y} \Pr[b = x, k = y \mid S = s].$$

For any $s_0 \in \mathcal{M}_S$, let us consider

$$\begin{aligned}
&\max_x \Pr[b = x, S = s_0] 2^{-L_R - L_{R_f}} \\
&= \max_x \sum_y \Pr[b = x, S = s_0, k = y] 2^{-L_R - L_{R_f}} \\
&\leq \max_x \left(\max_y \Pr[b = x, S = s_0, K = y] 2^{L_R + L_{R_f}} \right) 2^{-L_R - L_{R_f}} \\
&= \max_{x,y} \Pr[b = x, S = s_0, k = y]
\end{aligned}$$

The inequality holds because for any x, there will be at most $|\mathcal{K}_{x,s_0}| \leq 2^{L_R+L_{R_f}}$ non-zero terms in the summation, hence the sum will be at most $2^{L_R+L_{R_f}}$ times the largest term in the summation. As a result, we have

$$A \leq \sum_s \max_{x,y} \Pr[b = x, S = s, k = y] = B.$$

This is equivalent to

$$\widetilde{\mathbf{H}}_\infty(b \mid S) + L_R + L_{R_f} \geq \widetilde{\mathbf{H}}_\infty(b, k \mid S).$$

By applying the bound on overall entropy loss (Lemma 6), and considering that the recoverable randomness includes the L_R bit R and L_{R_f} bit R_f, we have

$$\widetilde{\mathbf{H}}_\infty(b \mid S) \geq \widetilde{\mathbf{H}}_\infty(b, k \mid S) - L_R - L_{R_f} \geq \mathbf{H}_\infty(b) + \mathbf{H}_\infty(k) - L_S$$

Therefore the theorem holds as claimed.

The theorem holds for any distributions of X and K, and for uniformly distributed K, the theorem implies that $\widetilde{\mathbf{H}}_\infty(b \mid S) \geq \mathbf{H}_\infty(b) + L_k - L_S$. Let us compare the remaining entropy if we use the simple concatenation method, which is as follows,

$$\widetilde{\mathbf{H}}_\infty(b \mid S) \geq \mathbf{H}_\infty(b) + L_R - L_S \tag{7.2}$$

Now, coming back to the question that whether it is beneficial to use a cascading function when the secret k is short compared with S_1. Clearly, from Theorem 7 and inequality (7.2), we can see that when $\mathbf{H}_\infty(k) - L_S \geq L_R - L_{S_1}$, or equivalently, $\mathbf{H}_\infty(K) \geq L_R + L_{R_f}$, the R.H.S in

(7.1) is larger then the R.H.S in (7.2), i.e. the entropy bound when using a mixing function is no worse than not using it. In particular, consider a deterministic sketch scheme (i.e. $L_R = 0$), and a "length preserving" mixing function (thus $L_{S_1} = L_S$), the difference in the right hand side of the inequality (7.1) and (7.2) is $\mathbf{H}_\infty(K)$. In other words, the bound on entropy loss of b given S can be reduced by $\mathbf{H}_\infty(k)$. Viewing from another direction, information loss on b is "diverted" to k.

Now, we consider only the non-fuzzy secret k and analyze the entropy loss.

Theorem 8 *Given a sketch scheme with encoder* Enc *that use randomness* R*, and a mixing function* f *using randomness* R_f*, and let* b*,* k*,* R*,* S_1*,* S*,* f*,* R_f *be as defined in Theorem 7, we have*

$$\widetilde{\mathbf{H}}_\infty(k \mid S) \geq \mathbf{H}_\infty(k) + \mathbf{H}_\infty(R) - L_{S_1}. \tag{7.3}$$

Proof Since $S = f(S_1, k, R_f)$, we can regard S as a sketch of k where the cascading function f is an encoder, and $S_1 = \mathsf{Enc}(b, R)$ and R_f are the "randomness" invested in computing Q, which are recoverable. Since R is recoverable, we have

$$\mathbf{H}_\infty(b) + \mathbf{H}_\infty(S_1) \geq \widetilde{\mathbf{H}}_\infty(b, S_1) \geq \mathbf{H}_\infty(b) + \mathbf{H}_\infty(R)$$

which means that $\mathbf{H}_\infty(S_1) \geq \mathbf{H}_\infty(R)$. and then we can apply the general bound (2.2) on k and S, and hence the inequality holds as desired.

It is worth to note that the bound in Theorem 8 is tight in the sense that there exists random variables and functions such that the equality in (7.3) holds. We will see an example of such case in Section 7.3.2.

Therefore, if L_{S_1} is large but the min-entropy $\mathbf{H}_\infty(S_1)$ is low, the quantity $\mathbf{H}_\infty(k) + \mathbf{H}_\infty(R) - L_{S_1}$ may be reduced to 0, in which case S may reveal all information about k.

7.3 Examples of Improper Mixing

In this section we give examples to illustrate the scenarios where mixing function may not be beneficial: (1) in scenarios where the sketch construction employs randomness, mixing function may not always provide protection on X. (2) when the sketch contains high redundancy from the adversary point of view, mixing function may reveal information of k.

7.3.1 Randomness Invested in Sketch

This section gives a simple example to illustrate the idea that mixing function may not always provide protection on b, if the sketch construction contains randomness. Hence, as a general guideline, when choosing a sketch scheme to be used in the cascaded mixing framework, it is better to select one that requires no randomness.

Consider a non-fuzzy k in $\{0,1\}^{L_k}$, and a fuzzy b in $\{1 \ldots 2^{L_b}\}$, where b_1 is close to b_2 if they differ only at the last bit. That is, b_1 and b_2 are close if $b_1 - (b_1 \mod 2) = b_2 - (b_2 \mod 2)$. Hence, a noisy copy of b could be either b or b with the last bit flipped.

Consider the following two sketch constructions: a deterministic construction $\mathsf{Enc}_1(b) = b \mod 2$, and a probabilistic construction $\mathsf{Enc}_2(b, R) = b + R \mod 2^{L_b}$, where R is a uniform random even number in $\{2, 4, \ldots, 2^{L_b}\}$.

Without mixing, sketches output from both constructions reveal at most one bit of b.

Given a one bit secret k, let the mixing function $f(S_1, k, R_f)$ be as following: it first generates with seed R_f a set $\mathbf{R} = \langle \mathbf{r}_1, \mathbf{r}_2 \rangle$ of random strings of length L_b, then it output $S_1 + \mathbf{r}_k \mod 2^{L_b}$.

Consider the case when Enc_1 is used, the mixing function is one-time pad encryption, by Theorem 7, there will be no entropy loss on b i.e. $\mathbf{H}_\infty(b) - \widetilde{\mathbf{H}}_\infty(b \mid S) = 0$. However, when Enc_2 is used, there could be cases where $\mathbf{r}_i$ has same parity, for example, $\mathbf{R} = \langle 0, 2 \rangle$. In that case, the information of the sketch is not protected and $\mathbf{H}_\infty(b) - \widetilde{\mathbf{H}}_\infty(b \mid S) = 1$ and there is no gain nor loss in mixing the secrets compare to the straightforward method. In other words, the secret k is unable to provide additional protection as desired.

Note that, by Lemma 6, the overall entropies $\widetilde{\mathbf{H}}_\infty(b, k \mid S)$ are the same in the aforementioned two cases, as well as in the straightforward method of not mixing the secrets. Note that in the second case, the adversary is unable to infer any information on k, where as in the first case he know that whether k and b has the same parity from the published data S.

Hence, when given two choices of sketch constructions where one is deterministic and the other is probabilistic, it is advisable to employ the deterministic method to achieve the protection provided by mixing function.

7.3.2 Redundancy in Sketch

When the sketch has redundancy, that is, the entropy of the sketch is smaller than the length of the sketch, information on k will be leaked from the mixed sketch. There are a few known sketch constructions where the "support" of the sketch (i.e. the number of sketches which non-zero probability of occurrences) is significantly smaller than the description size $2^{L_{S_1}}$ and thus their sketches contain redundancy. One example is the chaff-based method (CKL03) proposed to protect the biometric fingerprint. Here, a fingerprint is the secret b and can be represented as a set of 2D points. The chaff-based method gives its sketch which is the original x union with a set of random 2D points, constrained by the requirement that no two points are close to each other (w.r.t Euclidean distance). It is not easy to derive a compact description of the sketch that has size close to $\mathcal{M}_{S_1}$. Now, suppose that the sketch is mixed with a short k. Given a mixed sketch S, it could be highly likely that among all possible k's in inverting S, only one give a point set that satisfies the constrain. Thus, immediately, the secret k and the sketch is revealed, and the remaining entropy of the combined $\widetilde{\mathbf{H}}_\infty(b, k \mid S) = \widetilde{\mathbf{H}}_\infty(b \mid S_1)$. Hence, by mixing, not only there is no further protection of x, the k is revealed.

We also conducted experiment to illustrate that, even when the description of sketch is compact in 1D pointset, i.e. the description size equals the support $\mathcal{M}_{S_1}$, the chaff-based sketch still contains significant redundancy that leads to lost of information on k.

Consider the chaff-based method for 1D points, which is easy to

derive a compact description. We simulated the chaff-based method in $\mathbb{Z}_{24}$ with a minimum distance 3. There are in total 605 possible sketches, and we randomly generated 10^5 sketches. Figure 7.3 shows the numbers of occurrences for all 605 sketches with x-axis descendingly sorted by the number of occurrence (and we call the position of a sketch in this descending list the *rank* of it).

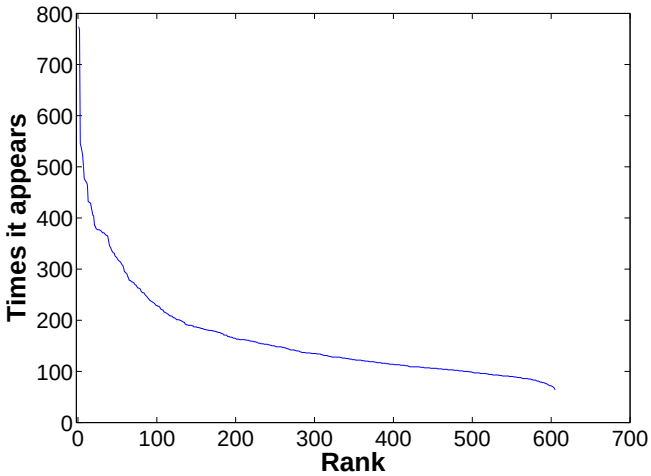


Figure 7.3: Histogram of sketch occurrences.

Suppose the sketch is then protected by a 5 bits key k, and a mixing function f such that the inverts are always valid sketches. We then simulate an adversary who try to guess k when given $S = f(S_1, k, R_f)$, where k and R_f are randomly chosen from their domain and S_1 is chosen according to the distribution approximated by Figure 7.3. We simulated 10^5 guesses and the adversary can succeed with probability slightly more than 0.052, instead of $1/(2^5) = 0.03125$ as in random guessing.

7.4 Extensions

7.4.1 The Case of Two Fuzzy Secrets

When both secrets are fuzzy and may not be uniform, we show that the bounds of Lemma 6, Theorem 7 and 8 can be obtained with slight modifications.

Suppose there are two independent secrets $b_1 \in \mathcal{M}_{b_1}$ and $b_2 \in \mathcal{M}_{b_2}$, and two sketch construction schemes with encoder Enc_1 and Enc_2 respectively. We assume that the first secret b_1 is more important than b_2. In this case, we can use the following steps to construct the sketch for the two secrets.

1. Compute $S_1 = \mathsf{Enc}_1(b_1, R_1)$ and $S_2 = \mathsf{Enc}_2(b_2, R_2)$.

2. Extract a key k_2 from b_2 using an extractor Ext.

3. Compute $Q = f(S_1, k_2, R_f)$ using a mixing function f.

4. Output the final sketch $S = Q \| S_2$.

It is possible to design Ext such that K_2 and S_2 are independent, and $\mathbf{H}_\infty(K_2)$ is only slightly smaller than $\widetilde{\mathbf{H}}_\infty(b_2|S_2)$ (DRS04). Let δ be a small extractor-dependent value such that $\mathbf{H}_\infty(K_2) \geq \widetilde{\mathbf{H}}_\infty(b_2|S_2) - \delta$.

The bound in Theorem 7 still applies on b_1 and K_2. Consider random variables b_1 and b_2, corresponding sketches S_1 and S_2, mixed sketch Q, andfinal sketch S, it's not difficult to show that

$$\widetilde{\mathbf{H}}_\infty(b_1|S) \geq \mathbf{H}_\infty(b_1) + \mathbf{H}_\infty(b_2) + \mathbf{H}_\infty(R_2) - L_{S_2} - \delta - |S|,$$

where R_2 is the recoverable randomness used in computing S_2. In this case, the small δ can be considered as the overhead of using the extractor Ext.

As a comparison, if we treat the two secrets independently, and consider $S = S_1 \| S_2$, we have $\widetilde{\mathbf{H}}_\infty(b_1|S) = \widetilde{\mathbf{H}}_\infty(b_1|S_1) \geq \mathbf{H}_\infty(b_1) + \mathbf{H}_\infty(R_1) - L_{S_1}$.

Similar to the example, we can conclude that if $\mathbf{H}_\infty(K_2) \geq L_{R_1} + L_{R_f}$, we can obtain a better bound on the entropies when we choose to mix b_2 with b_1. Otherwise, doing so may reveal more information about b_2.

The entropy loss on the second secret b_2 can be obtained using the bound in Theorem 8:

$$\widetilde{\mathbf{H}}_\infty(b_2|S) \geq \mathbf{H}_\infty(b_2) + \mathbf{H}_\infty(R_2) + \mathbf{H}_\infty(R_1) - L_{S_1} - L_{S_2} - \delta$$

The overall entropy loss in Lemma 6 applies to the general case. That is,

$$\widetilde{\mathbf{H}}_\infty(b_1, b_2|S) \geq \mathbf{H}_\infty(b_1) + \mathbf{H}_\infty(b_2) + \mathbf{H}_\infty(R_1) + \mathbf{H}_\infty(R_2) - L_{S_1} - L_{S_2}.$$

7.4.2 Cascaded Structure for Multiple Secrets

In some systems, it may be desirable to use more than two secrets. For example, in a multi-factor system, a user credential may include a fingerprint, a smartcard and a PIN, or two fingerprints and a password. Unlike the two secret case, there are many different cascaded strategies to mix the secrets.

Given secrets $b_1, b_2, \ldots, b_s$ and the corresponding sketches $S_1, S_2, \cdots, S_s$, the following are the main strategies to mix them, assuming we have mixing functions $f_1, \cdots, f_{s-1}$.

1. (Fanning) Apply mixing functions f_i on K_1 and S_{i+1} for all $1 \leq 1 \leq s-1$.

2. (Chaining) Apply mixing function f_i on K_i and S_{i+1} for all $1 \leq 1 \leq s-1$.

3. (Hybrid) Use a combination of fanning, chaining and independent encoding. For example, we can mix K_1 with S_2 and S_3, and further mix K_2 with S_4, but b_5 is encoded independently, and so on.

With the fanning approach, the entropy loss would be mostly diverted to the first secret, which may be the most easily revocable and replaceable secret. However, this approach requires that the first secret has sufficiently high entropy, since otherwise it may be relatively easy to obtain the first secret from the mixed sketch. In practice, this approach can be used when a long revocable key is available, such as key stored in a smartcard.

On the other hand, using the chaining approach only requires that the entropy of the i-th secret is sufficient to mix with the $(i+1)$-th sketch. In this case, the secrets should be mixed in the order of their "importance", which could be, for example, the ease of revocation and replacement, or the likelihood of being lost or stolen. Note that in this approach, it is crucial to determine the exact order of importance of the secrets.

If no single secret is of sufficient entropy, and the order of importance among secrets is not always clear, a hybrid approach may become more appropriate. As a special case, when all secrets are short and no secret is

more important than others, it would not be advisable to use the mixing approach and a straightforward method can be better.

7.5 Summary and Guidelines

In this chapter, we describe and compare different approaches to handle multiple secrets in biometric authentication application. Our analysis shows that with proper construction, the information leakage of the more important secret can be "diverted" to the less important ones. We give some guidelines for the application of cascaded mixing functions to two secrets. The same principles apply to multiple secrets.

1. If the importance of the secrets cannot be determined or is the same for both secrets, mixing is not recommended.

2. For the more important secret, if there are two secure sketch schemes that differ only in the amount of randomness used in the construction; choose the one that uses less randomness.

3. If the randomness invested cannot be decoupled from the sketch, cascaded mixing is not advisable unless the length of consistent key is longer than the length of the sketch.

Chapter 8

Conclusion

In this dissertation, we studied the problem of privacy protection of sensitive personal data. We focus on information theoretic secure mechanisms that provide unconditional security on controlling information leakages in two scenarios: data publishing and biometric authentication. In both scenarios, we seek to extend the existing privacy protection techniques to cater for some commonly deployed setting. In the enhanced construction, we show that we can achieve a better privacy-utility tradeoff.

We extend biometric protection mechanisms to cater for asymmetric setting and multi-factor setting. The extensions provide better privacy protections with respect to the remaining entropy of the biometric secret. We also give a differentially private mechanism for publishing pointset data. We proposed a notion of δ-neighbourhood that can be more appropriate under certain scenarios, and we give constructions for spatial dataset and temporal dataset which the notion can provide a good tradeoff for better utility. It is interesting to study whether our proposed notions can be

applied in other domains to provide stronger privacy protection.

References

[Ada00] J. Adams. Biometrics and smart cards. *Biometric Technology Today*, pages 8–11, 2000.

[Agg05] C. C. Aggarwal. On k-anonymity and the curse of dimensionality. *31st International Conference on Very Large Data Bases*, pages 901–909, 2005.

[BCD+07] B. Barak, K. Chaudhuri, C. Dwork, S. Kale, F. McSherry, and K. Talwar. Privacy, accuracy, and consistency too: a holistic solution to contingency table release. *symposium on principles of database systems*, pages 273–282, 2007.

[BDK+05] X. Boyen, Y. Dodis, J. Katz, R. Ostrovsky, and A. Smith. Secure remote authentication using biometric data. In *Eurocrypt*, pages 147–163, 2005.

[BDMN05] A. Blum, C. Dwork, F. McSherry, and K. Nissim. Practical privacy: the sulq framework. *ACM Symposium on Principles of Database Systems*, pages 128–138, 2005.

[Boy04] X. Boyen. Reusable cryptographic fuzzy extractors. In *Computer and Communications Security*, pages 82–91, 2004.

[BWJ05] C. Bettini, X. Wang, and S. Jajodia. Protecting privacy against location-based personal identification. *Secure Data Management*, pages 185–199, 2005.

[CKL03] T.C. Clancy, N. Kiyavash, and D.J. Lin. Secure smartcard-based fingerprint authentication. In *ACM Workshop on Biometric Methods and Applications*, pages 45–52, 2003.

[CL06] E.C. Chang and Q. Li. Hiding secret points amidst chaff. In *Eurocrypt*, pages 59–72, 2006.

[CPS+12] Graham Cormode, Cecilia Procopiuc, Divesh Srivastava, Entong Shen, and Ting Yu. Differentially private spatial decompositions. In *International Conference on Data Engineering*, pages 20–31, 2012.

[CST06] E.C. Chang, R. Shen, and F.W. Teo. Finding the original point set hidden among chaff. In *ACM Symposium on Information, computer and communications security*, pages 182–188, 2006.

[DKM+06] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor. Our data, ourselves: Privacy via distributed noise generation. *Advances in Cryptology-EUROCRYPT*, pages 486–503, 2006.

[DMNS06] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. *Theory of Cryptography*, pages 265–284, 2006.

[DNP+10] C. Dwork, M. Naor, T. Pitassi, G.N. Rothblum, and S. Yekhanin. Pan-private streaming algorithms. *Innovations in Computer Science*, 2010.

[DNPR10] C. Dwork, M. Naor, T. Pitassi, and G.N. Rothblum. Differential privacy under continual observation. *Proceedings of the 42nd ACM symposium on Theory of computing*, pages 715–724, 2010.

[DRS04] Y. Dodis, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In *Eurocrypt*, pages 523–540, 2004.

[Dwo06] C. Dwork. Differential privacy. *Automata, languages and programming*, pages 1–12, 2006.

[FFKN09] D. Feldman, A. Fiat, H. Kaplan, and K. Nissim. Private coresets. *Theory of computing*, page 361, 2009.

[FH07] D. Florencio and C. Herley. A large-scale study of web password habits. In *Proceedings of the 16th international conference on World Wide Web*, pages 657–666, 2007.

[FWCY10] B. Fung, K. Wang, R. Chen, and P.S. Yu. Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys*, pages 14–57, 2010.

[FWY05] B. Fung, K. Wang, and P. Yu. Top-down specialization for information and privacy preservation. *International Conference on Data Engineering*, pages 205–216, 2005.

[GL96] C. Gotsman and M. Lindenbaum. On the metric properties of discrete space-filling curves. *IEEE Transactions on Image Processing*, pages 794–797, 1996.

[GL04] B. Gedik and L. Liu. A customizable k-anonymity model for protecting location privacy. In *ICDCS*, pages 620–629, 2004.

[GW84] S.J. Grotzinger and C. Witzgall. Projections onto order simplexes. *Applied mathematics and optimization*, pages 247–270, 1984.

[HA03] P. Ho and J. Armington. A dual-factor authentication system featuring speaker verification and token technology. In *Audio- and Video-Based Biometric Person Authentication*, pages 128–136, 2003.

[HJK+08] S. Hong, W. Jeon, S. Kim, D. Won, and C. Park. The vulnera-

bilities analysis of fuzzy vault using password. In *Future Generation Communication and Networking*, pages 76–83, 2008.

[HRMS10] M. Hay, V. Rastogi, G. Miklau, and D. Suciu. Boosting the accuracy of differentially private histograms through consistency. *VLDB Endowment*, pages 1021–1032, 2010.

[JRP04] A.K. Jain, A. Ross, and S. Prabhakar. An introduction to biometric recognition. *Circuits and Systems for Video Technology*, pages 4–20, 2004.

[JS06] A. Juels and M. Sudan. A fuzzy vault scheme. *Designs, Codes and Cryptography*, pages 237–257, 2006.

[JW99] A. Juels and M. Wattenberg. A fuzzy commitment scheme. In *Computer and communications security*, pages 28–36, 1999.

[KGK+07] E. Kelkboom, B. Gókberk, T. Kevenaar, A. Akkermans, and M. van der Veen. "3d face": Biometric template protection for 3d face recognition. *Advances in Biometrics*, pages 566–573, 2007.

[Kle90] D.V. Klein. Foiling the cracker: A survey of, and improvements to, password security. In *USENIX Security Workshop*, pages 5–14, 1990.

[KM11] D. Kifer and A. Machanavajjhala. No free lunch in data privacy. *Management of data*, pages 193–204, 2011.

[KMD+10] B. Kaluža, V. Mirchevska, E. Dovgan, M. Luštrek, and M. Gams. An agent-based approach to care in independent living. *Ambient Intelligence*, pages 177–186, 2010.

[KY08] A. Kholmatov and B. Yanikoglu. Realization of correlation attack

against the fuzzy vault scheme. *Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, 2008.

[LGC08] Q. Li, M. Guo, and E.C. Chang. Fuzzy extractors for asymmetric biometric representations. In *Computer Vision and Pattern Recognition Workshops*, pages 1–6, 2008.

[LHR+10] C. Li, M. Hay, V. Rastogi, G. Miklau, and A. McGregor. Optimizing linear counting queries under differential privacy. *symposium on Principles of database systems of data*, pages 123–134, 2010.

[LLV07] N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, pages 106–115, 2007.

[LT03] J.P. Linnartz and P. Tuyls. New shielding functions to enhance privacy and prevent misuse of biometric templates. In *Audio-and Video-Based Biometric Person Authentication*, pages 1059–1059, 2003.

[MD86] G. Mitchison and R. Durbin. Optimal numberings of an n x n array. *Algebraic Discrete Methods.*, pages 571–582, 1986.

[Mey08] M. C. Meyer. Inference using shape-restricted regression splines. *Annals of Applied Statistics*, pages 1013–1033, 2008.

[MKA+08] A. Machanavajjhala, D. Kifer, J. Abowd, J. Gehrke, and L. Vilhuber. Privacy: Theory meets practice on the map. *International Conference on Data Engineering*, pages 277–286, 2008.

[MKGV07] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasub-

ramaniam. ℓ-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, pages 3–15, 2007.

[MRW99] F. Monrose, M. Reiter, and S. Wetzel. Password hardening based on keystroke dynamics. In *Proceedings ACM Conf. Computer and Communications Security*, pages 73–82, 1999.

[MT79] R. Morris and K. Thompson. Password security: A case history. *Communications of the ACM*, pages 594–597, 1979.

[MYCC04] YS. Moon, HW. Yeung, KC. Chan, and SO. Chan. Template synthesis and image mosaicking for fingerprint registration: An experimental study. In *Acoustics, Speech, and Signal Processing*, pages 405–409, 2004.

[NNJ07] K. Nandakumar, A. Nagar, and A.K. Jain. Hardening fingerprint fuzzy vault using password. In *Advances in Biometrics International Conference*, pages 927–937, 2007.

[NRS97] R. Niedermeier, K. Reinhardt, and P. Sanders. Towards optimal locality in mesh-indexings. *Fundamentals of Computation Theory*, pages 364–375, 1997.

[NRS07] K. Nissim, S. Raskhodnikova, and A. Smith. Smooth sensitivity and sampling in private data analysis. *ACM Symposium on Theory of Computing*, pages 75–84, 2007.

[PHIS96] V. Poosala, P.J. Haas, Y.E. Ioannidis, and E.J. Shekita. Improved histograms for selectivity estimation of range predicates. *ACM SIGMOD Record*, pages 294–305, 1996.

[PPJ08] Unsang Park, Sharath Pankanti, and AK Jain. Fingerprint verification using sift features. In *SPIE Defense and Security Symposium, Biometric Technology for Human Identification*, pages 1–9, 2008.

[PSC84] G. Piatetsky-Shapiro and C. Connell. Accurate estimation of the number of tuples satisfying a condition. *ACM SIGMOD*, pages 256–276, 1984.

[RGT97] Y. Rubner, L.J. Guibas, and C. Tomasi. The earth movers distance, multi-dimensional scaling, and color-based image retrieval. *ARPA Image Understanding Workshop*, pages 661–668, 1997.

[RU11] C. Rathgeb and A. Uhl. A survey on biometric cryptosystems and cancelable biometrics. *EURASIP Journal on Information Security*, pages 1–25, 2011.

[Sha01] C.E. Shannon. A mathematical theory of communication. *Mobile Computing and Communications Review*, 5(1):3–55, 2001.

[Sil75] S.D. Silvey. *Statistical inference*, volume 7. Chapman & Hall/CRC, 1975.

[SLM07] Y. Sutcu, Q. Li, and N. Memon. Protecting biometric templates with sketch: Theory and practice. *Transactions on Information Forensics and Security*, pages 503–512, 2007.

[SR01] R. Sanchez-Reillo. Including biometric authentication in a smart card operating system. In *Audio and Video Based Biometric Person Authentication*, pages 342–347, 2001.

[SRS+99] C. Soutar, D. Roberge, A. Stoianov, R. Gilroy, and B.V.K.V.

Kumar. Biometric encryption. *ICSA Guide to Cryptography*, pages 649–675, 1999.

[Sto00] Q. F. Stout. Optimal algorithms for unimodal regression. *Computer Science and Statistics*, pages 109–122, 2000.

[STP09] K. Simoens, P. Tuyls, and B. Preneel. Privacy weaknesses in biometric sketches. In *Symposium on Security and Privacy*, pages 188–203, 2009.

[Swe02] L. Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.

[TAK+05] P. Tuyls, A. Akkermans, T. Kevenaar, G.J. Schrijen, A. Bazen, and R. Veldhuis. Practical biometric authentication with template protection. In *Audio-and Video-Based Biometric Person Authentication*, pages 436–446, 2005.

[TG04] P. Tuyls and J. Goseling. Capacity and examples of template-protecting biometric authentication systems. *Biometric Authentication*, pages 158–170, 2004.

[TTT99] K.C. Toh, M.J. Todd, and R.H. Tütüncü. Sdpt3–a matlab software package for semidefinite programming, version 1.3. *Optimization Methods and Software*, pages 545–581, 1999.

[TTT03] R.H. Tütüncü, K.C. Toh, and M.J. Todd. Solving semidefinite-quadratic-linear programs using sdpt3. *Mathematical programming*, pages 189–217, 2003.

[UPPJ04] U. Uludag, S. Pankanti, S. Prabhakar, and A.K. Jain. Biomet-

ric cryptosystems: Issues and challenges. *Proceedings of the IEEE*, pages 948–960, 2004.

[web] Twitter census: Twitter users by location. `http://www.infochimps.com/datasets/twitter-census-twitter-users-by-location`.

[WL08] X. Wang and F. Li. Isotonic smoothing spline regression. *Booktitle Computational and Graphical Statistics*, pages 21–37, 2008.

[XWG10] X. Xiao, G. Wang, and J. Gehrke. Differential privacy via wavelet transforms. *IEEE Transactions on Knowledge and Data Engineering*, pages 1200–1214, 2010.

[XXY10] Y. Xiao, L. Xiong, and C. Yuan. Differentially private data release through multidimensional partitioning. *Secure Data Management*, pages 150–168, 2010.

[XY04] S. Xu and M. Yung. k-anonymous secret handshakes with reusable credentials. *11th ACM Conference on Computer and Communications Security*, pages 158–167, 2004.

[YF04] G. Yao and D. Feng. A new k-anonymous message transmission protocol. *5th International Workshop on Information Security Applications*, pages 388–399, 2004.