PRIVACY PRESERVING AND ROBUST WATERMARKING ON SEQUENTIAL GENOME DATA USING BELIEF PROPAGATION AND LOCAL DIFFERENTIAL PRIVACY

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR

THE DEGREE OF

MASTER OF SCIENCE

IN

COMPUTER ENGINEERING

By

Abdullah Çağlar Öksüz

August 2020

Privacy Preserving and Robust Watermarking on Sequential Genome Data using Belief Propagation and Local Differential Privacy
By Abdullah Çağlar Öksüz
August 2020

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Uğur Güdükbay(Advisor)

A. Ercüment Çiçek

Öznur Taştan Okan

Approved for the Graduate School of Engineering and Science:

Ezhan Karaşan
Director of the Graduate School

ABSTRACT

PRIVACY PRESERVING AND ROBUST WATERMARKING ON SEQUENTIAL GENOME DATA USING BELIEF PROPAGATION AND LOCAL DIFFERENTIAL PRIVACY

Abdullah Çağlar Öksüz
M.S. in Computer Engineering
Advisor: Uğur Güdükbay
Co-Advisor: Erman Ayday
August 2020

Genome data is a subject of study for both biology and computer science since the start of Human Genome Project in 1990. Since then, genome sequencing for medical and social purposes becomes more and more available and affordable. For research, these genome data can be shared on public websites or with service providers. However, this sharing process compromises the privacy of donors even under partial sharing conditions. In this work, we mainly focus on the liability aspect ensued by unauthorized sharing of these genome data. One of the techniques to address the liability issues in data sharing is watermarking mechanism. In order to detect malicious correspondents and service providers (SPs) -whose aim is to share genome data without individuals' consent and undetected-, we propose a novel watermarking method on sequential genome data using belief propagation algorithm.

In our method, we have three criteria to satisfy. (i) Embedding robust watermarks so that the malicious adversaries can not temper the watermark by modification and are identified with high probability (ii) Achieving ϵ-local differential privacy in all data sharings with SPs and (iii) Preserving the utility by keeping the watermark length short and the watermarks non-conflicting. For the preservation of system robustness against single SP and collusion attacks, we consider publicly available genomic information like Minor Allele Frequency, Linkage Disequilibrium, Phenotype Information and Familial Information. Also, considering the fact that the attackers may know our optimality strategy in watermarking, we incorporate local differential privacy as plausible deniability factor that induces malicious inference strength. As opposed to traditional differential privacy-based

data sharing schemes in which the noise is added based on summary statistic of the population data, noise is added in local setting based on local probabilities.

Keywords: Watermarking, local differential privacy, genomics, belief propagation.

ÖZET

DİZİSEL GENETİK VERİLER İÇİN İNANÇ YAYIMI VE LOKAL DİFERANSİYEL GİZLİLİK KULLANILARAK OLUŞTURULAN GÜÇLÜ VE GİZLİLİK KORUYUCU FİLİGRAN TEKNİKLERİ

Abdullah Çağlar Öksüz
Bilgisayar Mühendisliği, Yüksek Lisans
Tez Danışmanı: Uğur Güdükbay
İkinci Tez Danışmanı: Erman Ayday
Ağustos 2020

Genetik veriler 1990 yılında başlayan İnsan Genetik Verisi Projesi'nden beri hem biyolojinin hem de bilgisayar bilimlerinin çalışma alanlarından biri olmuştur. O zamandan bu yana genetik veri dizilimi, hem sağlık sektörü için hem de sosyal kullanım için gittikçe daha ulaşılabilir ve karşılanabilir hale gelmiştir. Üzerlerinde araştırma yapılabilmesi adına bu genetik veriler hem halka açık internet sitelerinde hem de servis sağlayıcıları aracılığıyla paylaşılabilmektedir. Ancak, bu paylaşımlar kısmi yapıldığı durumlarda bile paylaşımcıların veri gizliliklerini (mahremiyetlerini) tehlikeye atmaktadır. Bu çalışmamızda, verilerin izinsiz paylaşım durumundaki sorumlu tutulabilme ilkesini odaklanılmıştır. Sorumlu tutulabilmeyi yüksek olasılıklarla garanti edebilmek için kullanılan yöntemlerden biri de filigran tekniğidir. Paylaşımcıların izni olmaksızın, verileri ifşa olmadan paylaşabilmek isteyen servis sağlayıcılarına karşı, inanç yayımı tekniği aracılığıyla genetik verinin üzerine uygulanabilecek yeni bir filigran metodu öneriyoruz.

Yeni yöntemimizde, sağlanması hedeflenen üç kriter belirlenmiştir. Birincisi, kötü niyetli servis sağlayıcılarının silme ve değiştirme yapma ihtimaline karşın dayanıklı, onları yüksek olasılıklarla tespit edebilecek filigranlar üretmektir. İkincisi, herhangi bir servis sağlayacısıyla paylaşım yapmak için oluşturulan bütün filigranların ϵ-lokal diferansiyel gizliliği sağlamasıdır. Üçüncü kriter ise filigranların oluşturulması sırasında filigran uzunluğunu -veriden sağlanan faydayı yüksek tutmak adına- olabildiğince kısa ve etkili tutmak, ve bu filigranların gerçek genetik veriden ayırt edilememesini sağlamaktır. Oluşturulan filigranların servis sağlayıcıları tarafından "tek servis sağlayıcı saldırısı" ve "iş birliği saldırısı"

kullanılarak bozulmasını engellemek için genetik verilerle ilgili halka açık istatistiki değerler kullanılmaktadır. Bu değerler çekinik gen frekansı, bağlayış denksizliği, fenotip özellikleri ve aile bireylerinin genetik dizilimleri olabilir. Ayrıca servis sağlayıcılarının filigran oluşturma yöntemimizi ayrıntılarıyla bildiklerini varsayarak, olasılıksal bir inandırıcı yadsınabilirlik garantileyen ve okunan verinin kesinliğini azaltan lokal diferansiyel gizlilik yöntemi sistemimizde yer almaktadır. İstatistiki bilgilere göre verinin içine rastgele gürültü ekleyen, geleneksel diferansiyel gizlilik yöntemlerinden farklı olarak sistemimiz gürültüyü lokal olasılıklara bağlı kalarak eklemektedir.

Anahtar sözcükler: Filigran, lokal diferansiyel gizlilik, genetik, inanç yayımı.

Acknowledgement

First of all, I would like to thank my advisors Erman Ayday and Uğur Güdükbay for their continuous supports and encouragements throughout my research. It would have been impossible to complete this thesis without their guidance. Also, they taught me how to be a good academician which I'm thoroughly grateful for.

I would like to thank Ahmet Furkan Güç and Lütfi Kerem Şenel for their over 10 years of amazing friendship and for setting excellent examples to me in academia.

I would like to thank Yücel Şanlı and Onur Karakaşlar (a.k.a Balan'ars) for all the fun times that usually ended up in Corvus Pub. Without them and the eventful nights we had, this thesis couldn't be written. Those were the days my friends.

I would like to thank my office friends Alper, Cihan, Furkan, Gizem, Miray and Ömer for all the good times and coffee breaks. Then, I would like to thank Sinem Sav for her continuous guidance and support as an academician and friend whenever I struggled.

Finally, I would like to thank to the people I value above all else, my family. They've helped me to become a person who I am right now. No words are enough for me to describe how amazing they are and they've always been. Thanks to their immeasurable support and joy I believe, I've accomplished all the things I wanted and will continue to do so.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Digital watermarking is one of the most important technological milestones in digital data hiding. It is used as a technique to hide a message or pattern within the data itself for various reasons like copyright protection or source tracking of digitally shared data. Watermarks may contain information about the legal owner of data, distribution versions, and access rights [1]. Although watermarking has a wide range of applications, implementation schemes require different configurations for each use case and data type. For embedding copyright information in data and source tracking, robustness [2] against modifications is the crucial factor to preserve. The factors influencing such configurations alter depending on the characteristics of data, as well. Noise intolerance, the existence of correlations and prior knowledge for inference, and the preservation of utility requirements in sequential data are such factors that prevent the explicit implementation of digital data watermarking methods on sequential data. Noise intolerance refers to the non-existence of redundancy likewise digital data (e.g., slight color differences in image data) in which the watermark may be hidden, or may not be present in sequential data. The existence of correlations and the prior knowledge refer to the increased conditional probabilities of inference that may not be present in digital data. Finally, utility preservation refers to the differing informativeness weights of each data point so that some points must be avoided from slight changes, again may not be present in digital data.

We propose a novel watermarking scheme for sharing sequential genomic data consisting of three Single Nucleotide Polymorphism (SNP) (see § 3.1.2) states to be used by medical service providers (SPs). Each SP has access to the uniquely watermarked version of some individuals' genomic data. The preliminaries expected from this watermarking scheme are robustness against watermark tampering attacks such as modification and removal, imperceptibility for not revealing watermark locations, utility preservation of original data through a minimum number of changes and satisfy local differential privacy in watermarks so that the watermarked versions of the data are indistinguishable from actual human genomic data. By doing so, the watermarked data will not be shared in an unauthorized way and the source(s) of a leak will be easily identified by the data owner. To solve this multi-objective optimization problem, we use the belief propagation algorithm (see § 3.2), which helps us to determine optimal watermarking indexes on data that may preserve robustness with the highest probabilities when attacked and decrease the utility of data the least when changed. Apart from utility preservation and robustness concerns, public knowledge about the human genome like minor allele frequencies (MAF) (see § 3.1.3) of SNPs, point-wise correlations between SNPs, called Linkage Disequilibrium (LD) (see § 3.1.4), and the prior knowledge of genotype and phenotype information potentially leak probabilities about watermarked points if not considered. Through converting each prior information (MAF, LD, and so on) into the marginal probability distribution of the three SNP states for the belief propagation algorithm, we manage to infer the state probabilities of each SNP.

Our contributions are as follows:

1. We introduce a novel method for watermarking sequential data concerning the privacy of data and the robustness of watermark at the same time. We present the method's strengths and weaknesses in various attack scenarios and provide insight into the weaknesses.

2. Our method uses prior knowledge (MAFs, phenotype information, and so on) and inherent correlations to infer the state probabilities of SNPs. Using these inferred probabilities, we select SNPs that satisfy the following

two criteria in a non-deterministic setup: a low probability of robustness decrease (change resistance) when attacked and a low utility loss (efficient index selection) when changed. By giving priority to these SNP points for watermarking, we guarantee the preservation of robustness and utility in data against various attacks. Besides, the identification probabilities of single SNPs using prior knowledge are decreased with this method.

3. We test the robustness and limitations of our method using collusion (i.e., a comparison using multiple watermarked copies of the same data) and modification attacks and demonstrate how to reach a high probability of detection with various parameters, such as the watermark length, the number of SPs, the number of malicious SPs and the ϵ coefficient of local differential privacy.

4. We introduce randomly distributed non-genome-conflicting noise generated for the data to act naturally as watermarks and create imperceptible watermark patterns from the normal human genome if not attacked with collusion. Hence, rather than creating a fixed number of point-wise changes and tracking these changes for source tracking, we evaluate the whole data and reach a high probability of detection with a minimum number of changes.

5. We introduce watermarking schemes that satisfy ϵ-local differential privacy and plausible deniability in data along with it for data owners who value additional manners of enhanced privacy at the expense of robustness.

The rest of the thesis is organized as follows. In Chapter 2, we discuss related works on digital watermarking, its use in digital data hiding, genomic data privacy and security, and the concept of local differential privacy. Chapter 3 provides background knowledge for the sequential genomic data and the inference model we use. Chapter 4 introduces the problem definition and objective function. We elaborate on data and system models along with attack scenarios. In Chapters 5 and 6, we propose our solution and give the details of the implementation setup. In Chapter 7, we evaluate the metrics and the results of our proposed algorithm. Chapter 8 concludes the thesis by giving insight on possible future work.

Chapter 2

Background

We provide preliminary information required to comprehend the algorithmic setup that we used to test our solution on in the sequel.

2.1 Genomics

We implement our sequential data watermarking system using genomic data obtained from the 1000 Genomes Project [3]. In this section, we briefly introduce some genomics concepts that are essential to grasp the algorithmic setup.

2.1.1 Mendel's Law of Segregation and Law of Dominance

Mendel's Law of Segregation and Law of Dominance [4] are two of the hereditary principles collectively known as Mendelian Inheritance discovered by Gregor Mendel in the 1860s. Four main concepts related to these laws are (i) the existence of a gene in the form of more than one allele, (ii) the inheritance of two alleles for each trait, (iii) the separation of alleles in sex cell production (meiosis), and (iv) the explanation of different alleles in a pair as dominant and recessive.

Besides, according to the law of independent assortment, it is known that these alleles are inherited from the mother and the father each with equal probabilities without the interference of other genes even for the different allele pairs.

2.1.2 Single Nucleotide Polymorphism

Single Nucleotide Polymorphism (SNP) [5] is a point variation on the DNA when a single nucleotide (e.g., adenine, thymine, guanine, cytosine) difference occurs between individual members of populations. For example, fragments of sequenced DNA obtained from two individuals are AAGCCTG to AAGACTG. Variance occurring in the fourth index is SNP and it has two alleles, C and A, and almost all common SNPs have only two alleles that each one is inherited from the mother and the father according to Mendel's law of segregation. In two randomly selected human genomes, there is a similarity rate of 99.9% and SNP is the most common type of variation in the remaining 0.1%. However, 0.1% difference consists of ten million SNPs which are associated with differences in phenotype (e.g., eye color and hair type) and genotype (e.g., susceptibility to diseases like diabetes and schizophrenia) features [6].

2.1.3 Minor Allele Frequency

Two alleles that are inherited from the mother and the father might be the same or not. The genetic condition is called *homozygous* if the alleles are the same. Otherwise, the condition is called *heterozygous* [7]. Depending on the occurrence rate of a particular allele in the population, the more frequent allele is called the *major allele*, whereas the other is called the *minor allele*. That being said, homozygous genes are further divided into two categories as *homozygous major* and *homozygous minor*, based on which allele pair is inherited from the parents. The frequency of minor alleles prompts an immense amount of selection in heritability and therefore, recorded as publicly available data for medical institutions [8].

Using these publicly available Minor Allele Frequency (MAF) values, the probability of each genetic state in populations can be inferred using the following equations:

$$AA = 0 : P(Homozygous\ Major) = (1 - MAF)^2$$
$$Aa = 1 : P(Heterozygous) = 2 \times (MAF) \times (1 - MAF)$$
$$aa = 2 : P(Homozygous\ Minor) = (MAF)^2$$

2.1.4 Linkage Disequilibrium

Linkage disequilibrium (LD) is the non-random heritable associations of alleles in different loci [9]. Because it has an indication of population genetic forces on the genome formation, it is a widely investigated and exploited research topic in evolution and demographics studies [10]. Factors that have an impact on LD may vary due to genetic reshuffling, mutation rate, allelic drift, and so on. In genomic privacy, LD can be used to infer the state probabilities and hence values of multiple SNPs in correlated loci given the state value of a single SNP. Therefore, highly correlated states can be used for the enhancement of beliefs in belief propagation setup on the other SNPs. By exploring all the coexisting pairs of SNPs in the large sample population, linkage disequilibrium loci of high correlation values can be identified.

2.2 Belief Propagation Algorithm

Belief Propagation (BP), also known as sum-product message passing, is a message-passing algorithm used for the inference of networks and graphs like Bayesian Networks and Markov Networks [11]. BP calculates marginal probability distributions of unknown variables in factor graphs in an iterative manner using the information from previous states. In a factor graph, two types of nodes are used: (i) Factor Nodes, and (ii) Variable Nodes (cf. Chapter 5). BP is a widely

used technique in graphs because the marginal probability computation of variables that have a dependency on multivariate data (factors) gets exponentially complex as the number of factors increases. Moreover, marginal probabilities of factors must be re-computed given the new distribution of variables. With a finite number of iterations, BP approximates the actual distribution with less complexity.

2.3 Local Differential Privacy

Differential Privacy is a system of public data sharing that uses the patterns of groups in the dataset and while doing so without compromising the privacy of individuals in the dataset [12]. The main intuition is that an algorithm is differentially private if the use of any particular individual's data cannot be inferred from the computations. If any inference probability is limited to the upper bound of $\rho < \epsilon$ in the dataset, the algorithm is ϵ-differentially private. ϵ-differential privacy is derived for a process A if Equation 2.1 is satisfied in any two neighboring databases D_1 and D_2 with an outcome O.

$$P[A(D_1) = O] \leq e^{\epsilon} \times P[A(D_2) = O]. \tag{2.1}$$

Equation 2.1 is symmetrical and valid for any two neighboring databases D_1 and D_2. So, this equation can also be written as:

$$e^{-\epsilon} \times P[A(D_2) = O] \leq P[A(D_1) = O] \leq e^{\epsilon} \times P[A(D_2) = O]. \tag{2.2}$$

Local Differential Privacy (LDP) is the localized version of differential privacy that targets not the datasets or databases but the data indices. In LDP, the data is intentionally perplexed by the data owners so that plausible deniability is ensured without a "trusted party." The privacy assured by data owners is expressed as ϵ-local differential privacy. This ϵ value can be thought to provide

$\frac{100}{e^{\epsilon}+1}\%$ plausible deniability. As ϵ gets smaller, although the outcomes become less likely to be different from one another, more privacy is ensured. In summary, LDP is the local implementation and satisfying Equation 2.2 on every single data point or sequential data in our use case. It benefits our system as an additional privacy preservation measure. It is a technology adopted by major technology firms like Google, Apple, and Microsoft for collecting mass anonymized data like web browsing behaviors, typing behaviors, and telemetry data [13].

Chapter 3

Related Works

We present the related work on the security and privacy of genomic data and digital watermarking.

3.1 Security and Privacy of Genomic Data

Recent advances in the field of molecular biology and genetics and next-generation sequencing increased the amount of genomics data significantly [14]. While achieving a breakthrough in the genomics field, genomics data posses an important privacy risk for individuals by carrying sensitive information, i.e., kinship, disease, disorder, or pathological conditions [14, 15]. Thus, collecting, sharing, and conducting research on the genomic data became difficult due to privacy regulations [16, 17]. Further, Humbert et al. [18] show that sharing genomic data also threatens the relatives due to kin genomic data. To this end, several works have been conducted to find emerging ways of privacy-preserving collection and analysis of the genomic and medical data in the last decade.

Along the research direction of privacy-preserving medical data collection, several works have focused on using well-known privacy techniques such as

k-anonymity, l-diversity, de-identification, perturbation, anonymization, or t-closeness [19, 20, 21, 22, 23, 24, 25, 26, 27]. These methods, however, provide limited privacy protection, are prone to inference attacks, and tend to decrease the utility of the data [22]. Ayday et al. [28] propose obfuscation methods in which the output domain is divided into several sections and one section is returned as an output for genomic data protection.

It is shown applying anonymization techniques to genomic data still reveals significant information due to the successful inference attacks [29, 30]. To this end, cryptographic techniques, e.g., have been proposed [31, 32, 33, 34]. Furthermore, Karvelas et al. [35] propose a technique based on oblivious RAM to access genomic data and Huang et al. [36] propose an information-theoretical scheme for the storage of genomic data.

3.2 Digital Watermarking

Digital watermarking is a technique usually used for copy protection by inserting a pattern to the digital signal such as song, image, or video [37]. It is an attack counter-measure for the case of leakage or sharing without consent. It is worth mentioning that watermarking does not prevent leakage and it is used as a detection technique for the malicious parties.

By implementation, watermarking techniques might be classified by robustness, perceptibility, and features not-related to our implementation methods like capacity and embedding techniques. In terms of robustness, types of watermarking are fragile, semi-fragile, and robust watermarking. Robust watermarks are resistant to modifications and used for source tracking. On the other hand, fragile watermarks are the complete opposite of robust watermarks so that any slight change on the data will shift the watermark undetectable and they are used for tamper detection. Semi-fragile watermarks are in-between forms and resistant to benign modifications like robust watermarks but not resistant to malignant like fragile watermarks. In terms of perceptibility, types of watermarks are perceptible

and imperceptible watermarks. Perceptible watermarks are used as logos, opaque images for mainly authentication reasons. Imperceptible watermarks, however, can be used as source tracking agents, and their implementation is expected to be indistinguishable from the original data on that is used.

Digital watermarks are generally used for copy protection on multimedia data [38, 39, 40]. In such works, watermarking is used to encode copy information and to detect non-licensed copies of the multimedia. Another application field of watermarking is images [41, 42] by modifying the pixel values, substituting the least significant (LSB) bit of the pixels [43, 44], or using signal transforms such as Discrete Fourier Transform (DFT) [45] and Discrete Cosine Transform (DCT) [46, 47, 48].

The state-of-the-art solutions to apply watermark on audio signals usually rely on time-domain techniques such as the substitution of the LSB [49] or adding echo [50]. Quantization is another technique used for audio watermarking [51, 52]. Watermarking the text documents, on the other hand, requires different techniques such as using a line-shift or word-shift algorithm to move the lines/words upward or downwards and adds extra spaces in between them [53, 54]. Topkara et al. [55] propose a watermarking method that is based on features of the sentences and orthogonality between them. Atallah et al. [56, 57] propose the natural language watermarking scheme for text documents. Their solution makes use of a watermark bit string in the syntactic structure of the sentences. All these methods, however, are prone to collusion attacks where a malicious party might collude with other parties to detect the watermarking.

Boneh and Shaw [58] propose a fingerprinting approach for digital data that provides security against collusion attacks. Their method creates fingerprinting schemes that no combination of attackers may detect. However, in practicality, their method has some drawbacks for sequential data. First and foremost, their method does not address the sequential data inherent correlations or prior information known about the data and vulnerable to the attacks that use this information. Secondly, their method may create fingerprints so long to preserve robustness at the expense of utility [59] This long fingerprinting scheme may be

useful in data types in which the redundancy does not impact utility too much, but sequential data especially genomic data loses utility even in slight changes.

3.3 Watermarking Genomic Data

Watermarking schemes proposed for sequential data are limited and for genomic data specifically even more limited. Kozat et al. [60] proposed a steganography-based watermarking scheme for sequential electrocardiography data to hide private meta-data like the patient's social security number or birth date for data ownership authentication. Iftikhar et al.[61] proposed a robust and distortion-free watermarking scheme called GenInfoGuard for genomic data. Iftikhar's scheme uses features selected from the data for embedding a watermark on. Similar to our approach, Liss et al. [62] proposed a permanent watermarking scheme in synthetic genes that embeds binary string messages on open-frame synonymous amino-acid codon regions. Finally, Heider et al. [63] proposed the use of artificial dummy strands to act like watermarks on DNA.

Most recently, Ayday et al. [59] proposed a robust watermarking scheme for sharing sequential data against potential collusion attacks by using non-linear optimization. Our objective model is similar to theirs. However, different from their study, we consider the additive type of prior information scheme in which besides correlations, all sequential genomic data related information like familial genomes, phenotype states can be included by using factor nodes in belief propagation algorithm. Also, we designed a collusion attack that incorporates all the information that can be gained from single sp attacks and correlation attacks within so that the worst-case scenario is assumed and the attack model becomes more inclusive. Another difference between our method and theirs is the incorporation of ϵ-local differential privacy as an extra measure of privacy without impacting security. Andres et al. [64] proposed a method of embedding noise in sequential location data for geo-indistinguishability without violating the differential privacy. Inspired from their study and their new differential privacy criteria, we implemented a local set up so that extra criteria in the watermarking

process checks every data index against the differential privacy violations locally and prevents the violating versions from getting shared.

Chapter 4

Problem Definition

We present the data, system, and threat models, and the objective of our system. Frequently used symbols and notations are presented in Table 4.1.

4.1 Data Model

Sequential data contain ordered data points $x_1, x_2, \ldots, x_{d_l}$, where d_l is the length of the data. The values of x_i can be in different states from the set $\{y_1, y_2, ..., y_m\}$ depending on the type of the data. For example, x_i can be an hour, minute, or second triplets ranging from 0 to 23, 59, 59, respectively, for timestamp data. For our system, we will use 0, 1, and 2 for the SNP states of *homozygous major*, *heterozygous*, and *homozygous minor*, respectively. The length of the data is d_l and the number of points that will be watermarked at the end of the algorithm is w_l. For the remaining notations, please refer to Table 4.1.

Table 4.1: Frequently used symbols and notations

$x_1, ..., x_{d_l}$	Set of data points
$y_1, ..., y_m$	Possible values (states) of a data point
d_l	Length of the data
w_l	Length of the watermark
h	Total number of SPs
I_k	Index set of data points that are shared with k^{th} SP
J_k	Index set of data points that are watermarked for k^{th} SP
Z_k	Watermark pattern of k^{th} SP
W_k	Watermarked data shared with k^{th} SP
ϵ	Local differential privacy coefficient
$S_{I_i}^k$	Set of states for index i that are shared with the first k SPs

4.2 System Model

In our proposed system, we consider a system between data owner (Alice) and multiple service providers (SPs) with whom Alice shares sequential data as shown in Figure 1. Sequential data shared might vary, e.g. human genome data, text data, location data. For text data, the SP can be any service provider working on Natural Language Processing. For genome data, service providers can be medical researchers, medical institutions, or bio-technical companies. Alice may decide to share the whole data or parts of the data to receive different services. Also, the parts shared may differ for each SP.

For all the cases listed above, Alice wants to ensure that her data will not be shared unauthorized by the service providers and if shared anyway, she wants to preserve a degree of differential privacy and detect the malicious SP(s) who shared the data. Hence, she uses watermarking and shares a different watermarked version of the data all of which satisfy the degree of privacy desired by Alice with each SP. These different versions are produced through removing of certain parts or modifying the data. Therefore data indices most optimal for satisfying the criteria given above should be calculated beforehand with care by considering the structure, distribution, and vulnerabilities of the data. To calculate the complex probability distributions of multi-variable sequential data that satisfy Alice's demands, we use Belief Propagation(BP). Other graph inference

methods could have been used for the calculations but BP is adapted because of its fast approximation efficiency in non-loopy graph networks.

Watermarking is mostly done by changing the values (status) of data indices. Adding dummy variables is an example of methods that do not change the actual values but common methods used for watermarking are usually removal or modification. Since a slight addition in sequential data causes a shift in the other indices, it may impact the rest of the retrieval and embedding process like the butterfly effect. Therefore, we stuck with the watermarking method by removal or modification. Also in a broader sense, non-sharing can be considered as modifying the status of a certain index into "non-available". Normally, the security of a watermarking scheme increases along with the length of the watermark against the attacks discussed in the threat model section. However, a robust watermark should be short and as efficient as possible to maximize the detection probability of malicious SPs without reducing utility (percentage of data changed) by a lot.

Malicious SPs try to lower their chance of getting detected while leaking the data. If the system can not identify the source of leakage due to various attacks, SPs will avoid getting caught. To do so, SPs may tamper the watermarks via the same processes of embedding; by removal or modifications. Sharing a portion of data is an example of removal to avoid detection; however, this reduces the amount of information data contains. On the other hand, if the watermarked indices are known or inferred, changing the values of watermarked states rather than albeit removing them will help SPs to share the data undetected. Watermarked indices on the data can be found by the collaboration of multiple SPs who compare their versions of the data with each other by a collusion attack. Another method for finding the indices is using prior knowledge to infer the actual states of the data and looking for discrepancies by a single SP attack. Therefore, the belief propagation algorithm helps us to find the optimum indices that are vulnerable to these attacks very little and satisfy the conditions of minimum utility loss and maximized probability of detection against various attacks.

4.3 Threat Model

In the threat model, the goal of malicious SPs is to share the data undetected. This goal can be achieved by decreasing the robustness of the watermark which prevents the identification of the leakage source(s). Malicious SPs may identify high probability watermark points and tamper the watermark pattern by removal or modification. For such scenarios, we presume that malicious SPs will not do blind attacks without the prior knowledge of watermarked indices. These types of attacks will decrease the utility of data more than the robustness of the watermarking scheme and render the data useless. Hence, we introduce three attack models based on probabilistic identification that test the robustness of the watermarks that our proposed method generates.

Single SP Attack: In this attack, a single malicious SP is expected to use the prior information available to infer the actual states of the data and identify the watermarked indices without collaborating with other SPs. Examples of prior information include *minor allele frequencies*, *genotype* and *phenotype* information of parents for the genomic data or movement patterns, and frequently visited locations for the location data. For each data point, malicious SP finds the posterior probability of each state given the prior information $Pr(x_i = y|$ *prior information*$)$ and compares it with the expected probability of given state $x_i = y, y \in \{y_1, y_2, y_3, ..., y_m\}$. If the difference between the posterior and expected probabilities for the given state is high, it may be an indication of a watermarked index. We assume that the malicious SP knows the watermark length w_l, hence SP may select the top w_l indices with the highest differences in probability as watermarked and implement an attack.

Another vulnerability that malicious SPs may exploit is the use of inherent correlations and their values in the data to infer the actual states of correlated indices. For location data, these correlations might be previous location data within a close time interval. For genomic data, *linkage disequilibrium (LD)*; non-random association of certain alleles is an example of such correlations. LD is a property of certain alleles; not their loci. Therefore, correlation of alleles

$\{A, B\}$ in loci $\{I_A, I_B\}$ will not hold if either A or B changes. The asymmetric correlation observed in LD is a valid method of representation for other sequential data types. Hence, for the generalization of correlations in the implementation phase, our proposed system considers the correlations in the data as pairwise and asymmetric.

Collusion Attack: In addition to the knowledge obtained via a single SP attack, multiple SPs that receive the same proportion of data may vertically align their data to identify watermarked points. When SPs align their data, there will be indices with different states which can be considered as definitely watermarked. Normally, the proportion of data shared with SPs may differ, which will decrease the efficiency of alignment. However, for the construction of a stronger model against worst-case scenarios, the system considers the same data is shared with all SPs. Potentially watermarked indices received from collusion attack can be used along with prior information obtained from running a single SP attack and this attack type detects further more watermarked indices than the single SP attack.

4.4 Objective Model for the Detection of Malicious SP(s)

The objective of our proposed system and watermarking, in general, is identifying the source(s) of leakage when the data is shared unauthorized. An additional objective is to preserve a degree of privacy when these shared indices are compromised called ϵ-local differential privacy. In doing so, watermarks must be resilient against the attack models suggested in the Threat Model. In such cases, Alice can compare the leaked version with the original data and all the other versions shared with SPs. Later, points that are identified as different by our detection algorithms in the leaked version from the original data can be considered as watermarked/modified. As the watermark pattern is unique for each SP, Alice tries

to identify which points are modified or removed by someone else other than herself. Later she assesses the differences between the leaked version and the shared versions of SPs. Finally, she can infer the probability of which SPs are malicious.

For example, Alice shared her data with SPs $\{SP_1, SP_2, SP_3, SP_4\}$ with unique patterns $\{Z_1, Z_2, Z_3, Z_4\}$, respectively. By looking at the distinctive indices in which a watermark is present for one particular SP or combination of those indices, malicious SPs can be distinguished. However, as discussed in the collusion attack section, a collaboration between multiple SPs is a possible scenario. When SPs collaborate, a distinctive index that is watermarked in one SP will not appear on the other one. With this knowledge, SPs may find, remove, or modify these distinctive indices so that they won't get caught. If these collaborators are not caught it is a true negative case. Using a limited number of distinctive indices to keep the watermark short, puts the robustness of watermark at risk. When the watermark is too short, modification by malicious SPs may blame unauthorized sharing on the other SPs, resulting in even worse false-positive cases. Therefore, the watermark implemented on the real data must be long and scattered enough to give sufficient information even with its absence. On the other hand, the watermark modifies the data, and hence, decreases the utility of the data. Therefore, we want to ensure a watermarking scheme with small w_l to preserve the utility of data and watermark with long w_l to ensure the robustness as much as possible.

In the proposed system, we want to ensure watermarks that are both robust and not-violating differential privacy conditions. We will describe a novel method that regards robustness and privacy simultaneously (cf. Chapter 5). Then, we will evaluate the robustness of the watermarking scheme by precision results against the attack model and explore how much privacy can be achieved in a privacy-robustness trade-off (cf. Chapter 6).

Chapter 5

Proposed Solution

In this chapter, we describe the proposed watermarking scheme in detail. When Alice wants to share her data with SP_i, they employ the following protocol. The SP_i sends a request to Alice providing the indices required from her data, denoted as I_i. Then, Alice generates a list of available indices most suitable for watermarking J_i that satisfies $J_i \subset I_i$ and $|J_i| = w_l$. J_i is generated by the Belief Propagation Algorithm, which will be discussed in detail in the sequel. Finally, Alice inserts watermark into the indices of J_i. If the data is in binary form, it is as simple as changing 0 to 1 or vice versa. Otherwise, for the given state x_i, a different state y_i from the set $y_i \in \{y_1, y_2, ..., y_m\}$ and $y_i \neq x_i$ is chosen to be a part of the watermark pattern. In non-binary selection, if the given index contains correlation with other indices, the selection is determined by the probabilities and statistics of the correlated indices so that the watermark would not be vulnerable to the correlation attacks. Otherwise, it is a random selection with uniform distribution.

Our proposed method makes use of the belief propagation algorithm that uses prior information and previous shared versions of the data to identify indices with minimized utility loss and maximized detection probabilities of malicious SPs when modified for watermarking. Belief Propagation (BP), as discussed in § 2.2 and § 4.2, is an iterative message-passing algorithm used for the inference

of networks. The reason for using this algorithm is to infer the probability distributions of indices given the multi-variable prior information, attack scenarios, and privacy criteria. Normally, the factorization of prior information marginal probabilities could be used for a part of the inference of state probabilities. However, probability calculation gets exponentially complex as the dimensions of the data and the variety of prior information increases. Because BP approximates to the actual state probabilities in a finite number of iterations, it is much more efficient than factorized calculation. The main idea is to represent the probability distribution of variable nodes by factorization into products of local functions in factor nodes.

The steps of the Belief Propagation algorithm are as follows:

- The algorithm starts in a variable node with an initial probability distribution.
- The algorithm collects messages from the factor nodes for updating the probability distributions of the targeted unknown variable nodes. In loopy bilateral networks, this process is handled in iterations until convergence. However, this approach is changed to a top-to-bottom approach with one or two iterations for tree-like graph networks like ours for efficient approximation.
- Variable nodes generate the factor node messages by multiplying all incoming messages from the neighbors except the receiver neighbor.
- Factor nodes generate the messages by using local functions and send them to corresponding variable nodes.
- At the end of each iteration, the marginal probability distribution of each variable node is updated by multiplying all incoming messages from neighbors.
- The algorithm approximately calculates the beliefs of the variable nodes and passes it to the AE-node.

- The AE-node acts as a secondary factor node and calculates a new message that considers both attack scenarios and local differential privacy criteria.
- Finally, the AE-node passes its message together with variable node messages as parameters into the watermarking algorithm.

5.1 Nodes and Messages

In this section, we give the general setup and the details of the Belief Propagation (BP) Algorithm for Genomics data. BP consists of factor nodes, variable nodes, and messages between them. Connections between variable nodes and factor nodes are given in the factor graph (see Figure 5.1).

The notations for the messages are as follows:
$\mu^{v}_{i\to k}$: Message from variable node var_i to factor or attack-eDP node k at the v^{th} iteration.
$\beta^{v}_{i\to k}$: Message from familial nodes fam_i to variable node k at the v^{th} iteration.
$\omega^{v}_{i\to k}$: Message from phenotype nodes phe_i to variable node k at the v^{th} iteration.
$\lambda^{v}_{i\to k}$: Message from correlation node $c_{i,k}$ to variable node k at the v^{th} iteration.
$\delta^{v}_{i\to k}$: Message from attack-eDP node ae_i to be used as parameters for watermarking algorithm.

5.1.1 Variable Nodes

Variable nodes (decision nodes) represent the unknown variables, and each variable node sends and receives messages from factor nodes to learn and update its beliefs. Its main purpose is to infer the marginal state probabilities of all indices that can be obtained from prior information. For genomic data, this information is the publicly known statistics such as linkage disequilibrium (ld) correlations, familial genomic traits, and phenotype features. For each node we have a marginal probability distribution of states $y_1, y_2, ..., y_m$. Each variable node var_i represents

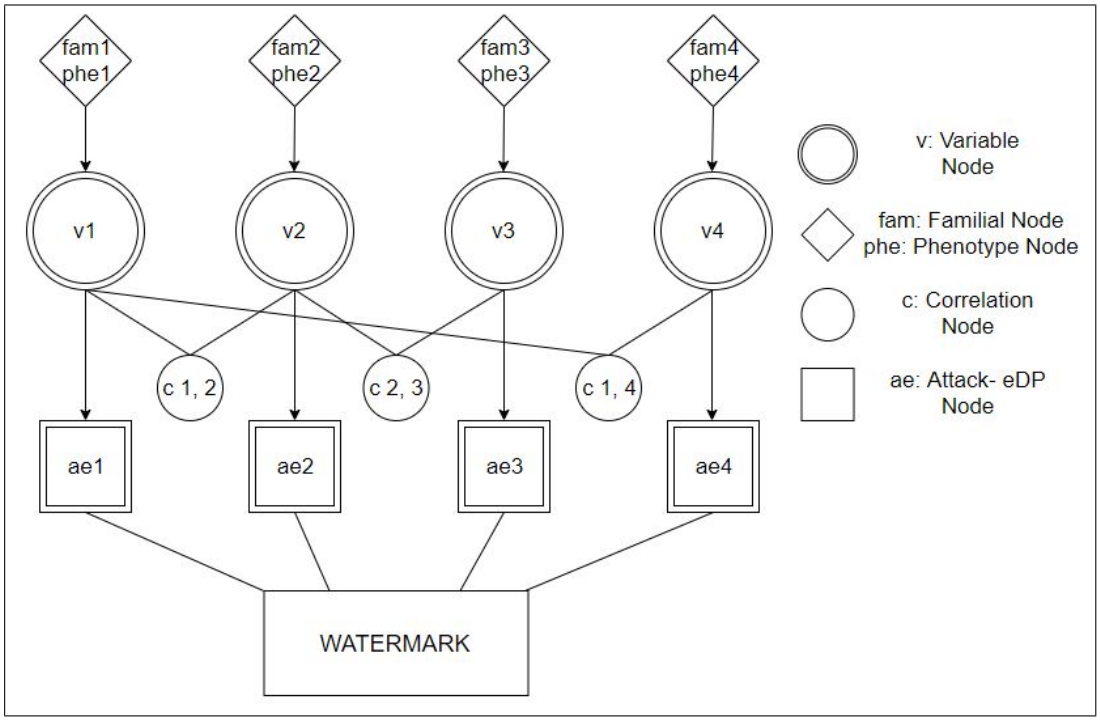


Figure 5.1: Factor graph representation of Variable Nodes and Attack-eDP interactions with other factor nodes: *Familial Nodes*, *Phenotype Nodes*, and *Correlation Nodes*.

the marginal probability distributions of i^{th} unknown variable in the format of $[P(x_i = y_1), P(x_i = y_2), \ldots, P(x_i = y_m)]$ so that each P corresponds to the probability of one y and all sums up to 1. For example, $x_3 = [0.6, 0.25, 0.15]$ for genomic data means the probability of the third SNP x_3 being homozygous major (AA / 0) is 0.6, heterozygous (Aa / 1) is 0.25 and homozygous minor (aa / 2) is 0.15. Probability distributions in variable nodes are calculated by multiplying the probability distributions coming from the neighboring factor nodes such as correlation nodes, familial nodes, and phenotype nodes. The message $\mu^v_{i\to k} P(x_i = y)$ from variable node i to factor node k indicates that $P(x_i = y)$ at v^{th} iteration where $y \in \{0, 1, 2\}$. The initial condition is $P(x_i^1 = y) = 0.33$ for all variable nodes to prevent the bias in inferring the probabilities. Equation 5.1 provides the function for the representation of a message from variable node i to correlation factor node k:

$$\mu^{v}_{i\to k}P(x_i = y)=\frac{1}{Z}\times\beta^{v-1}_{z\to i}P(x_i = y|fam_i)\times\omega^{v-1}_{z\to i}P(x_i = y|phe_i)\times\prod_{\substack{s=1,\\ s\neq i^n}}\delta^{v-1}_{s\to i}\ , \quad (5.1)$$

where Z is a normalization constant and $\sum \mu^{v}_{i\to k}P(x_i = y)$ for all y must be equal to 1.

5.1.2 Factor Nodes

Factor nodes represent the functions of factorized joint probability distributions of variable nodes. Factor nodes might be dependent (messages received or sent) on multiple variable nodes as well as a single variable node. Factor nodes might also be independent and fixed from the start. For genomic data, the correlation between SNPs called Linkage Disequilibrium can be given as the example of the first case. In such scenario, variable node var_i is connected to a correlation factor node $c_{i,j}$ along with the correlated variable node var_j. For the second case of dependency on a single variable, a message passed into the AE-node is determined by the current state of any variable node can be given as an example. For the third case of independency, family genomic information predetermined from the start can be given as an example. Let's assume for an SNP x, genomic information obtained from the family (father and mother) of certain individual L is $x_{L,f} = 0$ (homozygous major) and $x_{L,m} = 1$ (heterozygous). Then, we can safely predict the marginal probability distribution of that individual's SNP as $P(L, x)$=[0.5 0.5 0] using the Mendelian Law of Segregation [4]. This probability distribution is constant and not dependent on any value that the variable node might get. Therefore, throughout the algorithm, this probability distribution is propagated unchanged for any SNP x and receives no message $\mu^{v}_{i\to k}$ from its corresponding variable node.

5.1.2.1 Correlation Factor Nodes

In genomic data, we use Linkage Disequilibrium to enhance the privacy of the system against correlation attacks. Hence, malicious service providers will not be able to use the SNPs -which are correlated with other SNPs with high probability- for watermark detection. For every SNP pair, correlation coefficients are calculated before the iteration and the pairs with coefficients higher than σ_l threshold are marked as correlated and sensitive. Correlation coefficients may differ dependent on the states of each data point and their impact on estimating the probability distributions are typically asymmetric. For each sensitive SNP pair, there is one correlation node and these nodes keep track of the correlations inside the data.

The intuition used for calculating the message that shall be sent by correlation node is derived from the definition of r-squared, or the coefficient of determination, that explains how good the proportion of variance in the dependent variable predicts the proportion of variance in the independent variable [65]. Since our system uses and infers marginal probability distributions in BP, we used σ_{sj}^2 as a metric of how well we can predict the probability distribution of one state using the probability distributions of other correlated states. This intuition is supported in [66], too. For example, $\sigma_s j = 0.9$ can be used in our system as $\sigma_{sj}^2 = 0.81$. This means, $0.81 \times P(x_i = y)$ of the variance in j can be explained by the correlated node s for the particular states they correlate and it is used as probability distributions in the system. The unexplained proportion of other probabilities are distributed equally to all states. The messages from correlation node $c_{s,j} = i$ to j^{th} variable node $\lambda_{i\to j}^v$ are calculated as follows:

$$\lambda_{i\to j}^v P(x_j = y) = \sigma_{sj}^2 \times \mu_{s\to i}^v P(x_s = t),\ y,\ t \in \{0, 1, 2\}, \tag{5.2}$$

$$\lambda_{i\to j}^v P(x_j = y) = \frac{1 - (\sigma_{sj}^2 \times \mu_{s\to i}^v P(x_s = t))}{3},\ y,\ t \in \{0, 1, 2\}. \tag{5.3}$$

In these equations,

$$\{s = t, j = y\} \implies \sigma_{sj} \text{ (Equation 5.2)},$$

$$\{s = t, j = y\} \not\Longrightarrow \sigma_{sj} \text{ (Equation 5.3)},$$

where s is the neighbor variable node, σ_{sj} denotes the correlation coefficient and σ_{sj}^2 denotes the coefficient of determination. Figure 5.2 shows how correlation nodes are connected with variable nodes and how they send messages to one another.

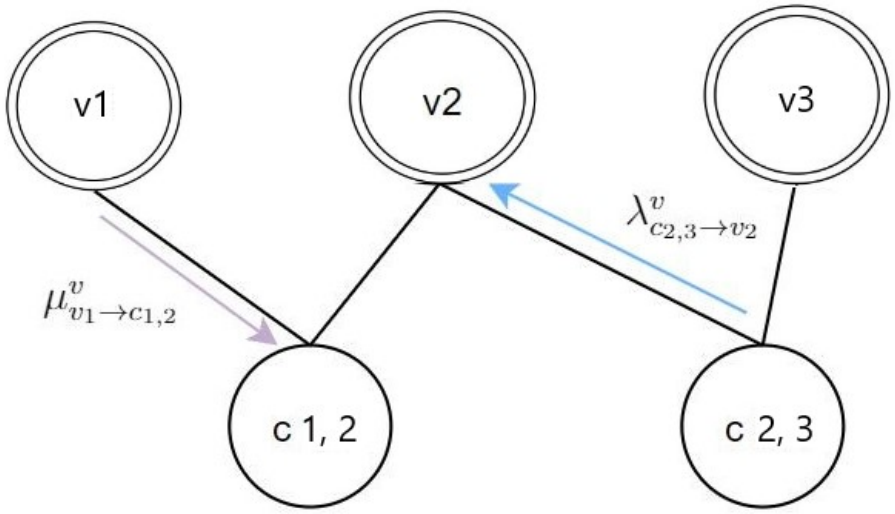


Figure 5.2: The relationship between variable nodes and correlation nodes. Both nodes may receive and send messages. For simplicity, one message for each type is shown.

For example, SNP_1 is connected with SNP_2 via $c_{1,2} = i$ for $x_1 = 0, x_2 = 2$ with $\sigma_{1,2} = 0.9$ and $\mu^v_{1\to i} P(x_1 = y) = [0.3, 0.6, 0.1], y \in \{0, 1, 2\}$ at v^{th} iteration. Then, we may calculate the message from i to 2^{nd} variable node, $\lambda^v_{i\to 2} P(x_2 = y)$ as:

$$\lambda^v_{i\to 2} P(x_2 = y) =$$
$$\left[\frac{1 - (0.9^2 \times 0.3)}{3}, \quad \frac{1 - (0.9^2 \times 0.3)}{3}, \quad \frac{1 - (0.9^2 \times 0.3)}{3} + (0.9^2) \times (0.3),\right]$$

$$\lambda^v_{i\to 2} P(x_2 = y) = [0.2523, 0.2523, 0.4954].$$

5.1.2.2 Familial Factor Nodes

Familial factor node fam_i calculates the message $\beta^v_{i\to k}P(x_k = y|f_i, m_i), y \in \{0, 1, 2\}$ using the Mendelian Inheritance Law of Segregation and sends it to variable node k. The probabilities are given in Table 5.1. In the message, f_i and m_i corresponds to the i^{th} *SNP* values of father and mother, respectively.

Table 5.1: Mendelian inheritance probabilities using the Law of Segregation.

SNP Father (Rows) Mother (Columns)	0	1	2
0	[1, 0, 0]	[0.5, 0.5, 0]	[0, 1, 0]
1	[0.5, 0.5, 0]	[0.25, 0.5, 0.25]	[0, 0.5, 0.5]
2	[0, 1, 0]	[0, 0.5, 0.5]	[0, 0, 1]

For example, if the father has $SNP^f_i = 1$ and the mother has $SNP^m_i = 2$ for the i^{th} *SNP*, the message from familial node fam_i is as follows:

$$\beta^v_{i\to k}P(x_i = y|f_i = 1, m_i = 2) = [0, 0.5, 0.5], y \in \{0, 1, 2\}.$$

5.1.2.3 Phenotype Factor Nodes

Phenotype factor node phe_i calculates the message $\omega^v_{i\to k}P(x_k = y|phe_i), y \in \{0, 1, 2\}, phe_i \in \{dominant, recessive\}$ using the Mendelian Inheritance Law of Dominance and sends it to variable node k. Probabilities are given in Table 5.2. In the message, phe_i corresponds to the dominance trait of the observed phenotype in i^{th} *SNP*. phe_i can be either dominant or recessive.

Table 5.2: Mendelian inheritance probabilities using Law of Dominance.

Observed phenotype trait	*SNP* distribution
Dominant (AA or Aa)	[0.5, 0.5, 0]
Recessive (aa)	[0, 0, 1]

For example, if the data owner is known to have blue eyes (recessive gene), which is encoded in the i^{th} *SNP*, the message from phenotype node phe_i is as

$$\omega^{v}_{i\to k} P(x_k = y | phe_i = recessive) = [0, 0, 1],\ y \in \{0, 1, 2\}.$$

5.1.2.4 Attack-eDP Node

The attack-eDP node is designed to simulate the inference power of the attackers on data and calculates the inverse probabilities that will keep the attacker uncertainty at maximum against single SP and collusion attacks while keeping the local differential privacy criteria intact by eliminating the watermarked state options which violate ϵ-local differential privacy. This node receives a message from the variable node. Although acting as another factor node, it does not send the message to the variable node. Instead, the attack-eDP node sends its message along with a variable node message to the watermarking algorithm as parameters.

Inside the attack-eDP node, the attack part re-calculates the watermarking probabilities of all indices based on the variable node probability distributions and previously shared versions of states to simulate single SP and collusion attack potential. In every SP_k's watermarking, a set of previous sharings S_i^{k-1} for each index i or set of indices I are used as a prior condition. Then the probabilities of the potential next states are calculated using binomial distribution given S. Finally, updated probability distributions are sent to the watermarking algorithm as the watermarking probability of each state. The calculation procedure followed by the node's attack part is described in the sequel:

$\alpha = |(x_i = y)| \in S_i^k, \quad \alpha$ is the number (cardinality) of states equal to y in set S_i^k.

$Binomial(S_i^k | x_i^k = y) = \binom{k}{\alpha} \times P(x_i = y)^{\alpha} \times P(x_i = y')^{k-\alpha}$.

$P(x_i = y)$ and $P(x_i = y')$ are calculated from the variable node's message.

$$a_0(x_i^k = 0 | S_i^{k-1}) \approx P(S_i^k | x_i^k = 0) = Binomial(S_i^k | x_i^k = 0),$$
$$a_1(x_i^k = 1 | S_i^{k-1}) \approx P(S_i^k | x_i^k = 1) = Binomial(S_i^k | x_i^k = 1),$$
$$a_2(x_i^k = 2 | S_i^{k-1}) \approx P(S_i^k | x_i^k = 2) = Binomial(S_i^k | x_i^k = 2).$$

$A_i^k = Normalized([a_0, a_1, a_2])$, where A is the updated marginal watermarking probability distribution of i^{th} index for the SP_k.

For example, let's assume for the index i, $var_i = \mu^v_{i \to ae_i} P(x_i = y) = [0.6, 0.4, 0]$ sends the following message to the attack-edp node ae_i and $S_i^{k-1} = \{0, 0, 1, 0, 1, 0\}$ where $k = 7$. We may calculate the watermarking probabilities of index i for SP_7 as follows:

For $x_i^7 = 0$, $\alpha = 5$ and $Binomial(S_i^7 | x_i^7 = 0) = \binom{7}{5} \times (0.6)^5 \times (0.4)^{7-5} = 0.261$,
For $x_i^7 = 1$, $\alpha = 3$ and $Binomial(S_i^7 | x_i^7 = 1) = \binom{7}{3} \times (0.4)^3 \times (0.6)^{7-3} = 0.290$,
For $x_i^7 = 2$, $\alpha = 1$ and $Binomial(S_i^7 | x_i^7 = 2) = \binom{7}{1} \times (0)^1 \times (1)^{7-1} = 0$.

$A_i^7 = Normalized([0.261, 0.290, 0]) \approx [0.474, 0.526, 0]$ is the updated watermarking probability distribution of index i for SP_7.

As an extra measure of privacy, we have incorporated the condition of satisfying Local Differential Privacy (LDP) [67] for Alice who wants to have a plausible deniability factor for the versions of data she shares. This incorporation creates watermarks for all the *SNPs* of the data owner and acts as a lower bound of privacy ensured along with lower and upper bounds on confidence degree by its very definition. A watermarking algorithm with ϵ-local differentially privacy must normally satisfy Equation 2.2, for all sharings of all *SNPs*. This condition is used for limiting the amount of information gained by the exclusion of each shared data from the total set of sharings.

In Equation 2.2, as ϵ increases, the uncertainty also increases at the expense of privacy. As ϵ decreases, more privacy is ensured. However, Equation 2.2 does not cover a localized setup but counts on databases or datasets as a whole. Therefore, we use the variant version of differential privacy adapted from the geo-indistinguishability study of Andres et al. that is both localized in *SNP* level and more suited to our sequential data since they tested the formula on sequential location data [64]. In e-DP part, our framework eliminates the watermarking options that violate the local differential privacy condition of Andres et al. and his modified privacy criteria is given below as:

$$\frac{P(x|S)}{P(x'|S)} \leq e^{\epsilon \times r} \times \frac{P(x)}{P(x')}, \quad \forall r > 0, \quad \forall x, x' : d(x, x') \leq r. \tag{5.4}$$

In Equation 5.4, just like the attack part, S represents the set of previous sharings of data (as we used in § 4.1) and r represents the distance between the states. In location data, r is calculated as the maximum Euclidean distance between states. Since our data is sequential genomic data and the states have no priority over one another, we used Hamming distance for r, which is equal to one all the time. It is important to note that the first part of Equation 5.4 refers to the newly updated probabilities obtained from modifications such as adding and removing noise and it corresponds to the ratio between ae_is. The second part refers to the unchanging probabilities of states given the prior information and it corresponds to the ratio between var_is. Hence, we can compare the results for each $x_i = y, y \in \{0, 1, 2\}$, and decides which states should be discarded for not violating the privacy condition.

Continuing from the example used in the attack part; we can calculate the privacy conditions as follows:

$$\text{For } x_i^7 = 0, \ \frac{P(x|S)}{P(x'|S)} = \frac{0.474}{0.526} = 0.901, \text{ and } \frac{P(x)}{P(x')} = \frac{0.6}{0.4} = 1.500.$$

This means $0.901 \leq e^{\epsilon} \times 1.500$ must be satisfied for not violating the ϵ-local differential privacy. Since $\frac{0.901}{1.500} \leq 1$ and $e^{\epsilon} \geq 1$ for all $\epsilon \geq 0$, this condition always holds and the state $x_i^7 = 0$ never violates the privacy.

$$\text{For } x_i^7 = 1, \ \frac{P(x|S)}{P(x'|S)} = \frac{0.526}{0.474} = 1.110, \text{ and } \frac{P(x)}{P(x')} = \frac{0.4}{0.6} = 0.667.$$

This means $1.110 \leq e^{\epsilon} \times 0.667$ must be satisfied for not violating the ϵ-local differential privacy. Since $\frac{1.110}{0.667} = 1.664$ and $ln(1.664) = 0.223$, if $\epsilon \leq 0.223$ the state probability of $P(x_i^7 = 1)$ must be updated for not violating the privacy. This update is done by calculating the minimum state probability that satisfy the condition as:

$$\frac{P(x|S)}{P(x'|S)} \leq \frac{P(x)}{P(x')} \times e^{\epsilon} \implies \frac{1-P(x)}{P(x)} \leq \frac{1-P(x|S)}{P(x|S)} \times e^{\epsilon} \implies \frac{1-P(x)}{P(x)} \leq (\frac{1}{P(x|S)} - 1) \times e^{\epsilon}$$

$$\implies P(x|S) \leq \frac{1}{(\frac{1}{P(x)}-1)\times e^{\epsilon}+1} = \frac{1}{\frac{e^{\epsilon}}{P(x)}-e^{\epsilon}+1}$$

After $P(x)$ is set, distribution is continuously normalized to converge into the probability that satisfies the condition.

For $x_i^7 = 2, \frac{P(x|S)}{P(x'|S)} = \frac{0}{1} = 0,$ and $\frac{P(x)}{P(x')} = \frac{0}{1} = 0.$

This means $0 \leq e^{\epsilon} \times 0$ and it never violates the local differential privacy for any ϵ just like the case of $x_i^7 = 0$. However, we know that $P(x_i = 2) = 0$ and it is an impossible watermarking case regardless of the violation.

In the end, if Alice set her privacy criteria to $\epsilon < 0.223$, ae_i's final marginal probability distribution will be equal to the distribution enforced by the eDP part. Otherwise, the distribution remains the same as attack part determined which is equal to $[0.474, 0.526, 0]$.

5.2 Watermarking

SNP state inferences are assumed to be conducted by the malicious SPs as well, given their prior information on the data for Single SP Attack and Correlation Attack (cf. § 4.3). Therefore, our system considers attacker inference strength and privacy criteria at the same time while watermarking. In watermarking, changing the actual state of the data is mandatory. Changing multiple indices than necessary results with losing utility on data and these changes increase the detection probability of changed indices by malicious SPs and decrease efficiency. Furthermore, these changes must be reflected like actual data, in order not to give no more means to malicious SPs for detecting the watermarked indices. For example, watermarking a SNP_i with $MAF_i = 0$ is meaningless. Because any other state of the SNP_i besides homozygous major is not observed in the population,

any change will be artificial and interpreted as watermarked.

Another point to be considered in our watermarking scheme is to keep the watermarking pattern probabilistic rather than deterministic. This means for each SP, we shall use a different set of indices and states to be watermarked. If the watermarked indices set is kept fixed, it presents a risk of compromising watermark robustness against modifications and removals in single SP attacks and collusion attacks. If the watermarked states are fixed for each index, the data does not reflect the population distribution and probabilistic inference of attackers may identify the indices that show discrepancies with the population.

Given these criteria, we calculate a watermark score *wScore* that helps us to list indices better to watermark in descending order. This score is calculated by comparing the attack-eDP marginal probability distributions with the original states of data. Firstly, the probability of the actual state in attack-eDP distribution is subtracted from one. This will give us the probability of that index being watermarked. Then these indices are sorted in descending order to give priority on indices most likely to be watermarked. For further insight, the watermark algorithm is given in Algorithm 1.

Algorithm 1 Watermarking Algorithm

Watermark(*data*, *atks*, *vars*, *wScore*, w_l)

```
1: j ← 1
2: k ← 0
3: newdata ← data
4: while k < w_l do
5:    i ← wScore(j, 4) {index of SNP}
6:    temp ← data(i) {actual state of SNP}
7:    flag ← true
8:    while flag do
9:       r ← random(0, 1)
10:      if atks(i, 1) ≥ r then
11:         newdata(i) ← 0
12:         if temp ≠ 0 then
13:            k + +
14:         end if
15:         flag ← false
16:      else if atks(i, 1) + atks(i, 2) ≥ r then
17:         newdata(i) ← 1
18:         if temp ≠ 0 then
19:            k + +
20:         end if
21:         flag ← false
22:      else if atks(i, 1) + atks(i, 2) < r then
23:         newdata(i) ← 2
24:         if temp ≠ 0 then
25:            k + +
26:         end if
27:         flag ← false
28:      end if
29:   end while
30:   if j == length then
31:      j ← 1
32:   else
33:      j + +
34:   end if
35: end while
36: return newdata
```

Chapter 6

Evaluation

We evaluated the proposed watermarking scheme in various aspects using genomic data. The most important aspects of data that are evaluated are watermark security against detection (robustness) and privacy guarantees. These aspects and their correspondence to the dependent variables are also given. We give the details of the data model, the experimental setup used, and the results of the experiments in the sequel.

6.1 Data Model and Experimental Setup

For the evaluation, we used the Single Nucleotide Polymorphism (SNP) data of 1000 Genomes Project [3]. The obtained data set contains the 7690 SNP-long data of 99 individuals in the form of 0s, 1s, and 2s. This means we have a 99×7690 matrix with elements $\{0, 1, 2\}$. This data set is used for learning the linkage-disequilibrium and MAF statistics of the data along with parental data generation based on the method proposed in [68]. Later on, these statistics are employed in the Belief Propagation Algorithm for probabilistic state inference. The threshold of pairwise correlations used for the results is specified as $\rho = 0.9$. Throughout the experiments, the length of data d_l is fixed to 1000, the number of

service providers h is fixed to 20, and w_l values vary between 10 and 100. In some exceptional cases, watermarks with $w_l > 100$ are also tested but those results were almost identical to watermarks with $w_l \approx 100$ and therefore not included.

6.2 Evaluation Metrics

We evaluated the data by calculating precision values and ϵ-privacy achieved for various attack types, parameter configurations, and sets of predicted SPs. In collusion attacks, two SPs collusion scenario contains all 190 pairs of 20 SPs since, h is fixed to 20 and $C(20, 2) = 190$. This number increases rapidly as the number of collusion SPs increases. To keep the computational cost low, we took the number of malicious SPs scenarios as 190 unique random sets for each case. Besides, we kept the number of malicious SPs up to $k = 10$, since we assumed to know the k and $k > 10$ increased our detection results back. We find the malicious set of SPs by checking the watermark patterns.

For detection, we compare the attacked data produced by malicious SPs with each of our previously shared data and watermark patterns. Assuming, we know the number of malicious SPs, we use two detection methods *Hamming Distance (H)* and custom *spPenalizer (E)* and their relaxed versions that use the variance of differing indices as the likelihood of maliciousness scorer. Both *Hamming Distance (H)* and *spPenalizer (E)* detection methods report the top number of malicious SP guesses, whereas the relaxed versions of them, *Hamming Distance Relaxed (HR)* and *spPenalizer Relaxed (ER)*, try to find the malicious SPs in the top "number of malicious SPs + 2" guesses. In single SP attacks, the precision results of detection algorithms are obtained by 190 random malicious SPs tests just like collusion attacks.

6.3 Results of Attacks

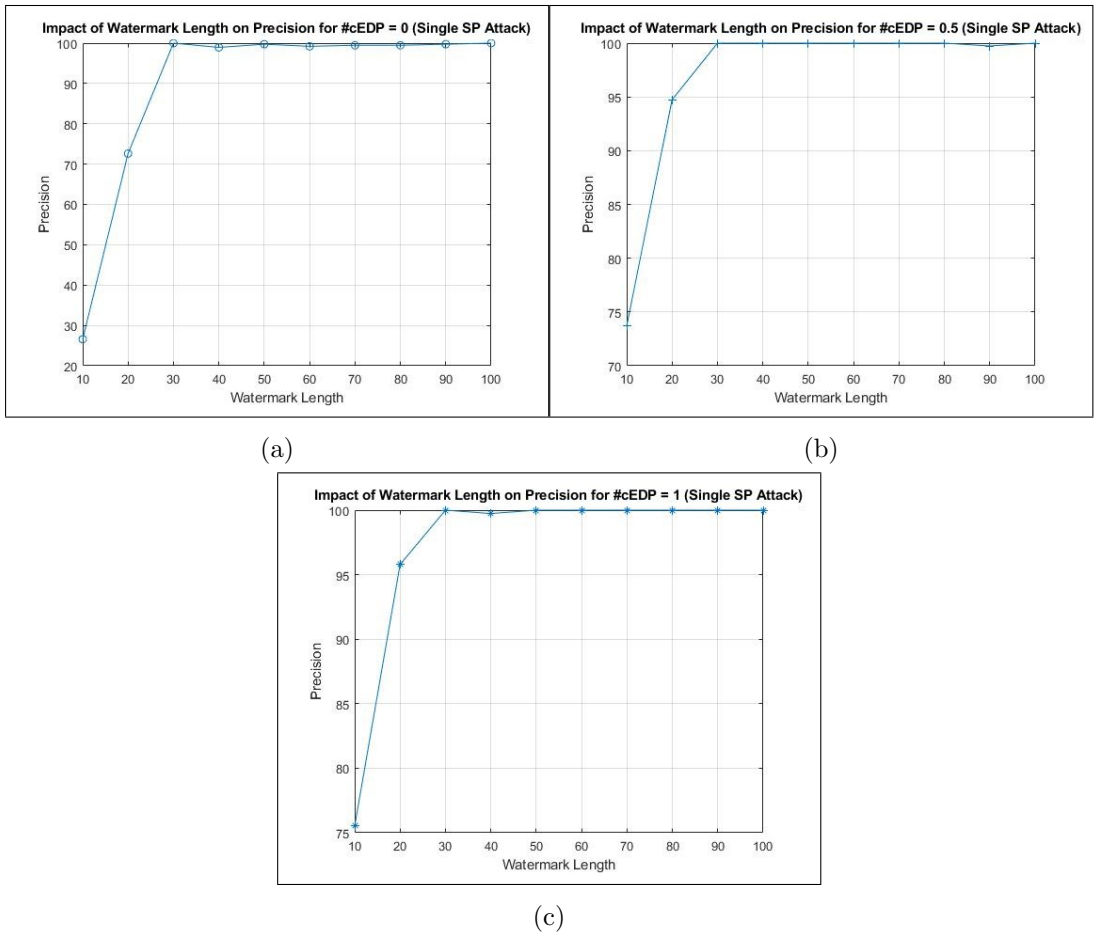
We evaluated the proposed scheme for the attack model described in § 4.3. The robustness of the watermarks is evaluated against the single SP attacks and collusion attacks in which the knowledge of single SP and correlation attacks are incorporated to reflect the worst-case scenario. In these experiments, we assume worst-case scenarios to create lower bounds. The assumptions that give maximum malicious SP information are as follows.

- Malicious SPs know the exact value of watermark length (w_l).
- For every SP, I_k is identical $k \in \{1, 2, ..., h\}$. It means all SPs have the same set of indices of data.
- Malicious SPs have all the population information e.g. correlations, MAFs, frequency of states.
- Malicious SPs know the SNPs of the data owner's father and mother (familial information).
- Malicious SPs know the phenotypical features of the data owner and correspondent SNP states.

6.3.1 Single SP Attack

In single SP attacks, a single SP uses all the knowledge available to itself for inferring the marginal state probabilities of SNPs. This process is similar to the calculations in the belief propagation part of our watermarking scheme. Later on, malicious SP identifies the top w_l SNPs with least probabilities $P(x_i = y), y \in \{0, 1, 2\}$ as watermarked, and modifies them to their most likely states for the prior knowledge available to itself. One thing to note here is that these SNPs can be removed or partially modified as a variant attack scenarios. However, the total modification almost always yielded the best results in our experiments in favor

of malicious SP and decreased our detection precision the most. Therefore, we assume the worst-case scenario and provide the precision results of our detection algorithms *Hamming Distance (H)*, *spPenalizer (E)*, and their relaxed versions (*HR*) and (*ER*) against modifying attacks.



(a) (b)

(c)

Figure 6.1: The impact of watermark length on precision for a single SP attack with different privacy preservation coefficient (ϵ) values. (a) $\epsilon = 0$, (b) $\epsilon = 0.5$, and (c) $\epsilon = 1$.

Figure 6.1 shows the impact of watermark length on precision for various values of the privacy criteria (ϵ). In all cases, $w_l \geq 30$ seems to be the breaking point where the precision reaches to almost 100%. Among 20 SPs, a single SP could be identified with almost full precision after $w_l \geq 30$. In our data, we think that

this corresponds to the minimum amount of change needed for distinguishing a version of shared data from another effectively. For example, this w_l may not suffice for comparison among 50 SPs. Another point to notice is that the results reflected here are only the precision results of the E method.

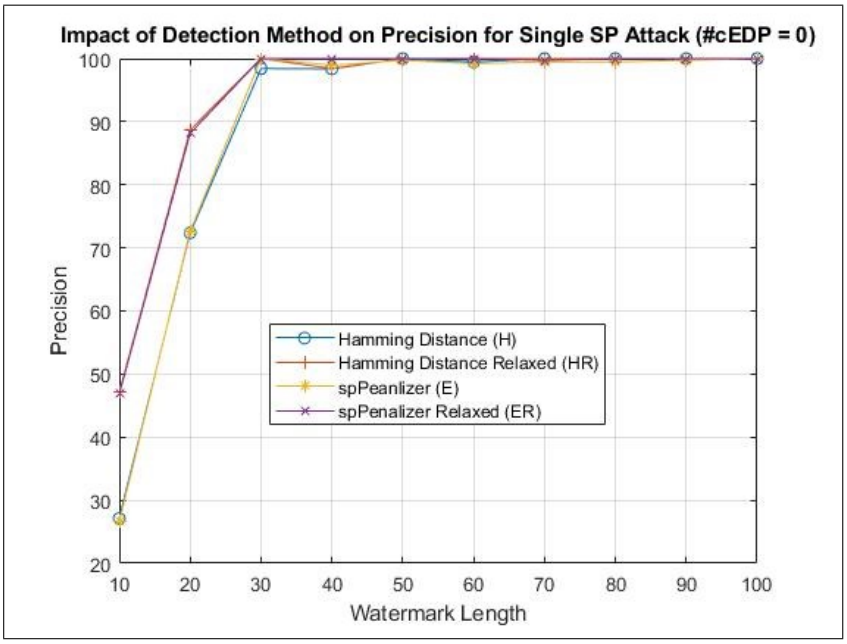


Figure 6.2: The impact of different watermark lengths and detection methods on precision for a single SP attack ($\epsilon = 0$).

Figure 6.2 shows a comparison of the detection methods. Not surprisingly, we observe that the relaxed versions HR and ER outperformed the H and E methods. Surprisingly though, the results of H and E and HR and ER were very close to each other. This might be explained with the lack of variety in more restrictive watermarking, which is enforced by the $\epsilon = 0$ privacy criteria.

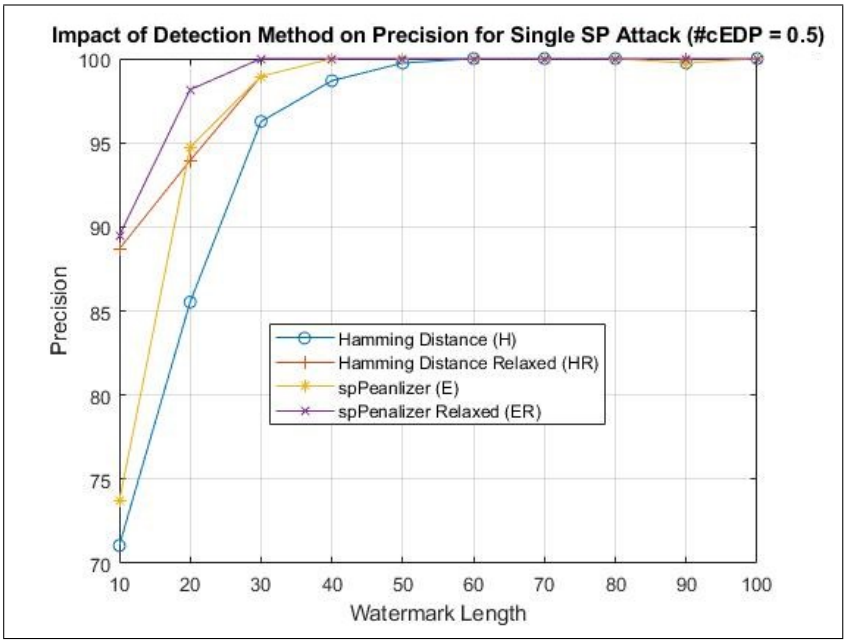


Figure 6.3: The impact of different watermark lengths and detection methods on precision for a single SP attack ($\epsilon = 0.5$).

Figure 6.3 compares the detection methods when we relaxed the local differential privacy criteria of ϵ to 0.5. For this case, we observe that the E method outperformed not only the H method but also its relaxed version HR for $w_l \geq 20$. Before making concluding remarks, we need to examine whether the impact of ϵ is an ongoing trend that increases the precision further or it stabilizes at some point.

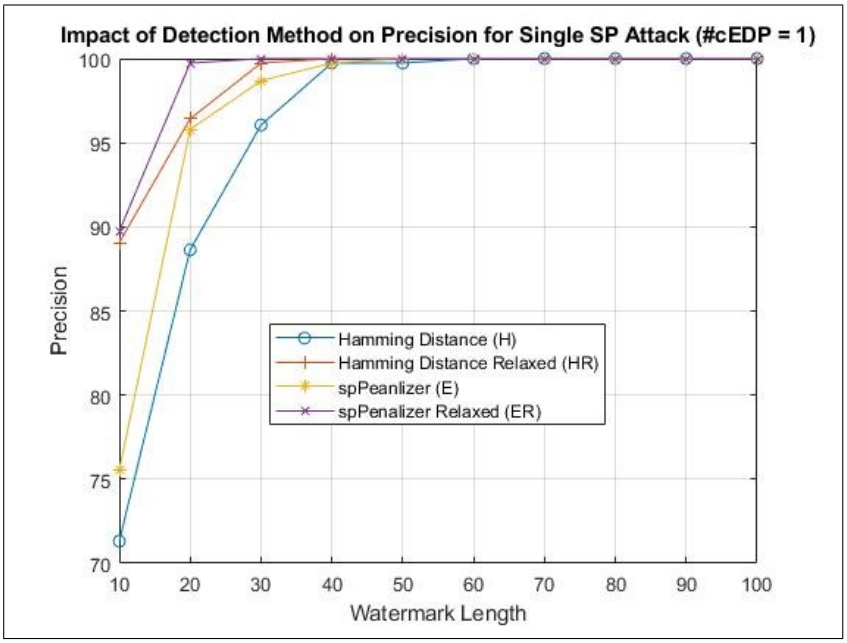


Figure 6.4: The impact of different watermark lengths and detection methods on precision for a single SP attack ($\epsilon = 1$).

Figure 6.4 shows that the precision results for $\epsilon = 1$ do not significantly differ from the precision results for $\epsilon = 0.5$, except for minor improvements.

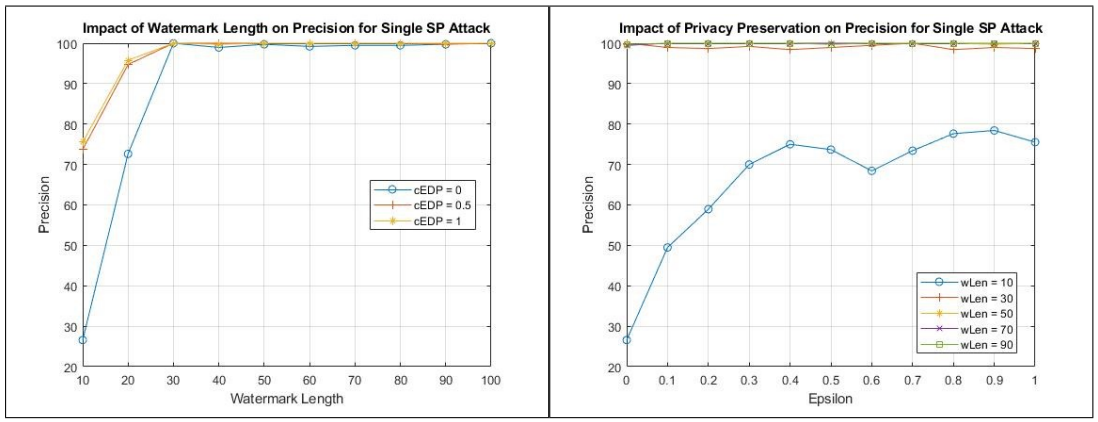


Figure 6.5: The impact of watermark length (left) and privacy preservation (ϵ) (right) on precision for a single SP attack.

Figure 6.5 shows that ϵ does not influence the precision of the E method,

except for $w_l = 10$ and $w_l = 20$ (left side). However, while explaining Figure 6.1, we have clarified this issue with not-satisfied indistinguishability.

Overall, it is safe to assume that ϵ has little to no impact on watermarks with $w_l \geq 30$ and the precision rates are close to 100% when the attack is conducted by single SP. Although it is not recommended, if the watermark is intended to be kept shorter than $w_l < 30$, ϵ should be raised at least to 0.5 to keep the robustness somehow high still.

6.3.2 Collusion Attack

In collusion attacks, multiple SPs collude and bring their data together to detect and modify watermarked indices. Firstly, the states different for the same SNP are identified as watermarked because the watermark pattern is unique for each SP. Therefore, the maximum number of indices that can be identified as watermarked by malicious SPs is $w_l \times k$ where k is the number of malicious SPs. Secondly, when the malicious SPs find fewer points in collusion attack than $w_l \times k$, they target additional indices as watermarked using the prior information on the data, e.g., *MAFs* and *LD* correlations, similar to the single SP attack. These indices are usually the least likely states when prior information is considered. At the end of the collusion attacks, malicious SPs change the states of data in two ways. The states which are not the same across all malicious SPs are modified to the most frequent ones. Then, the states which are the same across all malicious SPs but having the least likelihoods are modified to the most likely states possible. Our initial detection method, modification setups, and assumptions on the collusion attack are similar to those of a single SP attack. In addition to those, since collusion attacks contain much more information about the probability of a state than the single SP attack, we expect the precision results of the collusion attacks to be lower than single SP attacks. We expect the decreasing precision impact with the increasing number of malicious SPs.

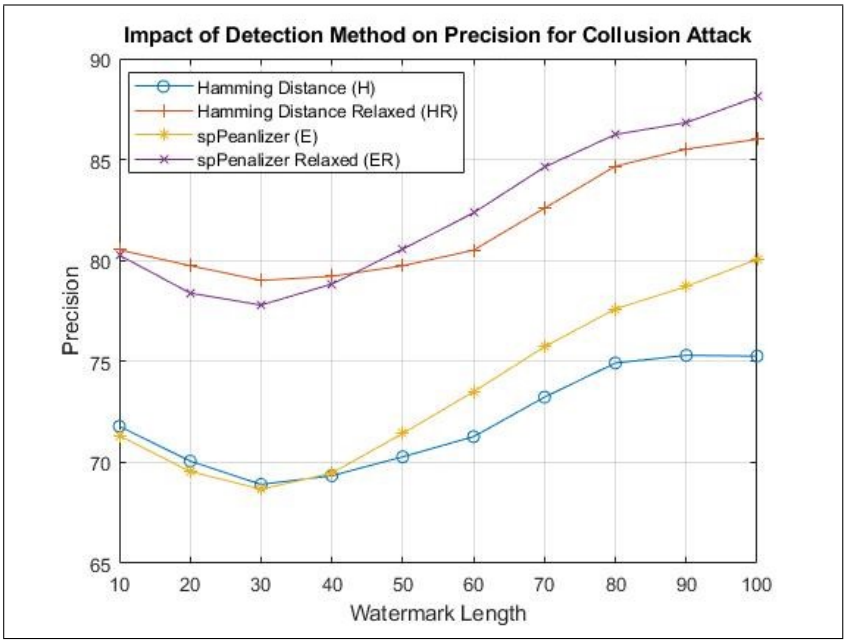


Figure 6.6: The impact of detection methods on precision for a collusion attack ($\epsilon = 0.5$ and $k = 10$).

First, we examined which detection method performs the best in detecting the malicious SPs of collusion attack. Similar to single SP attacks, the relaxed versions of *Hamming Distance (H)* and *spPenalizer* (E) methods, *HR* and *ER*, outperformed the former. This observation holds for Figure 6.6. Besides, we noted that the *E* method is better at detecting malicious SPs than the *H* method in general. A similar observation is also valid for the relaxed versions of *HR* and *ER*. Therefore, we considered the precision results of the *ER* method for the rest of the experiments.

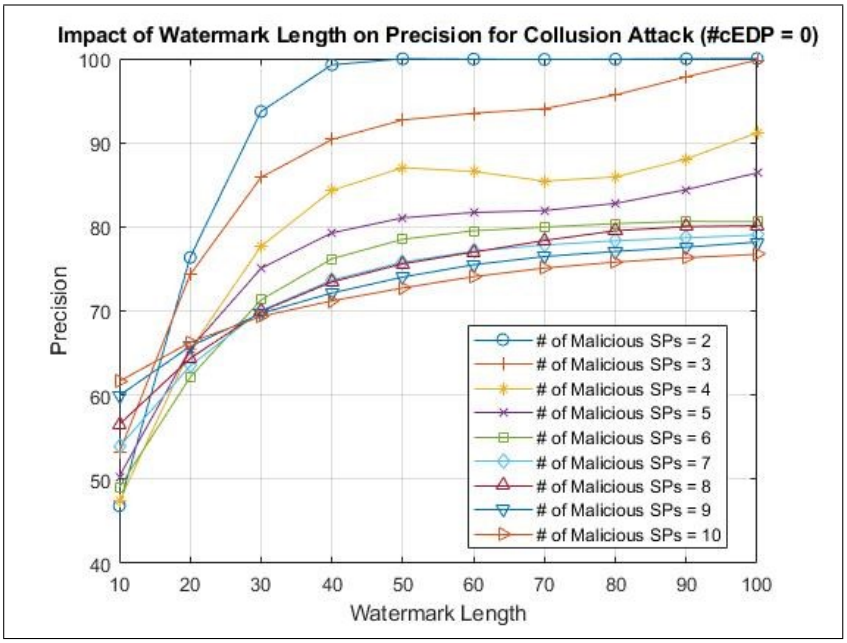


Figure 6.7: The impact of different watermark lengths on precision for a collusion attack ($\epsilon = 0$).

Figure 6.7 shows the impact of watermark length on precision keeping the privacy criteria ϵ at 0. In all cases, $w_l \geq 30$ seems to be the breaking point where all k malicious SP scenarios are detected with precision rates higher than 70%, and the increase in precision rates slow down. Among 20 SPs, $k = 10$ gives the worst results as expected with 75% precision rate even for a long watermark of $w_l = 100$. Similar to the explanation in a single SP attack, this low precision results can be explained with the very restrictive privacy criteria ($\epsilon = 0$) which decreases the robustness of the watermark. By sacrificing the privacy expected, higher precision results can be achieved. Another point to notice is that the minimum watermark length needed for distinguishing the SPs argument presented in a single SP attack show its impact on the rather chaotic precision results in $w_l < 30$.

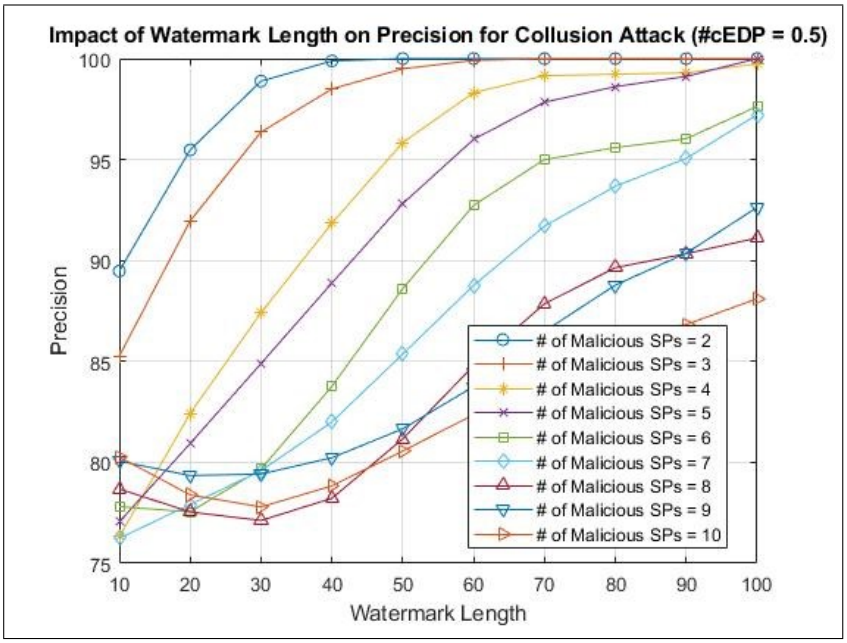


Figure 6.8: The impact of different watermark lengths on precision for a collusion attack ($\epsilon = 0.5$).

Figure 6.8 shows that the relaxation of the privacy criteria ϵ to 0.5 results in higher precision rates for almost all k SP scenarios. Apart from the case of $k = 8$, the precision results seem to be in order after $w_l \geq 30$. For the worst case of $k = 10$, precision rates close to 90% can be achieved with $w_l = 100$. However, we need to examine whether the relaxation of privacy criteria even further will result in increased precision, which was not the case for a single SP attack.

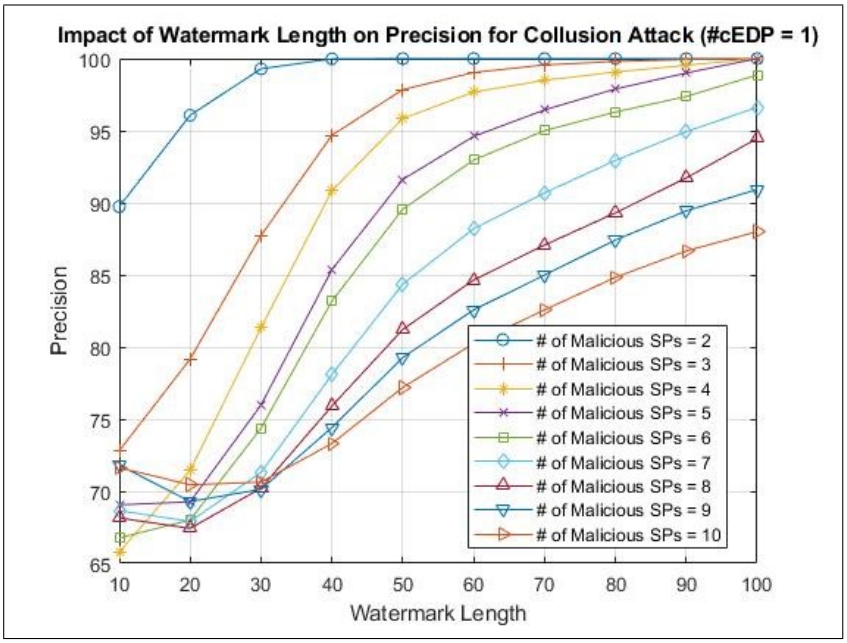


Figure 6.9: The impact of different watermark lengths on precision for a collusion attack ($\epsilon = 1$).

Figure 6.9 shows that further relaxation of privacy criteria ϵ to one does not result in increased precision, similar to the results of a single SP attack. In some of the experiments, the precision rates higher than 90% even 95% can be achieved for $k = 10$ and $w_l = 100$, but these are very rare cases.

With w_ls ≈ 200, precision rates up to 95% are achieved. However, increasing the $w_l \geq 100$ provides marginally small precision benefit compared to the utility it decreases. For the cases of $k = 9, 10$, this benefit is $\approx 5 - 6\%$ with doubled utility loss. Therefore, a reasonable upper bound of precision in our system for $k = 10$ (the worst-case scenario) can be determined as 90%.

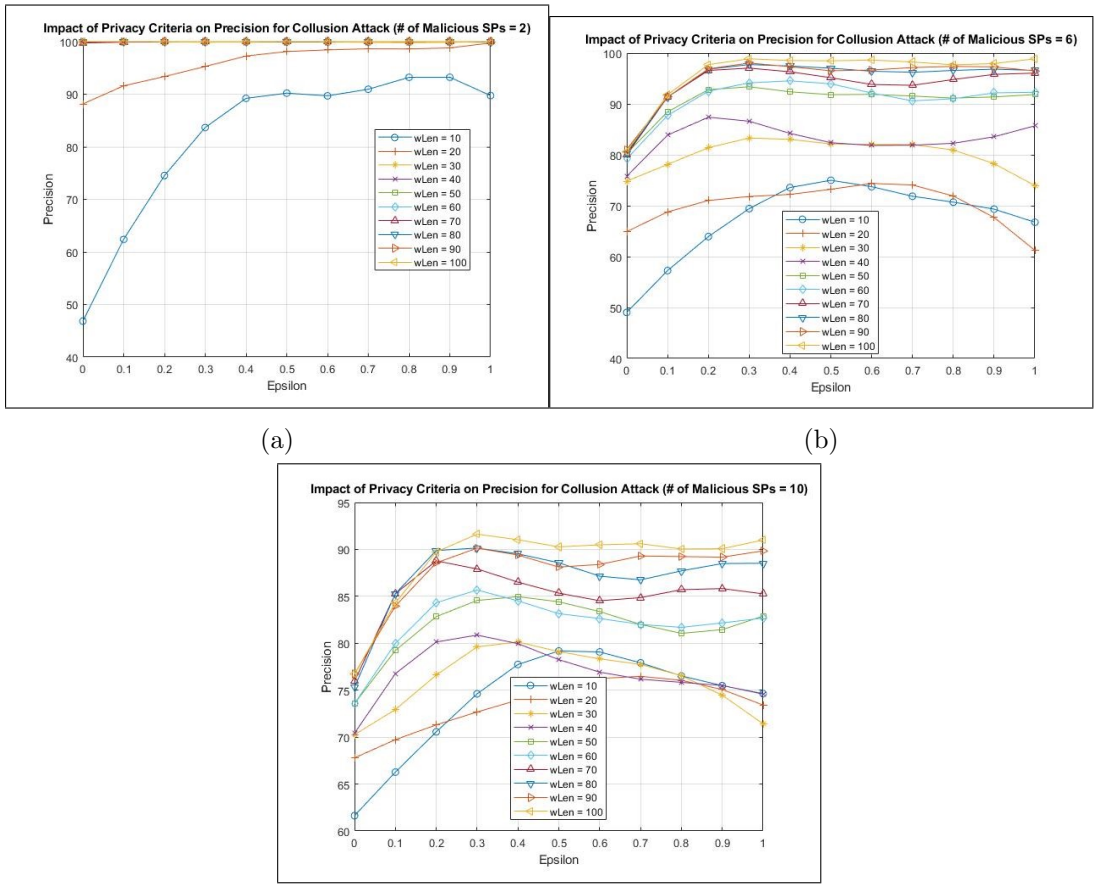


Figure 6.10: The impact of privacy preservation coefficient (ϵ) on precision for collusion attacks with different number of malicious SPs (k). (a) $k = 2$, (b) $k = 6$, and (c) $k = 10$.

Figure 6.10 depicts the impact of privacy criteria (ϵ) on precision. The case for $k = 2$ provides very limited information because all of the $w_l \geq 30$ gives 100% precision. If $k = 6$, the increase in precision while ϵ increases from 0 to 0.3 can be observed for almost all $w_l \geq 30$. For $w_l \geq 50$, the precision remain stable but for $w_l \leq 50$ arbitrary decreases and increases are observed. For the case of $k = 10$, we have similar observations to those of $k = 6$ with naturally lower precision results. Therefore, $\epsilon = 0.3$ seems to be the ideal spot where most privacy is kept intact while the precision is not reduced at the expense of privacy. The absence of a certain trend in $w_l \leq 40$ can be explained as follows. By keeping the privacy criteria within the probabilistic watermarking process, slight alterations of the

order of watermarking index lists can both decrease or increase the precision against attacks drastically.

Chapter 7

Conclusion and Future Work

We proposed a novel scheme of watermarking sequential genome data employing belief propagation algorithm with ensured ϵ-local differential privacy. This system is designed to be used between the data owner and service providers some of whom are assumed to be malicious. We implemented the algorithm against the worst-case scenario of malicious SPs and assumed they will know almost all statistics of the data available and tested the robustness of watermarks against single sp attacks and collusion attacks (with information known from single sp attack). Unauthorized sharing risk is greatly mitigated by the belief propagation algorithm. The algorithm secured high precision rates even against worst-case scenarios by considering all potential prior information (e.g., *MAFs*, *LD* correlations, and Familial genomes) that malicious SPs can use. While doing so, we wanted to keep the changes on data minimum for keeping the utility of data by using short length watermarks. Our experiments showed that when w_l is kept higher than 50, even for a high number of malicious SPs, watermark robustness is preserved more than 80%. Besides, we have observed that there is a significant trade-off between privacy and watermark robustness for $\epsilon < 0.3$ in collusion-secure watermarking. When $\epsilon \geq 0.3$ is used, privacy is preserved without impacting the precision and it addressed a potential liability issue from data being known and created a privacy measure of plausible deniability needed especially for rare SNPs. The detection methods of the algorithm proposed do

not predict but know the exact number of malicious SPs in the attack model. As future work, an automated detection method with SP classifier that clusters the malicious SPs from the others by itself or a system that makes predictions like [59] might be implemented for Belief Propagation Watermarking.

Bibliography

[1] S.-J. Lee and S.-H. Jung, "A survey of watermarking techniques applied to multimedia," in *Proceedings of the IEEE International Symposium on Industrial Electronics Proceedings*, vol. 1 of *ISIE '01*, pp. 272–277, June 2001.

[2] M. Barni and F. Bartolini, *Watermarking Systems Engineering: Enabling Digital Assets Security and Other Applications*. CRC Press, 1 ed., 2004.

[3] The International Genome Sample Resource, "1000 Genome Project." `https://mathgen.stats.ox.ac.uk/impute/1000GP_Phase3.html`, 2013. [Online; accessed 25-October-2019].

[4] R. Bailey, "What Is Mendel's Law of Segregation?." `https://www.thoughtco.com/mendels-law-of-segregation-373472`, 2019. [Online; accessed 26-July-2020].

[5] International Society of Genetic Genealogy, "Single Nucleotide Polymorphism." `https://isogg.org/wiki/Single-nucleotide_polymorphism`, 2018. [Online; accessed 25-October-2019].

[6] B. S. Shastry, "SNP alleles in human disease and evolution," *Journal of Human Genetics*, vol. 47, pp. 0561–0566, Nov. 2002.

[7] R. C. Elston, J. M. Satagopan, and S. Sun, "Genetic terminology," in *Methods in Molecular Biology*, pp. 1–9, Humana Press, Dec. 2011.

[8] R. D. Hernandez, L. H. Uricchio, K. Hartman, C. Ye, A. Dahl, and N. Zaitlen, "Ultrarare variants drive substantial cis heritability of human gene expression," *Nature Genetics*, vol. 51, pp. 1349–1355, Sept. 2019.

[9] M. Slatkin, "Linkage disequilibrium — understanding the evolutionary past and mapping the medical future," *Nature Reviews Genetics*, vol. 9, pp. 477–485, June 2008.

[10] J. C. Stephens, "Haplotype variation and linkage disequilibrium in 313 human genes," *Science*, vol. 293, pp. 489–493, July 2001.

[11] A. Braunstein, M. Mézard, and R. Zecchina, "Survey propagation: An algorithm for satisfiability," *Random Structures and Algorithms*, vol. 27, no. 2, pp. 201–226, 2005.

[12] C. Dwork, "Differential privacy," in *Proceedings of the 33rd International Colloquium on Automata, Languages and Programming (ICALP '06), Part II* (M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, eds.), vol. 4052 of *Lecture Notes in Computer Science*, (Venice, Italy), pp. 1–12, Springer, 2006.

[13] G. Cormode, S. Jha, T. Kulkarni, N. Li, D. Srivastava, and T. Wang, "Privacy at scale: Local differential privacy in practice," in *Proceedings of the International Conference on Management of Data*, SIGMOD '18, (New York, NY, USA), pp. 1655–1658, ACM, 2018.

[14] A. B. Carter, "Considerations for genomic data privacy and security when working in the cloud," *The Journal of Molecular Diagnostics*, vol. 21, no. 4, pp. 542–552, 2019.

[15] D. Grishin, K. Obbad, and G. Church, "Data privacy in the age of personal genomics," *Nature Biotechnology*, vol. 37, pp. 1115—1117, 2019.

[16] "The EU General Data Protection Regulation." `https://eugdpr.org/`. [Online; accessed 26-July-2020].

[17] Centers for Medicare & Medicaid Services, "The Health Insurance Portability and Accountability Act of 1996 (HIPAA)." `http://www.cms.hhs.gov/hipaa/`, 1996. [Online; accessed 26-July-2020].

[18] M. Humbert, E. Ayday, J.-P. Hubaux, and A. Telenti, "Addressing the concerns of the lacks family: Quantification of kin genomic privacy," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, CCS'13, (New York, NY, USA), pp. 1141–1152, ACM, 2013.

[19] N. Li and T. Li, "t-closeness: Privacy beyond k-anonymity and l–diversity," *Proceedings of the 32nd International Conference on Very Large Databases*, pp. 139–150, 2006.

[20] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "L diversity: Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data*, vol. 1, no. 1, Article no. 3, 52 pages, 2007.

[21] P. Samurai and L. Sweeney, "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," tech. rep., Computer Science Laboratory, SRI International, 1998.

[22] M. S. Somy, K. S. Gayatri, and B. Ashwini, "Privacy preserving health data mining," *International Journal of Computer Science and Technology*, vol. 6, no. 4, pp. 52–56, 2015.

[23] J. E. Wylie and G. P. Mineau, "Biomedical databases: protecting privacy and promoting research," *Trends in Biotechnology*, vol. 21, no. 3, pp. 113–116, 2003.

[24] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar, "On the privacy preserving properties of random data perturbation techniques," in *Proceedings of the Third IEEE International Conference on Data Mining*, pp. 1–9, IEEE Computer Society Press, 2003.

[25] S. Oliveira and O. Zaane, "Protecting sensitive knowledge by data sanitization.," in *Proceedings of the Third IEEE International Conference on Data Mining*, pp. 613–616, IEEE Computer Society Press, 2003.

[26] T. Churches, "A proposed architecture and method of operation for improving the protection of privacy and confidentiality in disease registers," *BMC Medical Research Methodology*, vol. 3, Article no. 1, 13 pages, 2003.

[27] B. Malin, "Protecting genomic sequence anonymity with generalization lattices," *Methods of Information in Medicine*, vol. 44, pp. 687–92, 02 2005.

[28] E. Ayday, J. L. Raisaro, J. Hubaux, and J. Rougemont, "Protecting and evaluating genomic privacy in medical tests and personalized medicine," in *Proceedings of the 12th ACM Workshop on Workshop on Privacy in the Electronic Society*, WPES '13, pp. 95–106, IEEE Computer Society Press, 2013.

[29] M. Gymrek, A. L. McGuire, D. Golan, E. Halperin, and Y. Erlich, "Identifying personal genomes by surname inference," *Science*, vol. 339, no. 6117, pp. 321–324, 2013.

[30] N. Homer, S. Szelinger, M. Redman, D. Duggan, W. Tembe, J. Muehling, J. V. Pearson, D. A. Stephan, S. F. Nelson, and D. W. Craig, "Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays," *PLOS Genetics*, vol. 4, no. 8, pp. 1–9, 2008.

[31] M. Blanton and M. Aliasgari, "Secure outsourcing of DNA searching via finite automata," in *Data and Applications Security and Privacy XXIV* (S. Foresti and S. Jajodia, eds.), (Springer, Berlin, Heidelberg), pp. 49–64, Springer, 2010.

[32] Y. Chen, B. Peng, X. Wang, and H. Tang, "Large-scale privacy-preserving mapping of human genomic sequences on hybrid clouds," in *Proceedings of the 19th Annual Network and Distributed System Security Symposium*, NDSS '12, 2012.

[33] J. R. Troncoso-Pastoriza, S. Katzenbeisser, and M. Celik, "Privacy preserving error resilient dna searching through oblivious automata," in *Proceedings of the 14th ACM Conference on Computer and Communications Security*, CCS '07, (New York, NY, USA), pp. 519–528, ACM, 2007.

[34] P. Baldi, R. Baronio, E. D. Cristofaro, P. Gasti, and G. Tsudik, "Countering GATTACA: Efficient and secure testing of fully-sequenced human genomes," *CoRR*, vol. abs/1110.2478, 2011.

[35] N. Karvelas, A. Peter, S. Katzenbeisser, E. Tews, and K. Hamacher, "Privacy-preserving whole genome sequence processing through proxy-aided oram," in *Proceedings of the 13th Workshop on Privacy in the Electronic Society*, WPES '14, (New York, NY, USA), pp. 1–10, ACM, 2014.

[36] Z. Huang, E. Ayday, J.-P. Hubaux, J. Fellay, and A. J. Genoguard, "Protecting genomic data against brute-force attacks," in *Proceedings of IEEE Symposium on Security and Privacy*, 2015.

[37] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*. San Francisco, CA, USA: Morgan Kaufmann Publishers, Inc., 2 ed., 2008.

[38] J. A. Bloom, I. J. Cox, T. Kalker, J.-P. Linnartz, M. L. Miller, and C. Traw, "Copy protection for DVD video," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1267–1276, 1999.

[39] M. Maes, T. Kalker, J.-P. Linnartz, J. Talstra, F. Depovere, and J. Haitsma, "Digital watermarking for DVD video copy protection," *IEEE Signal Processing Magazine*, vol. 17, pp. 47–57, 2000.

[40] N. Memon and P. W. Wong, "A buyer-seller watermarking protocol," *IEEE Transactions on Image Processing*, vol. 10, pp. 643–649, 2001.

[41] B. Chen, G. Coatrieux, G. Chen, X. Sun, J. L. Coatrieux, and H. Shu, "Full 4-D quaternion discrete Fourier transform based watermarking for color images," *Digital Signal Processing*, vol. 28, pp. 106–119, 2014.

[42] F. Huo and X. Gao, "A wavelet based image watermarking scheme," in *Proceedings of the International Conference on Image Processing*, ICIP '06, pp. 2573–2576, 2006.

[43] C.-C. Chang, J.-Y. Hsiao, and C.-S. Chan, "Finding optimal least-significant-bit substitution in image hiding by dynamic programming strategy," *Pattern Recognition*, vol. 36, no. 7, pp. 1583–1595, 2003.

[44] R.-Z. Wang, C.-F. Lin, and J.-C. Lin, "Image hiding by optimal LSB substitution and genetic algorithm," *Pattern Recognition*, vol. 34, no. 3, pp. 671–683, 2001.

[45] J. J. K. O. Ruanaidh, W. J. Dowling, and F. M. Boland, "Phase watermarking of digital images," in *Proceedings of 3rd IEEE International Conference on Image Processing*, vol. 3 of *ICIP '96*, pp. 239–242, 1996.

[46] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for images, audio and video," in *Proceedings of 3rd IEEE International Conference on Image Processing*, vol. 3 of *ICIP '96*, pp. 243–246, 1996.

[47] A. Piva, M. Barni, F. Bartolini, and V. Cappellini, "DCT-based watermark recovering without resorting to the uncorrupted original image," in *Proceedings of International Conference on Image Processing*, vol. 1 of *ICIP '97*, pp. 520–523, 1997.

[48] M. Holliman, N. Memon, B.-L. Yeo, and M. Yeung, "Adaptive public watermarking of DCT-based compressed images," in *Proceedings of SPIE*, vol. 3312, pp. 284–295, SPIE, 01 1998.

[49] M. Asad, J. Gilani, and A. Khalid, "An enhanced least significant bit modification technique for audio steganography," in *International Conference on Computer Networks and Information Technology*, pp. 143–147, 2011.

[50] H. O. Oh, J. W. Seok, J. W. Hong, and D. H. Youn, "New echo embedding technique for robust and imperceptible audio watermarking," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3 of *ICASSP '01*, pp. 1341–1344, 2001.

[51] B. Chen and G. W. Wornell, "Quantization index modulation: a class of provably good methods for digital watermarking and information embedding," *IEEE Transactions on Information Theory*, vol. 47, no. 4, pp. 1423–1443, 2001.

[52] V. Bhat K, I. Sengupta, and A. Das, "An audio watermarking scheme using singular value decomposition and dither-modulation quantization," *Multimedia Tools and Applications*, vol. 52, no. 2, pp. 369–383, 2011.

[53] J. Brassil, S. Low, and N. Maxemchuk, "Copyright protection for the electronic distribution of text documents.," *Proceedings of the IEEE*, vol. 87, pp. 1181–1196, 1999.

[54] J. Brassil, S. Low, N. Maxemchuk, and L. O'Gorman, "Hiding information in document images," *Proceedings of Conference on Information Sciences and Systems*, pp. 482–489, 1994.

[55] M. Topkara, U. Topkara, and M. J. Atallah, "Words are not enough: Sentence level natural language watermarking," in *Proceedings of the 4th ACM International Workshop on Contents Protection and Security*, MCPS '06, (New York, NY, USA), pp. 37–46, ACM, 2006.

[56] M. J. Atallah, C. J. McDonough, V. Raskin, and S. Nirenburg, "Natural language processing for information assurance and security: An overview and implementations," *Proceedings of the Workshop on New Security Paradigms*, pp. 51–65, 2000.

[57] M. J. Atallah, V. Raskin, M. Crogan, C. Hempelmann, F. Kerschbaum, D. Mohamed, and S. Naik, "Natural language watermarking: Design, analysis, and a proof-of concept implementation," in *Proceedings of the 4th International Workshop on Information Hiding, IH '2001*, vol. 2137 of *Lecture Notes in Computer Science*, pp. 185–200, 2001.

[58] D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data," in *Advances in Cryptology — CRYPT0' 95* (D. Coppersmith, ed.), (Berlin, Heidelberg), pp. 452–465, Springer, 1995.

[59] E. Ayday, E. Yilmaz, and A. Yilmaz, "Robust optimization-based watermarking scheme for sequential data," in *Proceedings of the 22nd International Symposium on Research in Attacks, Intrusions and Defenses*, RAID '19, (Beijing, China), pp. 323–336, USENIX Association, 2019.

[60] S. S. Kozat, M. Vlachos, C. Lucchese, H. V. Herle, and P. S. Yu, "Embedding and retrieving private metadata in electrocardiograms," *Journal of Medical Systems*, vol. 33, pp. 241–259, Aug. 2008.

[61] S. Iftikhar, S. Khan, Z. Anwar, and M. Kamran, "GenInfoGuard–a robust and distortion-free watermarking technique for genetic data," *PLOS One*, vol. 10, No. 2, Article no. e0117717. 22 pages, 2015.

[62] M. Liss, D. Daubert, K. Brunner, K. Kliche, U. Hammes, A. Leiherer, and R. Wagner, "Embedding permanent watermarks in synthetic genes," *PLOS One*, vol. 7, No. 8, Article no. e42465. 10 pages, pp. 1–10, 08 2012.

[63] D. Heider and A. Barnekow, "DNA watermarks: A proof of concept," *BMC Molecular Biology*, vol. 9, Article no. 40, 10 pages, Apr 2008.

[64] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: Differential privacy for location-based systems," *CoRR*, vol. abs/1212.1984, 2012.

[65] S. A. Glantz, B. K. Slinker, and T. B. Neilands, *Primer of Applied Regression & Analysis of Variance*. McGraw-Hill Education, 2016.

[66] L. E. Miller, "Correlations: Description or inference?," *Journal of Agricultural Education*, vol. 35, no. 1, p. 5–7, 1994.

[67] P. Kairouz, S. Oh, and P. Viswanath, "Extremal mechanisms for local differential privacy," *CoRR*, vol. abs/1407.1338, 2014.

[68] I. Deznabi, M. Mobayen, N. Jafari, O. Tastan, and E. Ayday, "An inference attack on genomic data using kinship, complex correlations, and phenotype information," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 15, no. 4, pp. 1333–1343, 2018.