QUANTIFYING AND PROTECTING GENOMIC 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
Mohammad Mobayenjarihani
July 2018

Quantifying And Protecting Genomic Privacy
By Mohammad Mobayenjarihani
July 2018

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.

Erman Ayday(Advisor)

Atila Bostan

Altay Guvenir

Approved for the Graduate School of Engineering and Science:

Ezhan Karaşan
Director of the Graduate School

ABSTRACT

QUANTIFYING AND PROTECTING GENOMIC PRIVACY

Mohammad Mobayenjarihani
M.S. in Computer Engineering
Advisor: Erman Ayday
July 2018

Today, genome sequencing is more accessible and affordable than ever. It is also possible for individuals to share their genomic data with service providers or on public websites. Although genomic data has significant impact and widespread usage on medical research, it puts individuals' privacy in danger, even if they anonymously or partially share their genomic data. In this work, first, we improve the existing work on inference attack on genomic privacy using observable Markov model, recombination model between the haplotypes, kinship relations, and phenotypic traits. Then to address this privacy concern, we present a differential privacy-based framework for sharing individuals' genomic data while preserving their privacy. Different from existing differential privacy-based solutions for genomic data (which consider privacy-preserving release of summary statistics), we focus on privacy-preserving sharing of actual genomic data. We assume an individual with some sensitive portion on his genome (e.g., mutations or single nucleotide polymorphisms - SNPs that reveal sensitive information about the individual). The goals of the individual are to (i) preserve the privacy of his sensitive data, (ii) preserve the privacy of interdependent data (data that belongs to other individuals that is correlated with his data), and (iii) share as much data as possible to maximize utility of data sharing. As opposed to traditional differential privacy-based data sharing schemes, the proposed scheme does not intentionally add noise to data; it is based on selective sharing of data points. Previous studies show that hiding the sensitive SNPs while sharing the others does not preserve individual's (or other interdependent peoples') privacy. By exploiting auxiliary information, an attacker can run efficient inference attacks and infer the sensitive SNPs of individuals. In this work, we also utilize such inference attacks, which we discuss in details first, in our differential privacy-based data sharing framework and propose a SNP sharing platform for individuals that provides differential privacy guarantees. We show that the proposed framework does not provide sensitive information to the attacker while it provides a high data sharing utility. Through experiments on real data, we extensively study the relationship between utility and several parameters that effect privacy. We also compare the proposed technique with the previous ones and show our advantage both in terms of privacy and data sharing utility.

Keywords: Genomic Privacy, Inference Attacks, Differential Privacy.

ÖZET

TÜRKÇE BAŞLIK

Mohammad Mobayenjarihani
Bilgisayar Mühendisliği, Yüksek Lisans
Tez Danışmanı: Erman Ayday
Temuz 2018

Günümüzde, genom dizilimi her zamankinden daha erişilebilir ve hesaplıdır. Ayrıca bireylerin genom verilerini servis sağlayıcıları veya kamuya açık web sitelerinde paylaşmaları da mümkündür. Genomik verilerin tıbbi araştırmalarda önemli bir etkisi ve yaygın kullanımı olmasına rağmen, genetik verilerini anonim veya kısmen paylaşsalar bile bireylerin gizliliğini tehlikeye atmaktadır. Bu çalışmada, ilk olarak, gözlemlenebilir Markov modeli, haplotipler, akrabalık ilişkileri ve fenotipik özellikler arasındaki rekombinasyon modeli kullanılarak genomik gizlilik çıkarsama saldırısı üzerinde mevcut çalışmaları geliştiriyoruz. Daha sonra bu gizlilik konusunu ele almak için, bireylerin mahremiyetlerini korurken genomik verilerini paylaşmaya yönelik gizlilik temelli farklı bir yazilim çerçevesi sunuyoruz. Genomik veriler için var olan farklı gizlilik temelli çözümlerden farklı olarak (özet istatistiklerinin gizliliğin korunmasını da göz önünde bulundurarak), gerçek genomik verilerin gizliliğinin korunarak paylaşilmasina odaklanıyoruz. Kendi genomunda (örneğin, mutasyonlar veya tek nükleotid polimorfizmleri - bireyle ilgili hassas bilgileri açığa çıkaran SNP'ler) bazı hassas kısımları olan bir bireyi ele aliyoruz. Bireyin amaçları (i) hassas verilerinin gizliliğini korumak, (ii) birbirine bağlı verilerin gizliliğini korumak (kendi verileriyle ilişkili olan diğer bireylere ait veriler) ve (iii) veri paylaşiminin faydasini artirabilmek icin mumkun oldugunca fazla veri paylaşmak. Geleneksel farklı gizlilik temelli veri paylaşım şemalarının aksine, önerilen plan, verilere kasıtlı olarak gürültü eklemez; veri noktalarının seçici bir şekilde paylaşılmasına dayanır. Önceki çalışmalar, diğerlerini paylaşırken hassas SNP'leri gizlemenin, bireyin (ya da diğer birbirine bağlı halkların) gizliliğini korumadığını göstermektedir. Yardımcı bilgilerden yararlanarak, bir saldırgan, etkili çıkarım saldırıları gerceklestirebilir ve bireylerin hassas SNP'lerini çıkartabilir. Bu çalışmada, öncelikle gizlilik temelli veri paylaşımı çerçevemizde, ayrıntılı olarak tartıştığımız bu çıkarım saldırılarını ve farklı gizlilik garantileri sağlayan bireyler için bir SNP paylaşım platformu önermekteyiz. Önerilen çerçevenin, yüksek bir veri paylaşımi sağlarken saldırgana hassas bilgiler elde edemedigini gösteriyoruz. Gerçek veriler üzerinde yapılan deneyler sayesinde, fayda ile gizliligi etkileyen çeşitli parametreler arasındaki ilişkiyi kapsamlı bir şekilde inceliyoruz.Ayrıca, önerilen tekniği daha öncekilerle karşılaştırıyoruz ve hem gizlilik hem de veri paylaşımı yararı açısından avantajımız olduğunu gösteriyoruz.

Anahtar sözcükler: genom gizliliki, farklı mahremiyet, çıkarım saldırıları .

Acknowledgement

Foremost, I would like to thank my supervisor Asst. Prof. Erman Ayday for all of his support and guidance. Of course without his patience, help, and creative ideas none of these would be possible for me.

Also, I am thankful to my friends for their support: Saharnaz Esmailzadeh-Dilmaghani, Noushin Salek-Faramarzi ,Nazanin Jafari, Iman Deznabi, Hamed Rezanezhad-Asl-Bonab, Mina Elhami-Asl, Mohammad Javaheri, Omid Safarzadeh, Aytek Aman, Arif Usta, and all of the other officemates.

I would also like to acknowledge the financial and technical support of the Computer Engineering Department at Bilkent University. I would like to acknowledge the support of Asst. Prof. Ercument Çiçek during this thesis, his help has a significant impact in my thesis. I would thank department chair, Dr. Altay Güvenir, and administrative assistant Ebru Ateş for their kind helps.

I would like to thank my parents and my sister, Fatemeh, for all of their love, support, and motivation throughout these years. All of my academic achievements would have been impossible without the support of my family.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Taking benefits of low cost and accessible sequencing of genomes, nowadays, even ordinary individuals can obtain their digital genome sequences in an affordable way via some online services such as 23andme [2]. They also share their genomic data with medical institutions, on public repositories (such as OpenSNP [3]), and with other direct-to-consumer service providers. Individuals typically use such services to be informed about their predisposition to certain diseases (e.g., cancer) [4, 5], to find their ancestors, or even to find compatible genomic partners. Moreover, this wide availability of genomes opens a new horizon for research in medical field (e.g., treatment of genomic-related diseases or personalized medicine). Although these direct-to-consumer services and potential revolution in medicine look appealing, they also raise significant privacy concerns and ramifications.

Because genes have critical information about one's medical profile and predisposition to sensitive diseases, once the identity of a genome donor is revealed, he or she is faced with the risk of discrimination by employers or insurance companies. Therefore, almost all public genomic data sharing repositories hide the identities of their donors (or participants). However, it has been shown that anonymization is not an effective technique for privacy-preserving genomic data sharing [6, 7].

Despite such risks, users in some online platforms (such as OpenSNP) share their genomic data with their identities, or some scientists publish their own genomic data on their personal websites [8]. Such individuals tend to hide sensitive parts of their genome (e.g., parts that reveal their predisposition to a sensitive disease) while sharing their genomic data. However, it has been shown that hiding is not sufficient for privacy. One prominent example of such is the Apolipoprotein E (APOE) status of James Watson (co-discoverer of the DNA). James Watson has publicly shared his DNA sequence except for the Apolipoprotein E (APOE) gene, which is the main predictor for the development of Alzheimer's disease. Although Dr. Watson tried to hide his APOE status, later it has been shown that it is possible to predict his APOE status [9] using the pairwise correlations that exists between single nucleotide polymorphisms (SNPs) in the genome, also referred to as *linkage disequilibrium* (LD) [10].[1]. According to the mentioned works, we need to define a quantity ot measure the genomic privacy of individuals.

Humbert et al. previously proposed a framework to quantify genomic privacy of individuals considering (i) partial genomic data that is publicly shared by the individual and his family members, (ii) simple pairwise correlations in the genome (i.e., linkage disequilibrium), and (iii) other public genomic knowledge (e.g., minor allele frequencies) [11]. In a recent study, Samani *et al.* showed that higher order correlations in the genome actually enables stronger inference power compared to the pairwise correlations [45]. However, in that work, authors did not study the implications of this result on kin genomic privacy.

For the first part of our work, Chapter 3, we show the extend of privacy risk on the individuals and their family members due to (i) complex correlations (i.e., high order correlations) in the genome, and (ii) publicly available phenotype information (e.g., physical traits or disease information) about the individuals. The main objective of chapter is to develop a new unifying framework for quantification of genomic privacy of individuals. Similar to the previous work[11], we use a graph-based, iterative algorithm to build this framework efficiently. Our results show that the attacker's inference power (on the genomic data of individuals)

[1] All auxiliary information for such an attack (e.g., methodology and the dataset to compute such correlations) are publicly available.

significantly improves by using complex correlations and phenotype information (along with information about their family bonds). We show that hiding the genomic data partially is not sufficient to preserve the privacy of individuals and even their family.

Although public availability of genome sequences is a privacy threat, limiting access to public genomic datasets is a barrier for both medical research and all of the aforementioned benefits. Thus, we need a trade-off between utility and privacy. That is, we need a method to ensure individuals about their sensitive genes' privacy, while providing high genomic data sharing utility to the researchers. In this paper, we build a framework to protect the privacy of individuals' genomic data while providing high utility for genomic data sharing. Our proposed technique relies on the differential privacy concept [12] to control the trade-off between the utility and privacy.

Differential privacy technique has been already used in genomics literature to privately release summary statistics (e.g., for privacy-preserving genome-wide association studies - GWAS). Such works generally focus on secure sharing of summary statistics [13], finding associated SNPs to a disease and locating them [14], and scalable data sharing for GWAS [15]. Different from these works that focus on privacy-preserving sharing of summary statistics, in the second part of this work, Chapter 4 ,we use the differential privacy concept for privacy-preserving sharing of individuals' genome sequences (or a data sequence in general).

Inspired by Miguel *et al.*'s work on location privacy [16], we use differential privacy concept in order to establish a method to control the trade-off between the utility and privacy for genomic data sharing. In [16], Miguel *et al.* have used the differential privacy concept for obfuscating and sharing individuals' location data. In a nutshell, they have proposed obfuscating a location within a radius of r (by adding Laplacian noise) before sharing it with a location-based service provider. They have also proved that their proposed mechanism implies ϵ-differential privacy. Here, we propose a similar idea for genomic data sharing. The main differences of our proposed work from [16] are as follows: (i) we consider the inherent correlations in data while sharing it, and (ii) rather than adding noise (which implies modifying the content of genomic data, and hence is not acceptable among medical researchers), we selectively decide whether or not to share particular SNPs based on our formulation. Different from previous work on genomic data sharing [1], in our proposed mechanism, not sharing a SNP does not provide any information about the value of that SNP (or other sensitive SNPs) to the attacker. We also consider and preserve the privacy of interdependent data (e.g., genomic privacy of family members).

We assume an individual (called the donor) with a genomic data sequence that includes some sensitive SNPs (e.g., the ones revealing his predisposition to a sensitive disease).[2] Our goal is to protect the sensitive part of sequence from inference attacks while sharing as much as possible from the rest (non-sensitive part). The attacker tries to infer the individual's sensitive SNPs by using existing inference attacks (e.g., using kinship information and correlations among the SNPs) and it has access to the public genomic datasets of different ethnicities (e.g., from [17, 18, 3]) to build its statistical models for the inference attacks.

The donor sequentially decides whether or not to share each of his non-sensitive SNPs. For each decision, we quantify the risk of inference for the sensitive SNPs. Then, using our formulation of differential privacy concept, we check if the information available to the attacker (with the sharing of the corresponding SNP) exceeds a predetermined boundary both for the donor and other interdependent individuals (e.g., his family members). Based on this, we decide whether or not to share the corresponding SNP. To demonstrate the common scenarios that happen during the inference attack and sharing procedure, we also provide a toy example (in Chapter 4.1.5). We show how our proposed mechanism prevents the attacker from gaining any extra information about sensitive SNPs beyond a predetermined boundary. More importantly, we show how neither hide nor share decision (for a non-sensitive SNP) leak any information about the values of the SNPs in the sensitive SNP set. This is because the proposed SNP sharing mechanism does not consider the real values of the sensitive SNPs. We also formally prove that

[2] Sensitive part of the genome is not fixed, it may vary among individuals.

this formulation implies ϵ-differential privacy.

We evaluate the proposed mechanism on real genomic data belonging to Central European population [18]. We study the effects of various design parameters on the privacy and utility. We also compare the proposed scheme with the existing work of Humbert *et al.* that proposes an optimization-based solution for the same problem [1]. We experimentally show that our proposed scheme provides both higher privacy (in terms of entropy and error) and utility compared to [1].

The rest of this work is organized as follows. In Chapter 2, we bring a brief introduction about genomics and technical preliminaries, moreover, we summarize the related work in the literature. In Chapter 3, we discuss the quantifying of genomic privacy plus we explain our inference attack algorithm and evaluate it with different kinship datasets. In Chapter 4, we explain our privacy preserving framework in detail, we also evaluate it against the discussed inference attack algorithm. Finally, in Chapter 5, we conclude our work and discuss the future work.

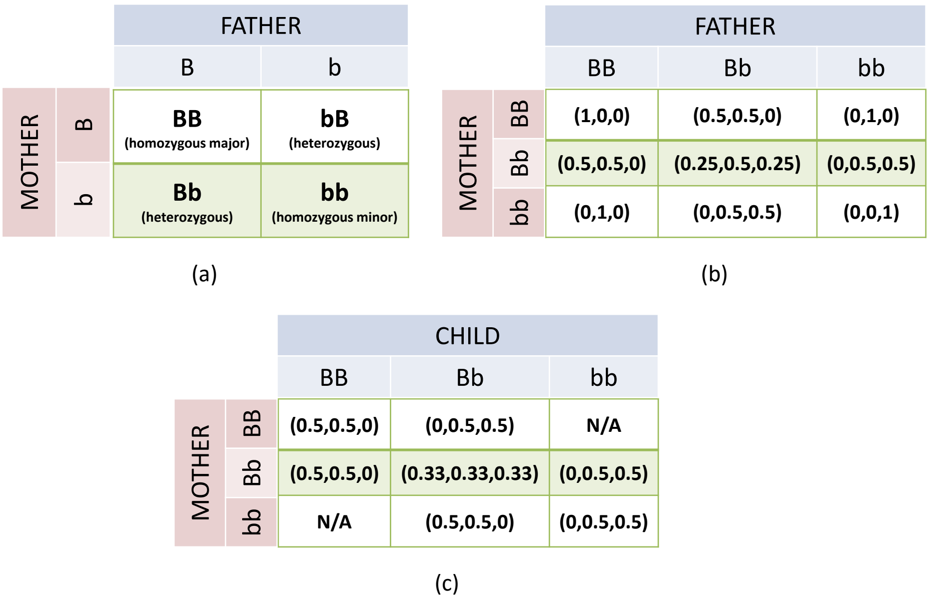
Chapter 2

Background and Related Work

2.1 Genomic

Single nucleotide polymorphism (SNP): Around 99.9% of an individual's genome is identical to the reference human genome and the rest is human genetic variation. The most common genetic variations in humans are the SNPs. SNP is a variation in the genome in which a single nucleotide (A, C, G, or T) differs between members of the same species or paired chromosomes of an individual. There are usually two different alleles (nucleotides) that are observed at a SNP position; one is called the *minor allele* and the other is the *major allele*. Furthermore, each SNP carries two alleles in total. Hence, the content of a SNP position can be in one of the following states: (i) BB (homozygous-major genotype), if an individual receives the same major allele from both parents; (ii) Bb (heterozygous genotype), if he receives a different allele from each parent (one minor and one major); or (iii) bb (homozygous-minor genotype), if he inherits the same minor allele from both parents (this is also shown in Fig. 2.1(a)). For simplicity, in the rest of the paper, we denote the value (content) of a SNP as the number of minor alleles it carries. Thus, we denote BB as 0, Bb as 1 and bb as 2.

Reproduction: The Mendel's first law, the Law of Segregation, states that a child's SNPs are independent from his ancestors', given the SNPs of his parents. Each child inherits one allele (nucleotide) of a SNP from his mother and the other one from his father, and each allele is inherited with a probability of 0.5. In [19] authors model this law by a function (introduced in Section 3.2) that simply considers the Mendelian inheritance probabilities as in Fig. 2.1(b). We also use this inheritance information in this work.



		FATHER	
		B	b
MOTHER	B	**BB** (homozygous major)	**bB** (heterozygous)
	b	**Bb** (heterozygous)	**bb** (homozygous minor)

(a)

		FATHER		
		BB	Bb	bb
MOTHER	BB	(1,0,0)	(0.5,0.5,0)	(0,1,0)
	Bb	(0.5,0.5,0)	(0.25,0.5,0.25)	(0,0.5,0.5)
	bb	(0,1,0)	(0,0.5,0.5)	(0,0,1)

(b)

		CHILD		
		BB	Bb	bb
MOTHER	BB	(0.5,0.5,0)	(0,0.5,0.5)	N/A
	Bb	(0.5,0.5,0)	(0.33,0.33,0.33)	(0,0.5,0.5)
	bb	N/A	(0.5,0.5,0)	(0,0.5,0.5)

(c)

Figure 2.1: (a) Mendelian inheritance for a child. (b) Inheritance probabilities for a SNP, given different genotypes for the parents. The probabilities of the child's genotype are represented in parentheses. (c) Inheritance probabilities for a SNP, given different genotypes for the child and the mother. The probabilities of the father's genotype are represented in parentheses (given the child and the father, the probabilities for the mother are also the same).

Correlations in the genome: It is shown that SNPs on the DNA sequence are correlated. For example, pairwise correlations between the SNPs in the genome are referred to as linkage disequilibrium (LD) [20]. In [19], the authors use the LD values between the SNPs as an input to their inference algorithm. In this work, we show that more complex, higher order correlations in the genome threaten kin genomic privacy more than the pairwise correlations.

Phenotypes: Phenotypes are observable characteristics of individuals (e.g., physical traits or diseases) that may be related to both their genotype and the environment. For example, SNP *Rs12821256* on chromosome 12 is associated with having blonde hair. If an individual has (C,C) nucleotide pair for this SNP, he is 4 times more likely to have blonde hair compared to other individuals. We use phenotype information of individuals to improve the inference power of the proposed algorithm.

2.2 Differential Privacy

Differential privacy [12] is a concept to preserve privacy of records in statistical databases. Its aim is to preserve a record's privacy while publishing statistical information about the database. Differential privacy assumes that any slight change in the database (e.g., addition or deletion of a single record) should have a negligible effect on the outcome of a query to the corresponding database. The general assumption about the attacker is that it knows the entire records in the database except for one and by issuing queries, it tries to perform a membership inference attack on that unknown record. More formally, differential privacy guarantees that an algorithm behaves approximately the same on two neighboring databases (that differ by a single record) as follows:

$$Pr[\mathcal{K}(D1) \in S] \leq exp(\epsilon) \times Pr[\mathcal{K}(D2) \in S], \tag{2.1}$$

where $D1$ and $D2$ are neighboring databases, $\mathcal{K}$ is a randomized algorithm, and S is the output of the randomized algorithm ($\mathcal{K}$). Function $\mathcal{K}$ is then called ϵ-differentially private if (2.1) holds for all neighboring databases.

Although the original formulation of the differential privacy considers neighboring databases, in [21], authors introduce a generalized version of differential privacy. Instead of neighboring databases, they consider vectors x, y in $\mathfrak{R}^n$ such that $|x - y|_1 \leq l$. Let a mechanism be defined as $M = \{\mu_x : x \in \mathfrak{R}^n\}$ with output from the set $S \in \mathfrak{R}^d$. Then, for every vector $x,\ y \in \mathfrak{R}^n$ such that $|x - y|_1 \leq l$, mechanism M is ϵ-differentially private if

$$\frac{\mu_x(S)}{\mu_y(S)} \leq exp(l\epsilon). \tag{2.2}$$

Differential privacy concept has been previously utilized in location privacy to share the location patterns of a user with a location-based service provider [16, 22]. In [16], Miguel *et al.* modify the original definition of differential privacy in order to establish a mechanism for location obfuscation. A user with a real location $x \in X$ obfuscates his location within a predetermined radius of r before sharing it with a location-based service provider. To do so, the user adds noise to his real location and obtains a noisy output $z \in Z$. Authors call this mechanism as ϵ-geo-indistinguishable. A mechanism satisfies ϵ-geo-indistinguishability *iff* for all priors and all observations $S \subseteq Z$

$$fracP(x|S)P(x'|S) \leq e^{\epsilon r}\frac{P(x)}{P(x')} \qquad \forall r > 0 \quad \forall x, x' : d(x, x') \leq r, \tag{2.3}$$

where the x and x' are the locations that are apart by at most r and $d(x, x')$ is the Euclidean distance between x and x'. Also, S is the set of noisy locations (noise is sampled from a Laplacian distribution). Authors also proved that (2.3) is equivalent to (2.2). We develop our proposed mechanism to share genomic data inspired from the generalized definition of differential privacy and its utilization for location patterns.

2.3 Inference Attack on Kin Genomic Privacy

Here, we briefly describe the inference attack on kin genomic privacy proposed in [11]. The attacker has access to the following resources:

(i) publicly available genomic datasets belonging to different populations [17, 23], (ii) family tree and family relationships between the individuals, and (iii) genomic data (partial or whole) that is shared by a subset of the family members. Besides these resources, the attacker uses Mendel's law (of inheritance) and high-order correlations between the SNPs [24].

The goal of the attacker is to infer the missing parts of the genomes of the family members (or a target individual in the family). All aforementioned resources and methodologies provide some information to the attacker about the probability distributions of the unknown SNPs. Thus, the attacker may use these resources to calculate the marginal probability distributions of unknown SNP values. To do this calculation in an efficient way, using a message passing algorithm (belief propagation [25, 26]) on a graphical model (factor graph) is proposed. A factor graph is a bipartite graph that includes two sets of nodes: (i) variable nodes that represent the SNPs of family members, and (ii) factor nodes that represent the dependencies between the resources of the attacker and the variable nodes. We discuss this attack methodology in In this setting, the factor nodes represent: (i) familial relationships (and hence the Mendel's law) between family members, (ii) high-order correlations between SNPs in the genome, and (iii) genotype-phenotype relationships between the SNPs and physical characteristics of individuals. The nodes on the factor graph are connected via edges (depending on the relationship between them) and through these edges, they iteratively exchange messages throughout the iterative algorithm. At the beginning, each variable node has its own belief about the marginal probability distribution of the corresponding SNP (computed using the MAF values). Then, the iterative algorithm starts and at each round, nodes generate and send messages (in the form of conditional probabilities) to their neighbors until the marginal probability distributions of the variable nodes converge.

2.4 Related Work

Genomic privacy topic has been recently explored by many researchers [27]. Several works have studied various inference attack against genomic data. Homer *et al.* showed that membership of an individual in a study group can be inferred using public statistics published about that group [28]. Later, Wang *et al.* showed that this attack can be even more severe by also considering the inherent pairwise correlations in the genome [29]. Recently, Shringarpure and Bustamante showed that presence of an individual in a genome sharing beacon (genomic datasets that only allow yes/no queries on the presence of specific alleles in the dataset) can be inferred using a likelihood-ratio test by repeatedly querying the beacon for SNPs of the victim [30]. Humbert *et al.* proposed an efficient inference attack to quantify kin genomic privacy using the family ties between individuals, pairwise correlations between the SNPs (LD), and publicly available statistics about DNA [11]. Samani *et al.* has shown that adversary can use high-order and complex correlation in the genome (e.g., Markov chain model and recombination) in order to infer the hidden parts of a targeted individual's genome more accurately compared to using LD [24]. Several countermeasures have been proposed to mitigate the aforementioned threats. Some researchers proposed using cryptographic techniques for privacy-preserving processing of genomic data. Jha *et al.* proposed a method for secure comparison of DNA sequences [31]. Blanton *et al.* focused on secure outsourcing of sequence comparisons [32]. Cassa *et al.* proposed a cryptographic scheme to securely transmit externally generated sequence data which does not require any patient identifiers [33]. Baldi *et al.* proposed cryptographic techniques for privacy-preserving computations on genomic data using private set intersection [34]. Ayday *et al.* proposed partially homomorphic encryption for privacy-preserving use of genomic data in clinical settings [35]. Recently, Wang *et al.* proposed private edit distance protocols to find similar patients (across several hospitals) [36]. Some researchers proposed using the differential privacy concept [21] to release summary statistics in a privacy-preserving way (to mitigate membership inference attacks). Fienberg *et al.* used the differential privacy concept for sharing the statistics such as minor allele frequencies, p-values, and chi-square values [13]. Johnson and Shmatikov proposed using the exponential mechanism for computation and release of (i) number of SNPs that are associated with the specific phenotype, (ii) the most significant SNPs related to a phenotype, (iii) p-values, and (iv) correlation between pairs of SNPs [14]. Yu *et al.* extended the work of Feinberg *et al.* and presented a scalable algorithm for any arbitrary number of SNPs [15]. Different from existing differential privacy-based approaches, in this work, we use the differential privacy concept to share the genomic sequence of an individual, not summary statistics. To share genomic sequences in a privacy-preserving way, Humbert *et al.* proposed an optimization-based technique that selectively hides portions of shared genomic data by considering the privacy budgets of both the donor and his family members [1]. Another goal of Humbert *et al.*'s work is to maximize the genomic data sharing utility (by maximizing the number of SNPs shared). This work is the closest in literature to ours. We compare our proposed mechanism with the work of Humbert *et al.* and show that our work outperforms [1] both in terms of privacy and utility.

Chapter 3

Attack On The Genomic Privacy

In this chapter first we talk about a message passing algorithm called belief propagation, then quantizing genomic privacy, next we discuss about the attack methodology, and finally we bring the evaluation and the results of an attack on the genomic privacy.

3.1 Belief Propagation

Belief propagation [37] is a message-passing algorithm for performing inference on graphical models (e.g., Bayesian networks or Markov random fields). It is typically used to compute marginal distributions of unobserved variables conditioned on the observed ones. Computing marginal distributions is hard in general as it might require summing over an exponentially large number of terms. The belief propagation algorithm can be described in terms of operations on a factor graph, a graphical model that is represented as a bipartite graph. One of the two disjoint sets of the factor graph's vertices represents the (random) variables of interest, and the second set represents the functions that factor the joint probability distribution (or global function) of the variables based on the dependencies between them. An edge connects a variable node to a factor node if and only if the variable is an argument of the function corresponding to the factor node. The marginal distribution of an unobserved variable can be exactly computed by using the belief propagation algorithm if the factor graph has no cycles. However, the algorithm is still well defined and often gives good approximate results for factor graphs with cycles (as it has been observed in decoding of LDPC codes) [38]. Belief propagation is commonly used in artificial intelligence and information theory.

3.2 Quantifying Kin Genomic Privacy

In [19], authors evaluate the genomic privacy of an individual threatened by his relatives revealing their genomes. Focusing on the SNPs in the genome, they quantify the loss in genomic privacy of individuals when one or more of their family members' genomes are (either partially or fully) revealed. They design a reconstruction attack, in which they formulate the SNPs, family relationships, and the pairwise correlations (LD) between SNPs on a factor graph and use the belief propagation algorithm for inference. Then, using various metrics, they quantify the genomic privacy of individuals and reveal the decrease in their level of genomic privacy caused by the published genomes of their family members. In the following, we briefly summarize the framework of [19] as we build the proposed scheme on top of this framework.

The goal of the adversary is to infer some *targeted SNPs* of a member (or multiple members) of a *targeted family*. Let $\mathbf{F}$ be the set of family members in the targeted family (whose family tree is $\mathcal{G}_{\mathbf{F}}$) and $\mathbf{S}$ be the set of SNP IDs (on the DNA sequence), where $|\mathbf{F}| = n$ and $|\mathbf{S}| = m$. Let also x^i_j be the value of SNP j ($j \in \mathbf{S}$) for individual i ($i \in \mathbf{F}$), where $x^i_j \in \{0, 1, 2\}$. Also, $\mathbb{X}$ is an $n \times m$ matrix that stores the values of the SNPs of all family members. Among the SNPs in $\mathbb{X}$, the ones whose values are unknown are in set $\mathbb{X}_{\mathrm{U}}$, and the ones whose values are known (by the adversary) are in set $\mathbb{X}_{\mathrm{K}}$. $\mathcal{F}_R(x^M_j, x^F_j, x^C_j)$ is the function representing the Mendelian inheritance probabilities (as in Fig. 2.1(b)), where (M, F, C) represent mother, father, and child, respectively. Finally, $\mathbf{P} = \{p^b_i : i \in \mathbf{S}\}$ represents the set of minor allele probabilities (or MAF) of the SNPs in $\mathbf{S}$.

The adversary carries out a reconstruction attack to infer $\mathbb{X}_{\mathrm{U}}$ by relying on his background knowledge, $\mathcal{F}_R(x_j^M, x_j^F, x_j^C)$, $\mathbb{L}^1$, $\mathbf{P}$, and on his observation $\mathbb{X}_{\mathrm{K}}$. The authors formulate this reconstruction attack as finding the marginal probability distributions of unknown variables $\mathbb{X}_{\mathrm{U}}$, and to run this attack in an efficient way, they formulate the problem on a factor graph and use the belief propagation algorithm for inference. In this work, we formulate the attack by also considering complex correlations in the genome and publicly available phenotype information. We show that the inference attack is significantly stronger when these additional factors are also considered. In the following, we provide the details of the proposed framework emphasizing the differences from [19].

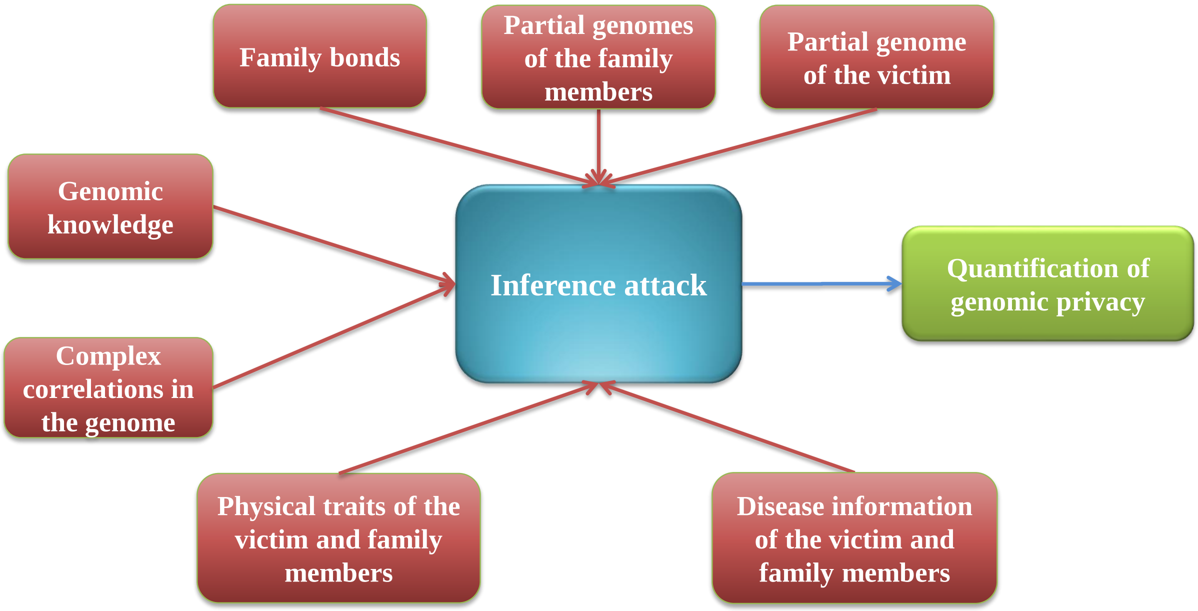

Family bonds
Partial genomes of the family members
Partial genome of the victim
Genomic knowledge
Inference attack
Quantification of genomic privacy
Complex correlations in the genome
Physical traits of the victim and family members
Disease information of the victim and family members

Figure 3.1: Overview of the proposed framework for quantification of genomic privacy.

[1] $\mathbb{L}$ is a $m \times m$ matrix representing the pairwise linkage disequilibrium (LD) between each pair of SNPs. Instead of the LD values, we use higher order correlations in this work for inference.

3.3 Proposed Framework

Our main objective is to develop a unifying framework for the quantification of the genomic privacy of individuals using all available public data on the Web and background knowledge on genomics. We assume that the attacker has access to the following resources about the target individuals: (i) the partial genomic data of individuals (from public genomic databases and genome sharing websites), (ii) phenotype information (physical characteristics) of individuals from OSNs, (iii) health related information of individuals from OSNs and health related social networks, and (iv) family bonds of individuals (e.g., their family trees) from OSNs or genealogy websites. Our proposed framework is also sketched in Fig. 3.1.

The objective is to infer the missing parts of the genomes of individuals in the target individuals set. For this, we use family bonds between the individuals in the target set, probabilistic relationship between the phenotype and genotype, similar relationship between diseases and the genotype, and some genomic tools for inference such as high order correlations in the genome and the recombination model. To run this inference attack efficiently, similar to the previous work, we rely on the belief propagation algorithm on a factor graph. Then, we quantify genomic privacy of individuals and show the risk for each individual.

Constructing the factor graph: A factor graph is a bipartite graph containing two sets of nodes (corresponding to variables and factors) and edges connecting these two sets. We form a factor graph by setting a variable node for each SNP x_j^i ($j \in \mathbf{S}$ and $i \in \mathbf{F}$). We use three types of factor nodes[2]: (i) *familial factor node*, representing the familial relationships and reproduction, (ii) *correlation factor node*, representing the higher order correlations between the SNPs either by using a Markov chain or hidden Markov model, and (iii) *phenotype factor node*, representing the correlation between the SNPs and the phenotypes (e.g., physical traits or diseases) of individuals. The factor graph representation of our

[2] There are two types of factor nodes in [19] representing the family relationships and the LD between the SNPs.

proposed framework is shown in Fig. 3.2. We summarize the connections between the variable and factor nodes below:

- Each variable node x_j^i has its familial factor node f_j^i if at least one parent of individual i is in the target family. Furthermore, x_j^k ($k \neq i$) is also connected to the familial factor node of x_j^i if k is the mother or father of i. If an individual i's both parents are not present in the target family, we do not assign familial factor nodes corresponding to the variable nodes of that individual. For example, in Fig. 3.2, all familial factor nodes belong to the child as his parents are present in the toy example. However, father's and mother's variable nodes do not have separate familial factor nodes.

- Variable nodes in set $\mathbf{C}$ are connected to a correlation factor node $g_{\mathbf{C}}^i$ (of individual i) if SNPs in $\mathbf{C}$ have correlation among each other. In particular, we consider higher order correlations in the genome. We model these correlations either using a Markov chain or a hidden Markov model, HMM (i.e., recombination model). When we use a Markov chainwith order of k the correlation set of node i is $\mathbf{C}_i = \{node_{i-k}, node_{i-k+1}, node_{i-k+2}, \ldots, node_{i-1}\}$ if $i > k$, and $\mathbf{C}_i = \{node_1, node_2, node_3, \ldots, node_{i-1}\}$ if $i \leq k$, and when we use HMM, $\mathbf{C}$ includes all SNPs in a chromosome.

- Variable nodes of individual i in set $\mathbf{H}_\alpha^i$ are connected to a phenotype factor node ph_α^i if SNPs in $\mathbf{H}_\alpha^i$ are associated with the phenotype ph_α. Note that more than one SNP can be associated with a given phenotype. Similarly, a SNP may be associated with more than one phenotype.

Messages between the nodes: As shown in [39], following the rules of belief propagation, the global probability distribution of the variable nodes can be factorized into products of local functions that are defined by the factor nodes following the rules of the belief propagation algorithm. The iterative belief propagation algorithm is based on exchanging messages between the variable and the factor nodes. We represent these messages as in the following:

- The message $\mu_{i\to k}^{(\nu)}(x_j^{i\,(\nu)})$ (from a variable node i to a factor node k) denotes the probability of $x_j^{i\,(\nu)} = \ell$ ($\ell \in \{0,1,2\}$), at the ν^{th} iteration.

- The message $\lambda_{k\to i}^{(\nu)}(x_j^{i\,(\nu)})$ (from a familial factor node to a variable node) denotes the probability that $x_j^{i\,(\nu)} = \ell$, for $\ell \in \{0,1,2\}$, at the ν^{th} iteration given $\mathcal{F}_R(x_j^M, x_j^F, x_j^C)$, $\mathbf{P}$, and the values of SNP j for the other two family members (other than individual i) that are connected to the corresponding familial factor node.

- The message $\beta_{k\to i}^{(\nu)}(\mathbf{C}, x_j^{i\,(\nu)})$ (from a correlation factor node to a variable node) denotes the probability that $x_j^{i\,(\nu)} = \ell$, for $\ell \in \{0,1,2\}$, at the ν^{th} iteration given the high order correlation between the SNPs in set $\mathbf{C}$.

- The message $\delta_{k\to i}^{(\nu)}(x_j^{i\,(\nu)})$ (from a phenotype factor node to a variable node) denotes the probability that $x_j^{i\,(\nu)} = \ell$, for $\ell \in \{0,1,2\}$, at the ν^{th} iteration given the phenotype ph_k for individual i and the association of the corresponding phenotype with SNP j.

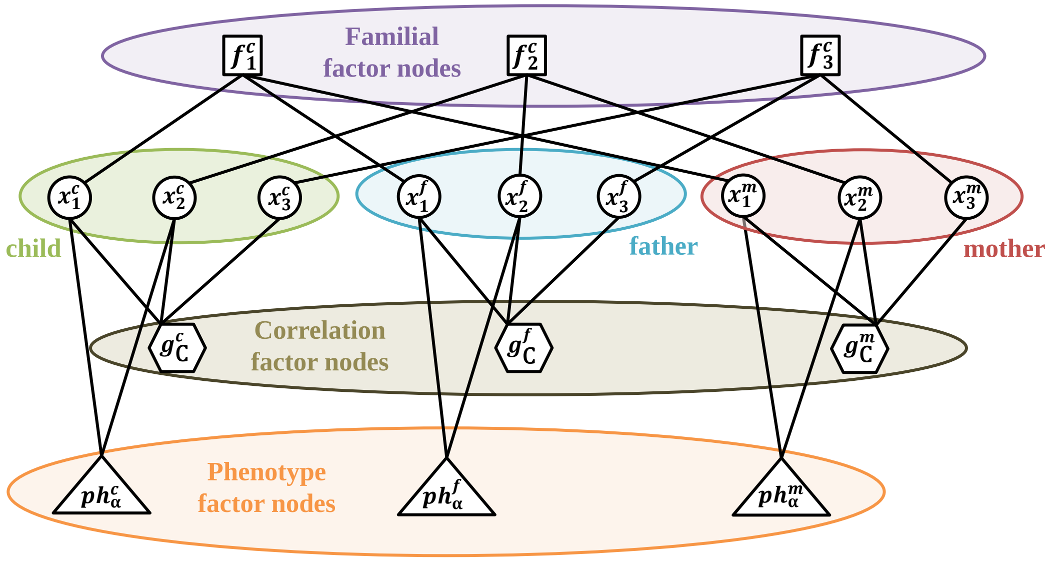


Figure 3.2: Factor graph representation of the proposed framework.

Toy example on a trio: Following [19], we choose a simple family tree consisting of a trio (i.e., mother, father, and child) and 3 SNPs (i.e., $|\mathbf{F}| = 3$ and $|\mathbf{S}| = 3$). In Fig. 3.2, we show how the trio and the SNPs are represented on a factor graph, where $i = m$ represents the mother, $i = f$ represents the father, and $i = c$ represents the child. Furthermore, the 3 SNPs are represented as $j = 1$, $j = 2$, and $j = 3$, respectively. We describe the message exchange between the variable node representing the first SNP of the mother (x_1^m), the familial factor node of the child (f_1^c), the correlation factor node $g_{\mathbf{C}}^m$, and the phenotype factor node ph_α^m (representing the phenotype α for the mother). Here we assume that variable nodes in set $\mathbf{C}$ are SNPs 1, 2, and 3. We also assume that the phenotype α is associated with SNPs 1 and 2 (that are in set $\mathbf{H}_\alpha^m$). The belief propagation algorithm iteratively exchanges messages between the factor and the variable nodes, updating the beliefs on the values of the targeted SNPs (in $\mathbb{X}_{\mathrm{U}}$) at each iteration, until convergence. For simplicity, we denote the variable and factor nodes x_1^m, f_1^c, $g_{\mathbf{C}}^m$, and ph_α^m with the letters i, k, z, and s, respectively.

Messages from variable nodes: Variable node i forms $\mu_{i\to k}^{(\nu)}(x_1^{m(\nu)})$ by multiplying all information it receives from its neighbors excluding the familial factor node k.[3] Hence, the message from variable node i to the familial factor node k at the ν^{th} iteration is given by

$$\mu_{i\to k}^{(\nu)}(x_1^{m(\nu)}) = \frac{1}{Z} \times \beta_{z\to i}^{(\nu-1)}(\mathbf{C}, x_1^{m(\nu-1)}) \times \delta_{s\to i}^{(\nu-1)}(x_1^{m(\nu-1)}), \tag{3.1}$$

where Z is a normalization constant. This computation is repeated for every neighbor of each variable node. If $x_1^m \in \mathbb{X}_{\mathrm{K}}$ (i.e., it is one of the SNPs that is observed by the attacker), then the message $\mu_{i\to k}^{(\nu)}(x_1^{m(\nu)})$ is constructed as a constant, depending on the value of x_1^m. Note that following the rules of belief propagation, to prevent self-bias, the message $\lambda_{k\to i}^{(\nu-1)}(x_1^{m(\nu-1)})$ is not used while generating $\mu_{i\to k}^{(\nu)}(x_1^{m(\nu)})$. Also, if the parents of the mother (m) were also in the graph, x_1^m would have its corresponding familial factor node f_1^m, and hence the λ message generated from this factor node would have been also used when generating $\mu_{i\to k}^{(\nu)}(x_1^{m(\nu)})$. Similarly, if SNP x_1 is associated with other phenotypes, δ messages from those phenotype factor nodes are also used while generating the message.

[3] Other messages from the variable node i to the other factor nodes (z and s) are also constructed similarly.

Messages from familial factor nodes: The message from the familial factor node k to the variable node i at the ν^{th} iteration is formed using the principles of belief propagation as

$$\lambda_{k\to i}^{(\nu)}(x_1^{m(\nu)}) = \sum_{\{x_1^f, x_1^c\}} f_1^c(x_1^m, x_1^f, x_1^c, \mathcal{F}_R(x_j^M, x_j^F, x_j^C), \mathbf{P}) \times$$
$$\prod_{y\in\{f,c\}} \mu_{x_1^y\to k}^{(\nu)}(x_1^{y(\nu)}), \tag{3.2}$$

where, $f_1^c(x_1^m, x_1^f, x_1^c, \mathcal{F}_R(x_j^M, x_j^F, x_j^C), \mathbf{P})$ is proportional to $p(x_1^m|x_1^f, x_1^c, \mathcal{F}_R(x_j^M, x_j^F, x_j^C), \mathbf{P})$, and this probability is computed using the table in Fig. 2.1(b). This computation is performed for every neighbor of each familial factor node.

Messages from correlation factor nodes: The message from the correlation factor node z to the variable node i at the ν^{th} iteration is formed as

$$\beta_{z\to i}^{(\nu)}(\mathbf{C}, x_1^{m(\nu)}) = \sum_{x_2^m, x_3^m} g_C^m(x_1^m, x_2^m, x_3^m) \times$$
$$\prod_{y\in\{2,3\}} \mu_{x_y^m\to k}^{(\nu)}(x_y^{m(\nu)}). \tag{3.3}$$

β messages are generated for every neighbor of each correlation factor node. As mentioned, as opposed to [19], in this work, we consider higher order correlations in the genome to make the inference stronger, and hence the function $g_C^m(x_1^m, x_2^m, x_3^m)$ depends on the correlation model we use. We consider two different correlation models on the genome: (i) Markov chain, in which we consider the genome as a sequence of SNPs, where the value of each SNP depends on the values of neighboring k SNPs. In this scenario, $g_C^m(x_1^m, x_2^m, x_3^m) = p(x_1^m|x_2^m, x_3^m)$, for $k = 2$ (note that LD is a special case of this formalization when $k = 1$). And, (ii) hidden Markov model (HMM), in which the genome is modeled as a Markov process with unobserved (hidden) states. We realize the HMM model for the genome by using the recombination model [40].

Messages from phenotype factor nodes: Finally, the message from the phenotype factor node s to the variable node i at the ν^{th} iteration is formed as

$$\delta_{s\to i}^{(\nu)}(x_1^{m(\nu)}) = \sum_{x_2^m} ph_\alpha^m(x_1^m, x_2^m) \times \mu_{x_2^m\to s}^{(\nu)}(x_2^{m(\nu)}). \tag{3.4}$$

Note that in this toy example, the phenotype α is associated with SNPs x_1 and x_2 only. The function $ph_{\alpha}^{m}(x_1^m, x_2^m)$ is computed based on the association of both SNPs with the corresponding phenotype. In some cases, it is observed that the associations of the SNPs to a phenotype are independent from each other. On the other hand, in some cases, we observe that the association depends on the values of both SNPs. Similarly, in some cases, the association is probabilistic, while in some cases the association may be deterministic. For example, having blonde hair color is associated with SNP *Rs12821256* [41]. If an individual has blonde hair, the probability distribution of the corresponding SNP is shown to be (0.01,0.4,0.59)[4], while if he does not have blonde hair, this distribution is shown to be (0.7,0.28,0.02). Thus, the attacker can improve his inference power by obtaining phenotype information about the individuals in the target family.

At each iteration of the algorithm, all variable and factor nodes generate their messages and send to all of their neighbors as described above. At the end of each iteration, we compute the marginal probabilities of each variable nodes (by multiplying all incoming messages), and we stop the algorithm when the values of the marginal probabilities stop changing. Note that the computational complexity of this inference attack is linear with the number of variable or factor nodes in the factor graph.

3.4 Evaluation

Here, we summarize our methodology to evaluate the proposed inference framework.

3.4.1 Datasets

In order to evaluate our method we used two datasets:

[4] Each entry represents the probability that the value of the SNP is 0, 1, and 2, respectively.

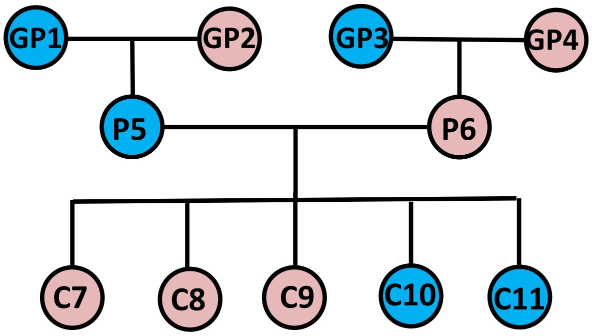


Figure 3.3: Family tree of CEPH/Utah Pedigree 1463 consisting of the 11 family members that were considered. The blue nodes (i.e., darker ones) represent the male and the pink ones (i.e., lighter ones) represent the female family members.

- CEPH/Utah Pedigree 1463
- Manuel Corpas Family Pedigree

3.4.1.1 CEPH/UTAH Pedigree 1463

To evaluate the proposed inference algorithm, we used the CEPH/Utah Pedigree 1463 dataset [42][5]. We obtained the SNP data both in the genome variant (GVF) and variant call (VCF) formats. Dataset contains partial DNA sequences of 17 family members and we used 11 of these 17 individuals (to be consistent with the previous work). The family bonds between these 11 individuals are illustrated in Fig. 3.3.

We focused on 100 neighboring SNPs (on the DNA sequence) of the target family on the 22nd chromosome. We also collected data for the MAF and also to model the higher order correlations in the genome. For this purpose, we used data of the CEU population from the 1000 Genomes Project and HapMap.

[5] The previous work by Humbert et al. also use the same dataset.

3.4.1.2 Manuel Corpas Family Pedigree

Manuel Corpas is a scientist, who released his family DNA dataset in variant call format (VCF) on his website [43]. The dataset consists DNA sequences of father, mother, son (Manuel Corpas), daughter, and aunt. The family tree of the individuals in this dataset is illustrated in Fig. 3.4. Similar to the CEPH/UTAH Pedigree dataset setup, for this dataset, we focused on the 22nd chromosome and selected 100 neighboring SNPs of each family member.

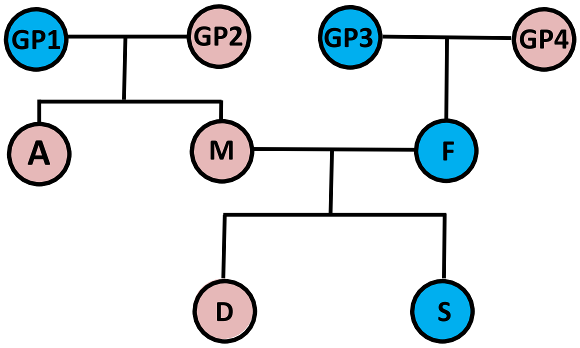


Figure 3.4: Family tree of Manuel Corpas consisting of the 9 family members that were considered. The blue nodes (i.e., darker ones) represent the male and the pink ones (i.e., lighter ones) represent the female family members. Genomic data for the grandparents (GP1, GP2, GP3 and GP4) is missing in the original dataset.

3.4.2 Evaluation Metrics

Similar to [19], we evaluated the proposed framework in terms of both attacker's *incorrectness* and *uncertainty*. Incorrectness quantifies the adversary's error in inferring the SNPs of the individuals in the target set. This metric can be expressed as follows:

$$E_j^i = \sum_{x_j^i \in \{0,1,2\}} p(x_j^i|\Psi)||x_j^i - \hat{x}_j^i||. \tag{3.5}$$

where, $\hat{x}_j^i$ is the true value of the inferred SNP, and Ψ includes all the information that is available to the attacker (as in Fig. 3.1). The incorrectness metric quantifies how far the adversary is away from the actual value of a SNP in his inference.

We also evaluated the proposed scheme based on the attacker's uncertainty. For this purpose, we used the following normalized entropy metric from [19]:

$$H_j^i = \frac{-\sum_{x_j^i \in \{0,1,2\}} p(x_j^i|\Psi) \log(x_j^i|\Psi)}{\log(3)}. \tag{3.6}$$

This can be described as the entropy of the adversary for an unobserved SNP. This metric quantifies the confidence of the adversary about his inference. Note that one needs the ground truth data in order to evaluate the incorrectness of the attacker. Here, by using both incorrectness and uncertainty metrics, we show the correlation between two, as in practice, it is not trivial to possess the ground truth data in order to evaluate the incorrectness of the attacker. That is, we show that one can also use the normalized entropy to quantify an individual's genomic privacy (and hence the strength of an inference attack). In fact, a recent work about genomic privacy metrics also reports that both incorrectness and uncertainty (normalized entropy) are suitable metrics to quantify genomic privacy (and hence the inference attack power) [44]. We compute the metrics in equations (3.5) and (3.6) for each SNP and then take the average for all the SNPs in the unknown set $\mathbb{X}_{\mathrm{U}}$.

3.4.3 Results

Due to the nature of kinship and characteristics of genomic data, we cannot avoid having cycles in our factor graph. Although there is no theoretical proof that our solution (and belief propagation algorithm in general) will converge to an optimal result in the presence of cycles, according to several runs of the algorithm on different SNPs, we observed that belief propagation converges with a significantly low error.

3.4.3.1 CEPH/UTAH Pedigree 1463

We conducted experiments for both high order correlation models (Markov chain and HMM). In the first experiment, among the 100 SNPs we considered, we

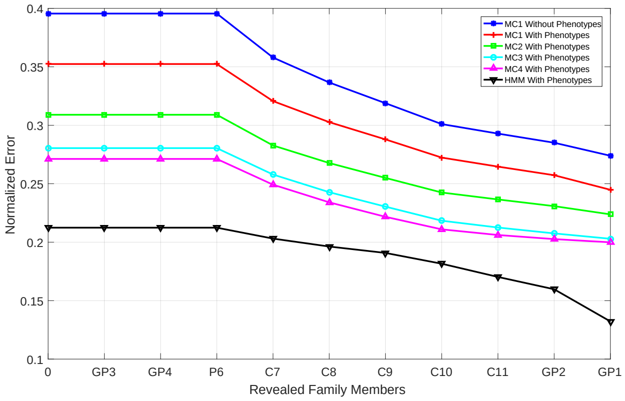


Figure 3.5: Decrease in genomic privacy of P5 (in Fig. 3.3) in terms of the *incorrectness* of the attacker. We reveal partial genomes of other family members for different high order correlation models in the genome. MC stands for the Markov chain model (with different orders) and HMM stands for the hidden Markov model.

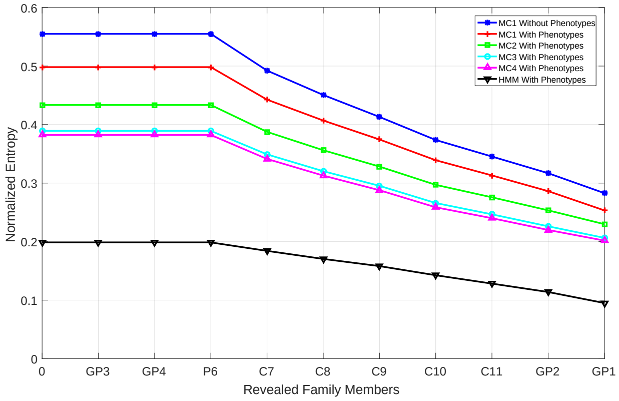


Figure 3.6: Decrease in genomic privacy of P5 (in Fig. 3.3) in terms of the *uncertainty* of the attacker. We reveal partial genomes of other family members for different high order correlation models in the genome. MC stands for the Markov chain model (with different orders) and HMM stands for the hidden Markov model.

randomly hide 50 SNPs belonging to P5 in the CEPH/UTAH family (in Fig. 3.3) and tried to infer them by gradually increasing the background information of the attacker. We also assumed that the attacker knows the following 3 phenotypes of each family member (that are associated with the considered SNPs) [41].

- Verbal declarative memory - associated to *Rs5747035*
- Neurofibromatosis - associated to *Rs121434260*
- Crohn's disease - associated to *Rs4820425*

Because the information about these phenotypes in family members are not publicly available, we probabilistically simulated these phenotypes for the family members (using real probabilities obtained from [41]) and used these simulated phenotypes for the inference. Thus, the contribution of the phenotype information to the inference attack will remain the same if we use the real phenotype information about the individuals as well.

We started revealing 50 random SNPs (out of 100) of other family members (starting from the most distant one to the P5 in terms of number of hops in Fig. 3.3) and observe how the inference power of the attacker changes. We run each experiment 50 times and take the average of each privacy metric. We modelled the high order correlations via both the Markov chain model (for different orders - k) and HMM. We show our results for the attacker's incorrectness and uncertainty in Figs. 3.5 and 3.6, respectively. Note that the case when $k = 1$ (with no phenotype information) represents the previous work by Humbert et al. We observed that both the incorrectness and uncertainty of the attacker decreases by revealing more data. More importantly, our results show that high order correlations and phenotype information contributes significantly to the inference power of the attacker. In both figures, we see that for the Markov chain model, attacker's inference does not improve much for orders of Markov chain (k) that is larger than 3. We further discuss the relation between the amount of unobserved (hidden) SNPs and this bottleneck (about the order of the Markov chain) in Appendix A.1. We also observed that the HMM increases the attacker's inference power compared to the Markov chain model. In all experiments, the accuracy of the HMM is better than the Markov chain's accuracy, which is also consistent with the previous work [45].

Next, to observe the effect of number of hidden SNPs to the high order correlation model, we run the same experiment for the Markov chain model and HMM by hiding different number of SNPs from the victim (P5) and the other family members. This time, we started revealing varying number of random SNPs (out of 100) of other family members (starting from the most distant one to the P5 as before) and observe the inference power of the attacker. In Figs. 3.7 and 3.8, we show our results for the Markov chain model when the order of the Markov chain (k) is 3. We observed that the inference power of the Markov chain model increases as more SNPs of the family members are observed. We obtained similar results for the HMM model (as before, we observed that HMM gives better accuracy compared to Markov chain for varying number of hidden SNPs). In order to show the standard deviations of the experiments, we also show the results with error bars in Appendix A.

3.4.3.2 Manuel Corpas Family Pedigree

We also evaluated our proposed attack on the Manuel Corpas Family Pedigree dataset. Here, we set our target as the mother (M in Fig. 3.4) and try to infer her unobserved SNPs. Unlike the previous experiment, here, we started revealing from the closest family members to the farthest member to show that the strength of the proposed inference attack is independent of the dataset and evaluation methodology. Similar to the previous experiment, we assumed that the attacker knows the same set of three phenotypes about each member of this family and we revealed 50 random SNPs (out of 100) of other family members. We run each experiment 50 times and take the average of each privacy metric.

The results for this experiment (in terms of normalized error and normalized entropy) are given in Figs. 3.9 and 3.10. Obtained results are consistent with

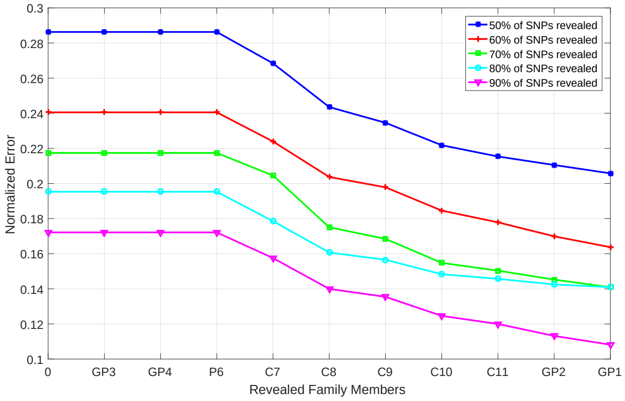


Figure 3.7: Decrease in genomic privacy of P5 (in Fig. 3.3) in terms of the *incorrectness* of the attacker. We reveal different number of random SNPs from other family members and use the Markov chain model (with $k = 3$) to model the high order correlation in the genome.

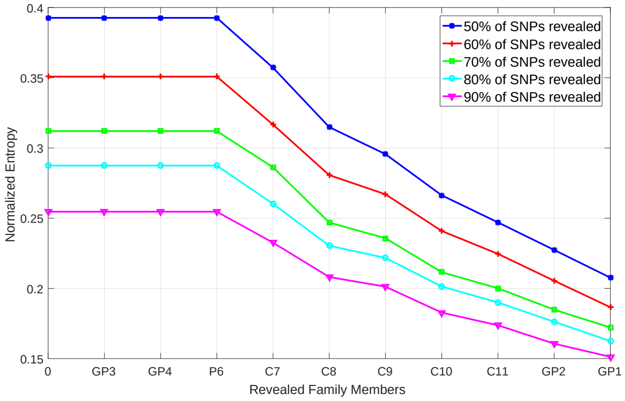


Figure 3.8: Decrease in genomic privacy of P5 (in Fig. 3.3) in terms of the *uncertainty* of the attacker. We reveal different number of random SNPs from other family members and use the Markov chain model (with $k = 3$) to model the high order correlation in the genome.

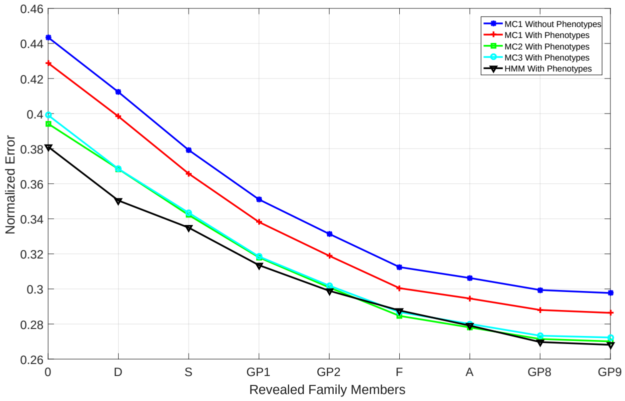


Figure 3.9: Decrease in genomic privacy of M (in Fig. 3.4) in terms of the *incorrectness* of the attacker. We reveal partial genomes of other family members for different high order correlation models in the genome. MC stands for the Markov chain model (with different orders) and HMM stands for the hidden Markov model.

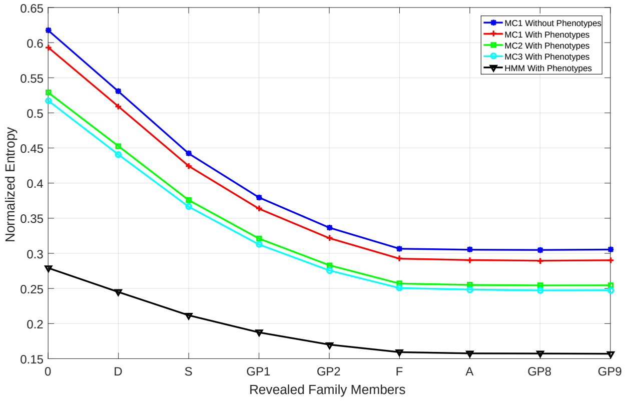


Figure 3.10: Decrease in genomic privacy of M (in Fig. 3.4) in terms of the *uncertainty* of the attacker. We reveal partial genomes of other family members for different high order correlation models in the genome. MC stands for the Markov chain model (with different orders) and HMM stands for the hidden Markov model.

our expectations (error and entropy decrease with each revealed family member). Similar to the previous results, it can be seen that high order correlation and phenotype information contributes significantly to inference power of the attacker. In general, we observed that the results are consistent with CEPH/UTAH pedigree experiments. However, since we changed the order of revealing family members, unlike the previous results, here we observed a continuous decrease in error and entropy for the genomic privacy of the victim. This is because each family member has a direct effect on our inference power.

Chapter 4

Defend The Genomic Privacy

In this chapter, first we explain our genome sharing privacy preserving methodology which is based on differential privacy, then we evaluate it against attack which may target individuals or their families. Finally, we compare our methodology with other method and discuss it robustness based on results.

4.1 Proposed Privacy-Preserving Framework

In this Chapter, we elaborate our proposed framework including our assumptions, notations we used, and the system model. First, we describe the general settings, assumptions, and the attacker model. Then, we provide a mathematical formulation of our solution and explain the general data sharing framework. Finally, we discuss some common scenarios via a toy example.

4.1.1 Assumptions and Notations

In this part, we explain our settings and notations. We have a set of family members denoted as $\mathbf{F}$. We represent the set of SNP IDs of an individual i

($i \in \mathbf{F}$) as $\mathbf{I^i}$. We represent the value of a SNP as the number of minor alleles it carries and we denote the value of a SNP j for individual i as x_j^i ($j \in \mathbf{I^i}$). Thus, x_j^i takes values from set $\{0, 1, 2\}$. Also, we denote a SNP j as x_j for general representation (regardless of its value in a specific individual). We denote the set of sensitive SNPs for individual i as $\mathbf{S^i}$. The SNPs in the sensitive set are never shared by the corresponding individual. However, as will be discussed later, information about these SNPs can be leaked either by sharing other SNPs that are not in the sensitive set or SNPs shared by other family members. Also, each family member may have his own sensitive SNP set.

During the SNP sharing (i.e., data sharing) procedure, by using our proposed mechanism, an individual decides to hide (or share) each of his SNPs. We denote the set of hidden SNPs of individual i as $\mathbf{H^i}$ and his set of shared SNPs as $\mathbf{R^i}$[1]. At the beginning of the sharing procedure (discussed in Section 4.1.4), all of the SNPs of i are hidden (i.e., $\mathbf{H^i} = \mathbf{I^i}$ and $\mathbf{R^i} = \phi$). Then, based on the result of the proposed mechanism on each SNP, we decide whether or not to add that SNP to the set of shared SNPs ($\mathbf{R^i}$). We list the frequently used notations in Table 4.1.

Table 4.1: Frequently used notations.

Definition	Notation
Set of family members	$\mathbf{F}$
Set of SNPs of individual i	$\mathbf{I^i}$
Value of SNP j of individual i	x_j^i
Set of sensitive SNPs of individual i	$\mathbf{S^i}$
Set of hidden SNPs of individual i	$\mathbf{H^i}$
Set of shared SNP of individual i	$\mathbf{R^i}$

4.1.2 Attacker Model

We assume that the attacker has background knowledge about public statistics about genomics and the relationship between the family members in $\mathbf{F}$. That is, the attacker has access to public resources including SNP data belonging to

[1] SNPs in the sensitive set of individual i are always hidden, and hence $\mathbf{S^i} \subseteq \mathbf{H^i}$.

different populations [23, 17]. Using such resources, the attacker can calculate the minor allele frequency (MAF) for each SNP (frequency at which the minor allele is observed in a given population). Using similar resources, the attacker can also compute high-order correlations between the SNPs and use this information for the inference of the SNPs in the sensitive sets of individuals [24]. For this, the attacker exploits the method introduced in [24] as follows:

$$P_k(x_j) = \begin{cases} 0 & \text{if } F(x_{i-k,i-1}) = 0 \\ \frac{F(x_{i-k,i})}{F(x_{i-k,i-1})} & \text{if } F(x_{i-k,i-1}) > 0. \end{cases} \tag{4.1}$$

Here, $P_k(x_j)$ is the probability distribution of SNP j computed using a Markov chain of order k. Also, $F(x_{i,j})$ is the frequency of the subsequence $x_{i,j}$ that includes all the SNPs between x_i and x_j. Furthermore, the attacker knows that the shared SNPs of an individual may threaten the kin genomic privacy of his family members. To utilize this information, the attacker mainly uses the Mendel's law.

To run its inference attack on the SNPs in the sensitive sets of individuals, the attacker uses the combination of all the aforementioned information as shown in [11] and chapter 3 by using a message passing algorithm on a graphical model [25, 26] (as introduced in Chapter 2.3). In attacker's favor, we assume that the correlation model (i.e., Markov chain order) used by the attacker is the same as the one we use during the proposed SNP sharing mechanism. Also, as we will show later via experiments, increasing the order of the correlation model beyond a value does not increase the inference power, and hence we do not assume that the attacker uses a higher-order correlation model than the SNP sharing mechanism. In our evaluations, we first focus on an individual's genomic privacy, then we also consider the kinship relationships between family members.

4.1.3 Mathematical Formulation

Miguel *et al.* have utilized the differential privacy notion to establish a method to protect a user's location privacy while sharing data with location-based service providers [16]. The authors have shown that a mechanism is differentially private if revealing any piece of information via the mechanism keeps the probability of a user's location within an exponential boundary as in (2.3). In this work, inspired by [16] and the generalized version of differential privacy [21], we introduce a novel formulation of differential privacy for sharing genomic data.

Similar to (2.2), we assume each SNP j takes a scalar value (from $\mathfrak{R}^1$) such that $x_j \in \{0, 1, 2\}$, and hence the L_1 norm between any two SNP values (i.e., l) is bounded by 2 (i.e., $|x_j - x_i| \leq 2$ for any two SNPs i and j). Unlike other differentially private data sharing mechanisms, which add noise to shared data points in order to protect privacy, we introduce a sharing mechanism for the hidden SNPs. That is, rather than sharing noisy SNP values, we prefer the proposed mechanism to decide whether or not to share each SNP (this is also the preferred methodology for medical data in general).

We assume the attacker's auxiliary information is denoted as $\mathbf{A}$. This information includes (i) minor allele frequency (MAF) values of the SNPs, (ii) high-order correlations between the SNPs, and (iii) the relationship between the family members in $\mathbf{F}$. Our proposed mechanism decides whether or not to share each hidden SNP of a donor i. For each SNP j in $\mathbf{H^i}$, the mechanism calculates the probability distribution of SNPs in $\mathbf{S^m}$ ($\forall m \in \mathbf{F}$) assuming x^i_j is shared and also using all the previously shared SNPs of both the donor and the other family members in $\mathbf{F}$. For this, we use the inference attack introduced in Chapter 2.3.

Let set $\mathbf{R}$ include all the SNPs that have been shared by the individuals in $\mathbf{F}$ so far. That is, $\mathbf{R} = \bigcup_{i \in \mathbf{F}} \mathbf{R^i}$. For individual i to share a new SNP j, we require that for all sensitive SNPs of all family members, the ratio between probabilities of different states should not be greater than a boundary as follows:

$$\begin{gathered}\frac{P(x^m_k | \mathbf{R} \cup x^i_j, \mathbf{A})}{P(x^{m\prime}_k | \mathbf{R} \cup x^i_j, \mathbf{A})} \leq e^{l \epsilon_m} \frac{P(x^m_k | \mathbf{A})}{P(x^{m\prime}_k | \mathbf{A})} \\ \forall m \in \mathbf{F}, \ \forall k \in \mathbf{S^m}, \ x^m_k, x^{m\prime}_k \in \{0, 1, 2\} \ x^m_k \neq x^{m\prime}_k.\end{gathered} \tag{4.2}$$

It is important to note that the above condition, and hence the sharing (or hiding) decision on a particular SNP is independent of the actual values of the sensitive SNPs of the donor and the other family members. We will further discuss the importance of this property in later sections.

Based on this formulation for SNP sharing, we have the following theorem:

Theorem 1. *Sharing SNPs of individual by following* (4.2) *implies generalized formulation of differential privacy that is shown in* (2.2)*.*

Proof Sketch. Similar to (2.2), we consider our data points (x_j) as scalars representing the SNP values. The measure $\mu_x(S)$ (in (2.2)) for general expression is equivalent to probability of observing a shared SNP sequence S given the auxiliary information of attacker about sensitive data points (x_j). Thus, by using Bayes' formula on (2.2), we can show that (2.2) implies posterior probability distributions should not differ from the priors by more than a boundary, and this is what we require in (4.2). We provide a more elaborate proof about the equivalency of our formulation with generalized differential privacy in Appendix B.1. Next, we discuss how we use this mathematical formulation for sharing genomic data between a donor and a service provider.

4.1.4 SNP Sharing Mechanism

The overview of the proposed mechanism is shown in Fig. 4.1. Let individual i be the donor that wants to share his SNPs with a service provider. At the beginning of the process, all of the SNPs of the individual i are hidden, (i.e., $\mathbf{R^i} = \phi$ and $\mathbf{H^i} = \mathbf{I^i}$). We first assign the set of sensitive SNPs ($\mathbf{S^i}$) and a privacy parameter (i.e., ϵ_i in (4.2)) for individual i. As discussed, these two parameters can be different for all individuals in $\mathbf{F}$. Then, we pick a SNP j from $\mathbf{H^i}$ and calculate its disclosure effect on the probability distribution of each SNP in the sensitive SNP set of i ($\mathbf{S^i}$) and the sensitive SNP sets of all the other family members in $\mathbf{F}$. If (4.2) holds true for all SNPs in $\mathbf{S^m}$ ($\forall m \in \mathbf{F}$), then we share the corresponding hidden SNP and add it to set $\mathbf{R^i}$, otherwise x^i_j remains in $\mathbf{H^i}$. The details of our proposed mechanism are also shown in Algorithm 1.

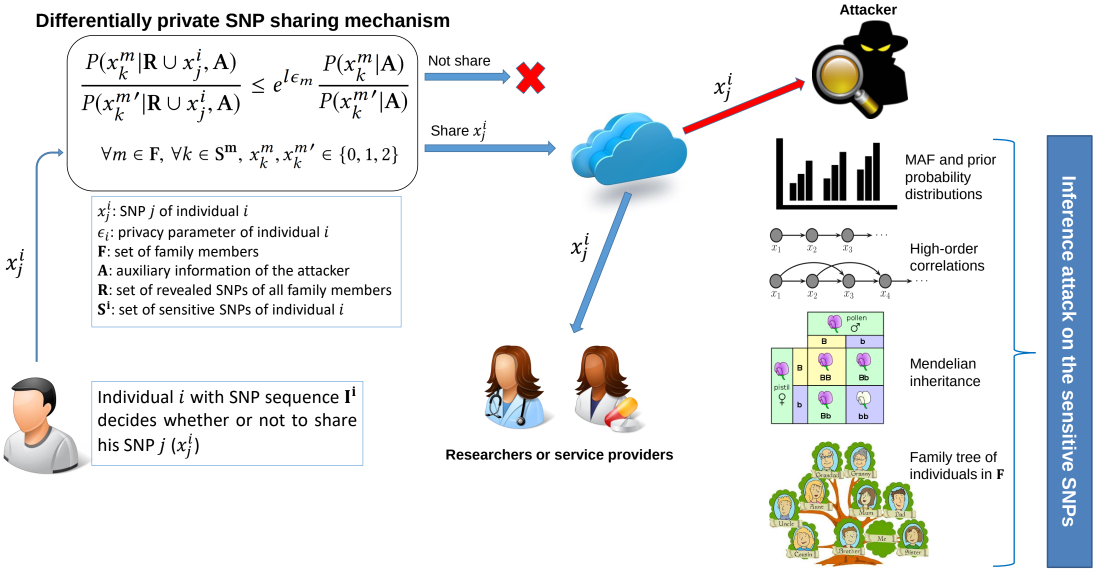

Differentially private SNP sharing mechanism
$\frac{P(x_k^m|\mathbf{R}\cup x_j^i,\mathbf{A})}{P(x_k^{m\prime}|\mathbf{R}\cup x_j^i,\mathbf{A})} \leq e^{l\epsilon_m}\frac{P(x_k^m|\mathbf{A})}{P(x_k^{m\prime}|\mathbf{A})}$
$\forall m\in\mathbf{F},\ \forall k\in\mathbf{S^m},\ x_k^m, x_k^{m\prime}\in\{0,1,2\}$
x_j^i: SNP j of individual i
ϵ_i: privacy parameter of individual i
F: set of family members
A: auxiliary information of the attacker
R: set of revealed SNPs of all family members
$\mathbf{S^i}$: set of sensitive SNPs of individual i
x_j^i
Individual i with SNP sequence $\mathbf{I^i}$ decides whether or not to share his SNP j (x_j^i)
Not share
Share x_j^i
x_j^i
x_j^i
Attacker
Researchers or service providers
MAF and prior probability distributions
x_1 x_2 x_3
High-order correlations
x_1 x_2 x_3 x_4
pollen ♂
B
b
B
b
pistil ♀
BB
Bb
Bb
bb
Mendelian inheritance
Grandad
Granny
Aunt
Mum
Dad
Uncle
Cousin
Brother
Me
Sister
Family tree of individuals in **F**
Inference attack on the sensitive SNPs

Figure 4.1: The donor (individual i) wants to share his SNP sequence with a service provider. We illustrate one instance of this sharing for SNP x_j^i.

ALGORITHM 1: SNP Sharing Mechanism

```
input  : Attacker's auxiliary information A: {MAF values of the SNPs,
         high-order correlations, the relationship between members in F},
         index of donor $i$, sensitive SNP sets $\mathbf{S^m}$ ($m \in \mathbf{F}$), shared SNP sets
         $\mathbf{R^m}$ ($m \in \mathbf{F}$), $\mathbf{R} = \bigcup_{i\in\mathbf{F}} \mathbf{R^i}$, privacy parameter $\epsilon_m$ of each family
         member $m$.
output: Set of shared SNPs of donor, $\mathbf{R^i}$
$\mathbf{R^i} \longleftarrow \phi$;
$\mathbf{H^i} \longleftarrow \mathbf{I^i}$;
forall the SNP $j$ in $\mathbf{H^i} \backslash \mathbf{S^i}$ do
    Run inference attack on $\mathbf{S^m}$, $\forall m \in \mathbf{F}$;
    Calculate $P(x_j^i|\mathbf{R} \cup x_j^i, \mathbf{A})$;
    $flag \longleftarrow 1$;
    forall the $m$ in F do
        forall the SNP $k$ in $\mathbf{S^m}$ do
            if (4.2) is violated then
                $flag \longleftarrow 0$;
            end
        end
    end
    if $flag = 1$ then
        $\mathbf{R^i} \longleftarrow \mathbf{R^i} \cup x_j^i$
    end
end
```

4.1.5 Toy Example

Here, we provide a toy example about the proposed SNP sharing mechanism to discuss some common scenarios that might happen. Notably, we show that sharing decision for a particular SNP is independent of the actual values of donor's (and family members') sensitive SNPs. Therefore, the reason for not sharing a SNP is not necessarily due to the decrease in the estimation error of the attacker when that SNP is shared, and hence the attacker cannot infer the actual values of the sensitive SNPs using the decisions of the donor.

Assume we have the population as shown in Table 4.2 that consists of 6 individuals $(i_1, \ldots, i_6)$. For simplicity, in this example, we do not consider the kinship information between the individuals. The SNP set has three SNPs, $\mathbf{I^i} = \{x^i_1, x^i_2, x^i_3\}$, and the sensitive set is $\mathbf{S^i} = \{x^i_3\}$ for all individuals.

	i_1	i_2	i_3	i_4	i_5	i_6
x_1	0	1	2	1	0	2
x_2	0	0	0	0	1	1
x_3	0	1	0	0	1	2

Table 4.2: The example population including 3 SNPs of 6 individuals. Each column shows the corresponding individual's SNP values.

The attacker's auxiliary information consists of MAF values of the SNPs and the correlation model between the SNPs. In Table 4.3, we show the prior probability distribution of each SNP x_j ($j \in \{1, 2, 3\}$) computed using its MAF value. We assume that the attacker uses the first order Markov chain to calculate the correlation model between the SNPs. That is, the attacker computes $P_1(x_j) = P(x_j|x_{j-1})$ (($j \in \{1, 2, 3\}$)), by using (4.1). We also show these correlation values (computed using the SNP sequences in Table 4.2) in Table 4.4.

	$P(x_1)$	$P(x_2)$	$P(x_3)$
Homozygous major (0)	0.33	0.67	0.50
Heterozygous (1)	0.33	0.33	0.33
Homozygous minor (2)	0.33	0.00	0.17

Table 4.3: Prior probability distributions of SNPs (in Table 4.2) computed using their MAF values.

	P(x_1)	P($x_2 \mid x_1$)	P($x_3 \mid x_2$)
$x_j = 0$, $x_{j-1} = 0$	0.33	0.50	0.75
$x_j = 0$, $x_{j-1} = 1$	0.33	1.00	0.00
$x_j = 0$, $x_{j-1} = 2$	0.33	0.50	0.00
$x_j = 1$, $x_{j-1} = 0$	0.33	0.50	0.25
$x_j = 1$, $x_{j-1} = 1$	0.33	0.00	0.50
$x_j = 1$, $x_{j-1} = 2$	0.33	0.50	0.00
$x_j = 2$, $x_{j-1} = 0$	0.33	0.00	0.00
$x_j = 2$, $x_{j-1} = 1$	0.33	0.00	0.50
$x_j = 2$, $x_{j-1} = 2$	0.33	0.00	0.00

Table 4.4: Correlation model between the SNPs for the first order Markov chain. The first column shows the different states of sequential SNPs and the remaining columns show the probabilities. In this example, we do not consider the SNPs before x_1, and hence in the correlation model, all states of x_1 are equally likely.

We set the privacy parameter $\epsilon = 0.3$ for all individuals. We consider the sharing of x_1^i for different individuals in the example population. For that, we compute how this disclosure changes the probability distribution of the sensitive SNP x_3^i and observe how this change may violate (4.2). We may observe three different cases (or a combination of them) for the left part of (4.2):

(a) Sharing x_1^i may change the ratio between states zero and one of x_3^i.

(b) Sharing x_1^i may change the ratio between states one and two of x_3^i.

(c) Sharing x_1^i may change the ratio between states two and zero of x_3^i.

For cases (a) and (b), consider individual 4 (i_4) as the donor. We show the prior probability distributions for x_3^4 in Table 4.3. We can compute the effect of sharing x_1^4 on the posterior probability distribution of x_3^4 as follows:

$$P(x_3^4|\mathbf{R} \cup x_1^4, \mathbf{A}) \propto \sum_{x_2} P(x_3^4|x_2^4)P(x_1^4 = 1), \quad x_3^4 \in \{0, 1, 2\}.$$

Thus, we compute the posterior distribution as $P(x_3^4 = 0|\mathbf{R} \cup x_1^4, \mathbf{A}) = 1/4$, $P(x_3^4 = 1|\mathbf{R} \cup x_1^4, \mathbf{A}) = 1/12$, and $P(x_3^4 = 2|\mathbf{R} \cup x_1^4, \mathbf{A}) = 0$. Setting $x_3^4 = 0$ and $x_3^{4\prime} = 1$, we observe that case (a) violates the condition in (4.2). Similarly, setting $x_3^4 = 1$ and $x_3^{4\prime} = 2$ (case (b)) also violates (4.2). As a result, in this scenario, we decide not to share x_1^4. As shown, regardless of the real value of x_3^4, condition in (4.2) may be violated due to several different cases. Therefore, not sharing x_1^4 would not provide additional information about the actual value of x_3^4 to the attacker.

To illustrate case (c), we can pick individual 3 as the donor and repeat similar computations. In this case, setting $x_3^3 = 2$ and $x_3^{3\prime} = 0$ violates (4.2). Therefore, even though sharing x_1^3 increases the estimation error of the attacker for the value of the sensitive SNP x_3^3, due to the violation of (4.2), our mechanism does not share x_1^3 and the attacker cannot infer further information about the value of x_3^3 due to this decision.

From this example, we come up with the following conclusions:

- Since we do not consider the real values of the SNPs in the sensitive set while computing (4.2), each of the aforementioned cases (or even combination of them) may occur, and hence the attacker cannot know which violation is the reason for not sharing a particular SNP. In Section 4.2.6, we show that not considering the real value of the sensitive SNPs increases the utility of shared data.

- Sharing a SNP may decrease or increase the estimation error of the attacker for the sensitive SNPs. Similarly, the decision may also either decrease or increase the entropy of the sensitive SNPs . Thus, the attacker cannot infer the real values of the sensitive SNPs from the decision our mechanism gives about sharing SNPs. We will discuss this further in Chapter 4.3.2.

4.2 Evaluation

In this section, we evaluate our proposed mechanism using a real-life dataset and study the effects of various parameters to both privacy and utility. We also compare the proposed mechanism with a similar work by Humbert *et al.* that has a similar goal as ours [1].

We use a dataset that consists of 1000 SNPs belonging to 99 people from Central European ethnicity [18]. Using this dataset, first, we compute the auxiliary information of the attacker. Thus, we generate the correlation model (Markov chain) on the SNPs of the individuals in the population using (4.1) and also compute the prior probability distributions of the SNPs using their MAF values. We define the utility of the shared data as the number (or fraction) of SNPs that are shared as a result of our proposed algorithm. We study the following parameters that have effect on the privacy and utility (i.e., amount of shared SNPs).

- **Order of Markov chain.** We study the correlation model between the SNPs (i.e., the order of Markov chain) on the inference power of the attacker and on the utility.
- **Privacy parameter.** We study the effect of ϵ parameter in (4.2) on both privacy and utility.
- **Size of sensitive SNP set.** We study the relationship between the fraction of sensitive SNPs to the whole SNPs and the utility.
- **Attacker's error and entropy.** We study the relationship between the success of attacker's inference attack and utility.
- **Entropy of the SNPs in the sensitive SNP set.** We compare the low entropy and high entropy sensitive SNPs in terms of utility.
- **Kinship relationships.** We study the effect of kinship inference attack which is discussed in [11] and Chapter 3 on the utility.

4.2.1 Order of Markov Chain and Privacy Parameter

Here, we choose 50 random SNPs as the sensitive ones (out of 1000 SNPs in the dataset), we repeat the experiment for 10 random individuals using the proposed SNP sharing mechanism, and report the average results. In Fig. 4.2, we show the relation between the privacy parameter (ϵ) and the utility for different orders of the Markov chain model (for the correlation between the SNPs). We observe that as expected, with increasing ϵ value, the average utility also increases. Also, with increasing Markov chain order, the utility decreases. In other words, higher-order correlation models improve the inference power of the attacker. We also observe that (i) for $\epsilon > 0.7$, the results for correlation models with Markov chain orders 3 and 4 overlap, and (ii) for correlation models of order higher than 4, the improvement in the inference power of the attacker is negligible (the results in Chapter 3 also support this finding).

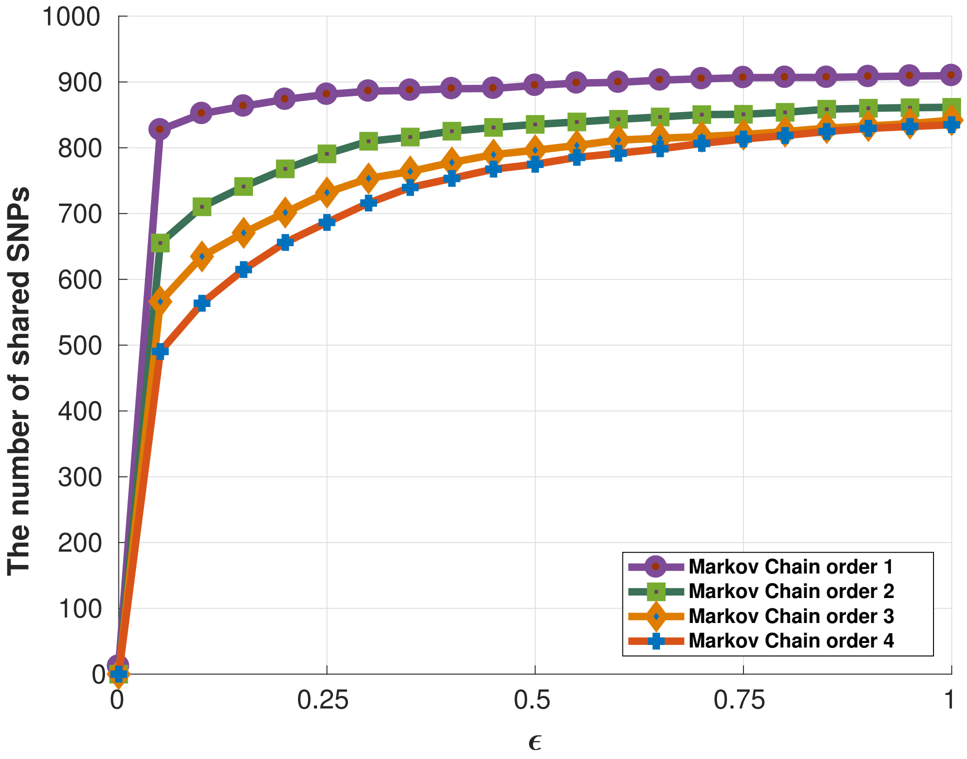
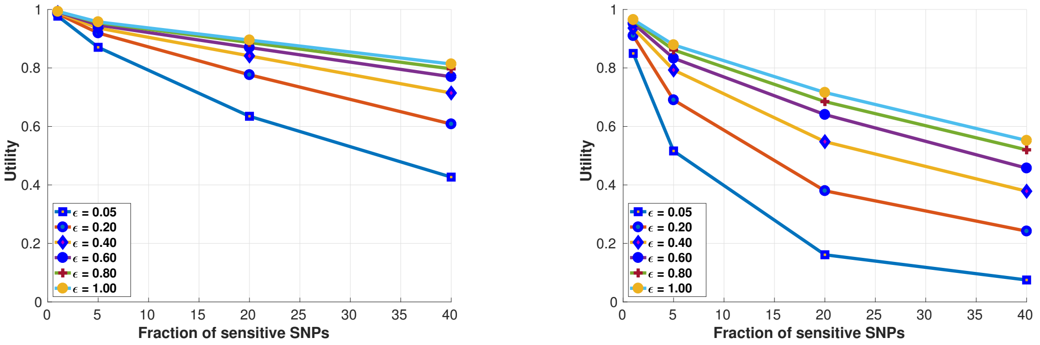


Figure 4.2: Relationship between utility (number of shared SNPs), privacy parameter (ϵ), and the correlation model (i.e., order of the Markov chain). Here, the donor has 1000 SNPs in total and the size of the sensitive SNP set is set to 50.

4.2.2 Size of the Sensitive SNP Set

Here, we study the effect of fraction of sensitive SNPs (to the whole SNPs) on the utility. For this study, we represent the utility as the fraction of the shared SNPs in the non-sensitive SNP set. Thus, the utility for the SNPs shared by individual i is defined as $|\mathbf{R^i}|/(|\mathbf{I^i}| - |\mathbf{S^i}|)$. As before, we repeat each experiment for 10 random individuals and report the average.

In Fig. 4.3, we show the effect of the sensitive SNP set size on the utility for different correlation models (for Markov chains of order 1 and 4) and for different privacy parameters. The x-axis shows the fraction of the sensitive SNPs to the whole SNPs in $\mathbf{I^i}$ (i.e., $|\mathbf{S^i}|/|\mathbf{I^i}|$) varying between 1% and 40%. Although



(a) Correlation model: Markov chain of order 1

(b) Correlation model: Markov chain of order 4

Figure 4.3: Relationship between utility (fraction of shared non-sensitive SNPs), fraction of sensitive SNPs, privacy parameter (ϵ), and the correlation model (i.e., order of the Markov chain). Utility is defined as $|\mathbf{R^i}|/(|\mathbf{I^i}| - |\mathbf{S^i}|)$ and the fraction of sensitive SNPs is defined as $|\mathbf{S^i}|/|\mathbf{I^i}|$.

the utility is defined as the fraction of shared SNPs from the non-sensitive SNP set, by increasing the fraction of the sensitive SNPs, we observe a decrease in the utility. This is because as the size of sensitive SNP set increases, more SNPs in the non-sensitive set becomes correlated with the sensitive SNPs. Also, the decrease in utility is higher for higher-order correlation models (which is consistent with the results in Fig. 4.2). We also observe that the improvement in utility gets smaller and the utility converges to a common value as the ϵ value gets closer to 1.

4.2.3 Estimation Error and Entropy

In order to evaluate our proposed SNP sharing mechanism in terms of attacker's success for inferring the sensitive SNPs, we use two metrics that has been previously proposed by Humbert *et al.* [11]: (i) the average distance of the attacker from true value of the sensitive SNPs (i.e., estimation error or incorrectness), and (ii) the entropy (or uncertainty) of the attacker based on inferred probability distributions of the sensitive SNPs. For attacker's incorrectness, we use the following metric:

$$E_i = \sum_{x_j^i} P(x_j^i)||x_j^i - \hat{x_j^i}||, \quad j \in \mathbf{S^i}, \quad x_j^i \in \{0, 1, 2\}, \tag{4.3}$$

where E_i is attacker's error for individual i's sensitive SNPs. Also, $P(x_j^i)$ is the probability distribution of SNP j of individual i that is inferred by the attacker as a result of the inference attack (as introduced in Chapter 2.3) and $\hat{x_j^i}$ is the true value of SNP j of individual i. For attacker's uncertainty, we use the following metric:

$$H_i = -\sum_{x_j^i} P(x_j^i) \log(P(x_j^i)), \quad j \in \mathbf{S^i}, \quad x_j^i \in \{0, 1, 2\}, \tag{4.4}$$

where H_i is attacker's uncertainty (entropy) for individual i's sensitive SNPs.

We study the effect of fraction of sensitive SNPs to the whole SNPs (i.e., $|\mathbf{S^i}|/|\mathbf{I^i}|$) on the error and entropy (as before, we run 10 experiments and report the average). In Table 4.5, we show how attacker's (average) estimation error changes with different fractions of sensitive SNPs for $\epsilon = 0.5$ (we did not observe much change for different ϵ values between 0.05 and 1). We observe that attacker's estimation error increases with increasing fractions of sensitive SNPs. The error increases fast for small fractions of sensitive SNPs and then it saturates for larger fractions. Also, the error does not change much for different correlation models. In fact, the error for Markov chain order 4 is sometimes larger than the one for order 1. This is because we share less SNPs for order 4 (as shown in Fig. 4.2), and hence higher order correlation model generates noisy inference results. Also, in Table 4.6, we show how attacker's (average) uncertainty changes with different fractions of sensitive SNPs for $\epsilon = 0.5$. As in error, we observe that attacker's uncertainty (entropy of the sensitive SNPs) increases with increasing fractions of sensitive SNPs. Note however that, as shown in Fig. 4.3, utility of the SNP sharing mechanism is different for different fractions of sensitive SNPs and correlation models.

	Fraction of sensitive SNPs			
	1%	5%	20%	40%
Markov chain order 1	0.9250	1.0428	1.0684	1.0781
Markov chain order 4	0.9540	1.0517	1.0694	1.0801

Table 4.5: Relationship between attacker's average estimation error and fraction of sensitive SNPs ($|\mathbf{S^i}|/|\mathbf{I^i}|$) for different correlation models. Privacy parameter (ϵ) is set to 0.5.

	Fraction of sensitive SNPs			
	1%	5%	20%	40%
Markov chain order 1	0.4387	0.4942	0.5111	0.5229
Markov chain order 4	0.4535	0.5027	0.5088	0.5218

Table 4.6: Relationship between attacker's average uncertainty and fraction of sensitive SNPs ($|\mathbf{S^i}|/|\mathbf{I^i}|$) for different correlation models. Privacy parameter (ϵ) is set to 0.5.

4.2.4 Entropy of the SNPs in the Sensitive SNP Set

Here, we study the relationship between the types of SNPs in the sensitive SNP set and the utility of the SNP sharing mechanism. For this, we categorize the SNPs as (i) high entropy SNPs, and (ii) low entropy SNPs. Entropy of a SNP is computed using its prior probability distribution that is computed using its (publicly known) MAF value (which varies between 0 and 0.5). Therefore, a SNP with low MAF value (close to 0) has low entropy (i.e., probability distribution of its states show high differences), whereas a SNP with high MAF value (close to 0.5) has high entropy (i.e., probability distribution of its states is almost uniform).

We construct the low-entropy sensitive SNP set by randomly selecting 50 SNPs whose entropy is less than 0.5 and we do the opposite for the high-entropy SNP set (we repeat the same experiment for 10 random individuals in the population). We show the results in Fig. 4.4 for different correlation models and different values of the privacy parameter (ϵ). We observe that in general, the proposed SNP sharing mechanism provides significantly more utility (i.e., number of shared SNPs are higher) when the sensitive SNP set includes low entropy SNPs. This is because the attacker already has a good knowledge (through the public MAF values) about the values of the low entropy SNPs in the sensitive SNP set. Thus, sharing other SNPs typically does not significantly improve this knowledge. On the other hand, attacker's knowledge about the values of high entropy SNPs (in the sensitive SNP set) is more likely to significantly increase with the sharing of other SNPs. Therefore, for a fixed ϵ value, we have higher utility when we include low entropy SNPs in the sensitive set. Furthermore, low entropy SNPs are expected to be the rare ones and rare SNPs typically have low correlations with the other ones. Thus, SNPs in the sensitive SNP set have low correlations with the non-sensitive ones, and hence sharing non-sensitive SNPs does not have much effect on the privacy of the SNPs in the sensitive set. On the other hand, high entropy SNPs typically have higher correlations with the other SNPs.

We also compute the attacker's estimation error for low and high entropy sensitive SNP sets. The average estimation error of the attacker is around 0.8 for the high-entropy sensitive SNP set and it is around 0.01 for low-entropy sensitive SNP set.[2] This is expected as the attacker already has a significant background knowledge (through MAF values) about the values of the low entropy SNPs.

4.2.5 Kinship Relationships

In this part, we evaluate our proposed SNP sharing mechanism by also considering the kin genomic privacy of individuals (as formulated in Chapter 4.1.3). Along with kinship relationships and Mendel's law, we also use higher-order correlations on the DNA as in Chapter 3. We use the inference attack introduced in Chapter 2.3 to compute the posterior probabilities in (4.2).

For the evaluation, we use a trio (father, mother, and son) from Manuel Corpas family DNA dataset [8]. We choose 100 neighboring SNPs of the considered family members. We set the size of the sensitive SNP set to 20 for all family members and we randomly choose 20 SNPs for each family member to construct their sensitive SNP sets (i.e., sensitive SNP set of each family member is different). We assume that the son is the donor and we use the proposed SNP sharing

[2] These values are almost the same for all correlation models and they slightly decrease with increasing ϵ value.

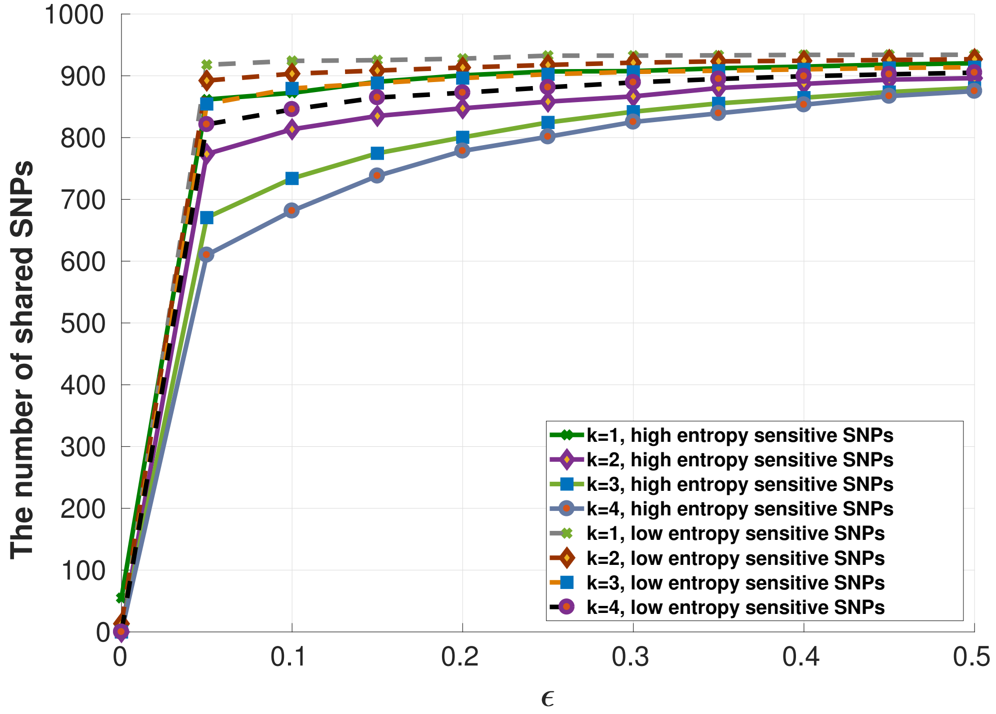


Figure 4.4: Relationship between the types of SNPs in the sensitive SNP set and the utility (number of shared SNPs) for different privacy parameter values (ϵ) and different correlation models. k represents the order of the Markov chain used for the correlation model. The dashed lines illustrate sensitive SNP sets with low entropy SNPs (SNPs whose entropy is less than 0.5) based on their MAF values and the continuous lines illustrate sensitive SNP sets with high entropy SNPs (SNPs whose entropy is equal to or higher than 0.5). In all experiments, the donor has 1000 SNPs in total and the size of the sensitive SNP set is set to 50.

mechanism to share the non-sensitive SNPs of son (by also considering genomic privacy of other family members). We use the same privacy parameter (ϵ) for all family members. We conduct each experiment for 10 times and show the results for utility (number of shared SNPs) for different ϵ parameters and correlation models in Fig. 4.5. In particular, we show that for high-order correlation models, the value of the privacy parameter should be high to have some utility when we also consider kin genomic privacy. Note that in this experiment, we assume 20% of total SNPs as sensitive for each family member and the utility increases as this fraction decreases.

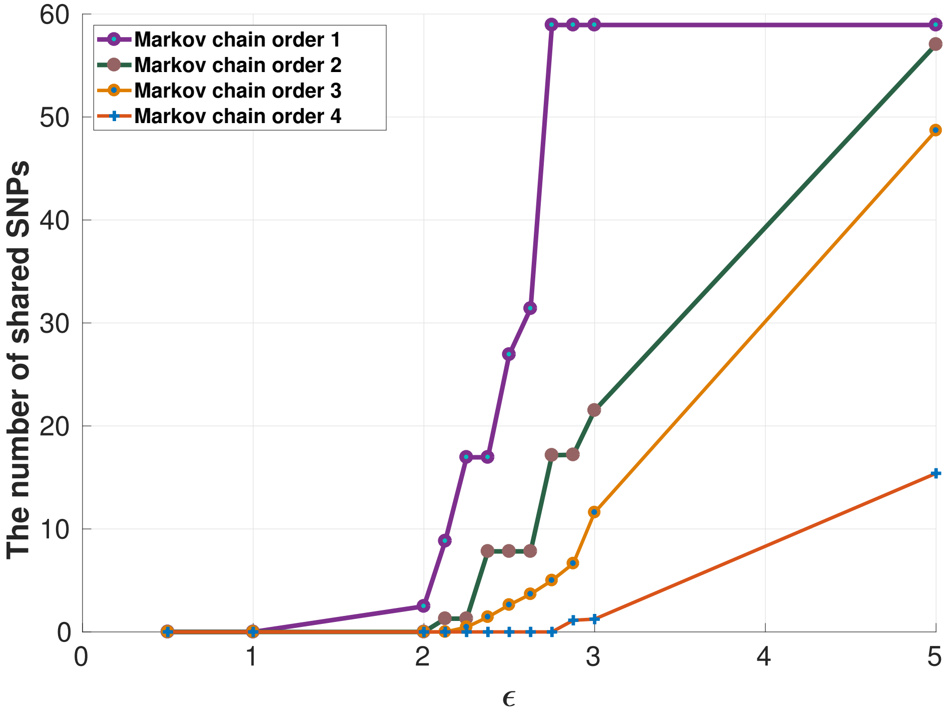


Figure 4.5: Relationship between the privacy parameter (ϵ) and the utility (number of shared SNPs) for different correlation models when we also consider kin genomic privacy. We consider a trio (father, mother, and son) and the donor is the son. Each family member has its own (randomly constructed) sensitive SNP set and the privacy parameter is the same for all family members. In all experiments, the donor has 100 SNPs in total and the size of the sensitive SNP set is set to 20 for all family members.

4.2.6 Comparison With Previous Work

We compare our proposed mechanism with Humbert *et al.*'s work [1] which has a similar goal as ours. Humbert *et al.* propose a SNP sharing mechanism by formulating the problem as an optimization problem, in which the goal is to maximize the utility while considering privacy constraints of the donor and his family members. In the rest of this section, we first briefly introduce the methodology of [1] and then, we compare our proposed mechanism with [1].

4.2.6.1 Methodology of Humbert *et al.* [1]

The set of family members is denoted as $\mathbf{F}$. A family member (referred as the donor) wants to share a set of his SNPs. The goal is to maximize the number of donor's shared SNPs while preserving his and the family members' genomic privacy up to a limit. The sharing decisions on SNPs of individual i are represented with vector $\mathbf{y^i} = \{y^i_j \colon j \in \mathbf{I^i}\}$, where $y^i_j = 1$ means individual i shares SNP j, and $y^i_j = 0$ means SNP j remains hidden. The sensitivity of a SNP j for individual i is denoted as s^i_j. The set of privacy sensitive SNPs of individual i is denoted as $\mathbf{P}^i_s$ and the privacy (loss) tolerance of individual i for the SNPs in $\mathbf{P}^i_s$ is denoted as $Pri(i, \mathbf{P}^i_s)$. The background information of the attacker includes the MAF values of the SNPs, pairwise correlation model between the SNPs (i.e., LD), and rules of Mendelian inheritance.

First (before sharing any SNPs of the donor), attacker's estimation error is computed for the sensitive SNPs of all individuals in $\mathbf{F}$ and it is denoted as $E^i_j(\mathbf{y} = 0)$ (for $i \in \mathbf{F}$ and $j \in \mathbf{P}^i_s$).[3] After the donor shares (reveals) some SNPs, the new estimation error is denoted as $E^i_j(\mathbf{y}^*)$ and the privacy loss for SNP j of individual i due to the sharing of the donor's SNPs is represented as $E^i_j(\mathbf{y} = 0) - E^i_j(\mathbf{y}^*)$. To quantify the effect of SNP sharing on the privacy of SNP j of individual i, a *privacy weight* (p^i_j) quantity is introduced as follows:

$$p^i_j = s^i_j \times (E^i_j(\mathbf{y} = 0) - E^i_j(\mathbf{y}^*)). \tag{4.5}$$

The utility of a SNP j is denoted as u_j. This quantity can be determined by researchers and genome studies. As discussed, the donor faces an optimization problem in which he wants to maximize the utility (i.e., number of shared SNPs). Formally, the optimization problem is defined as follows:

$$\begin{aligned} \underset{\mathbf{y}}{\text{maximize}} \quad & \sum_{j \in \mathbf{I^i}} u_j y^i_j \\ \text{subject to} \quad & \sum_{j \in \mathbf{P}^i_s} p^i_j \leq Pri(i, \mathbf{P^i_s}), \forall i \in \mathbf{F} \\ & y^i_j \in \{0, 1\}, \forall j \in \mathbf{P}^i_s. \end{aligned} \tag{4.6}$$

[3] Estimation error is computed as shown in (4.3).

In order to find a feasible solution, this optimization problem is first solved without considering the correlations between the SNPs (i.e., assuming that all SNPs are independent). Then, SNPs that are shared and violate the privacy tolerances of the individuals (when the correlations are considered) are hidden via a *fine-tuning algorithm*. To do so, first the family member k whose privacy constraint $Pri(k, \mathbf{P}_s^k)$ is violated the most as a result of the optimization part is identified. After identifying individual k, the next step is to hide some SNPs j from the shared SNP set of the donor ($\mathbf{R^i}$). To do this, a *global privacy weight* is computed for individual k due to each shared SNP j of the donor as follows:

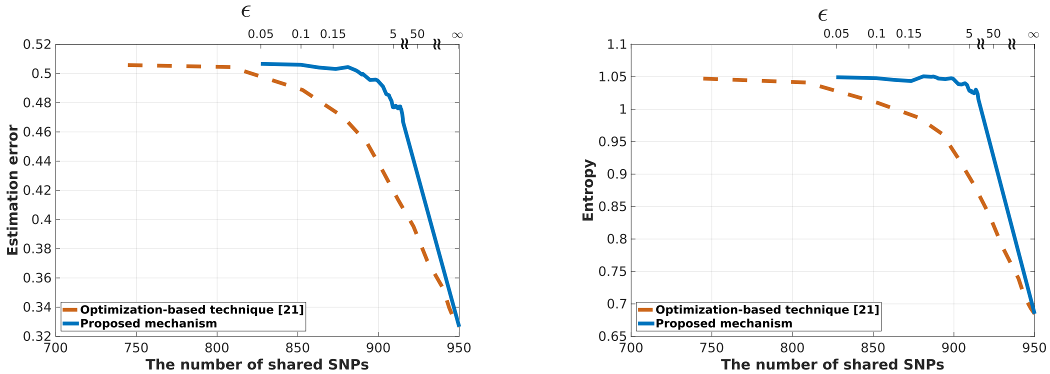
$$\delta_j^k = p_j^k + \sum_{l \in \mathbf{L_j}} p_l^k, \tag{4.7}$$

where set $\mathbf{L_j}$ includes all the SNPs that are in LD with SNP j. Then, for each shared SNP of the donor, the ratio between its global privacy weight and utility is computed to obtain $r_j^k = \delta_j^k / u_j$. The SNP j with the highest r_j^k value is removed from the set of shared SNPs of the donor (representing the SNP that causes high privacy loss for individual k while providing low utility). This process iteratively continues until the privacy constraints of all individuals in $\mathbf{F}$ are satisfied. We compare our proposed mechanism with Humbert *et al.*'s work [1] (hereafter, referred to as the "optimization-based mechanism") first without considering the kinship relationships between individuals and then, by also considering the kinship.

4.2.6.2 Comparison without considering kinship

For the donor i, we randomly choose 50 SNPs to construct his sensitive SNP set ($\mathbf{S^i}$) among 1000 SNPs ($\mathbf{I^i}$). As discussed before, our mechanism does not share the SNPs of individual i in $\mathbf{S^i}$, thus for sharing, we only consider SNPs in $\mathbf{I^i} \backslash \mathbf{S^i}$. To check the privacy constraints, we consider the SNPs in $\mathbf{S^i}$. Like Humbert *et al.*, we assume all of the SNP utilities and the sensitivities to be equal. We repeat the experiments for 10 random individuals and report the average.

We show the results of the comparison in terms of estimation error, entropy, and utility in Fig. 4.6. We observe that to achieve the same utility, the estimation error and the entropy provided by our proposed mechanism is significantly higher than the optimization-based method. On average, for the same utility, our mechanism provides 16% higher error and 18% higher entropy, which also means higher privacy. Moreover, the optimization-based mechanism always shares the SNPs that increase (or not change) the estimation error (and entropy) for the sensitive SNPs and hides the ones that decrease the error (and entropy). Thus, when a particular SNP j of the donor is hidden as a result of the optimization-based mechanism, the attacker can infer the value of that hidden SNP knowing that the actual value of that SNP reduces the error (and entropy) of the SNPs in the sensitive set. On the other hand, (as we have also shown via the toy example in Chapter 4.1.5) when deciding whether or not to share a SNP j, our proposed mechanism checks the change of the probability distributions for all the sensitive SNPs (regardless of the actual values of the SNPs) and if any of them violates (4.2), it do not share the corresponding SNP. Thus, our decision for sharing (or not sharing) a SNP does not provide extra information to the attacker.



(a) Error vs. utility for the proposed SNP sharing mechanism and the optimization-based technique.

(b) Entropy vs. utility for the proposed SNP sharing mechanism and the optimization-based technique.

Figure 4.6: Comparison between the proposed differential privacy-based SNP sharing mechanism and the optimization-based mechanism [1] when the kinship relationships between the individuals are not considered. The donor has 1000 SNPs in total and the size of the sensitive SNP set is set to 50. The utility is defined as the number of shared SNPs by the donor. In (a) we show the comparison in terms of the estimation error of the attacker and in (b) we show the comparison in terms of the uncertainty (entropy) of the attacker. The top x-axis in both plots shows the privacy parameter used for the proposed differential privacy-based mechanism. Privacy tolerances of individuals (i.e., $Pri(i, \mathbf{P_s^i})$ values) in [1] vary between 0 and 20.

Using this phenomenon, we also conduct attacker's inference attack by also using this additional auxiliary knowledge. That is, we assume that (i) attacker knows the sensitive SNP set of the donor (just the IDs of the SNPs, not the values), and (ii) attacker knows "not share" decision for a SNP means that the actual value of that SNP reduces the entropy of the SNPs in the sensitive set. Note that attacker cannot compute the estimation error (as it does not know the values of the SNPs in $\mathbf{S^i}$) but it can calculate the entropy (as (4.4) does not require the knowledge of the SNP values). In Fig. 4.7, we show the additional benefit of this attack for the attacker for both the proposed scheme and the optimization-based mechanism [1]. Here, we show the decrease in the estimation error from what we have shown in Fig. 4.6a. We observe that with this additional information, attacker's estimation error remains almost the same for the proposed mechanism. However, it decreases to almost 0 in the optimization-based mechanism, which shows the robustness of the proposed mechanism.

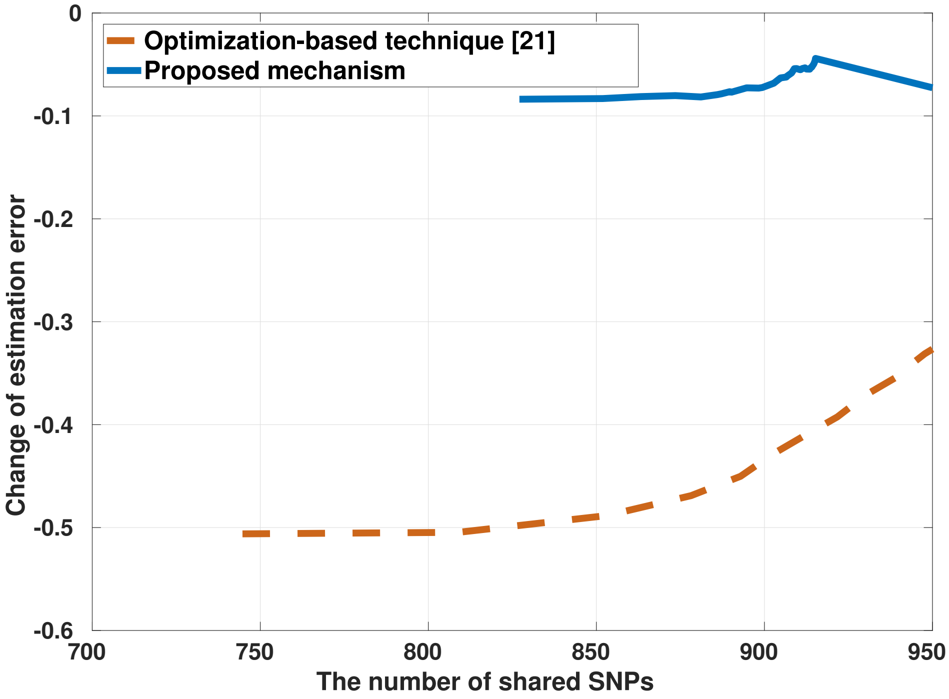


Figure 4.7: Decrease in the estimation error of the attacker from the values shown in Fig. 4.6a when the attacker uses additional auxiliary information about the decisions of the donor.

4.2.6.3 Comparison with considering kinship

Similar to Chapter 4.2.5, we use a trio (father, mother, and son) from [8]. We choose 100 neighboring SNPs, set the size of the sensitive SNP set to 20 for all family members, and randomly choose 20 SNPs for each family member to construct their sensitive SNP sets. We use both the proposed mechanism and the optimization-based mechanism to share the non-sensitive SNPs of the son (by also considering genomic privacy of mother and father). For the proposed mechanism, we use the same privacy parameter (ϵ) for all family members. Overall, for the same utility, we observe similar trends for entropy as in Fig. 4.6 and closer estimation error values for both schemes (we do not illustrate the results due to space constraints and due to the fact that the trend is similar to the previous experiment). Similar to before, when we also utilize the additional auxiliary information about the decisions of the donor in the inference attack, we observe that attacker's estimation error remains almost the same for the proposed mechanism. However, as before, it decreases to almost 0 in the optimization-based mechanism. This again shows the robustness of the proposed mechanism.

4.3 Discussion

Here, we discuss the proposed SNP sharing mechanism in terms of its functionality/practicality and robustness.

4.3.1 Functionality and Practicality

The proposed genomic data sharing mechanism provides privacy-preserving sharing of genomic data itself (not just the summary statistics from data as most previous works). As opposed to traditional differential privacy-based data sharing mechanisms, the proposed scheme does not introduce intentional noise to the shared data; it is based on selective sharing of data points while providing privacy guarantees for the sensitive parts of donor's genome. The proposed mechanism also considers a strong background knowledge of the attacker about the correlation model on the DNA and family relationships between individuals. We have shown that the proposed mechanism provides high utility while preserving individual and interdependent genomic privacy. In practice, a donor, depending on the entropy of the SNPs in his sensitive SNP set (not the values of those SNPs), may select a privacy parameter (ϵ) and share his non-sensitive SNPs with a service provider accordingly. It is important to note that the actual values of the SNPs in the sensitive set are not required for the sharing process (i.e., to check the condition in (4.2)).

When we do not consider the kinship relationships between the individuals, the time complexity to share a donor's SNP sequence of size n (of which m are in the sensitive set) is $O((n-m)n3^k + (n-m)m)$, where k is the order of the Markov chain that is used for the correlation model. When we also consider f

of the donor's family members during this process, the time complexity becomes $O((n-m)n^2 3^k f + (n-m)m)$. Thus, the time complexity scales quadratically (or cubic when the kinship is considered) with the number of SNPs to be shared by the donor. Time complexity scales exponentially with the order (k) of the Markov chain. However, as we discussed and showed via simulations (e.g., in Fig. 4.2), for correlation models of order higher than 4, the improvement in the inference power of the attacker is negligible, and hence we can assume the term 3^k as a constant. Considering the mechanism does not need to run in real-time, these complexity values are reasonable for practicality of the proposed mechanism.

4.3.2 Robustness

Our results illustrate the worst case scenarios in terms of attacker's power. We build a correlation model and compute the prior probability distributions of the SNPs to check the condition in (4.2) during SNP sharing. We use a population that is consistent with the donor's to build these models and we assume the attacker has access to the same population that also includes the victim (donor). In reality, the attacker may use a similar (but not the same) population for its inference attack. Therefore, its estimation error will be less than what we show in the evaluation.

The proposed SNP sharing mechanism considers the fraction of change in the probability distribution of all possible states of the sensitive SNPs (not the actual values of the sensitive SNPs of the donor and his family members). Therefore, not sharing a SNP from the non-sensitive SNP set does not mean that sharing that SNP would reduce the estimation error and entropy of the attacker about the SNPs in the sensitive set. In fact, as we have shown via the toy example in Chapter 4.1.5, sharing a SNP may actually decrease the estimation error (and entropy) of the attacker for the sensitive SNPs. Thus, the attacker cannot gain extra information by observing which SNPs are hidden by the mechanism. As another consequence of this property, for the proposed SNP sharing mechanism, attacker's estimation error and entropy do not monotonically decrease with the increasing privacy parameter (i.e., increasing ϵ value or increasing privacy budget for the donor). In Fig. B.1 (in Appendix B.2), we show the variation of estimation error and entropy with increasing privacy parameter. On the contrary, in Humbert *et al.*'s work [1], a SNP is shared only if it does not decrease the estimation error (and entropy) of the attacker. Also in [1], SNPs shared due to an increase in privacy budget always cause monotonic decrease in both estimation error and entropy of the attacker. With this knowledge, the attacker can actually infer the values of the SNPs the mechanism decides to hide. Our sharing mechanism is robust against this aforementioned attack.

Chapter 5

Conclusion And Future Work

We have proposed a privacy-preserving genomic data sharing mechanism based on differential privacy. Our method keeps an attacker's knowledge about sensitive parts of individuals' genomes within a boundary, while providing public availability of genomic data.

The proposed mechanism considers both individual and interdependent genomic privacy. That is, when a donor shares his genomic data, both his and his family member's genomic privacy are protected. One notable feature of the proposed scheme is that it selectively shares the SNPs of a donor without considering the real values of his sensitive SNPs. This prevents the attacker from initiating inference attacks based on the sharing decisions on the SNPs. We have studied and discussed the effects of different parameters on both utility and privacy of the proposed mechanism. Specifically, we have shown the relationship between the amount and type of sensitive SNPs and the utility.We have also shown that the proposed mechanism outperforms the previous work both in terms of privacy and utility.

As future work, we will explore more scenarios on different kinship relationships such as (i) the situation in which some family members already revealed some of their SNPs, and (ii) practicality of the proposed mechanism on an extended family (e.g., which family members to consider and how far to navigate in a family tree during the SNP sharing process). Furthermore, inspired by the idea about watermarking sequential data, we will explore adapting our proposed mechanism in such a way to provide privacy and address the liability issues at the same time.

Bibliography

[1] M. Humbert, E. Ayday, J.-P. Hubaux, and A. Telenti, "Reconciling utility with privacy in genomics," in *Proceedings of the 13th Workshop on Privacy in the Electronic Society*, pp. 11–20, ACM, 2014.

[2] `https://www.23andme.com/en-int/`, 2017. [Online; accessed 5-May-2017].

[3] `https://opensnp.org/`, 2017. [Online; accessed 5-May-2017].

[4] A. Jolie, "My medical choice," *The New York Times*, vol. 14, no. 05, p. 2013, 2013.

[5] `http://www.eupedia.com/genetics/medical_dna_test.shtml`, 2017. [Online; accessed 6-May-2017].

[6] 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.

[7] L. Sweeney, A. Abu, and J. Winn, "Identifying participants in the personal genome project by name," 2013.

[8] `https://personalgenomics.zone/2016/05/24/my-personal-exome-analysis-part-i-first-findings-2/`, 2017. [Online; accessed 6-May-2017].

[9] A. APOC, "On jim watson?s apoe status: genetic information is hard to hide," *European Journal of Human Genetics*, vol. 17, pp. 147–149, 2009.

[10] M. Slatkin, "Linkage disequilibrium?understanding the evolutionary past and mapping the medical future," *Nature Reviews Genetics*, vol. 9, no. 6, pp. 477–485, 2008.

[11] 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 2013 ACM SIGSAC conference on Computer & communications security*, pp. 1141–1152, ACM, 2013.

[12] C. Dwork, "Differential privacy: A survey of results," in *International Conference on Theory and Applications of Models of Computation*, pp. 1–19, Springer, 2008.

[13] S. E. Fienberg, A. Slavkovic, and C. Uhler, "Privacy preserving gwas data sharing," in *Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on*, pp. 628–635, IEEE, 2011.

[14] A. Johnson and V. Shmatikov, "Privacy-preserving data exploration in genome-wide association studies," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1079–1087, ACM, 2013.

[15] F. Yu, S. E. Fienberg, A. B. Slavković, and C. Uhler, "Scalable privacy-preserving data sharing methodology for genome-wide association studies," *Journal of biomedical informatics*, vol. 50, pp. 133–141, 2014.

[16] 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.

[17] `http://www.internationalgenome.org`, 2017. [Online; accessed 21-January-2017].

[18] `http://mathgen.stats.ox.ac.uk/impute/impute_v2.html`, 2017. [Online; accessed 19-January-2017].

[19] M. Humbert, E. Ayday, J.-P. Hubaux, and A. Telenti, “Addressing the concerns of the Lacks family: Quantification of kin genomic privacy,” *Proceedings of ACM CCS ’13*, 2013.

[20] D. S. Falconer and T. F. Mackay, *Introduction to Quantitative Genetics (4th Edition)*. Harlow, Essex, UK: Addison Wesley Longman, 1996.

[21] M. Hardt and K. Talwar, “On the geometry of differential privacy,” in *Proceedings of the forty-second ACM symposium on Theory of computing*, pp. 705–714, ACM, 2010.

[22] N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, “Optimal geo-indistinguishable mechanisms for location privacy,” pp. 251–262, 2014.

[23] `https://ghr.nlm.nih.gov/primer/genomicresearch/snp`, 2017. [Online; accessed 6-May-2017].

[24] S. S. Samani, Z. Huang, E. Ayday, M. Elliot, J. Fellay, J.-P. Hubaux, and Z. Kutalik, “Quantifying genomic privacy via inference attack with high-order snv correlations,” in *Security and Privacy Workshops (SPW), 2015 IEEE*, pp. 32–40, IEEE, 2015.

[25] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Elsevier, 2014.

[26] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, “Factor graphs and the sum-product algorithm,” *IEEE Transactions on information theory*, vol. 47, no. 2, pp. 498–519, 2001.

[27] M. Naveed, E. Ayday, E. W. Clayton, J. Fellay, C. A. Gunter, J.-P. Hubaux, B. A. Malin, and X. Wang, “Privacy in the genomic era,” *ACM Computing Surveys (CSUR)*, vol. 48, no. 1, p. 6, 2015.

[28] 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, p. e1000167, 2008.

[29] R. Wang, Y. F. Li, X. Wang, H. Tang, and X. Zhou, “Learning your identity and disease from research papers: information leaks in genome wide association study,” in *Proceedings of the 16th ACM conference on Computer and communications security*, pp. 534–544, ACM, 2009.

[30] S. S. Shringarpure and C. D. Bustamante, “Privacy risks from genomic data-sharing beacons,” *The American Journal of Human Genetics*, vol. 97, no. 5, pp. 631–646, 2015.

[31] S. Jha, L. Kruger, and V. Shmatikov, “Towards practical privacy for genomic computation,” in *Security and Privacy, 2008. SP 2008. IEEE Symposium on*, pp. 216–230, IEEE, 2008.

[32] M. Blanton, M. J. Atallah, K. B. Frikken, and Q. Malluhi, “Secure and efficient outsourcing of sequence comparisons,” in *European Symposium on Research in Computer Security*, pp. 505–522, Springer, 2012.

[33] C. A. Cassa, R. A. Miller, and K. D. Mandl, “A novel, privacy-preserving cryptographic approach for sharing sequencing data,” *Journal of the American Medical Informatics Association*, vol. 20, no. 1, pp. 69–76, 2013.

[34] P. Baldi, R. Baronio, E. De Cristofaro, P. Gasti, and G. Tsudik, “Countering gattaca: efficient and secure testing of fully-sequenced human genomes,” in *Proceedings of the 18th ACM conference on Computer and communications security*, pp. 691–702, ACM, 2011.

[35] E. Ayday, J. L. Raisaro, J.-P. 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*, pp. 95–106, ACM, 2013.

[36] X. S. Wang, Y. Huang, Y. Zhao, H. Tang, X. Wang, and D. Bu, “Efficient genome-wide, privacy-preserving similar patient query based on private edit distance,” in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 492–503, ACM, 2015.

[37] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann Publishers, Inc., 1988.

[38] A. T. Ihler, W. F. John III, and A. S. Willsky, "Loopy belief propagation: Convergence and effects of message errors," *Journal of Machine Learning Research*, vol. 6, no. May, pp. 905–936, 2005.

[39] F. Kschischang, B. Frey, and H. A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, 2001.

[40] N. Li and M. Stephens, "Modeling linkage disequilibrium and identifying recombination hotspots using single-nucleotide polymorphism data," *Genetics*, vol. 165, 2003.

[41] `http://www.snpedia.com/`.

[42] R. Drmanac, A. B. Sparks, M. J. Callow, A. L. Halpern, N. L. Burns, B. G. Kermani, P. Carnevali, I. Nazarenko, G. B. Nilsen, G. Yeung, *et al.*, "Human genome sequencing using unchained base reads on self-assembling DNA nanoarrays," *Science*, vol. 327, no. 5961, pp. 78–81, 2010.

[43] `https://manuelcorpas.com/2016/05/24/my-personal-exome-analysis-part-i-first-findings-2/`.

[44] I. Wagner, "Evaluating the strength of genomic privacy metrics," *ACM Transactions on Privacy and Security (TOPS)*, vol. 20, no. 1, p. 2, 2017.

[45] S. S. Samani, Z. Huang, E. Ayday, M. Elliot, J. Fellay, J.-P. Hubaux, and Z. Kutalik, "Quantifying genomic privacy via inference attack with high-order SNV correlations," *Proceedings of Workshop on Genome Privacy and Security (GenoPri'15)*, 2015.

Appendix A

Standard Deviation of the Conducted Experiments

We computed and plotted the standard deviations of the experiments. In Fig. A.1 and Fig. A.2 we show CEPH/UTAH pedigree results with error bars which represents the standard deviation of 50 runs over error and entropy. As shown, the results from the experiments do not have significant deviations from the average.

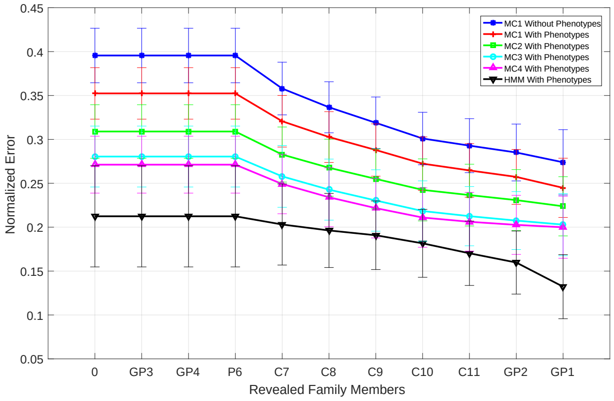


Figure A.1: Decrease in genomic privacy of P5 (in Fig. 3.3) in terms of the *incorrectness* of the attacker. We reveal partial genomes of other family members for different high order correlation models in the genome. MC stands for the Markov chain model (with different orders) and HMM stands for the hidden Markov model.

A.1 Bottleneck of the Markov Chain Order

We have conducted two experiments on the UTAH family in order to see the relation between bottleneck of the Markov chain order and number of hidden SNPs. We hide 10 and 90 percent of SNPs of each family member and then start to infer the missing SNPs. In Fig. A.3 and Fig. A.4 we show the effect of number of hidden SNPs on error (uncertainty) while inferring SNPs of P5. We conclude that the Markov chain bottleneck is related to the number of SNPs we try to infer. When the number of observed SNPs (by the attacker) is a lot, Markov models have more data to work with, and hence they converge to a small error value even with low order models. Thus, higher order models would not make the error any smaller. On the other hand, when the attacker observes fewer SNPs, increasing the order of the Markov chain model also increases the chance of inferring an unobserved SNP. For instance, in Fig. A.3, when we reveal 90 percent of each family member's SNPs (i.e., when the attacker already observes a significant amount of data), results obtained by Markov order 3 and 4 are totally overlapping. However, in Fig. A.4, when we reveal only 10 percent of each family member's SNPs, Markov order 4 does a significantly better job than Markov order

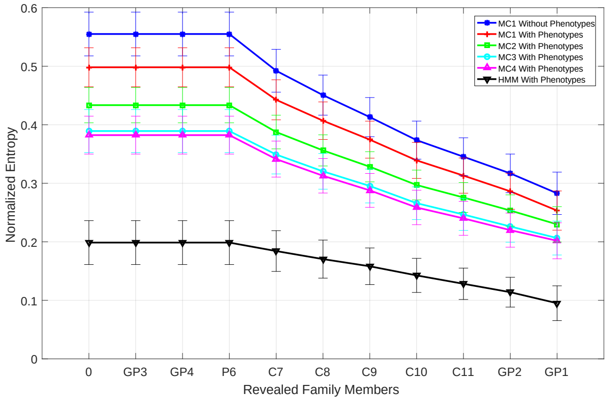


Figure A.2: Decrease in genomic privacy of P5 (in Fig. 3.3) in terms of the *uncertainty* of the attacker. We reveal partial genomes of other family members for different high order correlation models in the genome. MC stands for the Markov chain model (with different orders) and HMM stands for the hidden Markov model.

3.

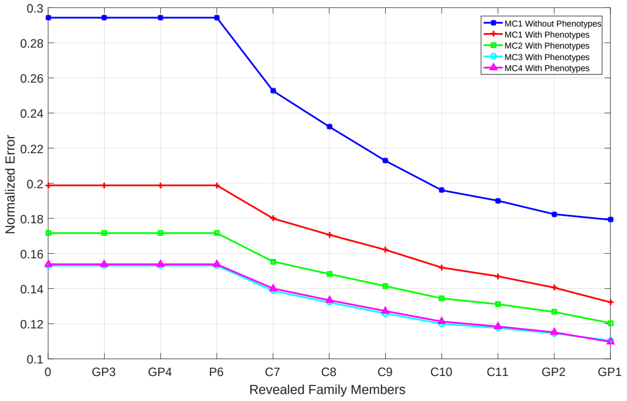


Figure A.3: Decrease in genomic privacy of P5 in terms of the *incorrectness* of the attacker, when we reveal 90 percent of random SNPs from other family members.

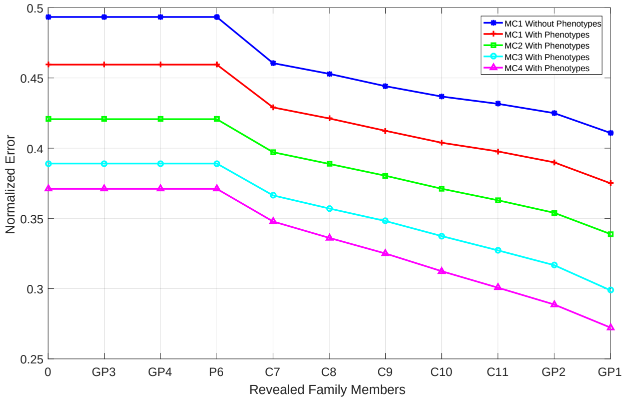


Figure A.4: Decrease in genomic privacy of P5 in terms of the *incorrectness* of the attacker, when we reveal 10 percent of random SNPs from other family members.

Appendix B

B.1 Proof of Theorem 1

In this section, we prove that our proposed mechanism for sharing SNPs implies generalized formulation of differential privacy.

Proof of Theorem 4.1. In the generalized formulation of differential privacy, Hardt and Talwar introduce a family of probability measures (M), such that, $M = \{\mu_x : x \in \Re^n\}$, where each measure μ_x is defined on $\Re^d$ [21]. A mechanism is called ϵ-differentially private if for all $x, y \in \Re^n$, such that $|x - y|_1 \leq l$, and for a measurable $S \in \Re^d$, we have $\dfrac{\mu_x(S)}{\mu_y(S)} \leq exp(l\epsilon)$.

In our formulation, the mechanism is the sharing procedure. Thus, M can be considered as the inference attack of the attacker and the S as the shared SNP sequence of the donor i ($\mathbf{R^i}$ in our formulation). x corresponds to each SNP j in the sensitive SNP set of the donor ($\mathbf{S^i}$) and other family members. The measure $\mu_x(S)$ can then be considered as the probability of observing S given the auxiliary information of attacker about x. By applying Bayes' law we have the following:

$$\frac{P(S|x)}{P(S|x')} = \frac{P(x|S)P(x')}{P(x'|S)P(x)}$$

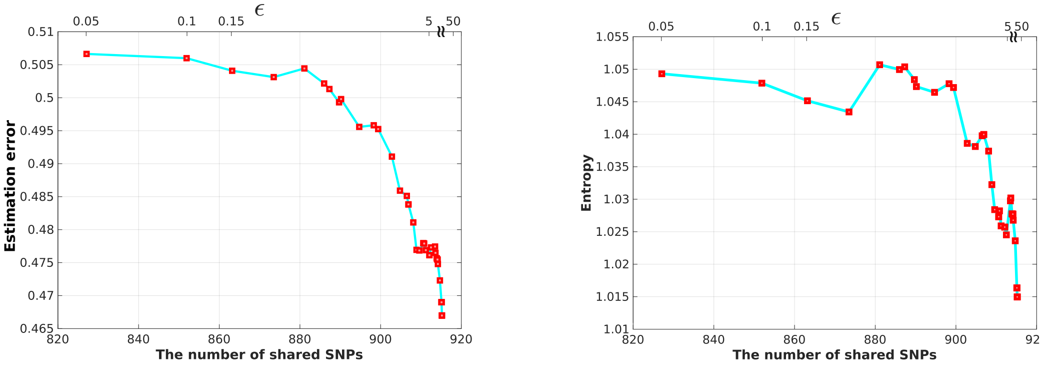
Substituting the right part of equation in the generalized formulation of differential privacy, we have:

$$\frac{P(x|S)P(x')}{P(x'|S)P(x)} \leq exp(\epsilon) \rightarrow \frac{P(x|S)}{P(x'|S)} \leq exp(\epsilon)\frac{P(x)}{P(x')}. \tag{B.1}$$

Thus, we can conclude the equivalence. □

B.2 Change in Attacker's Estimation Error and Entropy

As we discussed in Section 4.3.2, for the proposed SNP sharing mechanism, attacker's estimation error and entropy do not monotonically decrease with the increasing privacy parameter (ϵ). In Fig. B.1, we show the variation of estimation error and entropy with increasing privacy parameter. Due to this behavior, the attacker cannot infer the values of the SNPs the mechanism decides to hide.



(a) Estimation error vs. utility with increasing privacy budget (ϵ value)

(b) Entropy vs. utility with increasing privacy budget (ϵ value)

Figure B.1: Relationship between the estimation error and utility for increasing privacy budget (ϵ value) for the proposed SNP sharing mechanism.