A UTILITY MAXIMIZING AND PRIVACY PRESERVING APPROACH FOR PROTECTING KINSHIP IN GENOMIC DATABASES

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
Gülce Kale
March 2017

A UTILITY MAXIMIZING AND PRIVACY PRESERVING APPROACH FOR PROTECTING KINSHIP IN GENOMIC DATABASES

By Gülce Kale

March 2017

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.

Öznur Taştan Okan(Advisor)

Erman Ayday

Tolga Can

Approved for the Graduate School of Engineering and Science:

Ezhan Karaşan
Director of the Graduate School

ABSTRACT

A UTILITY MAXIMIZING AND PRIVACY PRESERVING APPROACH FOR PROTECTING KINSHIP IN GENOMIC DATABASES

Gülce Kale
M.S. in Computer Engineering
Advisor: Öznur Taştan Okan
March 2017

Rapid and low cost sequencing of genomic data enables widespread use of genomic information in research studies and personalized customer applications, where people share their genomic data in public databases. Although the identities of the participants are anonymized in these databases, sensitive information about individuals can still be inferred if the stored data is not shared in a privacy-preserving manner. Proper handling of kinship information is one such caveat that needs to be addressed to avoid exposure of privacy-sensitive information. In this work, we show that by using only the publicly available single nucleotide polymorphism (SNP) data of anonymized individuals, kinship relationships can be inferred. We present two scenarios that result in privacy leakage; one based on genomic similarity of the individuals; the other, through the outlier allele pair counts of the family members. In the proposed models, we assume that the family members join to the database sequentially and we systematically identify minimal portions of data to withhold as the new participants are added to the database. Choosing the proper positions to hide is cast as an optimization problem. Therein, the number of positions to mask is minimized subject to several privacy constraints that ensure the kinship information among any pair of the family members is not leaked. We evaluate the proposed technique on real genomic data of two different families of size five by considering different sequential arrival orders for the family members. Results indicate that concurrent sharing of data pertaining to a parent and an offspring results in high risks of privacy leakages, whereas the sharing data from further relatives together is often safer. We also show that different arrival orders of the members can lead to different levels of privacy risks and the utility of shared data can vary. Adoption of the proposed method shall allow safe sharing of genomic data in terms of kinship privacy in future research studies and public genomic services.

Keywords: Genomic privacy, optimization, family privacy, single nucleotide polymorphism.

ÖZET

GENOMİK VERİTABANLARINDA AKRABALIK İLİŞKİLERİNİN GİZLİLİKLERİNİ AZAMİ FAYDA SAĞLAYARAK KORUYAN BİR YAKLAŞIM

Gülce Kale
Bilgisayar Mühendisliği, Yüksek Lisans
Tez Danışmanı: Öznur Taştan Okan
Mart 2017

Genomik verilerin hızlı ve düşük maliyetli dizilimi, katılımcılara ait genomik bilgilerin saklandığı veri tabanlarını kullanan genetik araştırmaları ve kişisel servis uygulamalarını yaygınlaştırmaktadır. Bu veri tabanlarında kişilerin kimlikleri anonimleştirilse de, genomik veriler gizlilik korunmadan paylaşıldığında, kişiler hakkında hassas bilgiler edinilebilir. Akrabalık ilişkilerinin uygun şekilde saklanması güvenlik ihlallerinin engellenebilmesi için önemli noktalardan biridir. Bu çalışmada, yalnızca tek nükleotid polimorfizm (SNP) verilerinin kamuya açık kayıtlarını kullanıldığı bir durumda bile akrabalık ilişkilerinin tespit edilebilir olduğunu gösteriyoruz. Kişilerin genomik benzerlikleri ve aile üyelerinin arasındaki aykırı alel çift sayılarının varlığının, akrabalık ilişkilerini risk altına koyduğunu gözlemliyoruz. Çalışma kapsamında, riskleri en aza indirgemek için, akrabalık gizliliğinden ödün vermeden verilerin, maksimum fayda ile paylaşımını mümkün kılan hesaplama modelleri sunuyoruz. Bu modellerde, aile üyelerinin veri tabanına sırayla geldiklerini varsayıyoruz. Modeller, yeni aile üyeleri veri tabanına eklendikçe, sistematik olarak genomik veride saklanacak asgari bölümleri tespit ediyor. Hangi pozisyonların ne ölçüde saklanması gerektiğini, saklanan pozisyonlarının sayısını en aza indirildiği ve akrabalık bilgilerinin sızdırılmamasını engelleyen mahremiyet kısıtlamalarına tabi tutulduğu bir optimizasyon problemi ile buluyoruz. Beş bireyden oluşan iki farklı ailenin, aile bireylerinin veri tabanına geldiği farklı sıralarda, modelleri uyguladık. Aldığımız sonuçlara göre, bir ebeveyn ve bir çocuğun genomik verilerinin eşzamanlı paylaşımı, akrabalık ilişkisini yüksek risklerle açığa çıkartırken, daha uzak akrabalarda, güvenli veri paylaşımın mümkün olduğunu görüyoruz. Öte yandan, aynı aile üyeleri veri tabanına farklı sıralarla geldiklerinde, farklı derecede gizlilik riskleri ve veri paylaşım fayda değerleri ile sonuçlanabildiğini gösteriyoruz. Önerilen yöntemin benimsenmesinin, gelecek araştırmalarda ve kamu genom hizmetleri alanlarında, akrabalık gizliliği koruyarak güvenli genom veri paylaşımına izin vereceğini umuyoruz.

Anahtar sözcükler: Genomik gizlilik, optimizasyon, aile mahremiyeti, tek nukleotid farklılıkları.

Acknowledgement

Foremost, I would like to thank my supervisor Asst. Prof. Dr. Öznur Taştan Okan for her motivation, immense knowledge, and guidance. I have learnt many things since I became her student. I am also grateful to Asst. Prof. Dr. Erman Ayday for his creative ideas and his precious time to improve my research.

I am grateful to Yalım Baç who has read all this thesis for his continous help and support. He has a great influence on my personal and professional life.

I am thankful to my friends for their support, and the fun times we have spent; Ekin Demirci, Faruk Sevgili, Naz Şerifoğlu, Eylül Celtemen, Hazal Koptagel, Günce Uzgören. I would also like to thank my office friends Gökalp Urul, Onur Aydın, Istemi Bahçeci, Seher Acer, Elif Eser, Başak Ünsal, Gökçe Ayduğan, and Ebru Ateş.

Finally, my deep and sincere gratitude to my family for their continuous, love, help and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Since the completion of the human project in 2003 [1], significant progress has been made in sequencing technologies. With the advent of next-gen sequencing technologies, determining the complete sequence of an individuals genome is faster and cheaper than ever [2]. This progress rendered the extensive use of genome sequencing in biomedical research possible. Access to a large collection of genomic data is a precursor for potential scientific breakthroughs in medicine and genomics. While the use of genomic data in research studies gains traction, there is a concurrent increase in the number of web services that enable genomic data sharing (openSNP [3], 23andme [4], etc). Currently thousands of genomes are publicly shared online. Such a rise in the availability and use of genomic data raises important ethical, legal and social concerns. One immediate and pressing issue is the sharing of genomic data without compromising the privacy of the participants and their families.

An individuals genome is unique (except for the identical twins), encodes sensitive information and it is an inherently stable means for long-term storage. DNA carries personal information pertaining to its owner such as ethnicity, kin or predisposition to certain diseases such as cancer or schizophrenia. Moreover, some of the privacy risks are unanticipated and manifests themselves only after years of scientific progress. For example, association of a genomic variation with a certain disease can be discovered years after release of the data.

Even though most of the genomes on the Internet are anonymized, it has been shown that anonymization is not sufficient for protecting the identities of the genome donors [5, 6]. Once the owner of a genome is identified, she may face with genetic discrimination. An example case is reported by Dr. Noralane Lindo [7]. Dr. Lindor sequenced genomes of the grandchildren of a cancer patient. She found out that two of the grandchildren have the same mutation observed in the cancer patient, which predisposes them to cancer. Later one of the grandchildren carrying the mutation applied to army to pursue a career as a pilot. Upon disclosing the genetic test results, she was immediately rejected.

The scope of privacy concerns extend beyond the donors whose data are being shared. Henrietta Lacks family conflict is an exemplifying case [8]. Henrietta Lacks died due to cervical cancer in 1951. Some of her cancer cells are stored and have been used extensively for medical research named as HeLa cell line. Recently, Ms. Lacks genome was sequenced and publicly shared publicly online. The family was not asked for consent and they discovered this in a book called "The Immortal Life of Henrietta Lacks"[9]. The Lacks family objected to the unauthorized publication of the genome as they were concerned that the genome will reveal medical information and other personal information about the remaining members of the family. Data publisher claimed that since the family has changed over time, the privacy of the descendants was not at risk. However, short after the genome was made available, extensive information about the family was immediately discovered [10]. By the time the sequence was pulled off from the website, many people had already downloaded the data. Thus, the breach of privacy was irreversible.

Availability of shared data is both critical for carrying out large scale studies that relies on genomic information and for future hopes to offer personalized services and medicine. As the negative stories accumulate, and the fear of potential misuse of genomic data escalates, the public share of genomic data can be severely restricted by new regulations and/or by unwillingness among potential donors. Therefore, to support research that involves the handling of large-scale genomic data and to expand the ways in which genomic information can be used, privacy issues should be properly addressed. Implementing robust computational models that enable the privacy-preserving dissemination of data are critical ingredients. Towards this aim, in this thesis, we specifically focus on risks associated with the kinship information of the individuals in the genomic databases. Of particular interest are the algorithmic routes that render the maximal sharing of data possible without compromising kinsip privacy.

Once the genome of single member from a family is revealed, significant information about the genomes of the donors relatives can be revealed, putting their privacy at risk [11]. Thus, in addition to being sensitive information by itself, kinship information have the potential to comprise the genomic privacy of the family members in conjunction with other genomic attacks. In this thesis, we develop strategies for preserving the privacy in genomic databases. We show that the families in a database can be inferred by simple clustering techniques with the distance metrics calculated based on the genome dissimilarity. Main contribution of the thesis is providing privacy preserving and utility maximizing computational methods for sharing genomic data.

We assume that the family members arrive to the database sequentially. Our models involve masking certain part of the newly arrived member's genome. Which positions to hide are decided based on the inferred relationship of the individual to the other family members, and the genotype of the family members that are already in the database. We cast this problem as an optimization problem where the number of positions to mask is minimized subject to the privacy constraints that ensure the kinship information is not leaked. This technique lets us to systematically identify minimal portions of data to withhold as the new donors are added to the database. The proposed technique is evaluated in two different families; one that is inferred with an attack from the OpenSNP database and one that is publicly available.

The thesis is organized as follows:

- Chapter 1 introduces the thesis and describes the problem.

- Chapter 2 provides genetic background and discusses the related work.
- Chapter 3 details the method we developed; kinship inference from public genomic databases and its countermeasures.
- Chapter 4 evaluates the results on two different families.
- Chapter 5 recapitulates the key findings in this study and states possible future extensions of the proposed models.

Chapter 2

Background and Related Work

2.1 Genetic Background

DNA is the molecule that carries the genetic information. It is made up of building blocks called nucleotides which are attached together to form DNA's double helix structure. A DNA nucleotid can have four different values: $\{A, T, C, G\}$. All humans share at least 99.5% similar DNA [12] and if the genetic variation occurs in at least 1% of the population, it is called a *single nucleotid polymorphism (SNP)*. SNPs are the most common type of genomic variation. Each SNP position consists of two nucleotides; one is obtained from the mother and the other is obtained from the father. These nucleotids are called *alleles*. An allele pair is *homozygous* for a particular gene if the two alleles are the same and *heterozygous* if they are different from each other. For every SNP position, there is an identified nucleotide base which is called *reference allele* and other types of bases called are *non-reference or alternate allele*.

Mendel's law of segregation indicates that allele pairs are separated from each other for reproductive process and are randomly distributed into the gamete cells with equal probabilities; 0.5. The offspring has one allele pair for each gene; one obtained from the mother and the other from the father. In Figure 2.1, the possible allele pairs of an offspring can have are shown, given the genotype of the parents. Assume a SNP can carry two possible nucleotide values; A and a. If both of the parents are homozygous on the same nucleotide, i.e. AA *and* AA, child can only be homozygous; AA. If the parents are different homozygous i.e. AA and aa, child will be heterozygous Aa. The offspring can be heterozygous Aa or homozygous AA with equal probabilities, if one of the parents is homozygous AA and the other is heterozygous Aa. When both of the parents are heterozygous Aa and Aa, the child can posses the following allele pairs: AA, Aa, aa.

		Mother's genotype		
		AA	**Aa**	**aa**
Father's genotype	**AA**	AA	AA, Aa	Aa
	Aa	AA, Aa	AA, Aa, aa	Aa, aa
	aa	Aa	Aa, aa	aa

Figure 2.1: According to Mendel's law, possible allele pairs of an offspring, given the parents' genotype. A and a are the two possible nucleotides that a gene can have. A denotes dominant allele and a denotes recessive allele.

We label the alleles of an individual at a certain position with the number of non-reference alleles. The instances for which both alleles are the same as the reference genome is shown as 0, the positions wherein only one allele differ from the reference genome are denoted by 1, and finally the instances wherein an individual carry the two alternate alleles shall be denoted by 2. For two individuals, i and j, we denote the pairs of alleles with $[s_i, s_j]$. There are six possible combinations: both individuals have two reference alleles at the designated position: [0,0], one individual has one reference allele and the other has two: [1,0], the individuals are homozygous: [2,0], the individuals are heterozygous: [1,1], one individual has one alternate allele and the other one has two: [2,1], and finally both individuals have two alternate alleles [2,2]. First of all, note that without applying our methodology to hide the positions, it might be possible to detect relatives based on pairwise allele counts. For example [2,0] allele pair will almost always be zero in parent-offspring relations due to Mendel's law.

The frequencies of alleles for each SNP position in a population are known. The most frequently observed allele is major allele and the second frequently observed allele is called minor allele. *Minor allele frequency (MAF)* is commonly used in bioinformatics because it gives information about common and rare variants in the population. In DNA, many SNPs are linked with each other. If the association between different SNPs are non-random, it is called *Linkage disequilibrium (LD)* and it is determined from population's genetic history [13].

2.2 Familial Relationship Inference

Relationship inference techniques through the genotype data are common. Many studies have conducted in the literature. Moreover, there are personal services that provide people to track their relatives[4, 14, 15]. Relationship inference is basically based on calculation of IBS, IBD and/or kinship coefficients. If two person have identical alleles in a specific genome segment, this segment is called identical by state (IBS). When two people share a common ancestry in an IBS segment; the segment is identical by descent (IBD). Kinship coefficient is the probability of a randomly sampled allele from person p_1 and a randomly sampled allele from person p_2 being *IBD*.

Many tools and algorithms have been developed to calculate the kinship coefficient to find out unavailable pedigree information among people. Some of these tools are found in the literature are PLINK, KING, GRAB, ERSA, REAP and PC-Relate. GRAB algorithm splits the genome data into blocks, among them it finds the segments which are IBS, and infers the relationship degree by using a classification tree [16]. PLINK detects relatives by inferring IBD status as outputs of Hidden Markov Model (HMM) which uses IBS data [17] but it assumes the population is homogeneous. Manichaikul et al. [18] developed a very rapid algorithm; KING-robust to detect familial relationships which also works in populations with discrete substructure. ERSA predicts the degree the relationship degree using maximum likelihood estimates on IBD segments features [19]. REAP is developed to find familial relatives in admixed populations [20] to overcome biased estimates of KING. The most recent developed method is PC-Relate, which infers pedigrees based on principle component analysis (PCA) to predict the probabilities of IBD sharing segments and kinship coefficients [21].

Personal services like Ancestry.com [15] helps the customers to find their relatives. These services compare all the genomic data that in their genomic database and the genomic data of the client. For every pair of clients in the database, they determine IBD segments. Ancestry.com [22] has its own distribution of 24,362 clients pairwise IBD segment length and corresponding pedigree relationship information. They predict the kinship relations of a person by analyzing at that distribution. FamilyTreeDNA.com provide a service called FamilyFinder which finds the client's relatives [14]. The software runs on pre-clustered SNP sets that are 50-100 SNP long and then calculate SNP sets for matching. Finally, it analyzes the adjacent SNP sets to determine if they are IBD and it calculates the kinship relationship of two persons based on the number and size of the identified IBD segments.

2.3 Familial Privacy

When an individual share her own genomic data, it might disclose information about her relatives. In subsection 2.3.1, we explain what kind of attacks can be made to one's genomic data to infer the other relatives' genomic information, and the attacks that might reveal other relatives' real identites. In subsection 2.3.2, we provide the techniques that protect genomic privacy.

2.3.1 Threats that Violates Familial Privacy

This section summarizes attacks that violate genomic privacy of the family members. Section 2.3.1.1 explains how an individual's genomic data is reconstructed using other family members' data.

2.3.1.1 Reconstruction Attacks

Reconstruction attacks disclose an individual's unknown genomic data by using the data observed from the relatives. These attacks have prior information on:

1. Family member's genomic data
2. Genomic knowledge such as LD or MAF

Kong et al. [23] show that an individual's haplotype information can be derived from LD-based analysis of other family members' genotype data; i.e. genomic data of child can be inferred from the parents' data. Kahveci et al. [24] predict the mother's genomic data given only the child and father's genomes by using LD knowledge. Additionally, reconstruction of sibling's genome is also investigated; if the adversary knows one of the siblings, he can infer other sibling's genotype with 91.9% accuracy [25]. *Humbert et al.* [11] develops a technique that uses belief propagation algorithm in order to reconstruct further degree relatives having genotype of other family members' and LD knowledge.

2.3.1.2 Other Threats

Assume that an individual shares her genetic data anonymously in a public environment. Her identity can be revealed via re-identification attacks [26]. The attacker can trail her relatives using her real identity in social media [27]. This is very easy if people make their family information section public in Facebook. Additionally, Y chromosome inherits directly from father to child. Gymrek et al. [5] show that if an individual's genotype data is publicly available, his and his family members' identity can be revealed by considering the association between Y chromosome and surname.

2.3.2 Privacy Protection Techniques

There are many techniques to preserve an individual's genomic privacy. *Access control* shares the data exactly as it is, but stores the access to the data is restricted in a secure environment and the users who can reach the data is restricted to people who work in a specific research studies. dbGAP is one of the platforms that use this approach [26]. In public genomic data platforms like OpenSNP, users are able to publish their data *anonymously.* As it is mentioned, re-identification attacks can reveal user's real identity. *Cryptographic techniques* can protect a user's privacy but it is discussable how much utility they provide. All of these techniques are employed to protect an individual's privacy. In the literature, there is Humbert et al.'s study that aims to protect a family's genomic privacy [28].

Humbert et al. [28] developed a technique which enables individuals to share their genomic data with maximum utility meantime protecting personal and familial privacy. The technique protects the genomic privacy against an adversary aiming to estimate some target SNPs that are masked in one or more family members. They define a privacy metric to compute the adversary's error in inferring SNPs and the sensitivity of SNPs. The problem is described as an optimization problem which maximizes the utility subject to restricting the privacy risks under a predefined privacy threshold. This problem is a linear optimization problem if LD correlations are not given, and otherwise it is a non-linear optimization problem. They found the optimal SNPs to be hidden via branch-and-bound algorithm and compared their results with an exhaustive search over subset of SNPs. The exhaustive search could not find the SNPs as optimal as the method they developed in terms of utility but the branch-and-bound algorithm do not scale well for more than 50 SNPs.

When family members share their genetic data, an adversary can detect different genomic privacy leaks that might affect the family. It is difficult to conduct a study that provides protection against all types of privacy risks. Like Humbert et al.'s work, our goal is also to preserve genomic kin privacy, but the difference between these two studies is the protection of families against different types of privacy breaches. We have developed a framework which preserves the genomic kin privacy in terms of *not revealing familial relationships* for every incoming family member to the database. Humbert et al.'s technique protects the individual and kin genomic privacy against *an adversary who aims to infer some target and non-observed SNPs in family members.*

Chapter 3

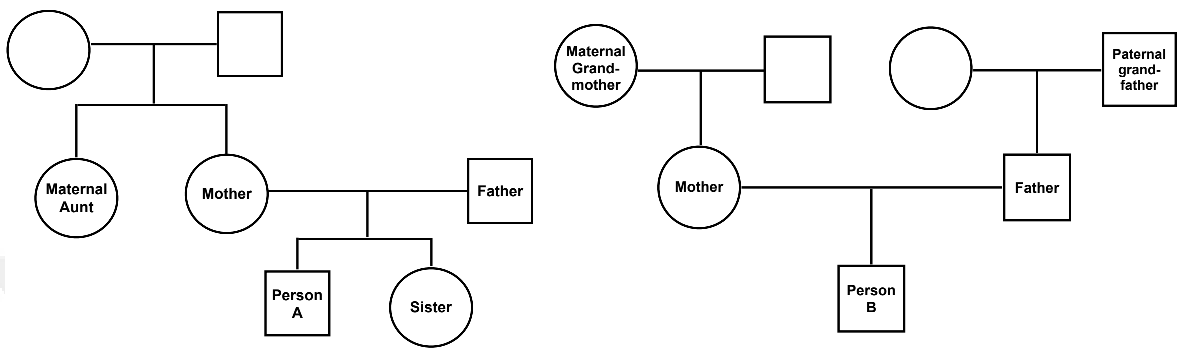
Kinship Inference from Public Genomic Databases and its Countermeasures

3.1 Datasets

3.1.1 OpenSNP Data

To infer family B and to calculate outlier pairwise count, we used 23andme data publicly available at the OpenSNP database (downloaded in March 2015). Individual identities are anonymized. Files with sizes less than 15 MB are eliminated, as the genomic data were limited. In total, 1000 individuals SNP data is available. To obtain reference and alternate allele information, each file is converted to VCF format by PLINK tool [17]. Reference SNP ids (rs), chromosome, position and genotype information are extracted from every VCF file. The genotype information are represented with 0, 1, or 2, the count of alternate alleles. We used the genomic positions that are not missing in all individuals in our analysis.

3.1.2 Families



(a) Family tree of family f_A **(b)** Family tree of family f_B

Figure 3.1: Two family datasets. (a) Family f_A consists of Person A, his father, mother and maternal aunt. **(b)** Family f_B consists of Person B, his mother, father, maternal grandmother and paternal grandfather. No genotype information is available for people denoted with empty squares or circles.

There are two family datasets that we used to test our method; we will refer them as f_A, and f_B. The genomic data of f_A members are publicly shared on a personal website by Person A [29]. The family consists of Person A, his mother, father maternal aunt and his sister; the pedigree is provided in Figure 3.1 A. The second family f_B (see Figure 3.1) is inferred from OpenSNP data via the inference methodology described in Section 3.2. Based on the pairwise kinship coefficients of family members, we set out to infer the pedigree. One possibility is that the family comprises Person B, the mother, the father, the maternal grandmother, and the paternal grandfather. Another possible family structure contains Person B, the mother, the father, the maternal aunt and the paternal uncle. This ambiguity stems from the similar kinship coefficient interval between parent-offspring and siblings in Table 3.1. We assume the first possibility holds and the family contains Person B, his mother, father, maternal grandmother and his paternal grandfather.

3.2 Motivational Attack

The objective of the adversary is to infer the relatives on anonymized genetic databases without any background knowledge about existing families in the database. We assume attacker knows all shared genotype data in the database and he is able to calculate the genotype similarity matrix. Clustering techniques are used to group data points which are more similar to each other. Likewise, people who are genetically closer or more related can be grouped together via clustering methods. Here, attacker applies hierarchical clustering where each person is a cluster himself in the beginning. By identifying the closest two members and combining them into one cluster, relatives are detected. Repeating this process till everybody is in a single cluster discovers relationships among the database members as a hierarchy.

The dendrogram in Figure A.1 shows genetic affinity of 1000 OpenSNP users that is obtained by applying hierarchical clustering. The distance between two individuals or clusters is calculated using the genetic dissimilarity defined by Weir and Zheng [30]. The linkage criteria is selected as average linkage method. The left axis denotes the height of the tree and the right axis denotes the kinship coefficient. We analyzed the members who are clustered together at higher points in the tree than the majority. 53 clusters are detected which consist of people who belong to same families. Among these clusters, one cluster has 5 members, two clusters have 4 members, 6 clusters have 3 members and 44 clusters have 2 members. Figure 3.2 shows a small fraction of the dendrogram in which the data points representing two families are encircled. The family in cluster c1 consists of two members; whereas the family in cluster c2, which will be referred to as f_B in the rest of this paper, consists of five members.

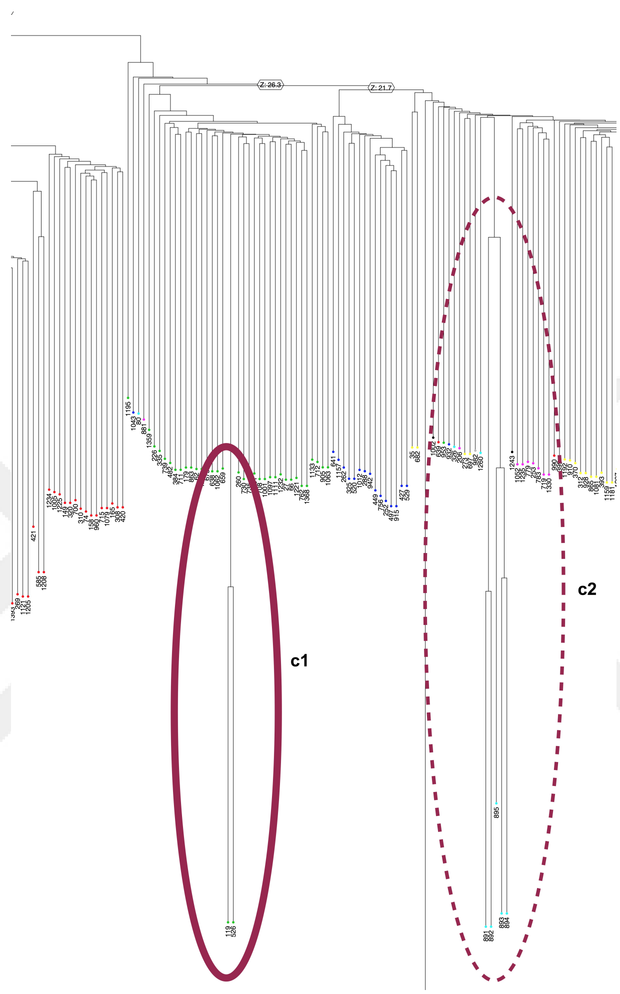


Figure 3.2: Two families that are found in the OpenSNP shown as clusters in dendrogram. A part of dendrogram is shown. The circled clusters denote the families c1 and c2. The cluster c1 consists of two members. The cluster c2 which is dashed circled represents a family with five member; f_B.

3.3 Routes Kinship Privacy Can Leak

We observe that familial relationships can be leaked through two different ways. In the following two subsections, we detail these leakage routes.

3.3.1 Privacy Leakage due to Genotype Similarity

The genomes pertaining to the members of a given family resemble each other more than the similarities observed among unrelated individuals. Therefore, the relatedness of two individuals can be inferred based on their genotype similarity. In this work, we use the metric defined by *Manichaikul et al.* [18] which is a robust estimator of kinship. In this metric the kinship between two individuals i and j is defined as followed:

$$\phi_{ij} = \frac{2n_{11} - 4(n_{02} + n_{20}) - n_{*1} + n_{1*}}{4n_{1*}} \tag{3.1}$$

Here, n_{11} is the number of genomic positions that are heterozygous in both individuals, n_{02} is the number of SNPs where the first individual i is homozygous dominant and the second individual j is homozygous recessive while n_{20} denotes the positions where j is homozygous dominant and i is homozygous recessive. n_1 and n_1 are the number of SNPs that are heterozygous for individual i and for individual j respectively. Without loss of generality, the i-th individual is assumed to have lower heterozygosity than the j-th individual that is $n_{1*} < n_{*1}$. Relationship inference criteria based on kinship coefficient is provided in Table 3.1.

Relationship	**Kinship interval**
Monozygotic twin	$0.353 < \phi < 0.500$
Parent-offspring	$0.170 < \phi < 0.353$
Full sibling	$0.176 < \phi < 0.353$
2nd degree	$0.088 < \phi < 0.176$
3rd degree	$0.044 < \phi < 0.088$
Unrelated	$\phi < 0.044$

Table 3.1: Kinship values and corresponding relationship degrees.

3.3.2 Privacy Leakage due to Outlier Allele Pair Counts

Our methodology involves hiding of genotype positions systematically to filter out relevant information to prevent discovering genomic similarity of family members. For example, positions wherein the two individuals are found to be heterozygous are frequently hidden in the database as it decreases the kinship between family members effectively. However, this alone can cause a privacy leakage. As we add new family members, the number of positions where the two family members are heterozygous will be very small. Simply comparing this number to the population, one could infer that the two individuals are indeed in the same family. We should also note that such a privacy leak can exist even without hiding any positions. Consider a parent-offspring relationship; positions where one person has two alternate alleles and the other has two reference alleles are not possible due to Mendels law, unless there is a mutation in position or an experimental artifact. Therefore, simply checking the number of such positions among individuals can reveal a parent-off spring relationship. We refer this privacy leakage due to pairwise allele outliers. To prevent such an outlier our model will take into account the number of counts in the databases for allele pairs and constrain the optimization models such that for each allele type family members' pairwise counts do not decrease such that they arise as outliers.

We sampled 1000 random individuals from the openSNP database data, for each $[s_i, s_j]$, we record the minimum counts observed between any two unrelated individuals. We will require that at any point in the database, the count of allele pairs are brought down to this recorded minimum threshold; thereby, ensuring that the two individuals count resemble to the rest of the population. The boxplot of genotype counts is shown in Figure 3.3. The outlier for different allele combinations are: $o_{10} = 27454$, $o_{11} = 27300$, $o_{12} = 15019$.

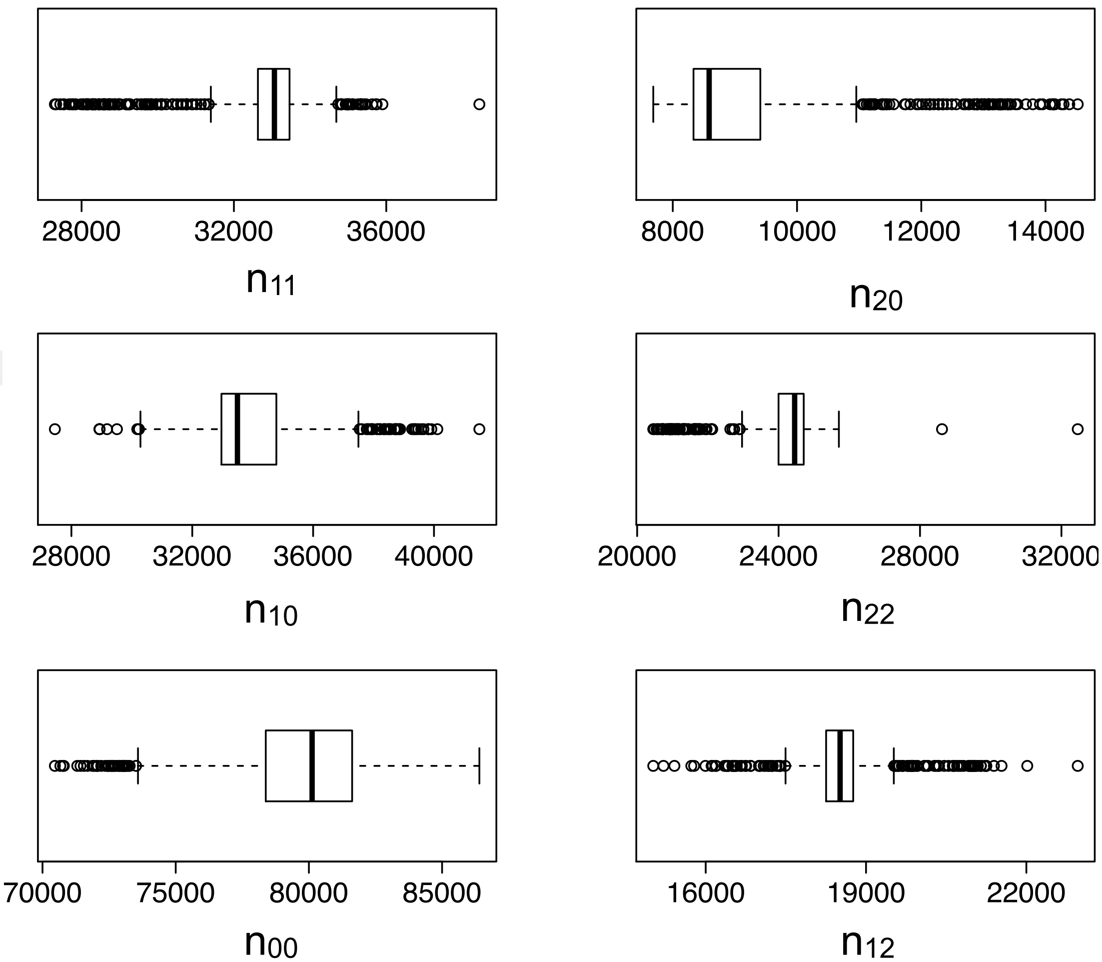


Figure 3.3: Number of different allele pairs in the population. $n_{s_is_j}$ is the number of genotype pairs where one individual has genotype s_i and the other individual has genotype s_j, respectively.

3.4 Protecting Kinship Privacy

We assume that individuals are added to the database sequentially. We further assume that the kinship privacy of the individuals who are already in the database is already protected. Our methodology comprises three steps. Upon the arrival of an individual at the database, we first check whether there is any kinship privacy risk associated with the addition of this individual's genome to the database. The model first infers if there is a family member already present in the database. If the individual does not have a relative, her genome can be safely added. If the

Table 3.2: Notation table.

s_i	SNP type of i_{th} user has where $s_i \in \{0, 1, 2\}$. If s_i is denoted with $*$ in any person, it covers all the SNP types: $* = \{0,1,2\}$
n_{s_i}	The number of genomic positions with a particular SNP configuration of the person i. For more than one person, it is a state vector to refer to the size of genomic positions with a particular SNP configuration of the family members i.e. for a three-membered family n_{101} indicates that the latest arrived member SNP type is 1, the second arrived member's is 0 and the first arrived family members is 1.
$x_{s_1 s_2 s_3 \ldots s_{\|f\|}}$	The number of positions that will be hidden with a particular SNP state sequence. For example, for a three-membered family $x_{1*1} = \{x_{101}, x_{111}, x_{121}\}$ is the state vector where $s_i = 1$, $s_j = \{0, 1, 2\}$ and $s_k = 1$
$o_{s_i s_j}$	Outlier lower bound value in the population. If a pair's $n_{s_i s_j}$ is below $o_{s_i s_j}$ number, they are likely to be outliers. i.e, o_{00} denotes the threshold value for [0,0] SNP pair.
ϕ_{ij}	Kinship value between person i and j
Φ	The maximum kinship value between any two people in a specific family.
f_k	k_{th} family in the database.
U	Utility value.

person does have a relative already in the database, in the ensuing step, the family structures in the database are updated accordingly. At a given time, assume that there are m families currently in the database and an individual i arrives with genotype g_i. If the person has at least one relative in the database and this can be inferred reliably, then the family structures can be reorganized in three different ways:

1. If user i has at least one relative in family f_k, then individual i is added to family f_k.

2. If individual i is identified as a kin of individual j who is not a member of any of the m families in the database, then a new family f_{m+1} is instantiated with members i and j.

3. If user i has relatives in both f_k and f_l families and they are not connected;

arrival of i will combine f_k and f_l families into a single family. This can arise in cases when the maternal family and the paternal family are added before an individual that combines the two sides.

Once the family of i is located and the family structures are updated, the new genotype of i will be added to the database in a privacy-preserving manner with the techniques which we shall detail in Section 3.4.2. Certain parts of g_i will be systematically masked and will not be visible to the outsiders. We denote this partially shared genome with g'_i. This overall process is illustrated in Figure 3.4.

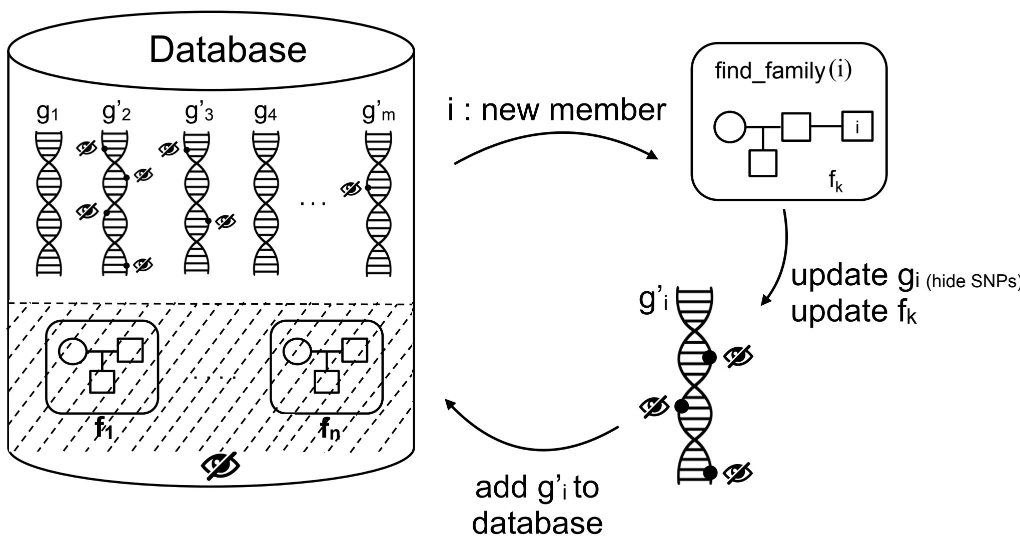


Figure 3.4: Overview of database addition. When a new person i with genotype g_i is arrived the database, i's relatives in the database are checked. The privacy of the family is protected by hiding a portion of g_i. The genotype of person i is now partially shared and denoted by g'_i.

The critical part is to protect privacy without impairing the utility of data sharing. Thus, we would like to maximize the amount of shared genomic data among the stored individuals. In Section 3.4.1 to motivate our optimization models, we will describe a Naïve approach for hiding the genomic positions and later contrast it with our privacy preserving and utility maximizing approach.

3.4.1 Naïve Approach

A greedy naïve approach can be constructed based on the minimization of the pairwise kinship coefficient of the individual i and its family members f_k . The KING kinship estimate shown in (3.4.1) indicates that decreasing the number of positions where two individuals genomes are heterozygotes, n_{11} decreases the kinship coefficient between the two individuals. B can be solved analytically to find the minimum number of genomic positions to hide in individual i so that the kinship coefficient reduces to zero. Once this number, which we refer to as x, is calculated that many random positions in individual i's genome can be hidden. This approach is summarized in Algorithm 1.

Algorithm 1 A Greedy Approach for Protecting Family Relationship

Inputs: new user i , family f_k
For all $j \in f_k$
    If $\phi_{ij} \geq 0$
      $x \leftarrow$ CALCULATENUMOFPOSITIONSTOHIDE(i,j)
      $X \leftarrow$ select x random genomic positions from the genotype positions of type Aa, Aa
      Remove X from person i

function CALCULATENUMOFPOSITIONSTOHIDE(i, j)
    $x \leftarrow (2n_{11} - 4n_{20} - n_{*1} + n_{1*}(1-4\phi))/(2(1-2\phi))$
    return x
end function

The greedy approach is mypophic and will lead to privacy leakages. We illustrate this with an example. Figure 3.5 illustrates the stepwise results of naïve approach on family f_A that consists of a maternal grandmother, a paternal grandfather, a mother, a father and a child. In each step, the number positions to hide is computed and the genomic data of the individual is updated by hiding that many random positions. Assume the paternal grandfather arrives at the database first, at this step his genome is added to the database witholding any positions as no relatives are present in the database. Secondly, the child arrives, 24 K heterozygotic positions should be hidden in order to decrease $\phi_{c,u}$, the kinship estimate of the paternal grandfather and the child to zero. As illustrated with green in the

Figure 3.5 the relationship is protected. In the subsequent step, the mother arrives at the database, the child and the mother can be added successfully when 26 K genome positions are withheld from the mother. At this point, family's privacy is protected. However, in step four, wherein the father arrives at the database, the privacy of the family is immediately impaired if his genomic information is to be shared without truncation. 22 K positions need to be removed from the father to conceal the relationship with the grand father. At the same time, to set $\phi_{f,c}$ to zero, 7 additional genomic positions need to be removed. However, the kinship coefficients still cannot be reduced. At best, the relationship between the father and child is hidden as if they are 2nd degree relatives and the father and grand parent are third degree relatives. In the fourth step, when the maternal grandmother arrives to the database, 15K positions are hidden from the mother, although this relationship is protected, the father child relationship is revealed. This is because the kinship is calculated over positions that are not missing in all the people in the database.

There are several weaknesses of this greedy approach:

1. Genomic positions of the newly added family member i is hidden. The number of positions are decided based on the pairwise relationship of i and with the other members. For example, in a three-membered family only relatedness between (i, j) and (i, k) are considered. Random selection does not consider the relationships among the other family members such as (j, k). This is the reason why in step 4 of Figure 3.5, the father and the child's relationship is removed.

2. The naïve approach decided the number of positions in a greedy fashion leading a large part of genomic data hidden, impeding the utility of sharing the data. When all family members are jointly considered, we can reduce the number of positions that are hidden.

3. This approach does not pay attention to the privacy leaks based on pairwise allele type counts.

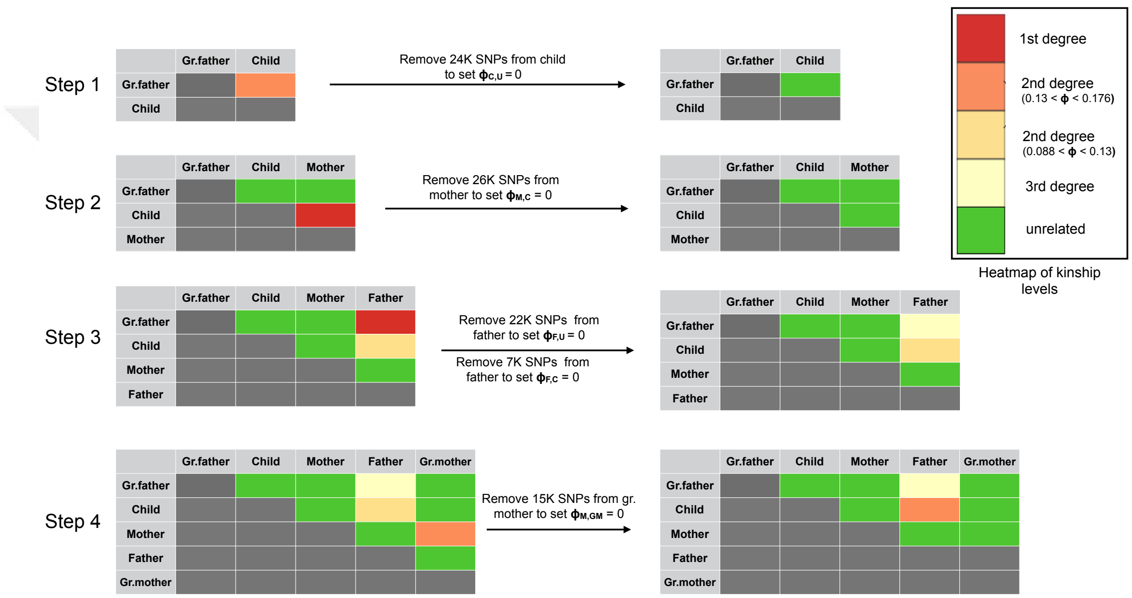


Figure 3.5: Stepwise illustration of the naïve approach on an example family. The family f_A is comprised of paternal grandfather, child, mother, father and maternal grandfather which arrives to the database in sequential time order. Pairwise relationships that can be inferred based on the shared genomic data and the kinship coefficient are represented with color before and after for arrival of the newcomer. Green denotes safe kinship values; i.e. unrelated members, beige represents 3rd degree relatives, i.e. cousins. Since second degree relatives have high kinship interval $0.088 < \phi < 0.176$, we split the colors into wheat yellow and orange at 0.13 kinship value. The former one denotes that kinship is greater than 0.13 and the latter means kinship smaller than 0.13. First degree members, i.e. parent-offspring relatives are illustrated with red. The naïve approach greedily hides the position in the new arriving member for the relationships where the relation is revealed.

We have developed a new methodology described in 3.4.2, to overcome the deficiencies of the naïve approach.

3.4.2 A Utility Maximizing Privacy Preserving Approach

A good solution should maximize the genomic data to be shared while minimizing the privacy risks associated with kinship among stored family members. We consider the two types of privacy risks described in Section (3.3.1) and (3.3.2). We model this problem as an optimization problem where the objective function is the number of positions to be witheld subject to the constraints that enforce privacy protection in terms of kinship and outlier allele counts. As outlined in the previous section, our methodology assumes sequential arrival of the family members and the genome of the newly added family member is protected by hiding portions of her genome. In order to maximize the data shared and to protect the privacy, we should take into account the SNP configuration of the family and select the positions based on these configurations. Before getting into the details of the model, we introduce a notation to describe the methodology.

We introduce a state vector, $\mathbf{s}$, to describe a particular SNP configuration for the family. Let $\mathbf{s} = s_m \ldots s_2 s_1$, represent the SNP configuration of the family based on the reverse chronological order of arrivals at the database, i.e. s_m denotes the SNP state for the latest arriving family member and s_1 denotes the SNP state of the first arriving member configuration and $s_i \in \{0, 1, 2\} \forall i \in \{1, 2, \ldots, m\}$. We will use the state vector to refer to the size of genomic positions with a particular SNP configuration of the m family members. We will denote the number of genomic positions with a particular SNP configuration with $n_{s_m \ldots s_2 s_1}$. For example for a two-member family, n_{10} will indicate that the latest arrived member SNP type is 1 where as the first arrived family members is 0. We use a star notation to denote any type of SNP in a particular person's genome. For instance, n_{1*} indicates the number of positions where the latest arrived person genome is 1 and the first-comer SNP can be of any type, 0, 1 or 2. Similarly, we will denote the number of positions that will be hidden with a particular SNP state sequence with $x_{s_m \ldots s_2 s_1}$.

To evaluate the solutions utility, we measure the utility of shared data for the first m incoming members over a M-membered family retrospectively as follows:

$$U = \frac{V * m - x}{V * M}, \tag{3.2}$$

x is the number of positions hidden in the family and can be written as a sum over all possible SNP state sequences for family: $x = \sum_{\mathbf{s} \in S} x_{\mathbf{s}}$, where $\mathbf{s} = \{s_t \in [0, 1, 2] | t = m, \ldots, 2, 1\}$ and S is the set of all possible state sequences. Here, V is the size of the set of genomic positions that are non-missing in all members. The denominator represents the total number of genomic positions shared for all family members if no genomic positions were hidden. The nominator represents the number of positions shared after hiding positions with each SNP configuration for the arrived family members. Thus, a utility value one means that all the data members are stored in the family, and all of their genomic positions are shared, wheres a value of zero indicates that no data is shared. Figure 3.6 illustrates how the utility score calculated.

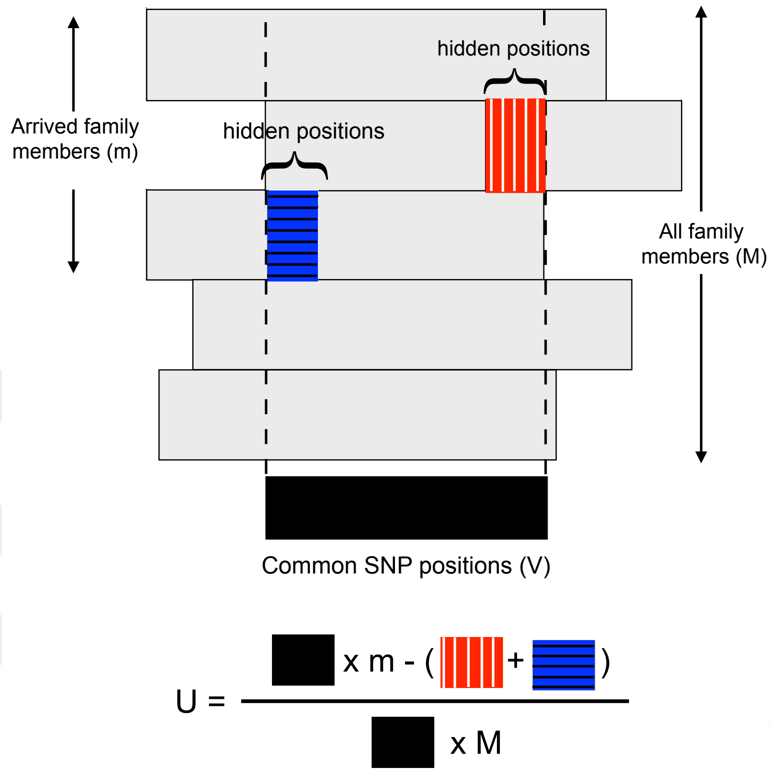


Figure 3.6: An illustration showing the calculation of the utility function. The family contains M members, m of which has arrived at the database. Addition of the first incoming member do not require to hide any SNPs. For the other arrived member, certain part of the genomes are masked that are shown as bars with vertical and horizontal lines. The SNP positions that are common in every family member is represented as a black bar; the size of which is V. The formula shows how the utility is calculated based on these numbers.

Notice that maximizing utility function (U) is equivalent to minimizing the sum of number of positions hidden with all possible SNP configuration, the term $\sum_{\mathbf{s} \in S} x_{\mathbf{s}}$. Moreover, due to the nature of the kinship estimates not all type of family SNP state sequences in S will be hidden. For example for a family of two, only hiding positions where both members SNPs are 1 will decrease the kinship estimate. Thus, among $x_{\mathbf{s}} \in S$ only x_{11} will be non-zero. From onwards, we outline the model for a three-member family; however, the formulation can be straightforwardly generalized to handle larger families. In the results section, we solve this problem for two families with five members each.

Consider a family f, whose members are the individuals i, j, k, and they arrive at times $t{+}2$, $t{+}1$, and t, respectively. The first incoming family member k has no relatives in the database, thus her genomic data, g_k, is shared without truncation. When the second family member j arrives, to conceal the relationship between j and k, certain parts of individual j's genome will be witheld. Because the kinship coefficient decreases only when n_{11} decreases, we will hide the positions of the genome, where $s_k = 1$ and $s_j = 1$. After hiding x_{11} positions, the new KING estimate, ϕ'_{jk}, will be:

$$\phi'_{jk} = \frac{2(n_{11} - x_{11}) - 4(n_{02} + n_{20}) - (n_{1*} - x_{11}) + (n_{*1} - x_{11})}{4(n_{*1} - x_{11})}$$

We can solve the equation for x_{11}:

$$x_{11} = \frac{2n_{11} - 4(n_{02} + n_{20}) - n_{1*} + n_{*1}(1 - 4\phi'_{jk})}{2(1 - 2\phi'_{jk})}. \tag{3.3}$$

Simply plugging in zero ϕ'_{jk} will give the sufficient number of genomic positions to be hidden in individual j. At this stage, model checks whether the outlier constraints are violated when certain positions are hidden. If that is the case, the database owner is alerted and the individual j is not added to the database. If no outlier constraint is violated, x_{11} number of positions are selected from the set of SNPs with 11 configuration and hidden. Finally, this protected version of the genome, g'_j, is stored in the database.

When the third individual, i, arrives at the database, the goal is to share i-th individual's genome without compromising the privacy of the entire family f, given that genomes g'_j and g_k are already in the database. To hide the relationship between i and j, we will need to genomic positions where $s_i = 1$ and $s_j = 1$, and there is no restriction on the third individual genotype. Similarly, to hide the relationship between i and k, we will need to remove certain number of positions, where the first and the last members SNPs are 1 and the middle comer can be of any SNP type. Thus, the number of positions should be selected from the set of SNPs such that where the latest family members SNP position is 1 and at least one of the two other members' SNP type is 1 to denote such or notations. we will use thi number, $x_{1\bullet\bullet}$, where it can be decomposed into five numbers $x_{1\bullet\bullet} = x_{110} + x_{111} + x_{112} + x_{101} + x_{121}$. Thus, to maximize the utility we would need to minimize $x_{1\bullet\bullet}$. We generate privacy constraints and outlier constraints in the following parts. The constraints are generated by the assumption of all the members are related with each other in family f, but if there are some members that are not blood-related, i.e. maternal aunt and paternal aunt, no privacy constraint need to be added for these pairs.

3.4.3 Constraints to Prevent Privacy Leakage due to Genomic Similarity

Our objective is to find a minimum size number of positions to hide where the following kinship constraints are satisfied for a specific Φ threshold level. In general, $\Phi = 0$ is individuals that are not in the same family. For all the relations between (i, j), (i, k), and (j, k) pairs, we will describe how our. In the equations below, we use $x_{1\bullet\bullet}$ for the positions that are removed from person i.

Let ϕ'_{ij} denote the new kinship estimate attained after hiding positions number of where i and j are both heterozygote, $x_{11*} = x_{110} + x_{111} + x_{112}$:

$$\phi'_{ij} = \frac{2(n_{11*}-x_{11*})-4(n_{20*}+n_{02*})-(n_{1**}-x_{1\bullet\bullet})+(n_{*1*}-x_{11*})}{4(n_{*1*}-x_{11*})}, \tag{3.4}$$

where $n_{*1*} < n_{1**}$ and $x_{1\bullet\bullet} = x_{110}+x_{111}+x_{112}+x_{101}+x_{121}$. If we require, this kinship estimate to be bounded with a preset kinship Φ, $\phi'_{ij} \leq \Phi$, the following inequality constraint can be derived:

$$2n_{11*}-4(n_{02*}+n_{20*})+(1-4\Phi)n_{*1*}-n_{1**} \leq (2-4\Phi)x_{11*}-x_{101}-x_{121} \tag{3.5}$$

Similarly, we derive two inequality constraints between i and k individuals and j and k individuals: ϕ'_{ij} denote the new kinship estimate attained after hiding positions number of where i and j are both heterozygote. This number is denoted with x_{11*}, where $x_{11*} = x_{110} + x_{111} + x_{112}$:

$$2n_{1*1} - 4(n_{2*0} + n_{0*2}) - n_{1**} + (1-4\Phi)n_{**1} \leq (2-4\Phi)x_{1*1} - x_{110} - x_{112} \tag{3.6}$$

Removal of $\{x_{110}, x_{111}, x_{112}, x_{101}, x_{121}\}$ alters ϕ_{jk}. Given that $\phi'_{jk} \leq \Phi$, the following inequality constraint can be derived:

$$2n_{*11} - 4(n_{*02} + n_{*20}) - n_{**1} + (1-4\Phi)n_{*1*} \leq (1-4\Phi)x_{11*} + 2x_{111} - x_{1*1}, \tag{3.7}$$

where $n_{*1*} < n_{**1}$.

These three constraints, if satisfied concurrently, will guarantee that the kinship estimates are Φ for all pairwise relationships.

3.4.4 Constraints to Prevent Privacy Leakage due to Pairwise Allele Outlier Values

As mentioned in Section 3.3.2, relationships can be revealed in the database by probing the pairwise allele counts in the population. Hiding positions from one of the family members decreases her pairwise allele counts with other family members and if they are too low, simply this count can reveal the relationship. We set a threshold value for a number to be outlier for each type of o_{10}, o_{11}, o_{12}. They are set as the minimum number of allele pair counts among the unrelated individuals. We define the outlier constraints that are enforced after hiding the positions so that the pairwise counts do not fall below these set threshold values. For example, when x_{110} the latest two arrived members SNPs are 1, and the first comer is 1. The outlier constraints will be defined as follows:

$$0 \leq o_{11} \leq n_{11*} - x_{110}$$
$$0 \leq o_{10} \leq n_{1*0} - x_{110}$$
$$0 \leq o_{10} \leq n_{*10} - x_{110}$$

There is also the trivial constraint of that the number of positions to hide in a certain SNP configuration cannot hide such positions. $0 \leq x_{110} \leq n_{110}$. We can rewrite these constraints in a more compact from as follows:

$$0 \leq x_{110} \leq u_{110}, \tag{3.8}$$

where $u_{110} = \min(n_{110}, (n_{11*} - o_{11}), (n_{1*0} - o_{10}), (n_{*10} - o_{10}))$. Similar constraints are derived for the other variables:

$$0 \leq x_{111} \leq u_{111}, \tag{3.9}$$

where $u_{111} = \min(n_{111}, (n_{11*} - o_{11}), (n_{1*1} - o_{11}), (n_{*11} - o_{11}))$.

$$0 \leq x_{112} \leq u_{112}, \tag{3.10}$$

where $u_{112} = \min(n_{112}, (n_{11*} - o_{12}), (n_{1*2} - o_{12}), (n_{*12} - o_{12}))$.

$$0 \leq x_{101} \leq u_{101}, \tag{3.11}$$

where $u_{101} = \min(n_{101}, (n_{10*} - o_{10}), (n_{1*1} - o_{11}), (n_{*01} - o_{10}))$.

$$0 \leq x_{121} \leq u_{121}, \tag{3.12}$$

where $u_{121} = \min(n_{121}, (n_{12*} - o_{12}), (n_{1*1} - o_{11}), (n_{*21} - o_{12}))$.

3.4.4.1 Optimization Model

Subject to the constraints defined in the previous two sections, finding the number of positions to hide in each different position type can be cast as a linear integer optimization problem.

$$
\begin{aligned}
\min \quad & x_{101}+x_{111}+x_{121}+x_{110}+x_{112} \\
\text{s.t.} \quad & \\
& 2n_{11*}-4(n_{02*}+n_{20*})-n_{1**}+(1-4\Phi)n_{*1*}\leq(2-4\Phi)x_{11*}-x_{101}-x_{121} \\
& 2n_{1*1}-4(n_{2*0}+n_{0*2})-n_{1**}+(1-4\Phi)n_{**1}\leq(2-4\Phi)x_{1*1}-x_{110}-x_{112} \\
& 2n_{*11}-4(n_{*02}+n_{*20})-n_{**1}+(1-4\Phi)n_{*1*}\leq(1-4\Phi)x_{11*}+2x_{111}-x_{1*1} \\
& 0\leq x_{110}\leq u_{110} \\
& 0\leq x_{111}\leq u_{111} \\
& 0\leq x_{112}\leq u_{112} \\
& 0\leq x_{101}\leq u_{101} \\
& 0\leq x_{121}\leq u_{121} \\
& x_{11*}=x_{111}+x_{110}+x_{112} \\
& x_{1*1}=x_{111}+x_{101}+x_{121} \\
& x_{101},x_{111},x_{121},x_{110},x_{112}\in Z_{\geq 0}
\end{aligned}
\tag{3.13}
$$

Kinship constraints for every pair of family members are written in the first three constraints. The remaining constraints represent the outlier constraints for every possible allele pair type that are removed.

As described in the results sections, in cases where this optimization problem does not have feasible solutions for the family of interest, we propose to relax the constraints. To find a feasible solution, we suggest relaxing one type of privacy constraints; the kinship or outlier constraints. The database owner can decide this. In this scenario one type of constraints is strictly satisfied whereas the other type can be relaxed. This procedure is explained in Figure 3.7.

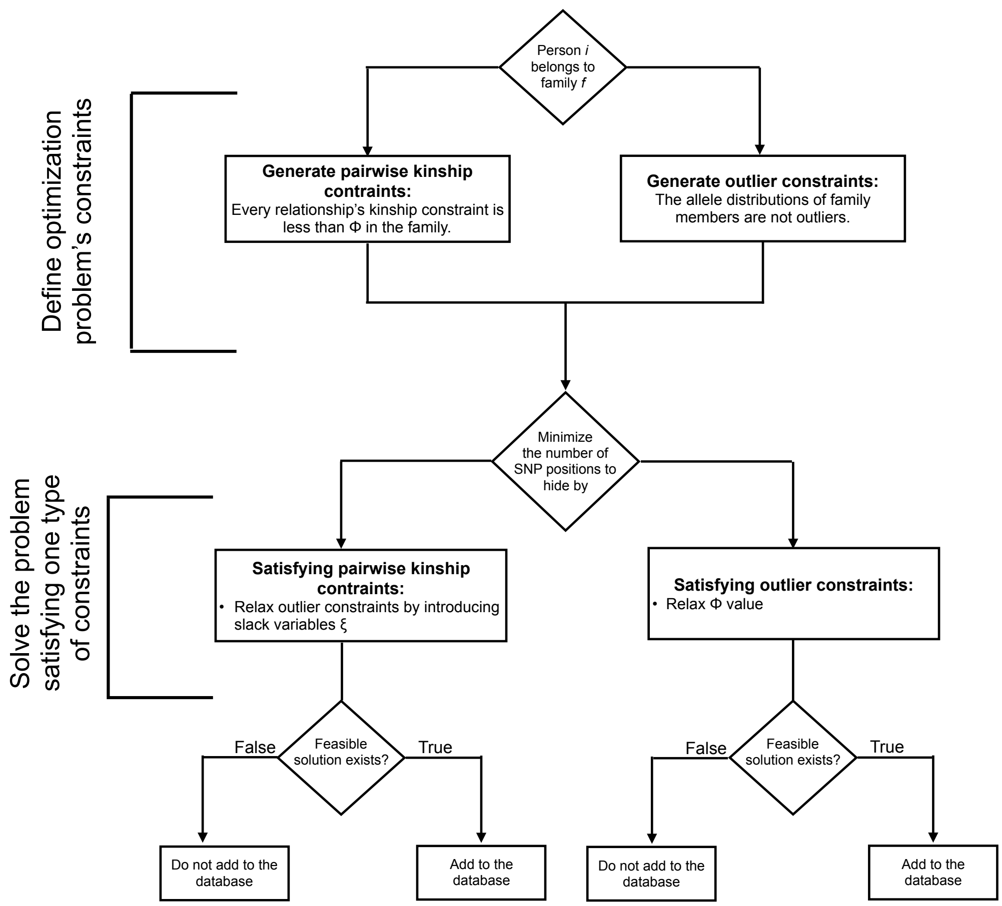


Figure 3.7: Protecting the privacy of a family when a new member i arrives at the database. There are two different ways to solve the problem: one is based on the relaxation of outlier constraints and the other is based on the relaxation of kinship constraints. For each method, if there is a feasible solution, person i can be added to the database safely.

3.4.5 Solution by Satisfying Kinship Constraints / Relaxing Outlier Constraints

If the problem is not feasible with all the original constraints that are strictly satisfied, one might seek nearby solutions that minimally sacrifice strict privacy by relaxing the outlier constraints. Thus, the returned solution will not be strictly below the set outlier threshold values, but we will ensure that these values shall deviate as small as possible from the original set outlier values. We achieve this by introducing slack variables for every type of allele pairs. The outlier constraints given in section 3.3.1 are relaxed as follows:

$$
\begin{aligned}
u_{110} &= \min\left(n_{110}, (n_{11*} - (o_{11} - \epsilon_1)), (n_{1*0} - (o_{10} - \epsilon_2)), (n_{*10} - (o_{10} - \epsilon_2))\right)\\
u_{111} &= \min\left(n_{111}, (n_{11*} - (o_{11} - \epsilon_1)), (n_{1*1} - (o_{11} - \epsilon_1)), (n_{*11} - (o_{11} - \epsilon_1))\right)\\
u_{112} &= \min\left(n_{112}, (n_{11*} - (o_{11} - \epsilon_1)), (n_{1*2} - (o_{12} - \epsilon_3)), (n_{*12} - (o_{12} - \epsilon_3))\right)\\
u_{101} &= \min\left(n_{101}, (n_{10*} - (o_{10} - \epsilon_2)), (n_{1*1} - (o_{11} - \epsilon_1)), (n_{*01} - o_{10} - \epsilon_2)\right)\\
u_{121} &= \min\left(n_{121}, (n_{12*} - (o_{12} - \epsilon_3)), (n_{1*1} - (o_{11} - \epsilon_1)), (n_{*21} - o_{12} - \epsilon_3)\right)
\end{aligned}
\tag{3.14}
$$

In the above inequalities, $\epsilon_1 >= 0$, $\epsilon_2 >= 0$, and $\epsilon_3 >= 0$ are slack variables that controls the relaxation of imposed constraints. We modify the optimization problem with these new constraints. Before solving the original optimization problem, we solve the optimization problem with the same constraints wherein the objective is to minimize $\epsilon_1 + \epsilon_2 + \epsilon_3$. Having found the minimum values of these variables, we plug in these values to obtain the relaxed outlier constraints and solve the original optimization problem where the aim is to minimize $x_{101} + x_{111} + x_{121} + x_{110} + x_{112}$. This integer linear programming problem can be solved optimally. We used IBM ILOG CPLEX Optimizer [31] in this work.

3.4.6 Solution by Satisfying Outlier Constraints / Relaxing Kinship Constraints

We can also solve the same problem where the outlier constraints are satisfied but the Φ, the maximum kinship estimate value allowed among family members, is not required to be 0 and but forced to deviate as small as possible from 0. For example, if outlier constraints are to be strictly satisfied, close family members can be shown as second or third degree relatives as opposed to requiring them to be unrelated. To solve this optimization problem, we should find the minimum Φ value that satisfies the constraints through a nonlinear optimization problem stated as follows:

$$\begin{aligned}
&\min \quad \Phi \\
&\text{s.t.} \\
&\quad 2n_{11*}-4(n_{02*}+n_{20*})-n_{1**}+(1-4\Phi)n_{*1*}\leq(2-4\Phi)x_{11*}-x_{101}-x_{121} \\
&\quad 2n_{1*1}-4(n_{2*0}+n_{0*2})-n_{1**}+(1-4\Phi)n_{**1}\leq(2-4\Phi)x_{1*1}-x_{110}-x_{112} \\
&\quad 2n_{*11}-4(n_{*02}+n_{*20})-n_{**1}+(1-4\Phi)n_{*1*}\leq(1-4\Phi)x_{11*}+2x_{111}-x_{1*1} \\
&\quad \Phi' \leq \Phi \leq 0.5 \\
&\quad \text{and the constraints derived from Section (3.4.4)}
\end{aligned} \tag{3.15}$$

In the above problem Φ is also unknown. We used the genetic algorithm solver under Global Optimization Toolbox in Matlab for nonlinear constrained optimization problems. After finding the optimal value of Φ from the model (3.15), the original optimization problem (3.13) can be solved to find the minimum number of positions to hide.

Chapter 4

Results and Discussion

We evaluate the suggested methodology for protecting kinship privacy on two families. We refer these two families, f_A and f_B, whose details are provided in Section 3.1.2. We present results of solving the problem using the two approaches; first based on strictly satisfying the outlier constraints while relaxing the pairwise kinship constraints and second, satisfying pairwise kinship constrains while relaxing the outlier constraints. We asses each solution by calculating the utility of shared data as described in Section 3.4.2. In the next subsections, we describe the results obtained using these two different approaches.

As the arrival sequence of the family members will lead to different solutions, we considered all possible orders the family members can arrive. Different arrival sequences and the results are displayed in a tree structure; each node represents a person that arrives at that particular time step. The root node displays the first arrived family member and each branch from root to leaf represents a different arrival sequence of the family members. At each branch, we only consider adding family members that are blood-related at a particular node, as other people can be trivially added. The branch halts if no feasible solution exists after adding the last member. In certain cases, the relationship can be brought down to the level of two unrelated person. In solving the case where the outlier constrains are satisfied and the kinship constraints are preserved, we still proceed if the relationship of the individual to the relatives can not be brought down to the level of two unrelated person, but can be brought down by at least one degree, i.e. the protected genomes of a parent, offspring relationship is inferred as a second degree relation.

In Section 4.2, we solve the optimization model by relaxing outlier constraints and satisfying kinship constraints. The new outlier values that are obtained from the solution of the optimization model is shown as how many standard deviations are away from the initial outlier values of the population. For example, if the solution denotes that the outlier value o_{10} should be 2.50σ lower to add the new member, the solution is written as $o_{10} - 2.50\sigma$. The standard deviation scores of the population for each $10, 11, 12$ allele pair counts is given in Table 4.1.

	n_{10}	n_{11}	n_{12}
σ	1432	1581	743

Table 4.1: Standard deviation scores found in the population. The standard deviation scores are calculated using the number of genomic positions with the particular SNP configurations $10, 11, 12$ of 1000 members.

4.1 Results by Solving the Optimization Problem by Satisfying Outlier Constraints

This section provides the results for the solution of the optimization model by relaxing the kinship constraints and satisfying outlier constraints on two different family datasets; f_A and f_B.

4.1.1 Results on Family A

4.1.1.1 Arrival Sequences where Person A Arrives First

Figure 4.1 displays the results for family A when Person A arrives the first and the solution is obtained by satisfying outlier constraints while relaxing the kinship constraints. The case that Person A arrives first is the most difficult case, as he has relationships with every other member in the family.

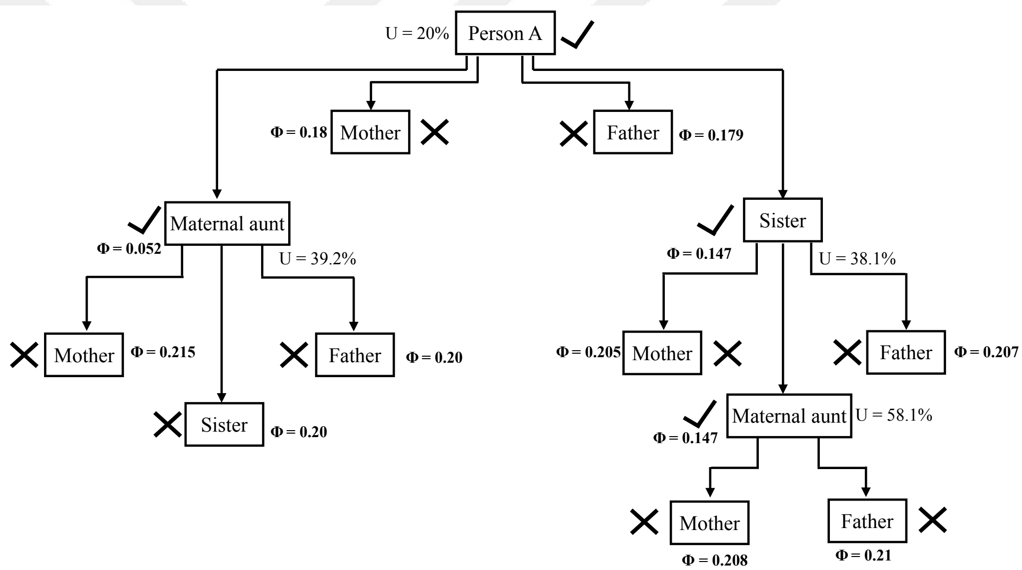


Figure 4.1: Solution for family f_A when the Person A is added first and when outlier constraints are satisfied and the kinship constraints are relaxed. A possible arrival sequence of f_A is shown, when the Person A is the first added member of f_A. Downward arrows point to the subsequent member arrived. Φ indicates the largest kinship estimate among all family members. Check mark next to a node indicates that the individual could be successfully added without compromising the familys privacy. A successful addition means at least one-degree decrease in relationship of the newest member with her relatives is attained. Cross mark indicates that even there is a feasible solution the new Φ value still reveals the relationship. If the family member can be added successfully, utility at that stage is provided at the bottom of the newly added family member's box.

As illustrated in Figure 4.1, we observe that if a parent of Person A arrives in the second step, it is impossible to hide this parent-offspring relationship. The two people are very close, solution requires masking a large portion of the parent. In that case the allele pair counts between the parent and the offspring are too low, violating outlier constraints, meaning this count itself will reveal the relationship. On the other hand, when more distant relatives such as a sister or a maternal aunt arrives at the database as the second person, the kinship coefficient cannot be decreased to the level of two unrelated individuals, but the relationship can be hidden such that they are identified as more distant relatives: i.e. Person A - aunt relationship can be deciphered as a 3rd degree relationship and sister - Person A relation can be decreased to a 2nd degree relative.

In the third level of the tree, the parents of Person A may not be added to the database since there is no feasible solution. Additionally, the sister cannot be added. Thus, first-degree members of Person A are not allowed to be added in the third order as well. At this stage of the tree, when the second added member is the sister, only the maternal aunt can be added as the third member safely. This scenario achieves 56% of utility. In that case Person A and the sister can be inferred as second relatives and the relationships of the maternal aunt to Person A and the sister are not revealed. In the 4th level of the tree, if the arrived member is the father, after him mother can be added to the database with 92.2% utility achievement. Note that this is the only arrival order where person A arrives first and that yields addition of all family members to database.

Successful arrival orders for the case where Person A arrives first:

{Person A - Sister - Maternal Aunt - Father - Mother}
{Person A - Sister - Mother - Maternal Aunt}
{Person A - Sister - Mother - Father}
{Person A - Sister - Father - Mother}
{Person A - Sister - Father - Maternal Aunt}
{Person A - Maternal Aunt - Sister - Father}

4.1.1.2 Arrival Sequences where Sister Arrives First

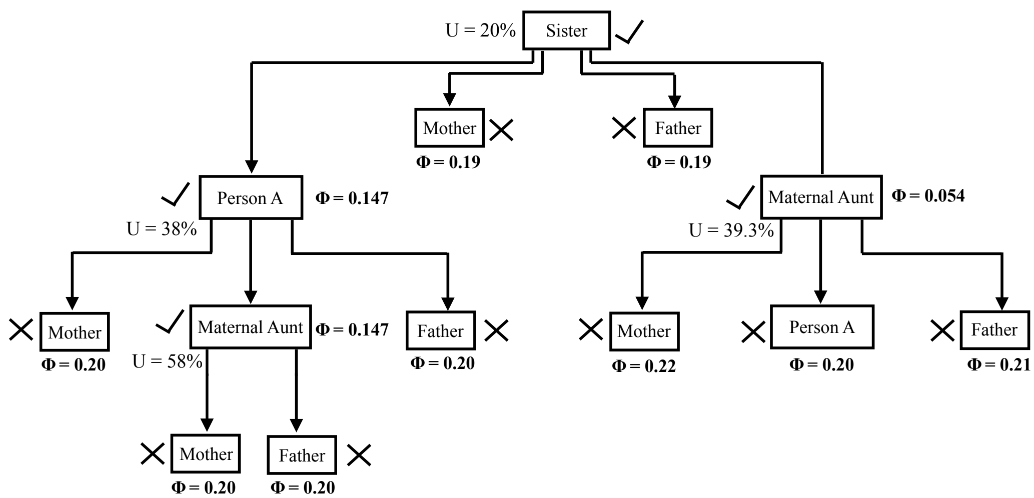


Figure 4.2: The first arrived member in f_A is the sister. The possible arrival sequences of family f_A is shown where the sister is the first arrived member. The optimization model is solved by relaxing kinship privacy constraints and satisfying outlier constraints.

We consider the arrival sequences where the sister arrives first to the database. The results for this set of sequences is displayed in Figure 4.2. As she is the first member of the family in the database, her genome is stored without any withholding. We observe that the parents of the sister cannot be added to the database without revealing the parent-offspring relationship whereas the maternal aunt or Person A can safely be added to he database if she or he arrives the second. If Person A is the second added member, the maximum kinship value between the family members is 0.14. The relationship between Person A and the sister, can be at most displayed as a second degree relative. In this arrival sequence {Sister - Person A}, the only subsequent member that can be added successfully is the maternal aunt. With the addition of maternal aunt, the Φ value in the family does not change, and the kinship value between siblings remains the same. There is no need to remove any SNPs from maternal aunt because she and her nephews are inferred as unrelated individuals due the low kinship values between them. This scenario achieves a maximum utility of 58% in the tree. Following the maternal aunt, father and mother can be added to the database subsequently. This arrival scenario achieves addition of all family members to the database when the sister is the first arrived member and it achieves 92.2% utility. If the data is shared without any tructation from these five people, the utility value would be 100%.

If the second added member is the maternal aunt, the Φ value increases to 0.054, where the sister and the aunt are denoted as third-degree relatives. However, after the maternal aunt, no family members can be added in the third order to the database.

Successful arrival orders for the sequences where the sister arrives first:
{Sister - Person A - Maternal Aunt - Father - Mother}
{Sister - Person A - Mother - Maternal Aunt}
{Sister - Person A - Mother - Father}
{Sister - Person A - Father - Mother}
{Sister - Person A - Father - Maternal Aunt}
{Sister - Maternal Aunt - Person A - Father}

4.1.1.3 Arrival Sequences where Maternal Aunt Arrives First

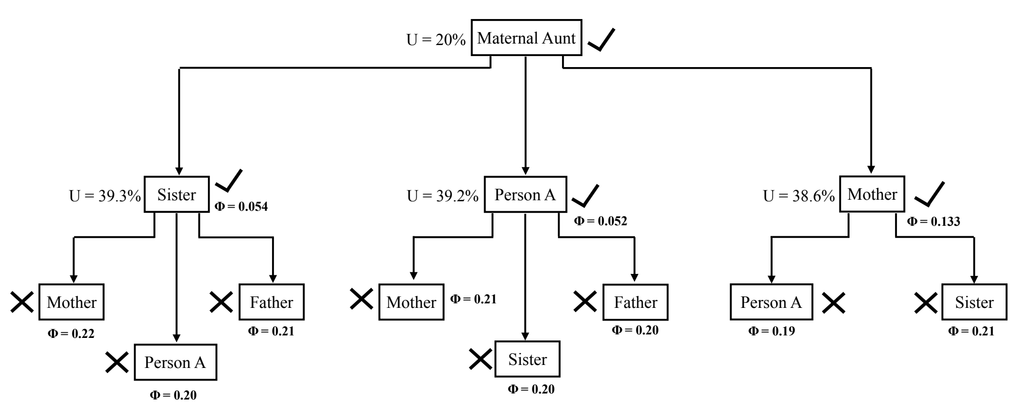


Figure 4.3: The first arrived member in f_A is the maternal aunt. Solutions for arrival sequences for f_A where the maternal aunt arrives first.

The tree of arrival sequences where the maternal aunt is the first comer is shown in Figure 4.3. Maternal aunt's closest relative is the mother. With the addition of the mother, the kinship value between the mother and the maternal aunt decreases up yo 0.133 which denotes the kinship value between second degree relatives. However, after adding these two very close members, adding another family members is impossible.

The second-degree relatives of the maternal aunt i.e. her nephews, can be added safely as the second person achieving approximately 39 % utility in both cases. In these cases, the maximum kinship coefficient among any member of the family will be around only 0.05; this is very close to the maximum kinship value that two unrelated individual can have which is 0.04 (see 3.1). Thus, this arrival sequence is quite favorable. However, in the following step when a third family member arrives, there is no feasible solution.

There are cases where two sequences involve the same people, but the order in these sequences affect the solution significantly. For example, although {Sister - Person A - Maternal Aunt} sequence can be added safely, {Maternal Aunt - Sister - Person A} cannot be added in the database in this order.

Successful arrival orders for the sequences where the maternal aunt arrives first:
{Maternal Aunt - Sister}
{Maternal Aunt - Person A}
{Maternal Aunt - Mother}

4.1.1.4 Arrival Sequences where Mother Arrives First

Figure 4.4 displays the solutions for the arrival sequences where the mother is the first arrived family member. In the left and right hand side of the tree shows the cases where the second arrived members is the offspring of the mother. We observe it is impossible to add a parent and her offspring concurrently in the database without leading a privacy breach. The middle branch in the tree shows that if it is the maternal aunt arrives subsequently, she can safely added to the

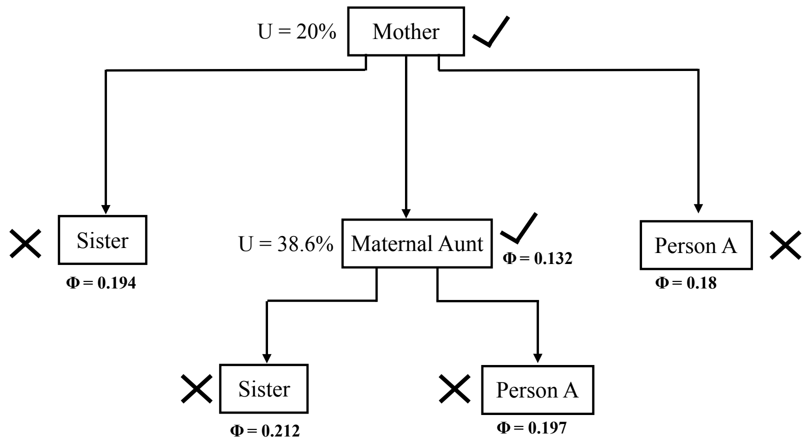


Figure 4.4: The sequence where the first arrived member in f_A is the mother. Solutions for arrival sequences for f_A where the mother arrives first for the case where kinship constraints are relaxed and the outlier constrains are satisfied.

database. However, after her addition, it is not possible to add any other member in the family. The reason behind this is once again is difficulty of preserving privacy in parent-offspring relations as their genomes bear extensive similarity. This situation is observed in the previously presented arrival sequences, a parent and an offspring should not be added concurrently to the database.

Successful arrival orders for the sequences where the mother arrives first:
{Mother - Maternal Aunt}

4.1.1.5 Arrival Sequences where Father Arrives First

Father has two blood-relatives in family f_A; Person A and the sister. Figure 4.5 shows that if children arrive to the database, they cannot be added to the database without a privacy leak. Φ value will be relaxed to 0.193 when the sister is added, and it will relaxed to 0.179 if Person A is the one to arrive. In both cases, parent-offspring relationship does not decrease at least by one degree. Thus successful addition is not possible.

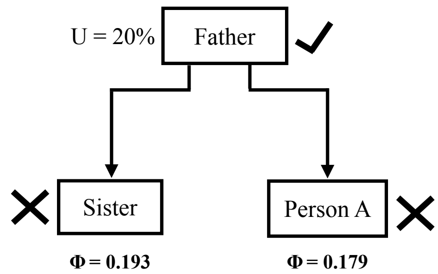


Figure 4.5: The first arrived member in f_A is the father. The possible arrival sequences of family f_A is shown, where it is the father that arrives first. The tree represents solutions obtained by satisfying outlier constraints and relaxing kinship constraints.

4.1.2 Results on Family B

4.1.2.1 Arrival Sequences where Person B Arrives First

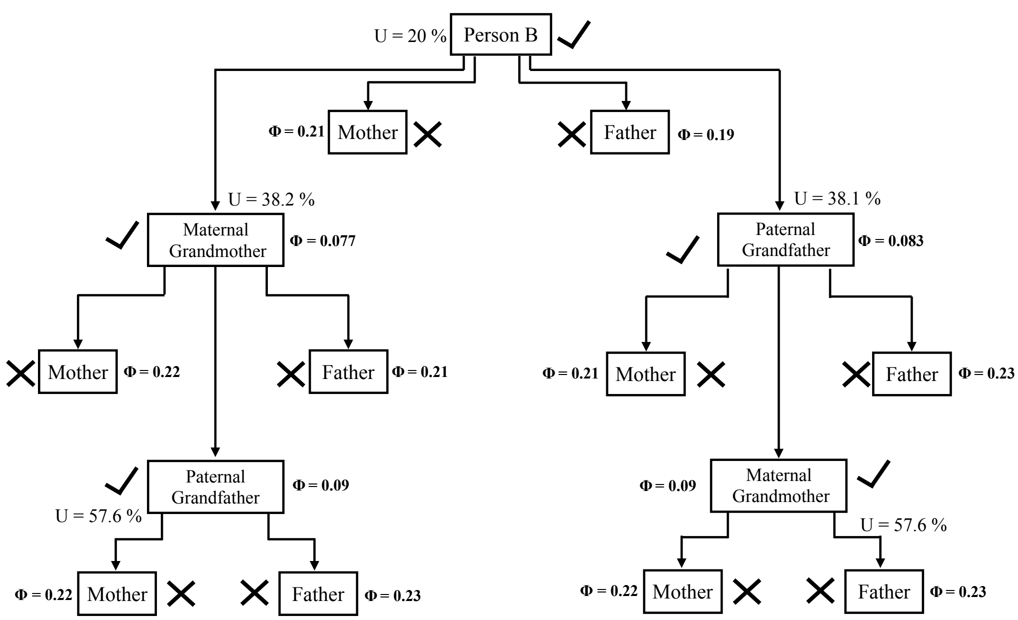


Figure 4.6: The first arrived member in f_B is Person B. The possible arrival sequences of family f_B is shown, when Person B arrives first. The optimization model is solved by relaxing outlier privacy constraints and satisfying kinship constraints.

In Figure 4.6, all the possible sequences where Person B arrives first is illustrated with the corresponding solutions. Similar to f_A case, we observe that when a parent of the Person B arrives as the second member, the relationship cannot be hidden. However, if the second arrived member is one of the grandparents, it is possible to add the grandparents genome to the database. The grandparents and the Person B can be displayed as a third degree relationship as opposed to the real second-degree relationship. Since the Person B has lower genomic similarity with his grandmother based on the kinship estimate, the addition of the grandmother results in a slightly lower Φ value and a higher utility compared to the addition of the paternal grandfather as the second member.

The third family member can only be added to the database if that member is a second degree relative of the Person B. As Figure 4.6 illustrates, if the second arrived person is the maternal grandmother, the third added person can only be the paternal grandfather. Similarly if the third arrived person is the paternal grandfather, only the maternal grandparent can be added as the third person. In both cases the utility will be as high as approximately 57.6% and a maximum kinship coefficient $\Phi = 0.09$ is in the family. Here too, we observe that adding a fourth family member is not possible without causing a kinship privacy leak.

Successful arrival orders for the sequences where Person B arrives first:
{Person B - Maternal grandmother - Paternal grandfather}
{Person B - Paternal grandfather - Maternal grandmother}

4.1.2.2 Arrival Sequences where Maternal Grandmother Arrives First

In this scenario, the grandmother is the first family member that arrives to the database. As shown in Figure 4.7, only family member that can be added is Person B. Provided that Person B is the second added member, the paternal grandfather can be added as the third family member. The maximum kinship value in the family will be 0.09, with a high utility 58.8%. Adding parents to the database at any time point makes the model infeasible.

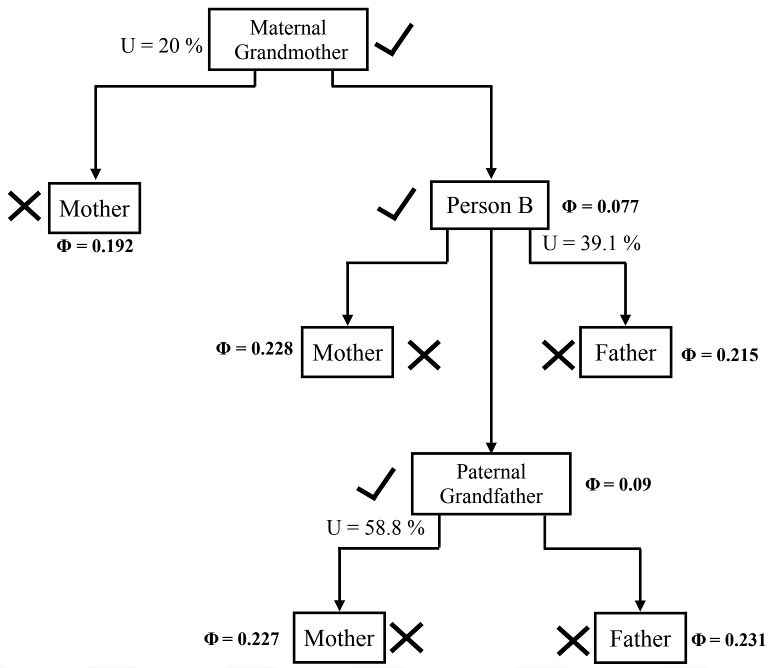


Figure 4.7: The first arrived member in f_B is the maternal grandmother. The possible arrival sequences of family f_B is shown where the maternal grandmother is the first arrived member. The optimization model is solved by relaxing kinship privacy constraints and satisfying outlier constraints.

Successful arrival orders for the sequences where the maternal grandmother first:

{Maternal grandmother - Person B - Paternal Grandfather}

4.1.2.3 Arrival Sequences where Paternal Grandfather Arrives First

The sequence tree of the arrival sequences where paternal grandfather arrives first is given in Figure 4.8. This tree is very similar to the arrival tree where maternal grandmother arrives first (Figure 4.7). The only difference between the two solutions is the difference in utility values, which is around 0.1%, when the third person is added. Note that since the number of SNP positions is large, 0.1% of this number corresponds to large number of SNPs.

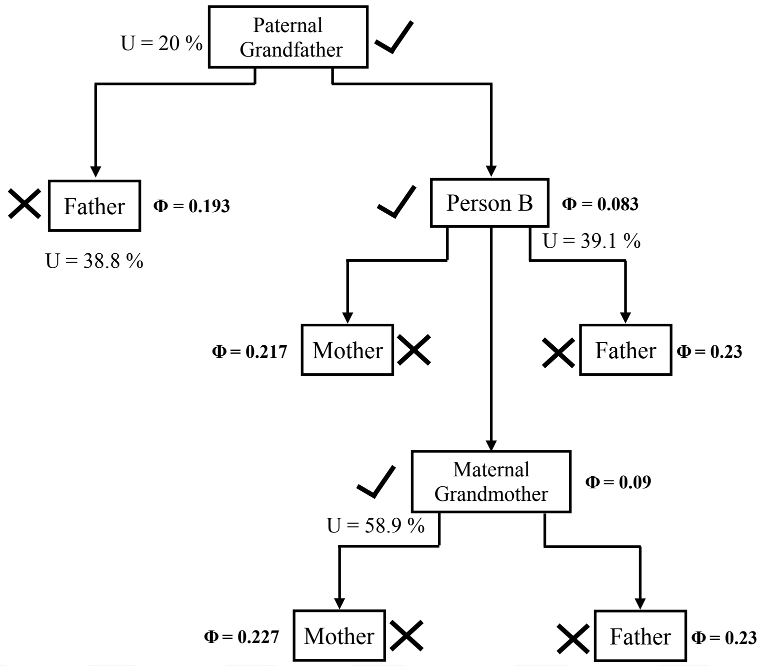


Figure 4.8: The sequence where the first arrived member in f_B is the paternal grandfather. Solutions for arrival sequences for f_B where the paternal grandfather arrives first for the case where kinship constraints are relaxed and the outlier constrains are satisfied.

Successful arrival orders for the sequences where the paternal grandfather arrives first:
{Paternal grandfather - Person B - Maternal grandmother}

4.1.2.4 Arrival Sequences where Mother Arrives First

Figure 4.9 displays the arrival sequences and their solutions for the case where the mother is arrived first. This case is very constraining, after her arrival, no relatives can be added to the database. Although the model has a feasible solution, it is not enough to reduce the parent-offspring relationships by at least one degree.

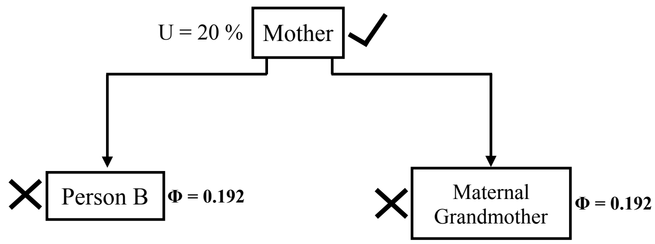


Figure 4.9: The first arrived member in f_B is the mother. The possible arrival sequences of family f_B is shown, where it is the mother that arrives first. The tree represents solutions obtained by satisfying outlier constraints and relaxing kinship constraints.

4.1.2.5 Arrival Sequences where Father Arrives First

Similar to the mother case, when the father arrives to the database first, no family member should be added to the database if the kinship privacy is to be preserved. The sequence tree of the father is given in Figure 4.10. None of the relationships between father and his relatives decreases at least one relationship degree.

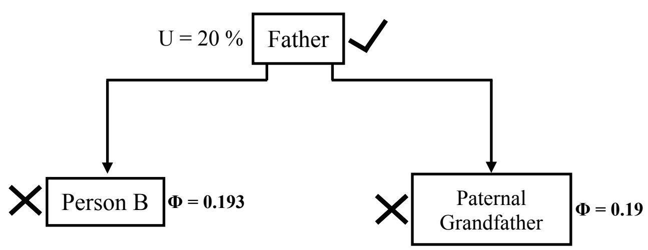


Figure 4.10: The first arrived member in f_B is the father. Solutions for arrival sequences for f_B where the father arrives first and where the outlier constraints are satisfied.

4.2 Results by Solving the Optimization Problem by Satisfying Outlier Constraints

In this section, we provide results obtained by solving the optimization model where the kinship privacy constraints are satisfied while outlier constraints are relaxed are provided. As in the previous solutions, we consider all possible arrival sequences of the family. In the current model, the relaxation of outlier constraints might lead the outlier constraints go down to zero. When the outlier values get value of zero, this corresponds to the case where the outlier constraints are eliminated. As this is not a preferable situation, in the results provided below, if a feasible solution is found where the outlier values are very low (at least 15σ away from the outlier values found in the population), it is considered as too much relaxation. We conclude the person can be added; however, do no proceed with the sequence to add any other member as the outlier constraints are already puts the privacy at risk. This cases are indicated by an arrow followed by a cross mark in place of the subsequent person.

4.2.1 Results on Family A

4.2.1.1 Arrival Sequences where Person A Arrives First

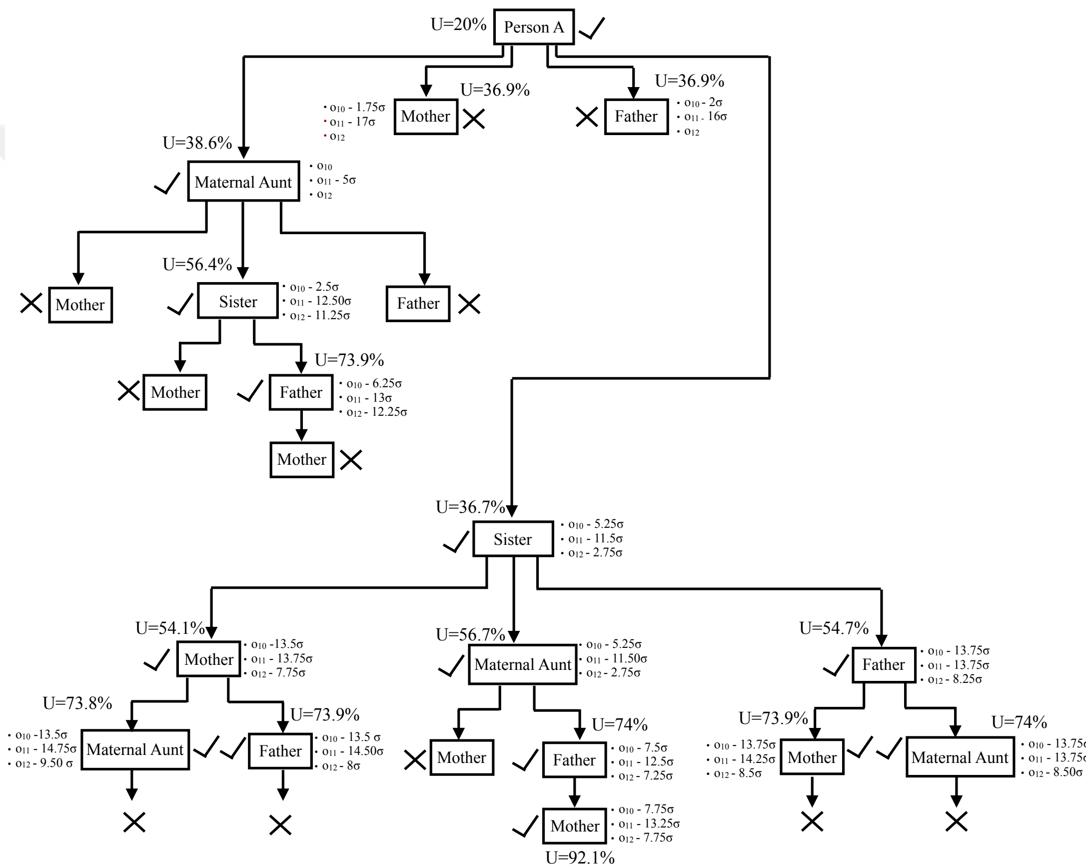


Figure 4.11: The solution for family f_A when the Person A is added first and when the kinship constraints are relaxed and the outlier constraints are satisfied. Downward arrows point to the subsequent member that arrives. Check mark indicates a successful addition of the member with the feasible solution that does not detriment privacy. Cross mark denotes that there is no feasible solution for the addition of this person. o_{10}, o_{11} and o_{12} are the initial outlier values found in the population.The relaxed outlier values returned from the optimization problem's solution are shown next to each box.

Figure 4.11 shows all the possible sequences of arrival of the family members to the database if Person A is to be the first member added while satisfying the imposed outlier constraints. Again, at the first level of the tree, the addition of parents is not allowed, because such an addition can only occur if the outlier thresholds are almost 0. If the outlier constraints were ignored and no family member pair solution exceeds 0, then approximately 36.9% utility would have been achieved.

Unlike addition of parents, addition of further relatives as second family member does not decrease outlier constraints down to zero. We observe that the solution for the adding the sister requires a much greater decrease in outlier threshold values compared to the solution for addition of the maternal aunt. The reason behind this difference is due to the relatively more distant relationship of the aunt and Person A. If the second added family member is the aunt, then the sister can be the added as the third person. Following the sister father can be added to the database. This arrival sequence {Person A- Aunt- Sister - Father} achieves 73.9% utility. If the second added family member is the sister, the addition sequence {Person A - Sister - Aunt - Father} (which consists of the same family members as the former sequence) achieves slightly higher utility value; 74% and the minimum outlier values are different from the former sequence. The o_{12} value is 5σ lower, o_{11} value is 0.5σ lower and o_{10} value is 1.25σ higher in the latter sequence. This result indicates that arrival sequences results with very different feasible solutions when the relatives arrive in different orders.

If the second added member is the sister and the third member is one of the parents, then following them a fourth member can be added but the outlier constraints have to be relaxed too much to attain a feasible solution at least 13.5σ, 13.75σ, and 7.75σ leakage in o_{10}, o_{11}, o_{12} values. Still, the outlier values are lower than 15σ but addition of a fifth members is not feasible. We observe that at any level of the tree, if a sequence includes one of the parents, the outlier constraints are rendered very low; therefore, the privacy is violated to a great extend in terms of the outlier constraints.

Successful arrival orders for the sequences where Person A arrives first:

{Person A - Father}

{Person A - Mother}

{Person A - Maternal Aunt - Sister - Father}

{Person A - Sister - Maternal Aunt - Father -Mother}

{Person A - Sister - Mother - Father}

{Person A - Sister - Mother - Maternal Aunt}

{Person A - Sister - Father - Maternal Aunt}

{Person A - Sister - Father - Mother}

4.2.1.2 Arrival Sequences where Sister Arrives First

Figure 4.12 displays all possible arrival scenarios in which the sister is the first arrived member of the family. We observe that the solution is very similar to the case where Person A arrives at the database as the first person (Figure 4.11). This is expected as the sister and Person A have similar kinship distance to the family members. Differences arises in the outlier and the utility attained as the kinship estimates of the Person A and the sister to other family member are not exactly the same. The most significant difference is observed in the outlier values of two sequences: {Sister - Person A} and {Person A - Sister}. The outlier value for successful addition of the former sequence should be 2.25σ away and for the latter sequence it should be 2.75σ away from the o_{12} value. However, the total number of removed SNPs is equal. Indeed, the solution to this nonlinear optimization problem is not unique. As CPLEX returns one of the solutions randomly from all the possible solutions, they can end up different. Actually, the different o_{12} value is one of the alternative solutions.

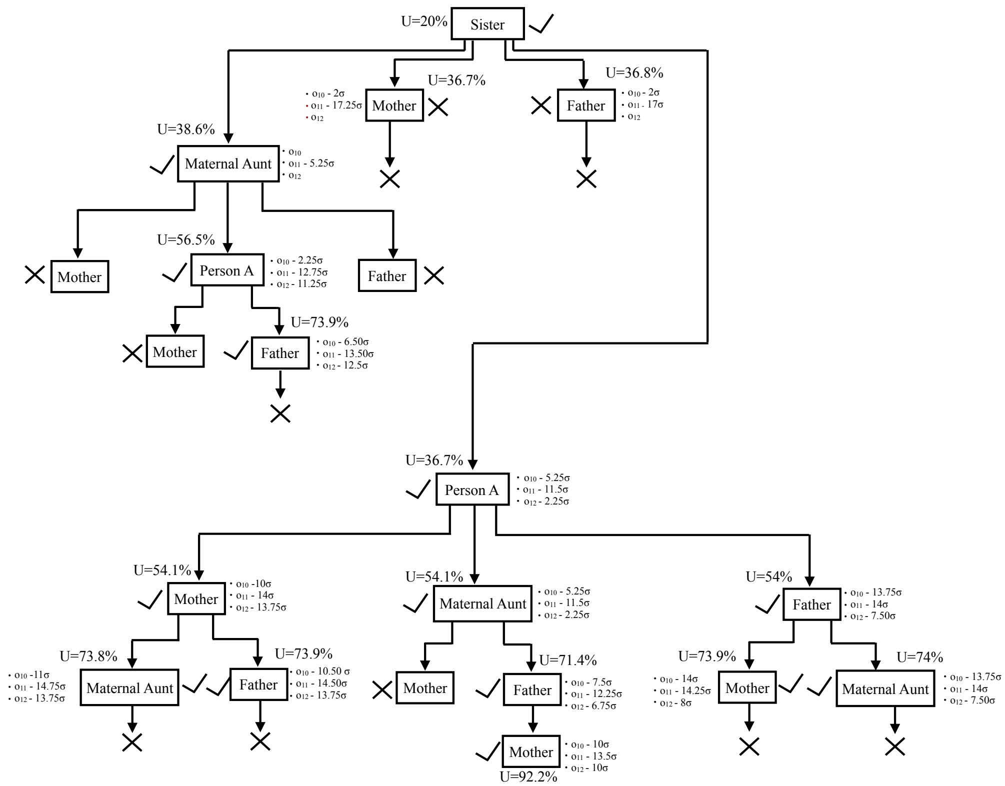


Figure 4.12: The first arrived member in f_A is the sister. Solutions for arrival sequences for f_A where the sister arrives first for the case where outlier constraints are relaxed and the kinship constrains are satisfied.

Successful arrival orders for the sequences where the sister arrives first:

{Sister - Father}

{Sister - Mother}

{Sister - Maternal Aunt - Person A - Father}

{Sister - Person A - Maternal Aunt - Father - Mother}

{Sister - Person A - Mother - Maternal Aunt}

{Sister - Person A - Mother - Father}

{Sister - Person A - Father - Maternal Aunt}

{Sister - Person A - Father - Mother}

4.2.1.3 Arrival Sequences where Maternal Aunt Arrives First

Figure 4.14 shows the arrival sequences and their corresponding solutions when the maternal aunt arrives at the databases as the first person. When the second added member is one of the nephews, four family members can be added. The four people sequence, {Maternal Aunt - Person A - Sister - Father} and {Maternal Aunt - Sister - Person A - Father}, sequences are similar as the last three people are nephews and father. However, here the two results of these sequences are slightly different from each other. The former sequence leads less outlier relaxation and privacy leakage.

If the mother is the second arrived family member, sister and Person A can successfully be added as the third member. However, if the nephews are added in the second and third place, it is observed that a fourth member can also be added to the database and this addition sequence ensures the maximum utility of 73.9% between all possible sequences.

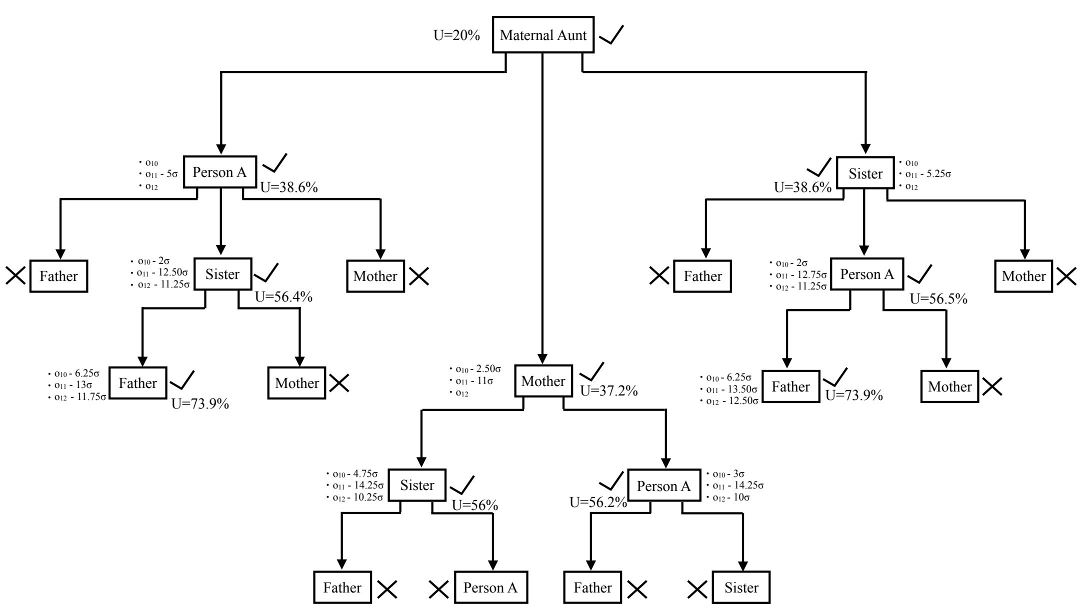


Figure 4.13: The first arrived member in f_A is the maternal aunt. The possible arrival sequences of family f_A is shown, where it is the maternal aunt that arrives first. The tree represents solutions obtained by relaxing outlier constraints and satisfying kinship constraints.

Successful arrival orders for the sequences where the maternal aunt arrives first:

{Maternal Aunt - Person A - Sister - Father}

{Maternal Aunt - Sister - Person A - Father}

{Maternal Aunt - Mother - Sister}

{Maternal Aunt - Mother - Person A}

4.2.1.4 Arrival Sequences where Mother Arrives First

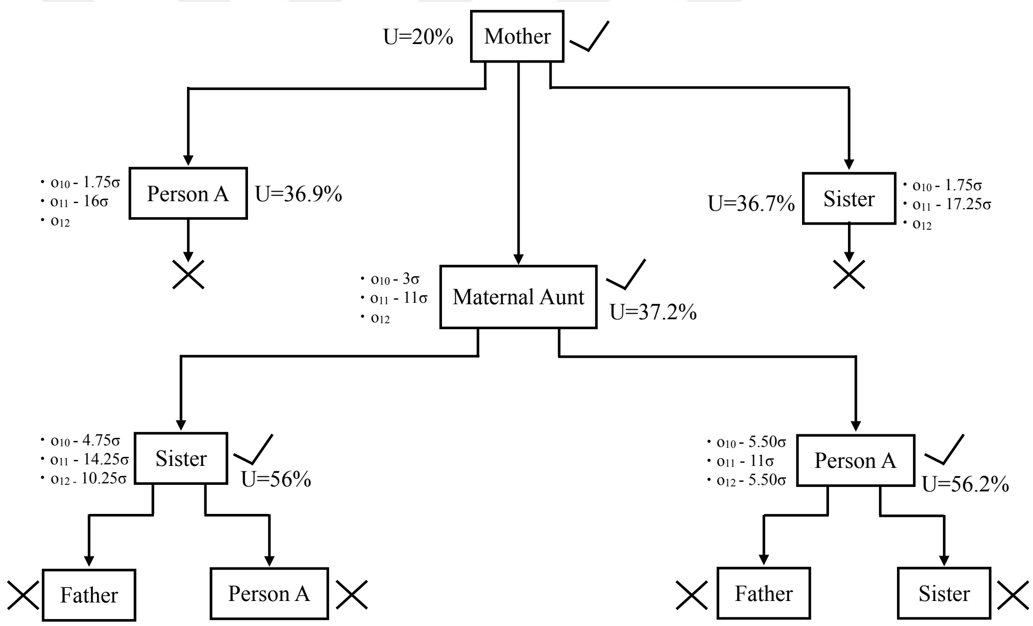


Figure 4.14: The sequence where the first arrived member in f_A is the mother. Solutions for arrival sequences for f_A where the mother arrives first for the case where outlier constraints are relaxed and the kinship constrains are satisfied.

When the mother is the first arrived family member in the database, Person A, or the maternal aunt can be added as the subsequent member. The addition of the Person A prevents further family members from being added as he is close to the other family members. On the other hand, the maternal aunt results with longer sequences: {Mother - Maternal Aunt - Sister} and {Mother - Maternal

Aunt - Person A}. These sequences achieve approximately 56% utility. The latter sequence causes significantly lower levels of privacy leaks in outlier values but addition of a fourth member makes the model infeasible.

Successful arrival orders for the sequences where the mother arrives first:
{Mother - Person A}
{Mother - Sister}
{Mother - Maternal Aunt - Sister}
{Mother - Maternal Aunt - Person A}

4.2.1.5 Arrival Sequences where Father Arrives First

The father is only related to Person A and the sister in the family. If the father is first arrived member to the database, addition of one of his offsprings results with very high privacy leaks in outlier o_{11} values. The father's tree is given in Figure 4.15.

Successful arrival orders for the sequences where the father arrives first:
{Father - Person A}
{Father - Sister}

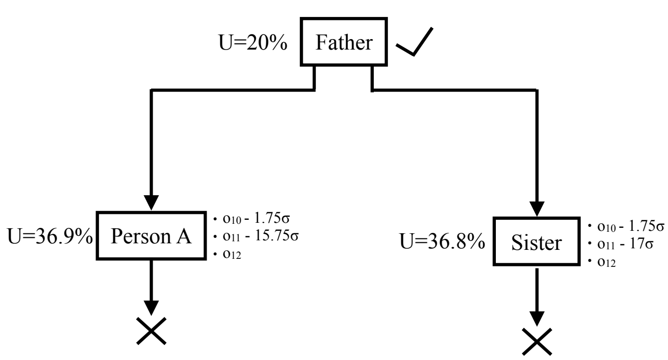


Figure 4.15: The first arrived member in f_A is the father. Solutions for arrival sequences for f_A where the father arrives first and kinship constraints are satisfied.

4.2.2 Results on Family B

4.2.2.1 Arrival Sequences where Person B Arrives First

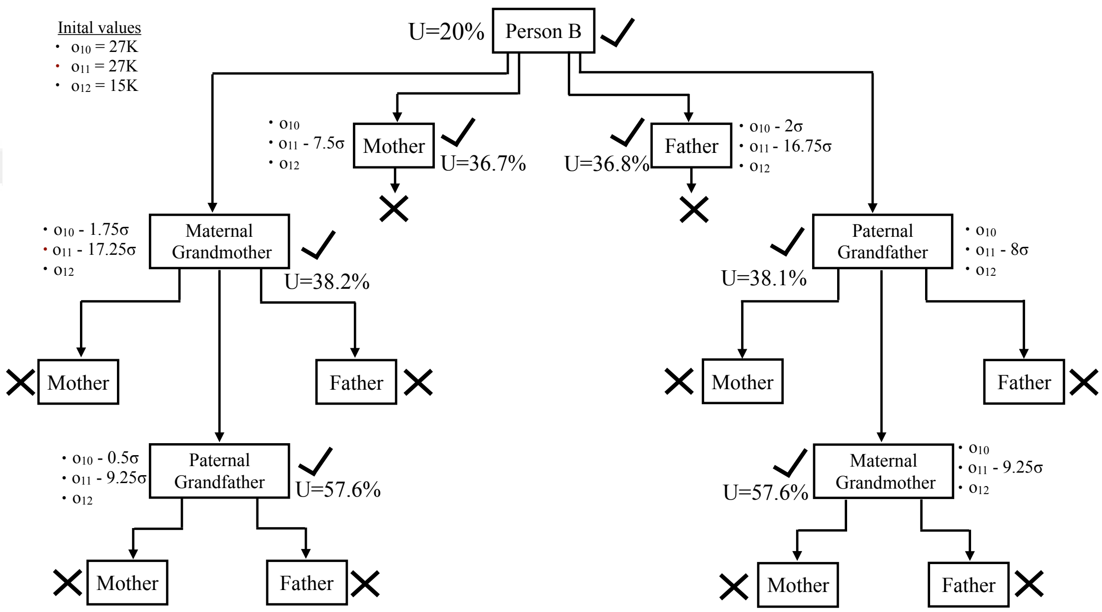


Figure 4.16: The sequence where the first arrived member in f_B is Person B. Solutions for arrival sequences for f_B where Person B arrives first for the case where outlier constraints are relaxed and the kinship constrains are satisfied.

Figure 4.16 illustrates all possible arrival sequences when the Person B is the first member to arrive at the database. The corresponding solutions are based on solving the model by relaxing the outlier constraints. Parents can be added but the outlier values will be very low, risking the privacy of the family. In this case, it is not possible to add a third family member because the model is infeasible for the given outlier constraints.

If the second added family member following Person B is a second degree relative such as paternal grandfather or maternal aunt, then the adding this member is possible without a privacy breach. This addition results in approximately 8σ leakage in o_{11} value and 38.1% utility. A small amount of difference is observed in the utility and outlier values pertaining to the addition of maternal grandmother and paternal grandfather as the second members. At the third level of the tree, two addition sequences are obtained: {Person B - Paternal Uncle - Maternal Aunt} and {Person B - Maternal Aunt - Paternal Uncle}. These sequences show that the second relatives of the Person B can be successfully added at any level of the tree when the outlier constraints are relaxed.

Successful arrival orders for the sequences where Person B arrives first:
{Person B - Mother}
{Person B - Father}
{Person B - Maternal grandmother - Paternal grandfather}
{Person B - Paternal grandfather - Maternal grandmother}

4.2.2.2 Arrival Sequences where Mother Arrives First

There are two relatives of the mother in f_B; maternal grandmother and Person B, with both of whom she has parent-offspring relationships. The solutions from the optimization model for addition of these members require almost 0 o_{11} values. Therefore, after the addition of second member, no one member should be added.

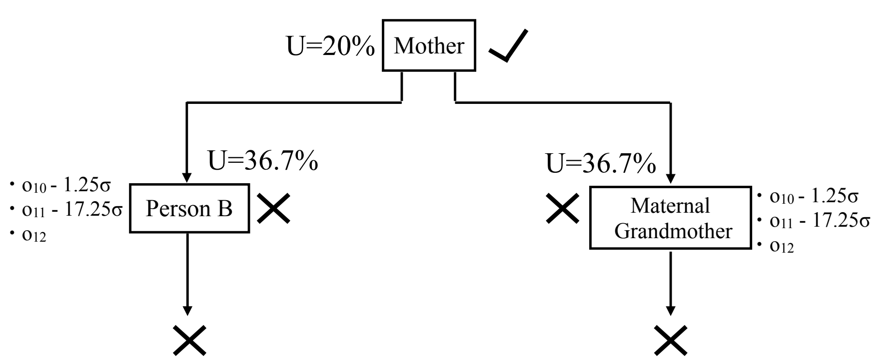


Figure 4.17: The first arrived member in f_B is the mother. The possible arrival sequences of family f_B is shown, where it is the mother that arrives first. The tree represents solutions obtained by satisfying kinship constraints and relaxing outlier constraints.

Successful arrival orders for the sequences where the mother arrives first:

{Mother - Person B}

{Mother - Maternal grandmother}

4.2.2.3 Arrival Sequences where Father Arrives First

This scenario is very similar to the previous scenario in which the father is the first arrived member. Father's relatives are paternal grandfather and Person B; the father has two parent-offspring relationships in the family. Addition of one of them requires masking a large number of SNPs, and a huge privacy leakage in terms of population statistics.

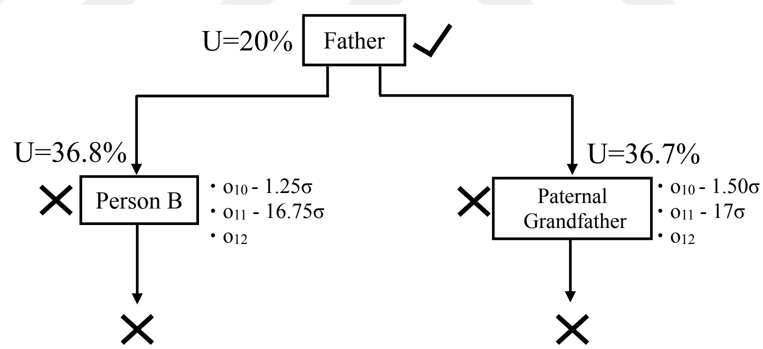


Figure 4.18: The first arrived member in f_B is the father. Solutions for arrival sequences for f_B where the father arrives first. The optimization model is solved by relaxing outlier privacy constraints and satisfying kinship constraints.

Successful arrival orders for the sequences where the father arrives first:

{Father - Person B}

{Father - Paternal grandfather}

4.2.2.4 Arrival Sequences where Maternal Grandmother Arrives First

Figure 4.19 illustrates the possible arrival sequences in which the maternal grandmother is arrived at the database first. Mother or Person B can be added as the subsequent person. Given that the mother is added second, a third family member cannot be added because the addition of the mother almost sets the o_{11} value to zero.

If the second incoming member is Person B, arrival of him produces a feasible solution with 38.2% utility and 7.5σ decrease in o_{11} value. Following Person B, it is not possible to add his parents. On the other hand, if o_{11} value decreases by 1.50σ and o_{10} decreases by 1.75σ, the paternal grandfather can be added.

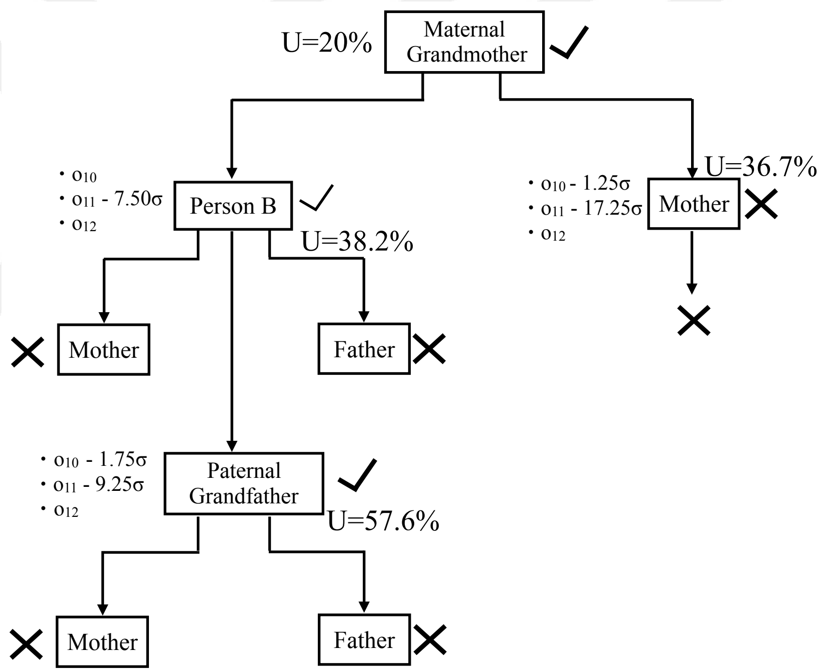


Figure 4.19: The sequence where the first arrived member in f_B is the maternal grandmother. Solutions for arrival sequences for f_A where the maternal grandmother arrives first for the case where kinship constraints are satisfied and the outlier constrains are relaxed.

Successful arrival orders for the sequences where the maternal grandmother arrives first:

{Maternal grandmother - Mother}

{Maternal grandmother - Person B - Paternal grandfather}

4.2.2.5 Arrival Sequences where Paternal Grandfather Arrives First

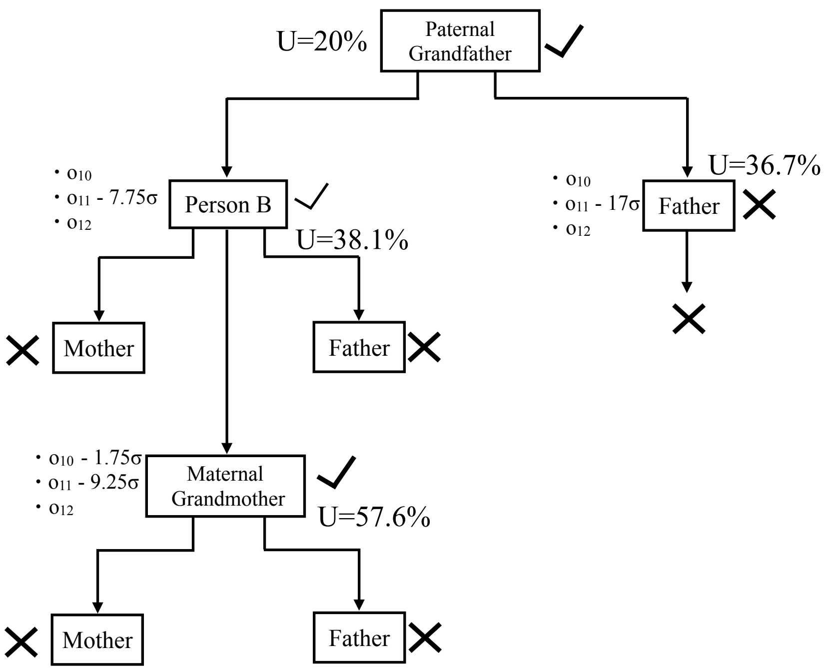


Figure 4.20: The first arrived member in f_B is the paternal grandfather. The possible arrival sequences of family f_B is shown where the paternal grandfather is the first arrived member. The optimization model is solved by satisfying kinship privacy constraints and relaxing outlier constraints.

Figure 4.20 displays the case where the paternal grandfather arrives first. In this scenario, we observe a tree very similar to Figure 4.19. The longest sequence is {Paternal Grandfather - Person B - Maternal Grandmother}; which results in 57.6% utility and 9.25σ decrease in o_{11} and 1.75σ decrease in o_{10} values. At the second level of tree, when Person B is added, the utility value is 1% lower and 0.25σ more privacy leakage occurs compared to the previous scenario described in Section 4.2.2.4. This is because of high genomic similarity between the paternal grandfather and Person B.

Successful arrival orders for the sequences where the paternal grandfather arrives first:

{Paternal grandfather - Father}

{Paternal grandfather - Person B - Maternal grandmother}

Chapter 5

Conclusion and Future Work

Everyday genomic data are incorporated in various domains including biomedical research, clinical care and customer services. Realizing the promise of genome sequencing in these domains often requires widespread participation to share genomic data. However, if unrestricted access to the genomic data is granted, critical information about the participants can be extracted without their consent. Potentially dire and irreversible consequences of privacy breach and the associated risks entail responsible data sharing policies to be put in place. This critical challenge not only necessitates implementing law and policies to protect individuals privacy rights and but also developing safeguarding computational tools that secure individuals privacy. In this work, we developed such a methodology to protect kinship information of the individuals who share their genomic data in a public database.

We show that the families in a database can be inferred by simple clustering techniques with the distance metrics calculated based on the genome dissimilarity. In our experiments we worked with SNP genomic data and evaluated the methodology on two families, one inferred from OpenSNP database by this clustering attack strategy and one publicly shared by the family. We proposed a technique to protect the familial relationships in public genomic databases. Two privacy risks that may reveal familial relations to be detected: genotype similarity and outlier SNP allele pair counts. We assumed a sequential arrival to the database and we protect privacy by hiding certain SNP loci in the newly arrived member. We propose a model that minimizes the kinship privacy risks while maximizing the amount of genomic data, thus the utility of it. We formulated this trade-off between utility of sharing data and preserving the privacy of the family as an optimization problem. The number of SNPs to hide and the category of the SNP loci, which depends on the allele type in other family members, is determined by solving an optimization model. This endows us with a method to systemically identify and withhold a minimal portion of the newly arrived members genome and share the remaining data without compromising kinship privacy.

On the two families we worked with, a solution to the optimization problem that satisfies both types of privacy constraints is often not found. For this reason, we build our optimization strategy on imposing one of the privacy constraints strictly, meanwhile relaxing the other. The method was tested against protecting two different families. The family f_B consists of parent-offspring, sibling and aunt-nephew relationships and the family f_A consists of parent-offspring and grandparent-grandchild relationships. Having different relationships in these two families enabled us to test the method in different scenarios.

Based on the results obtained from the two families, we observe that the concurrent presence of parent and offspring data in a database constitutes a risk. Sharing the genomic data of siblings is feasible without compromising privacy. However, reducing the two siblings kinship to the level of two unrelated individuals was not possible but they can be disguised as if they were in a second-degree relationship. Further distant relatives can be successfully shared together. We observe that the arrival sequence of the family members affect the results. For instance as shown in Figure 4.1, the sequence {Person A, Sister, Maternal Aunt} can be added to the database, but once the arrival sequence is set to {Person A, Maternal Aunt, Sister} a privacy preserving dissemination is not possible.

When the outlier constraints are strictly satisfied and the kinship constraints are relaxed, we observed that, for both families, it is possible to share genomic data up to three people. In the family f_A the siblings are interpreted as second-degree relatives such as aunt-nephew or half-siblings whereas aunt-nephew relations revealed as unrelated. In the family f_B, grandparent-grandchild relatedness can be hidden as if it is a third-degree relationship such as cousins or great-grandparents. For the family f_A, if the kinship constraints are strictly satisfied and the outlier constraints are relaxed up to five people can be added successfully without revealing the real relationships. For the family f_B, adding up to three family members is feasible.

The method developed here can be extended in future work. Potential extensions, ramifications and follow-up studies can be outlined as follows:

- The mechanisms we developed are based on KING kinship estimate [18]; however, the same framework can be adapted to other kinship estimate metrics by generating privacy constraints based on these estimates. The privacy constraints in the optimization models should be be updated accordingly.
- Currently, our methodology only exploits one type of genomic data, specifically the SNP data. As a future work other genomic variation information, such as the copy number variation, or other auxiliary information such as those that can be gathered from online social networks can be incorporated and a more wholistic protection mechanism can be developed.
- In certain populations, people from certain ethnic origins faces discrimination; protecting such vulnerable groups requires mechanisms to conceal the ethnicity of the individuals who share their data. A methodology with the same perspective can be developed to protect the ethnicity information of the individuals who share their data.
- One limitation of the current work is disregarding the statistical dependencies between genomic positions. If a position that is masked is strongly statistically dependent to another loci, it can be inferred from the correlated position thus obviating the extra protection gained from its absence. The proposed models could be improved by incorporating these correlation structures.

- We worked with kinship privacy risk in isolation of other genomic privacy risks. For future work, one can consider a joint optimization strategy to protect kinship at different folds. For example, certain positions reveal more information as they are shown to be associated with disease states or predisposition. In this work, all the genomic positions are treated as equally important. Based on the level of information that a position can reveal, we can assign importance weights to it. This information can then be incorporated in the model such that the critical positions are preferably masked more often in comparison to the other regions that are not associated with a certain phenotype. However, one should keep in mind that, we do not have the complete knowledge of the complex interactions of the genotype with the phenotype. As we learn more about these interactions, the assigned importance weights to each genomic position will be updated. A position that seems to release no information about the individual can be associated with a critical disease or behavioural treat in the future.

Bibliography

[1] J. C. Venter, M. D. Adams, E. W. Myers, P. W. Li, R. J. Mural, G. G. Sutton, H. O. Smith, M. Yandell, C. A. Evans, R. A. Holt, *et al.*, "The sequence of the human genome," *Science*, vol. 291, no. 5507, pp. 1304–1351, 2001.

[2] S. Goodwin, J. D. McPherson, and W. R. McCombie, "Coming of age: ten years of next-generation sequencing technologies," *Nature Reviews Genetics*, vol. 17, no. 6, pp. 333–351, 2016.

[3] `http://www.opensnp.org`. Last visited on 1-Feb-2017.

[4] `https://www.23andme.com/welcome/`. Last visited on 11-Feb-2017.

[5] 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.

[6] L. Sweeney, A. Abu, and J. Winn, "Identifying participants in the personal genome project by name," 2013. In: Harvard University. Data Privacy Lab. White Paper 1021 1.

[7] M. L. Noralane, "Personal autonomy in the genomic era." In Video Proceedings of Mayo Clinic Individualizing Medicine Conference, 2012.

[8] `http://www.nytimes.com/2013/03/24/opinion/sunday/the-immortal-life-of-henrietta-lacks-thesequel.html?pagewanted=all`. Last visited on 9-Aug-2015.

[9] R. Skloot, *The immortal life of Henrietta Lacks.* Broadway Books, 2011.

[10] E. Ayday and J.-P. Hubaux, "Threats and solutions for genomic data privacy," in *Medical Data Privacy Handbook*, pp. 463–492, Springer, 2015.

[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] S. Levy, G. Sutton, P. C. Ng, L. Feuk, A. L. Halpern, B. P. Walenz, N. Axelrod, J. Huang, E. F. Kirkness, G. Denisov, *et al.*, "The diploid genome sequence of an individual human," *PLoS Biol*, vol. 5, no. 10, p. e254, 2007.

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

[14] https://www.familytreedna.com/learn/autosomal-ancestry/universal-dna-matching/family-finder-test-work/. Last visited on 20-Feb-2017.

[15] `https://www.ancestry.com`. Last visited on 11-Feb-2017.

[16] H. Li, G. Glusman, C. Huff, J. Caballero, and J. C. Roach, "Accurate and robust prediction of genetic relationship from whole-genome sequences," *PloS one*, vol. 9, no. 2, p. e85437, 2014.

[17] S. Purcell, B. Neale, K. Todd-Brown, L. Thomas, M. A. Ferreira, D. Bender, J. Maller, P. Sklar, P. I. de Bakker, M. J. Daly, and P. C. Sham, "PLINK: a tool set for whole-genome association and population-based linkage analyses," *Am J Hum Genet*, vol. 81, pp. 559–575, Sep 2007.

[18] A. Manichaikul, J. C. Mychaleckyj, S. S. Rich, K. Daly, M. Sale, and W.-M. Chen, "Robust relationship inference in genome-wide association studies," *Bioinformatics*, vol. 26, no. 22, p. 2867, 2010.

[19] C. D. Huff, D. J. Witherspoon, T. S. Simonson, J. Xing, W. S. Watkins, Y. Zhang, T. M. Tuohy, D. W. Neklason, R. W. Burt, S. L. Guthery,

et al., “Maximum-likelihood estimation of recent shared ancestry (ERSA),” *Genome research*, vol. 21, no. 5, pp. 768–774, 2011.

[20] T. Thornton, H. Tang, T. J. Hoffmann, H. M. Ochs-Balcom, B. J. Caan, and N. Risch, “Estimating kinship in admixed populations,” *The American Journal of Human Genetics*, vol. 91, no. 1, pp. 122–138, 2012.

[21] M. P. Conomos, A. P. Reiner, B. S. Weir, and T. A. Thornton, “Model-free estimation of recent genetic relatedness,” *The American Journal of Human Genetics*, vol. 98, no. 1, pp. 127–148, 2016.

[22] https://www.ancestry.com/corporate/sites/default/files/AncestryDNA-Matching-White-Paper.pdf. Last visited on 20-Feb-2017.

[23] A. Kong, G. Masson, M. L. Frigge, A. Gylfason, P. Zusmanovich, G. Thorleifsson, P. I. Olason, A. Ingason, S. Steinberg, T. Rafnar, *et al.*, “Detection of sharing by descent, long-range phasing and haplotype imputation,” *Nature genetics*, vol. 40, no. 9, pp. 1068–1075, 2008.

[24] F. Balci, H. Kulan, C. Alkan, and E. Ayday, “A new inference attack against kin genomic privacy,” in *In Privacy-aware computational genomics (PRIVAGEN), Tokyo, Japan*, Sep 2015.

[25] C. A. Cassa, B. Schmidt, I. S. Kohane, and K. D. Mandl, “My sister’s keeper?: genomic research and the identifiability of siblings,” *BMC medical genomics*, vol. 1, no. 1, p. 32, 2008.

[26] Y. Erlich and A. Narayanan, “Routes for breaching and protecting genetic privacy,” *Nature Reviews Genetics*, vol. 15, no. 6, pp. 409–421, 2014.

[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] 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.

[29] M. Corpas, M. Cariaso, A. Coletta, D. Weiss, A. P. Harrison, F. Moran, and H. Yang, "A complete public domain family genomics dataset," *bioRxiv*, 2013.

[30] B. Weir and X. Zheng, "SNPs and SNVs in forensic science," *Forensic Science International: Genetics Supplement Series*, 9 2015.

[31] IBM ILOG CPLEX Optimizer. `http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/index.html`.

Appendix A

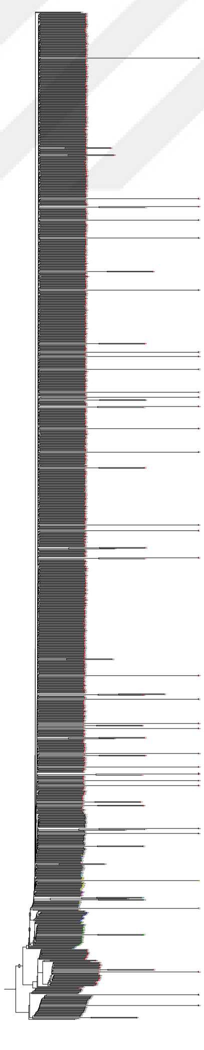


Figure A.1: Dendrogram of hierarchical cluster analysis of the OpenSNP data. The hanging lines show the families detected by applying hierarchical clustering on OpenSNP data.