Bioinformatics, 2017, 1–9
doi: 10.1093/bioinformatics/btx568
Advance Access Publication Date: 12 September 2017
Original Paper

OXFORD

Genome analysis

A utility maximizing and privacy preserving approach for protecting kinship in genomic databases

Gulce Kale, Erman Ayday* and Oznur Tastan*

Department of Computer Engineering, Bilkent University, Ankara 06800, Turkey

*To whom correspondence should be addressed.

Associate Editor: John Hancock

Received on May 11, 2017; revised on July 20, 2017; editorial decision on August 25, 2017; accepted on September 11, 2017

Abstract

Motivation: Rapid and low cost sequencing of genomes enabled widespread use of genomic data in research studies and personalized customer applications, where genomic data is shared in public databases. Although the identities of the participants are anonymized in these databases, sensitive information about individuals can still be inferred. One such information is kinship.

Results: We define two routes kinship privacy can leak and propose a technique to protect kinship privacy against these risks while maximizing the utility of shared data. The method involves systematic identification of minimal portions of genomic data to mask as new participants are added to the database. Choosing the proper positions to hide is cast as an optimization problem in which the number of positions to mask is minimized subject to privacy constraints that ensure the familial relationships are not revealed. We evaluate the proposed technique on real genomic data. Results indicate that concurrent sharing of data pertaining to a parent and an offspring results in high risks of kinship privacy, whereas the sharing data from further relatives together is often safer. We also show arrival order of family members have a high impact on the level of privacy risks and on the utility of sharing data.

Availability and implementation: https://github.com/tastanlab/Kinship-Privacy

Contact: erman@cs.bilkent.edu.tr or oznur.tastan@cs.bilkent.edu.tr

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

With the advances in sequencing technologies, obtaining the sequence of an individual's genome is faster and cheaper than ever (Goodwin *et al.*, 2016). This success and the reliability of sequencing rendered extensive use of genome sequencing in biomedical research and clinical care possible. While the use of genomic data in research studies gains traction, there is also a concurrent increase in the number of web services that enable genomic data sharing (openSNP and 23andme.com). Thus, today, 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 as a person's genome carries sensitive information pertaining to its owner such as ethnicity, kin, or predisposition to certain diseases. One immediate and pressing issue is the sharing of genomic data without compromising the privacy of the participants and their families (Erlich and Narayanan, 2014; Naveed *et al.*, 2015).

Even though most of the shared genomes are anonymized in genomic databases, it has been shown that anonymization is not sufficient for protecting the identities of the data donors (Clayton, 2010; Gymrek *et al.*, 2013; Homer *et al.*, 2008; Jacobs *et al.*, 2009; Lumley and Rice, 2010). Potentially dire and irreversible consequences of privacy breach and the associated risks not only necessitate implementing law and policies to protect individuals' rights but also developing safeguarding computational tools that secure individuals' privacy. Several methods have been proposed for protecting participants' identities in genome-wide association studies (Chen

et al., 2017; Wan *et al.*, 2017; Xie *et al.*, 2014). There are also methods that enable sharing of statistical analysis results conducted on genomic data in a privacy-preserving manner (Johnson and Shmatikov, 2013; Simmons and Berger, 2016; Tramèr *et al.*, 2015; Yu *et al.*, 2014) or identifying relatives in a privacy preserving manner (He *et al.*, 2014; Hormozdiari *et al.*, 2014). However, to the best of our knowledge, there is no work in the literature that aims at protecting privacy of kinship relationships of the members when genomic data is publicly shared.

Kinship information is a sensitive and its breach may lead to undesired parenthood issues, incidences of which have already been experienced. In his article titled 'With genetic testing, I gave my parents the gift of divorce' (Belluz, 2014), a researcher told his personal story. After conducting a genetic test and comparing his genome with others who took the same test, he accidentally found out that he had a half-brother. This eventually led to his mother and father getting a divorce. Kinship information can also be exploited in multilayer attacks. For instance, if an attacker obtains the genomic data of an individual, by inferring the kinship relationship between this individual and his/her family members in an anonymized genomic dataset, the attacker can easily de-anonymize the genomes of the family members. Additionally, if the kin of a person is identified together with her genome, critical information about the genomes of the family members can be inferred (Deznabi *et al.*, 2017; Humbert *et al.*, 2013), putting the whole family's privacy at risk. Thus, not only the deanonymized individual but also her family members may face discrimination on the basis of their genomic makeup, examples of which have already been experienced (Lindor, 2012). Therefore, in addition to being sensitive information by itself, kinship information has the potential to comprise the genomic privacy of the family members when used with other attacks.

In this work, we develop a methodology to protect kinship information of the individuals who share their genomic data in public databases. We define two ways that kinship privacy may leak. We present a computational model that renders the maximal sharing of genomic data possible while minimizing kinship privacy risks. We assume sequential arrival of individuals at the database and protect privacy by selectively hiding certain SNP loci in the newly arrived member. The number of SNPs to hide and the category of the SNP loci, which depends on the allele type in other family members, are determined by solving an optimization model. 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 systematically identify minimal portions of data to withhold as the new donors are added to the database. The proposed technique is evaluated with different arrival sequences in two different families.

2 Materials and methods

In the following sections, we first introduce the general framework we propose. Next, we define the two routes that kinship privacy can be revealed. Then, the proposed optimization model that maximizes the amount of data shared while minimizing the privacy leakage is explained. Finally, we describe how we solve these models.

2.1 General framework for protecting kinship privacy

As in the real life, we assume that individuals arrive at the database sequentially. At a given time, the privacy of the individuals who are already in the database is already protected. When a new person arrives, only the genome of this individual will be partially masked if needed. 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 by computing the kinship coefficient of the donor with the other people in the database. If the donor does not have a relative, her genome can be safely added without any withholding. If the person does have a relative already in the database, the family structures in the database are updated.

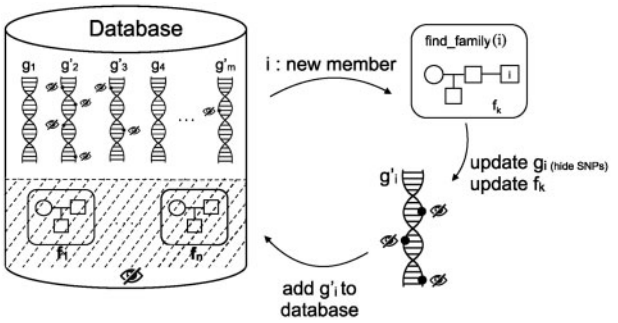

Fig. 1. Overview of the proposed scheme. When a new person i with genotype g_i is added to the database, we check for i's relatives in the database and determine the family i is related to. The privacy of the family f_k (to which i belongs to) is protected by selectively hiding a portion of g_i. The genotype of person i is then partially shared and this partially shared genotype is denoted as g'_i

At a given time, assume there are already families in the database and this family information is only known by the database. When an individual i arrives with genotype g_i, if the person has at least one relative in the database, then the family structures will be reorganized in one of the three different ways: (i) if i has at least one relative in a preexisting family, individual i is added to the family, (ii) if individual i is identified as a kin of individual j who is not a member of any of the families in the database, then a new family is instantiated with members i and j, and finally (iii) if user i has relatives in two different families and they are not connected; arrival of i will combine the families into a single family. This can arise in cases where the maternal and the paternal families are already in the database before the arrival of an individual.

Once the family of i is located and the family structures are updated, genotype data of i is added to the database in a privacy-preserving manner. Certain parts of g_i are systematically masked (with techniques which we shall detail in Section 2.4), and hence will not be visible to the outsiders. We denote this partially shared genome as g'_i. This overall process is illustrated in Figure 1.

2.2 Notations

Before discussing the details of the model, we introduce the frequently used notations. The SNP type of an individual at a certain position is represented with the number of its alternate alleles. Thus, the genome loci at which both alleles are the same as the reference genome are represented as 0, the positions wherein only one allele differs from the reference genome are denoted as 1, and the instances wherein an individual carries two alternate alleles are denoted as 2.

Our methodology assumes sequential arrival of the family members. We define a state vector, $\mathbf{s} = s_m \dots s_2 s_1$ that represents 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 for any SNP position) where

$s_i \in \{0, 1, 2\}$. We use this state vector while referring to the size of the genomic positions with a particular SNP configuration of the m family members. $n_{s_m \ldots s_2 s_1}$ denotes the number of genomic positions with the SNP configuration $s_m \ldots s_2 s_1$. For example, for a two-member family, n_{10} indicates the number of genomic loci where the latest arrived member's SNP type is 1 and the first arrived family member's SNP type is 0. Additionally, 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's SNP is of type 1 and the first comer's SNP can be of any type; 0, 1, or 2. Finally, we denote the number of positions that will be hidden with a particular SNP state sequence as $x_{s_m \ldots s_2 s_1}$.

2.3 Routes that leak kinship privacy

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

2.3.1 Privacy leak due to genotype similarity

Genomes pertaining to the members of a given family resemble each other more than the unrelated individuals. Therefore, the relatedness of two individuals can be inferred based on their genotype similarity. Several methods for estimating the relationship of a given pair of individuals based on genotype have been proposed in the literature (Huff *et al.*, 2011; Manichaikul *et al.*, 2010; Purcell *et al.*, 2007). KING kinship coefficient (Manichaikul *et al.*, 2010) is one such metric that has been demonstrated to be a robust estimator of kinship, which we utilize in this work. In this metric the kinship between two individuals i and j is defined as follows:

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

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. 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 this kinship coefficient is provided in Manichaikul *et al.* (2010).

2.3.2 Privacy leak due to outlier allele pair counts

Our methodology, as described in Section 2.4 in detail, involves hiding of genotype positions of a newly arrived member to prevent inference of kinship relationship of family members. To do so, positions wherein the two individuals are found to be heterozygous are frequently hidden as it decreases the kinship coefficient between two family members effectively. However, this alone will cause another privacy leakage as the number of positions where the two family members are heterozygous will be too small. Simply comparing this number to the population, one could infer that the two individuals are indeed in the same family.

To prevent such leakages, the model we propose chooses the regions to mask such that among family members the pairwise counts for each allele type do not decrease beyond the level of an outlier value. We set these threshold values for the pairwise SNP configurations that include at least one heterozygote genome, as these are the regions to be hidden to decrease the kinship coefficients. We denote these numbers as o_{10}, o_{11}, and o_{12}. Here, o_{10} indicates the outlier count for the allele pairs where one individual's SNP type is 1 and the other individual's SNP type is 0; and the other two numbers indicate the outlier values for the indexed SNP configurations. We estimate these outlier values from a population of randomly selected unrelated individuals from the openSNP database as described in Supplementary Material.

2.4 A utility maximizing privacy preserving approach for protecting kinship

2.4.1 Utility of sharing genomic data

A good solution should maximize the genomic data to be shared while minimizing the privacy risks associated with kinship among stored family members. We define the utility of shared data for the first m incoming members over a M-membered family retrospectively as follows:

$$U = \frac{V * m - x}{V * M}, \quad (2)$$

where x is the number of SNP positions that are masked in the family. Here, V is the size of the set of genomic positions that are not missing in all family members. The denominator represents the total number of genomic positions shared by all family members if no SNP positions were hidden. The nominator represents the number of positions shared after the masking. Supplementary Figure S1 illustrates how this utility score is calculated. As more family members' data is shared, the utility value increases. The maximum utility that can be achieved when m of the M members are in the database is m/M (when no positions are withheld), and the minimum utility is 0 (when all shared positions are hidden).

2.5 Protecting privacy for a three-member family

We would like to maximize utility for the family subject to privacy constraints that ensures that the kinship information of the family is preserved. We consider two types of privacy risks described in Sections 2.3.1 and 2.3.2 in deciding which portions of the genome of a newly added family member will be masked.

Maximizing utility in Equation (2) is equivalent to minimizing the sum of number of positions hidden, which can be represented as the sum of all positions masked with different SNP configurations. This can be represented as $x = \sum_{s \in S} x_s$, where S denotes the set of all possible state sequence vectors. From onwards, we describe the model for a three-member family for clarity; however, the formulation applies to handle larger families. In Section 3, we solve this problem for two families comprising five members each.

Consider a family f, whose members are the individuals i, j, k, in the order of latest arrived member to the first arrived member. 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 withheld. Because the kinship coefficient decreases significantly when n_{11} decreases, we hide the positions of the genome where $s_k = 1$ and $s_j = 1$.

After hiding x_{11} positions, the new KING estimate between the individuals j and k, ϕ'_{jk}, is calculated as follows:

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

Thus, we solve for x_{11} as below:

$$x_{11} = \frac{2n_{11} - 4(n_{02} + n_{20}) - n_{1*} + n_{*1}(1 - 4\phi'_{jk}))}{2(1 - 2\phi'_{jk})} \quad (3)$$

We can find the sufficient number of genomic positions to be hidden, x_{11} in individual j, by setting ϕ'_{jk} to the desired level and plugging the other numbers that are calculated from the two genomes. If ϕ'_{jk} is set to zero and $x_{11} < n_{11}$, this will decrease the relationship to the level of two unrelated person. At this stage, the system should also check whether the outlier constraints are violated. That is, $(n_{11} - x_{11}) \geq o_{11}$ should be satisfied. If that is not 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 and hidden from the set of SNPs of individual j where both k and j have SNP types equal to 1. Finally, this protected version of the genome, g'_j, is published in the database.

When the third individual i arrives at the database, the goal is to share the 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. The problem becomes more involved as the size of the family grows. To hide the relationship between i and j, we need to hide genomic positions where $s_i = 1$ and $s_j = 1$, and there is no restriction on the third individual's genotype. Similarly, to hide the relationship between i and k, we need to mask certain number of positions, where the first and the last members' SNPs are 1 and the second comer can be of any SNP type. Thus, the positions to be concealed should be selected from the set of SNPs such that the latest family member's SNP type is 1 and at least one of the two other members' SNP type is 1. To denote the number of such positions, we use the notation $x_{1\bullet\bullet}$ that is defined as $x_{110} + x_{111} + x_{112} + x_{101} + x_{121}$. These five configurations are the only configurations that will affect at least one of the pairwise relationship's kinship estimates. To maximize the utility we would need to minimize $x_{1\bullet\bullet}$ in a three-member family.

We generate privacy constraints and outlier constraints in the following sections for the case when the third member arrives. The constraints are generated under the assumption that all the members are related to 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 such pairs.

2.5.1 Constraints to prevent privacy leakage due to genomic similarity

As new positions are hidden [due to the nature of the kinship coefficient in Equation (1)], the kinship estimates between individuals are updated. We would like these estimates to be above a threshold value, Φ, to conceal the actual kinship relationship. If $\Phi = 0$, the relationships are hidden such that the people in the family are displayed as unrelated people. Recall i-th individual is the latest arrived member of the family, we use $x_{1\bullet\bullet}$ for the total number of positions that are masked from person i's genome. Below, for all the pairwise relations, (i, j), (i, k), and (j, k) pairs, we first derive expressions to indicate what the newly updated kinship estimates are after hiding x_{1**} positions. Then, based on this expression, we derive constraints that ensure the newly updated value is above the predefined Φ value.

Let ϕ'_{ij} denote the new kinship estimate attained after masking. ϕ'_{ij} is calculated as:

$$\frac{2(n_{11*} - x_{11*}) - 4(n_{20*} + n_{02*}) - (n_{1**} - x_{1\bullet\bullet}) + (n_{*1*} - x_{11*})}{4(n_{*1*} - x_{11*})}$$

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

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

Similarly, we derive an inequality constraint between individuals i and k after hiding positions where i and k are both heterozygote as below:

$$2n_{1*1} - 4(n_{2*0} + n_{0*2}) - n_{1**} + (1 - 4\Phi)n_{**1} \leq (2 - 4\Phi)x_{1*1} - x_{110} - x_{112} \quad (5)$$

Lastly, the inequality constraint between individuals j and k is derived:

$$2n_{*11} - 4(n_{*02} + n_{*20}) - n_{**1} + (1 - 4\Phi)n_{*1*} \leq (1 - 4\Phi)x_{11*} + 2x_{111} - x_{1*1} \quad (6)$$

These three constraints [Equations (4–6)], if satisfied concurrently, will guarantee that the kinship estimates are above Φ for all pairwise relationships.

2.5.2 Constraints to prevent privacy leakage due to pairwise allele outlier values

As mentioned in Section 2.3.2, 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, this count may reveal the relationship. Thus, we define a set of outlier constraints to guarantee that upon selecting which positions to hide, the pairwise allele counts do not fall below this set of outlier threshold values. The three outlier constraints are defined as follows:

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

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

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

There is also the trivial constraint that the number of positions to hide in a certain SNP configuration cannot exceed the total number of SNPs with that configuration, $0 \leq x_{110} \leq n_{110}$. We can rewrite these constraints in a more compact from as follows: $0 \leq x_{110} \leq u_{110}$, 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 type of positions to be held as well. These constraints together will ensure that as we hide certain positions, the population statistics are not outliers.

2.5.3 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 integer linear optimization problem as follows:

$$\begin{aligned} &\min \quad x_{101}+x_{111}+x_{121}+x_{110}+x_{112} \qquad (7)\\ &\text{s.t.}\end{aligned}$$

$$\begin{aligned} 2n_{11*}-4(n_{02*}+n_{20*})-n_{1**}+(1-4\Phi)n_{*1*} &\\ \leq (2-4\Phi)x_{11*}-x_{101}-x_{121} &\\ 2n_{1*1}-4(n_{2*0}+n_{0*2})-n_{1**}+(1-4\Phi)n_{**1} &\\ \leq (2-4\Phi)x_{1*1}-x_{110}-x_{112} &\\ 2n_{*11}-4(n_{*02}+n_{*20})-n_{**1}+(1-4\Phi)n_{*1*} &\\ \leq (1-4\Phi)x_{11*}+2x_{111}-x_{1*1} &\end{aligned}$$

$$\begin{aligned} &0 \leq x_{110} \leq u_{110}\\ &0 \leq x_{111} \leq u_{111}\\ &0 \leq x_{112} \leq u_{112}\\ &0 \leq x_{101} \leq u_{101}\\ &0 \leq x_{121} \leq u_{121}\\ &x_{11*}=x_{111}+x_{110}+x_{112}\\ &x_{1*1}=x_{111}+x_{101}+x_{121}\\ &x_{101},x_{111},x_{121},x_{110},x_{112} \in Z_{\geq 0}. \end{aligned}$$

The first three constraints are kinship constraints derived in Section 2.5.1. The next five inequality constraints represent the outlier constraints as derived in Section 2.5.2.

This problem can be solved optimally if there is a feasible solution. When there are many close relatives in the family, as was the case in the two families we tested the models on, the optimization problem may not have a feasible solution that satisfies all the constraints. In these cases, we propose to relax the constraints and alert the database owner about the amount of the privacy violation. Then, it is up to the database owner and/or the individual whether to share their data once they are informed about the risks. We solve the problem by relaxing one type of privacy constraints; the kinship or outlier constraints. In this scenario one type of constraints is strictly satisfied whereas the other type of constraint is relaxed. The overall idea is depicted in Supplementary Figure S2.

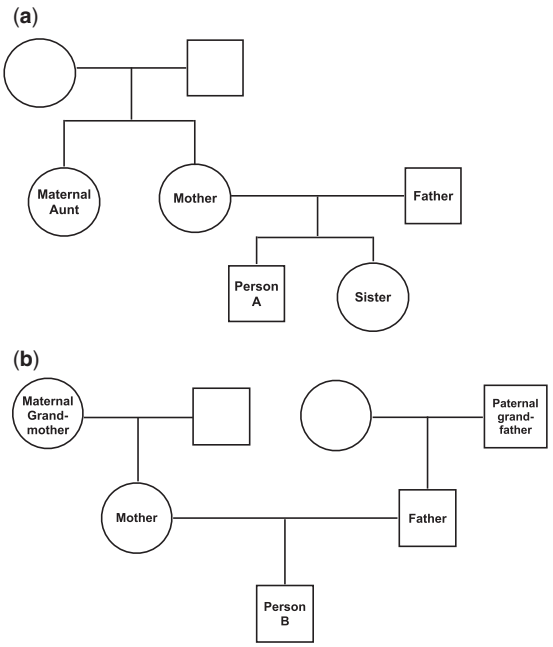

Fig. 2. Two family datasets. (**a**) Family A consists of person A, his father, mother, and maternal aunt. (**b**) Family B consists of person B, his mother, father, maternal grandmother, and paternal grandfather. No genotype information is available for people denoted with empty squares or circles

2.5.4 Solution by relaxing outlier constraints

If the problem is not feasible with all the original constraints are strictly satisfied, one might seek approximate solutions that minimally sacrifice strict privacy by relaxing the outlier constraints. Thus, the eventual solution will not be strictly below the outlier threshold values, but we ensure that these values shall deviate as small as possible from the original set of outlier values. We achieve this by introducing slack variables for every type of allele pairs. The outlier constraints given in Section 2.5.2 are relaxed as follows:

$$\begin{aligned} u_{110} &= \min\begin{pmatrix} n_{110}, (n_{11*}-(o_{11}-\epsilon_1)),\\ (n_{1*0}-(o_{10}-\epsilon_2)), (n_{*10}-(o_{10}-\epsilon_2)) \end{pmatrix}\\ u_{111} &= \min\begin{pmatrix} n_{111}, (n_{11*}-(o_{11}-\epsilon_1)),\\ (n_{1*1}-(o_{11}-\epsilon_1)), (n_{*11}-(o_{11}-\epsilon_1)) \end{pmatrix}\\ u_{112} &= \min\begin{pmatrix} n_{112}, (n_{11*}-(o_{11}-\epsilon_1)),\\ (n_{1*2}-(o_{12}-\epsilon_3)), (n_{*12}-(o_{12}-\epsilon_3)) \end{pmatrix} \qquad (8)\\ u_{101} &= \min\begin{pmatrix} n_{101}, (n_{10*}-(o_{10}-\epsilon_2)),\\ (n_{1*1}-(o_{11}-\epsilon_1)), (n_{*01}-(o_{10}-\epsilon_2)) \end{pmatrix}\\ u_{121} &= \min\begin{pmatrix} n_{121}, (n_{12*}-(o_{12}-\epsilon_3)),\\ (n_{1*1}-(o_{11}-\epsilon_1)), (n_{*21}-(o_{12}-\epsilon_3)) \end{pmatrix} \end{aligned}$$

In the above inequalities, $\epsilon_1 >= 0, \epsilon_2 >= 0$, and $\epsilon_3 >= 0$ are slack variables that control the relaxation of imposed constraints. This is equivalent to decreasing the outlier values than the original set values by some amount as determined by the slack variables. 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 them 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 as the solver.

2.5.5 Solution by relaxing kinship constraints

We can also solve the same problem where all the outlier constraints are satisfied but Φ (the maximum kinship estimate value allowed among family members) is not required to be ≤ 0. Instead, Φ is forced to deviate as small as possible from 0. For example, first-degree relationships 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 non-linear optimization problem stated as follows:

$$\begin{aligned}
&\min \quad \Phi \\
&\text{s.t.} \\
&2n_{11*} - 4(n_{02*} + n_{20*}) - n_{1**} + (1 - 4\Phi)n_{*1*} \\
&\qquad\qquad \leq (2 - 4\Phi)x_{11*} - x_{101} - x_{121} \\
&2n_{1*1} - 4(n_{2*0} + n_{0*2}) - n_{1**} + (1 - 4\Phi)n_{**1} \\
&\qquad\qquad \leq (2 - 4\Phi)x_{1*1} - x_{110} - x_{112} \\
&2n_{*11} - 4(n_{*02} + n_{*20}) - n_{**1} + (1 - 4\Phi)n_{*1*} \\
&\qquad\qquad \leq (1 - 4\Phi)x_{11*} + 2x_{111} - x_{1*1} \\
&\Phi' \leq \Phi \leq 0.5
\end{aligned}$$

and the constraints derived from (9) with the new definitions in (10). (9)

The Φ' is the kinship value attained when the optimization problem is solved for only two members. If that is already some value >0, in solving the three-member case, we only require Φ to be above that value instead of 0. In the above problem, in addition to the x values, Φ is also unknown. For this, we use the genetic algorithm solver under Global Optimization Toolbox in Matlab. After finding the optimal value of Φ from the model in Equation (9), the original optimization problem in Equation (8) is solved to find the minimum number of positions to mask.

2.6 SNP data and families

We evaluated our methodology on real genomic data of two families; we will refer these families as f_A, and f_B. The genomic data of f_A members are publicly shared on a personal website by person A (Corpas *et al.*, 2013). The family consists of a person A, his mother, father, maternal aunt, and his sister. The pedigree is provided in Figure 2a. The second family f_B (see Fig. 2b) is inferred from openSNP data via hierarchical clustering as described in Supplementary Material.

To infer family f_B and to calculate outlier pairwise count, we used 23andme data that is publicly available at the openSNP database (downloaded on March 2015) in which individual identities are anonymized. Files with sizes <15 MB are eliminated, as the genomic data were limited. In total, SNP data of 1200 individuals is used. To obtain the reference and alternate allele information, each file is converted to VCF format by PLINK tool (Purcell *et al.*, 2007). Reference SNP ids chromosome, position, and genotype information are extracted from VCF files. We used the genomic positions that are common in all individuals for our analysis.

3 Results

We present results of solving the optimization problem on two families f_A and f_B for different arrival sequences of the family members using two approaches separately. We considered all sequential arrival orders; here, we present the cases when person A (or person B) arrives the first as these sequences are more challenging as person A (person B) bears genomic similarity to all the other family members. Solutions for the other arrival orders are provided in the Supplementary Material.

The solutions below are displayed in a tree structure, the root indicates the first comer and each branch represents a different arrival sequence of the family members wherein each node represents a person that arrived at that particular time step. A branch stops if no feasible solution exists after the addition of the corresponding family member. At each branch, we only consider adding family members that are blood related at a particular node, as other people can be trivially added.

3.1 Results on solving optimization problem by satisfying kinship constraints

Here, 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 the distance to the outlier values of the population in terms of standard deviations (σ). 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 presented as $o_{10} - 2.50\sigma$. The standard deviation values of the population for each allele pair counts for which one member's SNP type is 1 is as follows; 1432 where one person has SNP type 1 and the other person has 0, 1581 where both individuals have SNP type 1 and finally, 743 where one person has SNP type 1 and the other person has SNP type 2.

Figure 3a shows all the possible sequences of arrival of the family members to the database if person A is the first member arrived while satisfying the imposed kinship constraints. 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 are ignored for $\Phi = 0$, which enforces all family members to be inferred as non-relatives, then ~36.9% utility would be achieved.

Unlike addition of parents, addition of further relatives as second family member does not decrease the outlier constraints drastically. We observe that the solution for the addition of the sister requires a higher decrease in outlier threshold values compared to the solution for addition of the maternal aunt. The reason behind these different outcomes is due to the 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. This arrival sequence {person A- aunt- sister} achieves 56.4% utility. If the addition sequence is {person A—sister—aunt}, the utility value is slightly higher, 56.7%, and the minimum outlier values are lower compared to the former sequence. The o_{12} value is 8.5σ lower, o_{11} value is 1σ lower, and o_{10} value is 2.75σ lower in the latter sequence. This result indicates that arrival sequences result in different feasible solutions when the relatives hold different relationship degrees.

If the second added member is the sister and the third member is one of the parents, then the outlier constraints have to be relaxed too much to attain a feasible solution. In this case, we observed at least 13.5σ, 13.75σ, and 7.75σ leakage in o_{10}, o_{11}, o_{12}, respectively. We also observed 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.

Supplementary Figure S6 illustrates all possible arrival sequences when person B is the first member to arrive at the database. Parents can be added at the second level but the outlier values will be very low, rising the risk of privacy impairment. 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 adding this member is feasible. 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

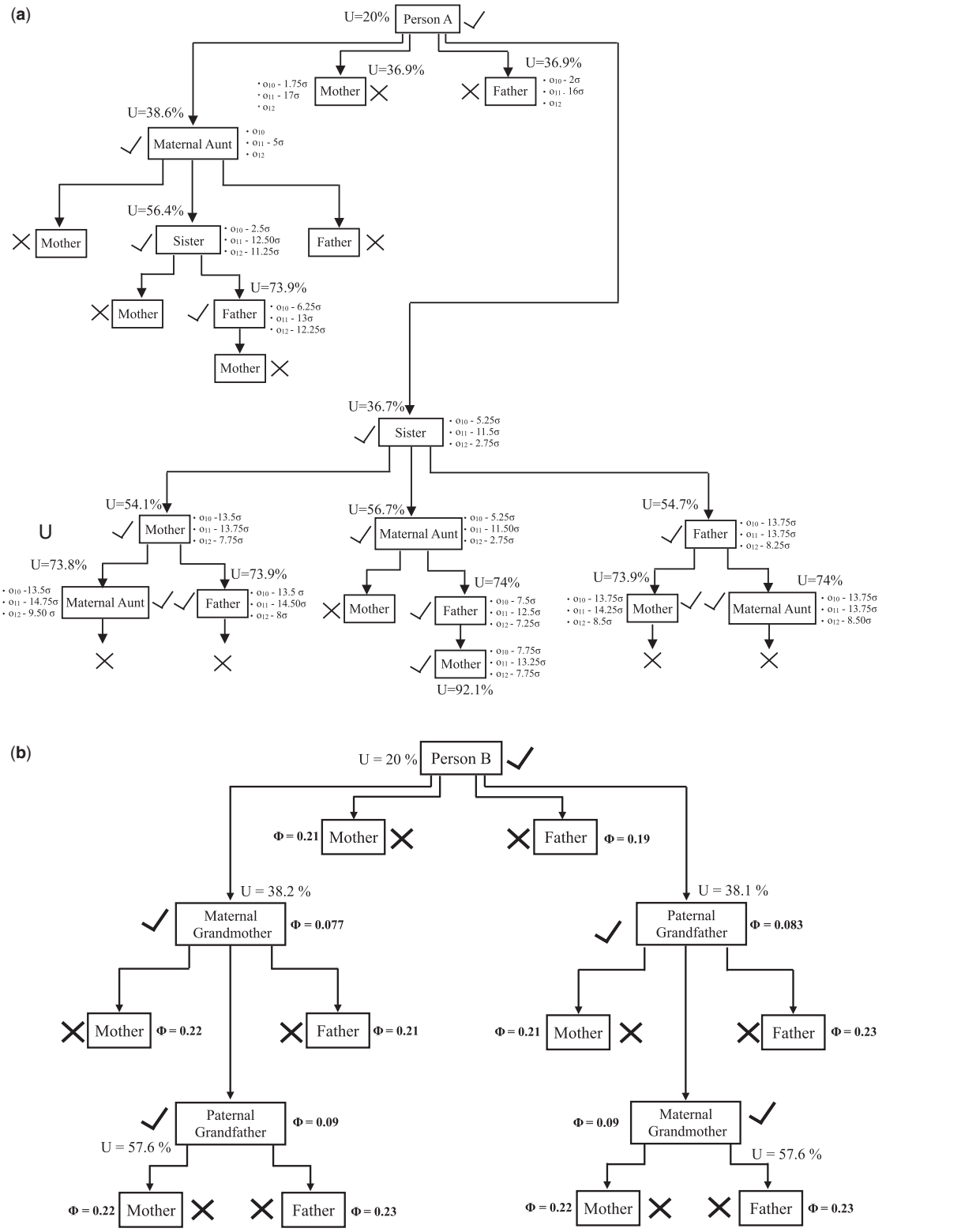

Fig. 3. Solutions when person A or person B arrives first. (**a**) The solutions for family A when the kinship constraints are preserved and the outlier constraints are relaxed. All possible arrival sequences are shown, when the person A is the first added member. Downward arrows point to the subsequent arriving member to the database. Check mark next to a node indicates that the individual could be successfully added without compromising the family's privacy. 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. The relaxed outlier values returned from the optimization problem's solution are shown next to each box. (**b**) Solution when outlier constraints are satisfied and the kinship constraints are relaxed. All possible arrival sequence of family B is shown, when the person B is the first added member. 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, there is at least one kinship value among the family members that reveals the relationship. The minimum possible kinship value, Φ, that can be attained in the solution is added next to each person

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 degree relatives of the person B can be successfully added at any level of the tree when the outlier constraints are relaxed.

3.2 Results on solving optimization problem by satisfying outlier constraints

We present the results when the optimization problem is solved by satisfying the outlier constraints and relaxing the kinship constraints. We proceed such that a person can be successfully added to the database if the relationship of the corresponding individual to the relatives can be reduced by at least one degree (e.g. a parent–offspring relationship is inferred as a 2nd or further degree relationship after the removal of the SNPs).

Figure 3b displays the results for family f_A when the optimization problem is solved based on satisfying the outlier constraints and relaxation of the kinship constraints. We observe that if the parents of person A arrive in the second step, it is impossible to hide the relationship. When more distant relatives such as a sister or a maternal aunt arrives at the database after the arrival of person A, 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: 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 relationship. 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 after the aunt. At this stage of the tree, only the maternal aunt can be added safely, if the second added member is the sister. This scenario achieves 56% utility. In this solution, person A and the sister can be inferred as second degree relatives and the relationships of the maternal aunt to person A and the sister are not revealed. In the 4th level of the tree, none of the family members can be added without hurting family's privacy.

In Supplementary Figure S6, for family f_B all the possible sequences corresponding to the addition of the family members to the database are illustrated, when the person B arrives first. Similar to the f_A case, we observe that when a parent of the person B arrives at the second step, the relationship cannot be hidden. However, if the second added member is one of the grandparents, the addition is rendered successfully; the grandparents and the person B can only be inferred as third degree relatives. Since the person B has lower kinship coefficient with his grandmother, 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 person B. As Figure 3b illustrates, if the second added person is the maternal grandmother, the third added person can only be the paternal grandfather or vice versa with an $\sim$57.6% utility and a maximum kinship coefficient $\Phi = 0.09$ in the family. The addition of a fourth member is not allowed, since none of the family members' relationships can be successfully concealed.

4 Discussion

On the two families we worked with, a solution to the optimization problem that satisfies both types of privacy constraints is often not found. The families we worked with have close relatives; such a solution might be feasible if all family members are more distantly related. We suggest two ways to solve the problem by imposing one of the privacy constraints strictly, meanwhile relaxing the other. Based on the results obtained from the two families, we observed that the concurrent presence of parent and off-spring data in a database constitutes a risk. On the one hand, sharing genomic data of siblings is feasible without compromising privacy. On the other hand, reducing the two siblings' kinship to the level of two unrelated individuals is not possible, but they can be disguised as if they have a second-degree relationship. Genomic data belonging to further distant relatives can be successfully shared together. We observe that the arrival sequence of the family members also affect the results. For instance as shown in Supplementary Figure S5, the sequence {person A, sister, maternal aunt} can be added to the database, but when the arrival sequence is {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 it is possible to share genomic data of up to three people for both families. In family A the siblings are interpreted as second-degree relatives such as aunt–nephew or half-siblings, whereas aunt–nephew relations revealed as unrelated. In family B, grandparent–grandchild relatedness can be hidden as if it is a third-degree relationship such as cousins or great-grandparents. For family A, if the kinship constraints are strictly satisfied and the outlier constraints are relaxed, up to four people can be added successfully without revealing the real relationships. Further addition of a fifth member is possible but this can only be achieved if the outlier constraints are almost neglected. For family B, adding up to three family members was observed to be feasible.

An alternative to the proposed framework would be using encryption techniques and secure computing platforms. However, encryption contradicts with the public availability of genomic data. Although encrypted computing techniques (e.g. homomorphic encryption) allows limited number of operations on encrypted genomic data, to run complex data mining techniques or genome-wide-association studies, researchers need publicly available datasets. Moreover, secure computing platforms relies on trust to a third party that may not be accepted by data owners or legislation.

5 Conclusion and future work

Everyday genomic data are incorporated in various domains including biomedical research, clinical care, and direct-to-consumer services. Realizing the promise of genome sequencing in these domains requires widespread motivation to share genomic data. As the negative stories accumulate, and the fear of potential misuse of genomic data escalates, the public availability 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 is the critical ingredient. Towards this aim, in this work, we specifically focus on privacy risks associated with the kinship of the individuals in genomic databases.

The method developed here can be extended in future work in different directions. In this work, we worked with kinship privacy risk in isolation of other genomic privacy risks. For example, certain positions reveal more information as they are shown to be

associated with disease states or predisposition. 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. However, one should keep in mind that, we do not have the complete knowledge of the genotype-phenotype interactions. A position that seems to release no information about the individual can be associated with a critical disease or behavioral treat in the future. The utility function we propose here is generic, depending on the application it can be modified such that certain positions are down weighted or up weighted; accordingly the objective of the optimization model would need to be redefined. The current work is disregarding the statistical dependencies between genomic positions. The proposed model could be improved by incorporating these correlation structures. As a fourth line of work, the model we developed are based on KING kinship estimator (Manichaikul *et al.*, 2010); thus, is limited with the assumptions of the KING, such as all positions affect the kinship equally, while we would expect rare variants to be more influential in inferring kins. As a future work, the framework can be adapted to other kinship estimates by deriving privacy constraints based on these kinship metrics and updating the optimization models accordingly.

Acknowledgements

The authors would like to thank to Dr Ozlem Cavus (Bilkent University) for valuable discussions.

Funding

Erman Ayday is supported by a funding from the European Unions Horizon 2020 research and innovation programme under the Marie Skodowska-Curie grant agreement No. 707135.

Conflict of Interest: none declared.

References

Belluz,J. (2014) With genetic testing, i gave my parents the gift of divorce. https://www.vox.com/2014/9/9/5975653/with-genetic-testing-i-gave-my-parents-the-gift-of-divorce-23andme (11 July 2017, date last accessed).

Chen,F. *et al.* (2017) Princess: privacy-protecting rare disease international network collaboration via encryption through software guard extensions. *Bioinformatics*, **33**, 871–878.

Clayton,D. (2010) On inferring presence of an individual in a mixture: a Bayesian approach. *Biostatistics*, **11**, 661–673.

Corpas,M. *et al.* (2013) A complete public domain family genomics dataset. *bioRxiv*, doi: 10.1101/000216.

Deznabi,I. *et al.* (2017) An inference attack on genomic data using kinship, complex correlations, and phenotype information. *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, pp. 1–1.

Erlich,Y. and Narayanan,A. (2014) Routes for breaching and protecting genetic privacy. *Nat. Rev. Genet.*, **15**, 409–421.

Goodwin,S. *et al.* (2016) Coming of age: ten years of next-generation sequencing technologies. *Nat. Rev. Genet.*, **17**, 333–351.

Gymrek,M. *et al.* (2013) Identifying personal genomes by surname inference. *Science*, **339**, 321–324.

He,D. *et al.* (2014) Identifying genetic relatives without compromising privacy. *Genome Res.*, **24**, 664–672.

Homer,N. *et al.* (2008) Resolving individuals contributing trace amounts of dna to highly complex mixtures using high-density snp genotyping microarrays. *PLoS Genet.*, **4**, e1000167.

Hormozdiari,F. *et al.* (2014) Privacy preserving protocol for detecting genetic relatives using rare variants. *Bioinformatics.*, **30**, i204–i211.

Huff,C.D. *et al.* (2011) Maximum-likelihood estimation of recent shared ancestry (ERSA). *Genome Res.*, **21**, 768–774.

Humbert,M. *et al.* (2013) 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.

Jacobs,K.B. *et al.* (2009) A new statistic and its power to infer membership in a genome-wide association study using genotype frequencies. *Nat. Genet.*, **41**, 1253–1257.

Johnson,A. and Shmatikov,V. (2013) Privacy-preserving data exploration in genome-wide association studies. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1079–1087. ACM.

Lindor,N.M. (2012) Personal autonomy in the genomic era. In *Video Proceedings of Mayo Clinic Individualizing Medicine Conference*.

Lumley,T. and Rice,K. (2010) Potential for revealing individual-level information in genome-wide association studies. *Jama*, **303**, 659–660.

Manichaikul,A. *et al.* (2010) Robust relationship inference in genome-wide association studies. *Bioinformatics*, **26**, 2867.

Naveed,M. *et al.* (2015) Privacy in the genomic era. *ACM Comput. Surv.*, **48**, 6.

Purcell,S. *et al.* (2007) PLINK: a tool set for whole-genome association and population-based linkage analyses. *Am. J. Hum. Genet.*, **81**, 559–575.

Simmons,S. and Berger,B. (2016) Realizing privacy preserving genome-wide association studies. *Bioinformatics*, **32**, 1293–1300.

Tramèr,F. *et al.* (2015) Differential privacy with bounded priors: reconciling utility and privacy in genome-wide association studies. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 1286–1297. ACM.

Wan,Z. *et al.* (2017) Expanding access to large-scale genomic data while promoting privacy: a game theoretic approach. *Am. J. Hum. Genet.*, **100**, 316–322.

Xie,W. *et al.* (2014) Securema: protecting participant privacy in genetic association meta-analysis. *Bioinformatics*, **30**, 3334–3341.

Yu,F. *et al.* (2014) Scalable privacy-preserving data sharing methodology for genome-wide association studies. *J. Biomed. Inform.*, **50**, 133–141.