Title
Identifying genetic relatives without compromising privacy

Permalink
https://escholarship.org/uc/item/80j24150

Journal
Genome Research, 24(4)

ISSN
1088-9051

Authors
He, Dan
Furlotte, Nicholas A
Hormozdiari, Farhad
et al.

Publication Date
2014-04-01

DOI
10.1101/gr.153346.112

Peer reviewed

Identifying genetic relatives without compromising privacy

Dan He,[1,4] Nicholas A. Furlotte,[1,4] Farhad Hormozdiari,[1] Jong Wha J. Joo,[2] Akshay Wadia,[1] Rafail Ostrovsky,[1,5] Amit Sahai,[1,5] and Eleazar Eskin[1,3,5,6]

[1]*Department of Computer Science, University of California, Los Angeles, Los Angeles, California 90095,USA;* [2]*Interdepartmental Bioinformatics PhD Program, University of California, Los Angeles, California 90095, USA;* [3]*Department of Human Genetics, University of California, Los Angeles, Los Angeles, California 90095, USA*

The development of high-throughput genomic technologies has impacted many areas of genetic research. While many applications of these technologies focus on the discovery of genes involved in disease from population samples, applications of genomic technologies to an individual's genome or personal genomics have recently gained much interest. One such application is the identification of relatives from genetic data. In this application, genetic information from a set of individuals is collected in a database, and each pair of individuals is compared in order to identify genetic relatives. An inherent issue that arises in the identification of relatives is privacy. In this article, we propose a method for identifying genetic relatives without compromising privacy by taking advantage of novel cryptographic techniques customized for secure and private comparison of genetic information. We demonstrate the utility of these techniques by allowing a pair of individuals to discover whether or not they are related without compromising their genetic information or revealing it to a third party. The idea is that individuals only share enough special-purpose cryptographically protected information with each other to identify whether or not they are relatives, but not enough to expose any information about their genomes. We show in HapMap and 1000 Genomes data that our method can recover first- and second-order genetic relationships and, through simulations, show that our method can identify relationships as distant as third cousins while preserving privacy.

The field of human genetics has undergone a revolution within the past 10 yr with the advent of high-throughput genomic technologies, which can measure human genetic variation at ever-decreasing costs (Matsuzaki et al. 2004; Gunderson et al. 2005; Wheeler et al. 2008). The development of these technologies was driven by the goal to perform genome-wide association studies (GWASs), where genetic variation information is collected from tens of thousands of individuals and correlated with disease status (Risch and Merikangas 1996; Manolio et al. 2008; Hardy and Singleton 2009). These studies have linked thousands of new genes to dozens of diseases (Hindorff et al. 2009). While GWASs have been the most visible application of high-throughput genotyping technologies, other areas have been revolutionized as well. For example, these technologies have allowed researchers to ask fundamental questions about human history (Liu et al. 2006; Reich et al. 2009; Tishkoff et al. 2009), to identify genetic relationships between individuals (Stankovich et al. 2005; Pemberton et al. 2010; Kyriazopoulou-Panagiotopoulou et al. 2011), and to characterize an individual's ancestry (Royal et al. 2010). Over the past few years, a personal genomics industry has been established that provides genetic sequencing, genotyping, and analysis services directly to consumers (Genetics and Public Policy Center 2011).

One service that is currently provided by several personal genomics companies is the identification of relatives. The idea behind this service is that individuals provide genetic samples that are genotyped and then stored in a database. Each of the samples is compared to the other samples, and any pair of individuals that appears to be genetically related is then notified of a genetic match. Unfortunately, this application requires that individuals release or share their genetic data with other individuals or organizations that they may not necessarily trust. Individual-level genetic data are extremely sensitive, because they are considered health information about an individual. Furthermore, since each individual's genetic makeup is unique, an individual can be identified even from only a small fraction of his or her genetic data.

The genetics community has already been shaken by privacy issues with the discovery by Homer et al. (2008) showing that individuals can be identified within a pool of DNA based only on aggregate statistics about the pool (in this case the frequency of variants). This result surprised the genetics community and the National Institutes of Health (NIH), which, in an effort to make the results of NIH research available to the public, had been publicly releasing variant frequency information on GWAS disease and healthy populations. Given an individual's DNA information, the observation of Homer et al. (2008) can be exploited to ascertain if the individual was part of any public GWAS studies and if the individual happened to be in a disease cohort. This would expose the disease status of that individual. Understandably, these observations changed the NIH policy overnight, were widely reported in the media (DNA databases shut down after identities were compromised [Editorial] 2008; Genetic privacy [Editorial] 2013), and initiated much research in the area (McGuire 2008; Jacobs et al. 2009; Sankararaman et al. 2009; Heeney et al. 2011; Kahn 2011; Knoppers et al. 2011). More recently, Gymrek et al. (2013) showed that they can reveal the identity of individuals in genetic reference

[4]**These authors contributed equally to this work.**
[5]**These authors contributed equally to this work.**
[6]**Corresponding author**
E-mail eeskin@cs.ucla.edu

Article published online before print. Article and publication date are at http://www.genome.org/cgi/doi/10.1101/gr.153346.112.

data sets by combining their genetic data with small amounts of data from the individuals—such as their approximate age—and taking advantage of publicly available genetic databases and other data available on the Internet. While it is critically important to protect an individual's privacy, restrictions on sharing genetic data severely limit the promise of high-throughput genomic technologies for personal genomics and medicine (Wang 2011).

In this article, we present a technological solution to the natural tension between privacy and the application of personal genomics technologies by capitalizing on recent breakthroughs in cryptography. We describe a framework that enables individuals who have access to their genomes to identify other individuals to whom they are related while keeping their genetic data private. In this framework, individuals release special-purpose cryptographically protected information about their genome which allows others to determine whether or not they are related to the individual. However, the released information does not contain any useful information about the individual's genome. We demonstrate our methods by inferring relationships in several HapMap populations (The International HapMap 3 Consortium 2010) and 1000 Genomes populations (The 1000 Genomes Project Consortium 2010). Through simulations, we show that our approach can detect relationships as distant as third cousins while preserving privacy.

Results

Identifying genetic relatives without compromising privacy

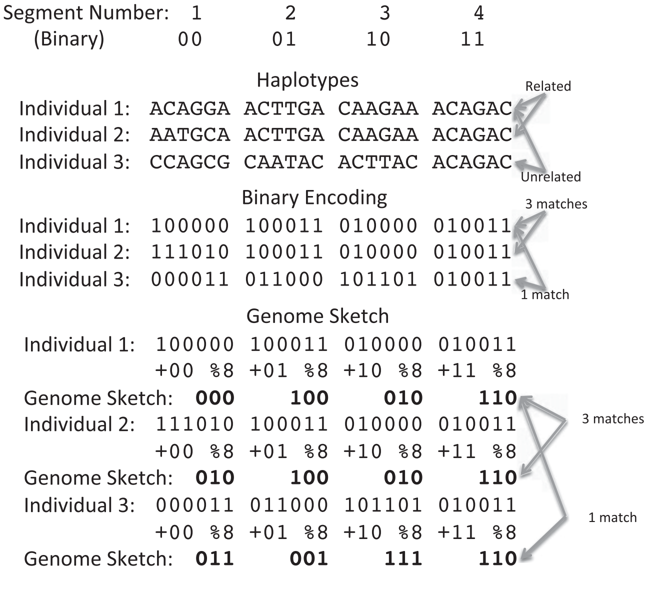
Here we describe a system that enables individuals to discover genetic relatives without revealing any information about their genomes. In our framework, we assume that each individual has access to their own genome. Each individual will publicly release an encrypted version of their genome to a public central repository. An individual will identify relatives by comparing their own genomes to each of the other encrypted genomes in the repository using an algorithm that will inform the individual who in the repository is related to the individual. A key aspect of the framework is that no information about the genomes of the individuals is revealed in the process of identifying relatives—due to the encryption. The way the algorithm works is that the pair of individuals is informed if they share at least a certain number of segments of their genomes, but they do not obtain any information if they do not share enough of the genome. We show below that the amount of segment sharing is related to estimating the fraction of the genome that is identical by descent, which is a traditional approach to identify relatedness.

Our method takes advantage of a new technology referred to as "fuzzy" encryption (Dodis et al. 2008). Our methodology is centered around the concept of a "secure genome sketch" (SGS), which is an encrypted version of an individual's genome and is released publicly. Because of the encryption, a SGS preserves privacy in the sense that it does not reveal information about an individual's genome. Informally, the main idea behind the SGS is that the SGS uses information from an individual's genome as the encryption "key" in the context of a fuzzy encryption scheme. Unlike traditional encryption schemes where the key required for decryption must be identical to the key used in encryption, in a fuzzy encryption scheme, the encryption key and decryption key must only be similar. Thus, other individuals can detect whether or not they are related to the individual by using information from their own genomes to try to decrypt the SGS. If two individuals are related, their genomes will be close enough so that the decryption attempt will allow them to identify that they are related. The threshold required for genome similarity that is required is specified at the time of encryption and is tuned to the level of relatives that the scheme can identify. For example, if the threshold is set at the level of similarity that occurs between first cousins, only individuals who are first cousins or closer can identify their relatives while more distant relatives will not identify each other.

Relative identification by segment matching

The standard approach to identifying whether or not a pair of individuals are closely related is to predict identity-by-descent (IBD) regions between the individuals and then use the amount of shared IBD regions to quantify the degree of relatedness between a pair of individuals (Pemberton et al. 2010). We propose a simple approximation to this scheme that we demonstrate is adequate for identifying relatives and is amenable to the encryption methods proposed. We partition each individual's genome into segments, each consisting of a fixed number of single nucleotide polymorphisms (SNPs). In our experiments on the HapMap and 1000 Genomes data, we use segments consisting of 300 SNPs. We phase each individual's genotypes to obtain the haplotypes for each segment. We approximate the relatedness of two individuals by computing the number of segments where one of the haplotypes matches exactly and refer to this quantity as the number of "segment" matches between a pair of individuals. Below we will explain how we perform this comparison securely.

Figure 1 shows a cartoon example of creating a genome sketch (GS) for three individuals. In this example, for simplicity we assume that each individual has only one chromosome consisting of 24 SNPs, which is split into four segments of six SNPs. In this example, individuals 1 and 2 are related, while individual 3 is unrelated to the two other individuals. In our example, we assume that to be related, two individuals must share the exact haplotype at three out of the four segments. While this example is obviously



```
Segment Number:   1        2        3        4
   (Binary)       00       01       10       11

                       Haplotypes
Individual 1:  ACAGGA  ACTTGA  CAAGAA  ACAGAC
Individual 2:  AATGCA  ACTTGA  CAAGAA  ACAGAC
Individual 3:  CCAGCG  CAATAC  ACTTAC  ACAGAC

                    Binary Encoding
Individual 1:  100000  100011  010000  010011
Individual 2:  111010  100011  010000  010011
Individual 3:  000011  011000  101101  010011

                     Genome Sketch
Individual 1:    100000  100011  010000  010011
                 +00 %8  +01 %8  +10 %8  +11 %8
Genome Sketch:   000        100     010     110
Individual 2:    111010  100011  010000  010011
                 +00 %8  +01 %8  +10 %8  +11 %8
Genome Sketch:   010        100     010     110
Individual 3:    000011  011000  101101  010011
                 +00 %8  +01 %8  +10 %8  +11 %8
Genome Sketch:   011        001     111     110
```

Figure 1. Overview of genome sketch (GS) construction. A simple example of private relative identification consisting of three individuals with their genome split into four segments, each consisting of six SNPs. In this example, individuals are related if they share all but one segment and are unrelated otherwise. The GS is constructed using sketch elements of length 3 bits.

much smaller in scale than the full genome, we can use it to illustrate our cryptographic scheme for relative identification.

Genome sketches

We define a "genome sketch" (GS) as a representation of an individual's segments that allows us to compute the number of segment matches between a pair of individuals without revealing the full genetic information of an individual. A GS is obtained by converting the values of the haplotypes for each segment into a pair of binary numbers, where each digit represents a SNP position in the segment, and where zero represents the major allele and one represents the minor allele.

In our example, the haplotypes correspond to 6 bit values. We also incorporate the location of the haplotype in the genome by considering the segment number also represented as a binary number. In our example, since we have four segments, the segment number can be represented by 2 bits with values for the first through fourth segment as 00, 01, 10, and 11. We combine the information of the haplotype alleles and haplotype location by summing the binary numbers corresponding to the haplotype value and segment location. We can compare two individuals by computing how many of these values are common to both individuals. A common value is an indicator that in some segment of the genome, the two individuals share the same haplotype.

The last step of the GS construction is to apply a technique called collision-resistant hashing to each of our values. A collision-resistant hashing function deterministically transforms a source binary value into a target binary value, typically of shorter length, which has the property that if two source values are close together (for example, they differ by only one position), the resulting target values will be very different from each other. The set of resulting hashed values compose an individual's GS, and each value is referred to as a sketch element. Since the hashing function is deterministic, if two individuals share a haplotype in their genome, their corresponding GS element will be identical. However, because of the collision-resistant hashing, if the haplotypes between a pair of individuals differ by even a single SNP allele, the corresponding sketch elements will be very different.

Comparing GSs from two individuals by counting the number of overlaps (also referred to as the "set distance") closely estimates the number of segments where the two individuals have a shared haplotype. This estimate is a slight overestimate because of the possibility that two different haplotypes in different locations in the genome can be hashed to the same sketch element. In the terminology of hashing functions, this is referred to as a collision. Below, we show that the number of collisions in our real data experiments is very small.

In our example in Figure 1, a GS is constructed by first summing the binary representation of the haplotype in each segment with the segment number. For clarity of the example, our hash function simply takes the last three digits of this sum as the GS element. This operation is referred to as "modular 8" and is abbreviated as "%8" in the figure. The GS is the set of these values for each individual. Note that for individual 2, there was a collision in the hashing between the first and third segment which resulted in only three GS elements.

The full GS of an individual can be represented either as a set or as a vector of size 2^k, where k is the number of possible sketch values. Figure 2 shows the conversion of the GSs for each individual into a binary vector of length 8. Each position in the vector corresponds to a potential sketch element, and the vector has a one if the individual's GS contains that element and has a zero otherwise. The number of positions that match between the GS vectors of a pair of individuals is closely related to the number of matching segments.

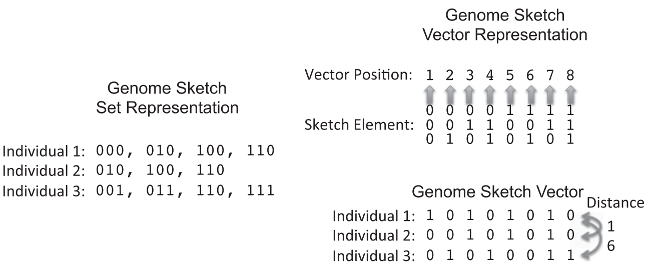

Genome Sketch
Set Representation
Individual 1: 000, 010, 100, 110
Individual 2: 010, 100, 110
Individual 3: 001, 011, 110, 111
Genome Sketch
Vector Representation
Vector Position: 1 2 3 4 5 6 7 8
Sketch Element:
0 0 0 0 1 1 1 1
0 0 1 1 0 0 1 1
0 1 0 1 0 1 0 1
Genome Sketch Vector
Individual 1: 1 0 1 0 1 0 1 0
Individual 2: 0 0 1 0 1 0 1 0
Individual 3: 0 1 0 1 0 0 1 1
Distance
1
6

Figure 2. Conversion of GS sets into vectors. The GS consisting of elements of length 3 bits can be converted into a vector representation of length $2^3 = 8$.

We note that reverse engineering an individual's GS to recover the individual's genome is very difficult because of the hashing. However, with access to an individual's GS, it is possible to query whether or not the individual has a specific haplotype. This is done by converting the haplotype to a GS element and then checking to see if that element matches an element in the GS. Since even unrelated individuals share some IBD regions, some genetic information will be compromised. For this reason, each individual keeps their GS private. In our example in Figure 1, if individual 3 has access to the GS of individual 2, individual 3 can infer that they have the same haplotype in the fourth segment because they share the GS value "110." Furthermore, an individual can use the GSs of publicly available genetic data sets, such as those from the 1000 Genomes project (The 1000 Genomes Project Consortium 2010) or HapMap (The International HapMap 3 Consortium 2010), and obtain genetic information about all regions that are IBD with any individual in the database.

Secure GSs

We address the privacy issue of GSs by using a relatively new cryptographic construct called a "secure sketch." A secure sketch is a construct that allows for the computation of a set distance between two sketches only if their distance is within a certain threshold (for a further discussion of secure sketches, see Dodis et al. 2008 and references therein). The ideas underlying our encryption scheme are closely related to the theory of error-correcting codes (ECCs) (Huffman and Pless 2003).

In our approach, users will have access to their own GSs, which they will keep private. Users will also create what we will call a "secure genome sketch" (SGS) using their GS as a starting point, which they will make public. The way users will determine whether they are not related to another individual is to obtain that individual's SGS and then attempt to use their own GS to check if they are related.

Figures 3 and 4 illustrate a simplified example of our system continuing the example from Figures 1 and 2. In our example, there are three individuals; the first two individuals are related and the third individual is unrelated. GSs are generated with the aid of an ECC matrix that is the same width as the length of the GS vector. Figure 3 shows an example of an ECC matrix, which in this case is the famous Hamming code (7,4) with a parity bit. Each row of the ECC matrix is referred to as a codeword. ECCs are widely used in wireless communications, where the goal is to transmit signals accurately and be robust to errors. This code is designed to send a 4-bit message (the first 4 bits of the code). The remaining four columns are designed in such a way that they allow for errors

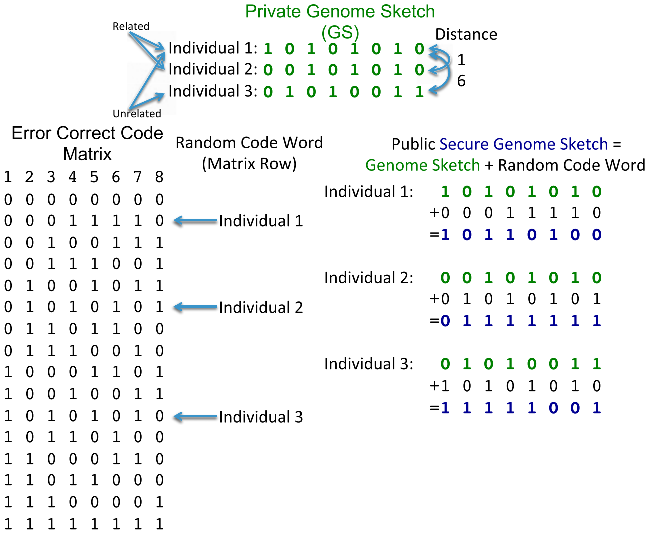


Figure 3. Encoding of GSs into secure genome sketches (SGSs). The GS for the three individuals is converted unto a SGS by adding a random codeword (matrix row) selected from an error-correcting code. Instead of addition, the figure uses the exclusive OR operation for clarity. These SGSs are then made public. Information that is kept private (GSs) is colored green; information that is publicly released (SGSs) is colored blue.

in the communications but still retain the ability for recovering the message. For example, if someone wanted to transmit the message "0010," they would use the coding matrix to convert the message to the 8-bit codeword "00100111" and transmit the codeword. If in the transmission there was an error in the fourth position that resulted in the received signal "00110111," the receiver can still recover the correct message by using the matrix to "decode" the transmission by finding the row that most closely matches the signal. In this case, the only row of the matrix that matches the signal with one error is the correct row, and this allows for the recovery of the message. On the other hand, if there were three errors in the signal that resulted in "10000110," that would mean that the signal could not be decoded since four rows would match with two errors.

To generate a GS, an individual randomly selects a row of the matrix and sums the row with his or her GS (Fig. 3). This resulting SGS is then made public. To then identify a relationship, an individual would obtain a public GS from another individual and subtract their own GS (Fig. 4), resulting in what is called a "relationship message." They would then attempt to use the code matrix to decode the resulting relationship message. If the decoding is successful—that is, the result closely matches a row in the coding matrix—this implies that the individuals are related. If the decoding is unsuccessful, this implies that the individuals are unrelated. The intuition is that if the two individuals are related, then the difference between their genomes is small and what is decoded will be close to a matrix row or codeword. On the other hand, if the individuals are unrelated, then their GSs are far apart, and thus with probability very close to 1, the relationship message will not be close to any codeword. The small probability of failure arises from the fact that if the sum of two GSs is a codeword, then, even if they are far apart, the resulting relationship message will itself be a codeword, leading to a false match. However, in practice, this happens with probability very close to zero because intuitively, the number of possible GSs is much larger than the number of possible codewords.

In the example, individual 1 randomly selected the second matrix row, individual 2 randomly selected the sixth matrix row, and individual 3 randomly selected the eleventh matrix row (Fig. 3). These choices were then summed to their GSs to make the public SGSs. In our example, we demonstrate the process of individual 1 to identify relatives. Individual 1 would obtain both public SGSs from individuals 2 and 3. Individual 1 then subtracts his or her own private GS from each of these SGS and attempts to decode the result using the coding matrix. Instead of addition and subtraction, we use the exclusive OR operation for clarity of the figure. The exclusive OR results in a zero when the two digits match and a one otherwise. Note that when attempting to decode the result from individual 2, the decoding is successful and identifies the sixth row as the closest match. This is exactly the row that individual 2 chose randomly when creating the SGS. The reason why this decoding is successful is that the difference between the GS of individual 1 and individual 2 is small enough for the ECC to still decode successfully. The fact that the decoding is successful allows individual 1 to know that individual 2 is a relative. When attempting to decode the result from individual 3, the decoding is unsuccessful and there are four rows that are equidistant from the result. This implies that the GSs of individual 1 and individual 2 are farther apart than the distance that the error correct can decode, and thus the individuals are unrelated. The ability to successfully decode a vector is related to the distance between rows or codewords in the ECC. A key idea behind our scheme is that we utilize an ECC such that the distance is set so that only pairs of individuals that are within the relatedness threshold can successfully decode their SGSs.

In our simple example, there are only four segments and sketch elements are only 3 bits long. However, in our real data experiments, we have a much larger number of segments and sketch elements are 24 bits long. This significantly increases the computational complexity of both encoding and decoding the GSs. In order to scale to the genome, we utilize an improved version of the Juels-Sudan construction (Dodis et al. 2008). Computing the similarity of GSs involves comparing the overlap of sets of 24-bit vectors. This can be thought of as computing the Ham-

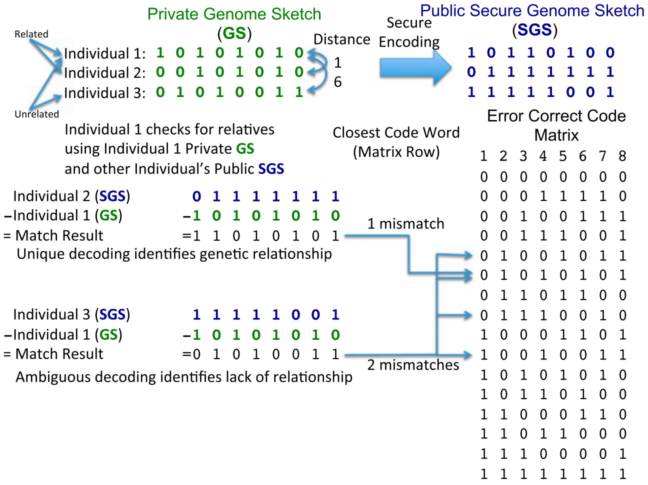

Private Genome Sketch
(GS)
Distance
Secure Encoding
Public Secure Genome Sketch
(SGS)
Related
Unrelated
Individual 1: 1 0 1 0 1 0 1 0
Individual 2: 0 0 1 0 1 0 1 0
Individual 3: 0 1 0 1 0 0 1 1
1
6
1 0 1 1 0 1 0 0
0 1 1 1 1 1 1 1
1 1 1 1 1 0 0 1
Error Correct Code Matrix
Individual 1 checks for relatives using Individual 1 Private GS and other Individual's Public SGS
Closest Code Word (Matrix Row)
1 2 3 4 5 6 7 8
0 0 0 0 0 0 0 0
0 0 0 1 1 1 1 0
0 0 1 0 0 1 1 1
0 0 1 1 1 0 0 1
0 1 0 0 1 0 1 1
0 1 0 1 0 1 0 1
0 1 1 0 1 1 0 0
0 1 1 1 0 0 1 0
1 0 0 0 1 1 0 1
1 0 0 1 0 0 1 1
1 0 1 0 1 0 1 0
1 0 1 1 0 1 0 0
1 1 0 0 0 1 1 0
1 1 0 1 1 0 0 0
1 1 1 0 0 0 0 1
1 1 1 1 1 1 1 1
Individual 2 (SGS) 0 1 1 1 1 1 1 1
−Individual 1 (GS) −1 0 1 0 1 0 1 0
= Match Result =1 1 0 1 0 1 0 1
Unique decoding identifies genetic relationship
1 mismatch
Individual 3 (SGS) 1 1 1 1 1 0 0 1
−Individual 1 (GS) −1 0 1 0 1 0 1 0
= Match Result =0 1 0 1 0 0 1 1
Ambiguous decoding identifies lack of relationship
2 mismatches

Figure 4. Decoding of SGSs to identify relatives. An individual identifies relatives by obtaining the public SGS from other individuals and subtracts his or her own GS and attempts to decode the result using the coding matrix. Instead of addition, the figure uses the exclusive OR operation for clarity. If the decoding is successful, the individuals are related. If the decoding is unsuccessful, the individuals are unrelated. Information that is kept private (GSs) is colored green; information that is publicly released (SGSs) is colored blue.

ming distance between length 2^{24}-bit vectors, each representing the GS of an individual where each position represents a specific 24-length vector, and the bit is 1 if the individual contains that GS element and zero otherwise. Similarly, the ECC matrix will have width 2^{24}. The distance between words of the code matrix is twice the difference of the threshold between related and unrelated individuals. A major advantage of our method is that it provides an efficient algorithm for both encoding and decoding a GS represented as a set.

Identification of parent–child relationships in the HapMap data

We demonstrate our methodology using two populations from the HapMap Phase 3 data which contain related individuals (The International HapMap 3 Consortium 2010). We use the CEU (European) and YRI (African) populations, which have different degrees of linkage disequilibrium, to highlight the robustness of our approach. The CEU population consists of 165 individuals made up of 96 related pairs and 13,434 unrelated pairs. The YRI population is made up of 167 individuals and contains 104 related pairs and 13,757 unrelated pairs. We filter SNPs with <5% minor allele frequency and any SNPs that have more than three alleles. This results in 1,387,466 SNPs, which are partitioned into 4625 segments of 300 SNPs. In our simulation, we assume that each individual has access to his or her genome and wants to identify any relatives without revealing their genetic information. In our simulations, each individual generates a SGS that they make public.

When the data set was constructed, it was assumed that the remaining individuals were unrelated, but recent studies have identified many unannotated relationships (Pemberton et al. 2010). We apply KING (Manichaikul et al. 2010), a method for predicting genetic relationships from whole-genome data sets, to identify the unannotated genetic relationships and eliminate these pairs from consideration. This results in the elimination of 27 unrelated pairs from the CEU data set and 12 unrelated pairs from the YRI data set, which is consistent with previous attempts to identify the unannotated relationships.

We first show that the number of segments matching differentiates related individuals from unrelated individuals. We partition each individual's genome into 4625 segments, each containing 300 SNPs. Figure 5, A and B, shows a histogram of the number of matches between the related and unrelated pairs of individuals in the HapMap samples. The threshold of 450 separates the related individuals from unrelated individuals. We note that shared IBD regions between close relatives are typically longer than our segments and would likely span several neighboring segments.

Figure 6, A and B, demonstrates that hash collisions have a very small effect on the relative distance between the related and unrelated individuals. For most pairs, the difference between the number of GS overlaps and the segment overlaps is less than 10 in the HapMap data. This is much smaller than the difference between the related and unrelated individuals (Fig. 5A,B).

In our scheme, each individual obtains the set of secure sketches from all of the remaining individuals and applies the decoding software to compare their own genome to the secure sketch of each of the other 321 individuals. The total number of comparisons performed is 109,892. We omit performing the comparisons on the 27 ambiguous relationships in the CEU population and the 12 ambiguous relationships in the YRI population. Forty-eight of the CEU individuals and 54 of the YRI individuals are children in trios, and we correctly identify both of their parents. The parents each correctly identify a genetic relationship with their children. In no cases do we incorrectly predict a genetic relationship among individuals who are not related. When performing the comparisons, no genetic information was revealed to the other individuals.

Identification of second-order genetic relationships in the 1000 Genomes data set

While the 1000 Genomes Project originally intended to sequence unrelated individuals, from the resulting sequence data generated in the project, it became apparent that some of the individuals are related in some of the African populations. Specifically in the ASW population, there are 61 individuals, and the relationships among the individuals contain two second-order relationships and three sibling relationships. In the LWK population, there are 97 individuals, and the relationships contain five second-order relationships, six sibling relationships, and four parent–child relationships. We merge these two populations with the YRI population and analyze them together to demonstrate that our approach can recover

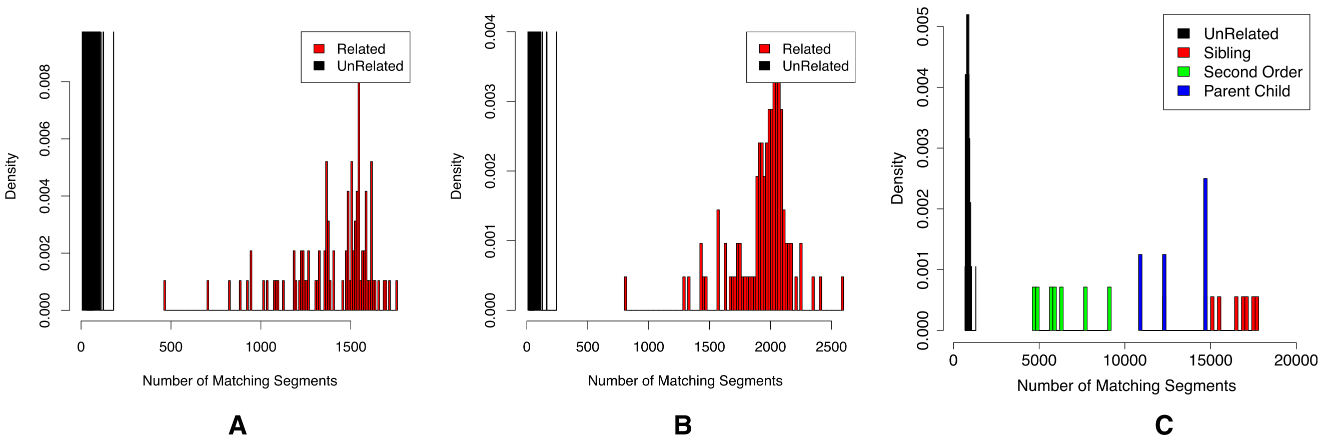


Figure 5. The number of segment matches can be used to determine if individuals are related. We split the genomes of each individual into segments of length 300 SNPs and then compared the number of segments where the haplotypes match exactly between any two individuals in the HapMap data and 1000 Genomes data. Related individuals have a much higher number of matches when compared to unrelated individuals. (*A*) CEU (HapMap). (*B*) YRI (HapMap). (*C*) AFR (1000 Genomes).

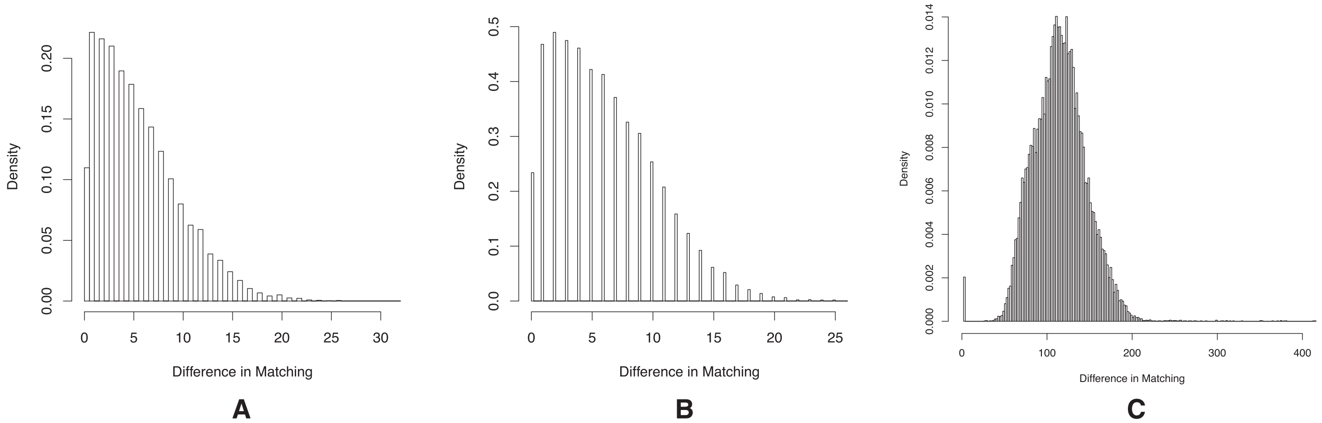


Figure 6. The number of common GS elements between two individuals is close to their number of segment matches. We measure the difference between the number of common GS elements and the number of segment matches in the HapMap data and 1000 Genomes data. The differences are small compared with the distance between related and unrelated individuals. (*A*) CEU (HapMap). (*B*) YRI (HapMap). (*C*) AFR (1000 Genomes).

more distant relationships. After filtering our SNPs with a minor allele frequency of <5% and any markers with more than two alleles, we partition the 1000 Genomes data into segments of 300 SNPs, resulting in 29,004 segments.

Figure 5C shows the number of segment overlaps between pairs of related and unrelated individuals. For clarity of the figure, we only plot the histogram for 20 unrelated individuals. We note a very large separation between the related and unrelated individuals. We note that many unrelated individuals share many segments, which is expected because the genomic region corresponding to 300 SNPs is much shorter in the 1000 Genomes data compared with the HapMap data, due to the difference in the total number of SNPs. We use a threshold of 3000 to separate related and unrelated individuals.

Similar to the HapMap data, the difference between the number of sketch overlaps and segment overlaps is very small compared to the difference between related and unrelated individuals. Figure 5C shows that for most pairs, the difference between the number of GS overlaps and the segment overlaps is less than 250.

Identification of more distant relatives in simulated data

The more distantly a pair of individuals is related, the fewer of their segments will have exactly matching haplotypes. We generated simulated data using the 1000 Genomes data as a starting point to identify at what point is the amount of segment overlaps between related individuals indistinguishable from unrelated individuals. The results of our simulation study are shown in Figure 7. As can be shown, our method is applicable up to third cousins since the threshold of 1500 separates them from unrelated individuals. We also generated pairs of individuals who are fourth cousins and observed that some of them had sharing at the same level of individuals who are unrelated (data not shown), which implies that third cousins are the limit for this encoding scheme.

Security of SGSs

A general question follows: How secure are SGSs? We refer to "security" in the cryptographic sense. This is equivalent to asking how difficult is it to reverse engineer a SGS to a GS and similarly how difficult is it to reverse engineer a GS into an actual genome? This question can be addressed in a very general way by considering the relative amount of information in the individual's GS compared to the amount of information publicly released in an individual's SGS. The "amount of information" is quantified in terms of "entropy," or the number of bits required to encode the information. The security of the encryption scheme will depend on the entropy in the original data, which we refer to as the "total entropy," as well as the amount of information that is released through the encryption scheme, which we refer to as the "entropy loss." Since entropy is additive, the "remaining entropy" is the difference between the "total entropy" and the "entropy loss."

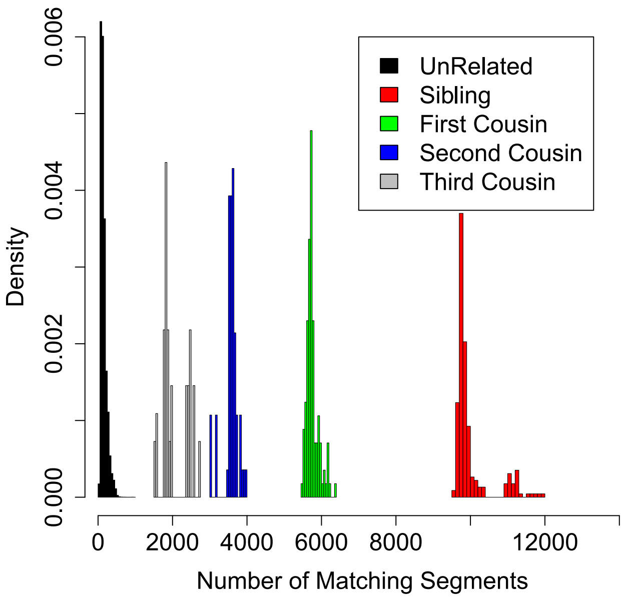


Figure 7. The number of segment matches for different degrees of relatives. We created simulated data by generating related individuals using the 1000 Genomes data as a starting point and tuning the simulated genotype error rate and haplotype phasing error rate to match observed amounts of segment matching for corresponding relationships in the real data. The simulation shows that our approach can distinguish between pairs of unrelated individuals and individuals related up to third cousins.

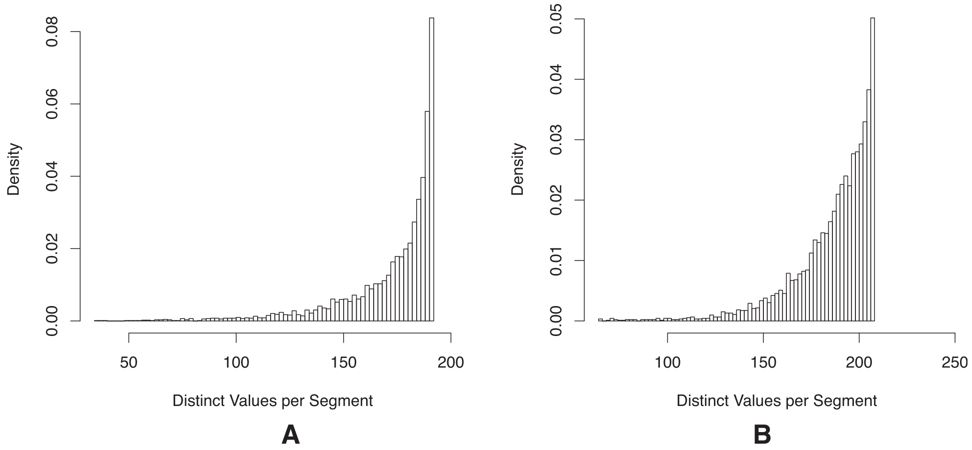


Figure 8. Histogram of the number of different values per segment in population for unrelated individuals. We consider the 96 parents in the CEU trios and the 104 parents in the YRI trios. For each segment, we count the number of different values within a segment. The maximum possible is twice the number of individuals (192 in CEU and 208 in YRI) in the case in which each individual has a different value on each chromosome. The histograms show that the vast majority of segment values differ between unrelated individuals. (*A*) CEU (HapMap). (*B*) YRI (HapMap).

The amount of information released as part of an individual's SGS, or entropy loss, depends on the cryptographic scheme used to perform the encryption and the "relatives" threshold that we must recognize. In the strategy that we are utilizing, the amount of entropy remaining is $\frac{t^2}{s}$, where t is the threshold required for matching relatives and s is the number of segments under the assumption that each sketch element itself contains m bits of entropy, where m is the length of the element.

In the HapMap data, a GS consists of 4625 segments, the threshold for similarity is 450, and the remaining entropy is 43 bits. For the 1000 Genomes simulations, we obtain much more secure encryption. In our experiments using the 1000 Genomes data, we have 29,004 segments with a threshold of 3000, and the remaining entropy is 310 bits. If we are searching for third cousins, we would use a threshold of 1500 and the remaining entropy would be 77 bits.

Our security rests on the assumption that the amount of entropy in each sketch element is more than 24 bits. If we were able to obtain a complete distribution for haplotypes for the human population in each segment, we could directly measure the amount of entropy in the GS. Unfortunately, since we only have access to a finite number of individuals, it is impossible to accurately measure this entropy. However, the amount of entropy is likely very high because in our data set almost every unrelated individual has unique values for most segments, as shown in Figure 8. Therefore, we expect the amount of entropy in the GS to far exceed the amount of entropy loss in our approach, thus providing a significant amount of security. We note that if the entropy of each sketch element is smaller than 24 bits, this scheme can be adjusted using different thresholds to still guarantee security.

Discussion

We have proposed a new approach for addressing the inherent tension between privacy and data sharing in personal genomics that leverages recent developments in cryptography, and we demonstrate how these developments can be used to identify genetic relationships while preserving privacy. The key idea of our approach is that each individual releases specially encrypted information about their genome, which allows for other individuals to identify if they are related, but the information does not reveal any information about the individual's genome in the event they are not.

We demonstrated our approach using two populations from HapMap and two populations from the 1000 Genomes Project with very different linkage disequilibrium structures and known genetic relationships. The HapMap data sets contain many parent–child relationships, which we are able to detect without any false positives. The 1000 Genomes Project contains a smaller number of more distant relationships such as cousins, and we demonstrate that we are able to recover these relationships as well. We also generated simulated data containing more distant relatives using the 1000 Genomes data as a starting point to show that our approach can detect genetic relationships as distant as third cousins.

In our experiments, we used haplotype segments of length 300 as the basic unit of identifying genetic relationships. In principle, we can use any segment length as long as there is an adequate separation in the amount of sharing between related and unrelated individuals and the segments are long enough to contain enough entropy after being hashed. In the data sets, we examined, segments of 300 were adequate to identify up to third cousins while preserving privacy. However, other strategies for encoding the genome, including changing the segment length, may lead to the ability to detect even more distant relationships.

We note that our method first infers haplotypes, and the method for determining relatedness measures the amount of haplotype segments shared. An error in haplotype inference will decrease the amount of segment sharing between related individuals because, due to the error, a similar segment will appear to be different. Since the reference data sets for haplotype inference are the HapMap and 1000 Genomes data sets, which are the same data sets that are utilized for our experiments, there is a possibility that the haplotypes used in our experiments may be more accurate than haplotypes that would be used in practice. Our experiments adjust for this possible bias by using only the haplotypes from other populations as the reference data set for haplotype inference. As haplotype inference techniques improve, the estimation of relatedness by segment sharing will become more accurate which will make it easier to identify more distant relationships. We note that genotype errors will also decrease the amount of segment sharing between related individuals.

The recent development of sequencing technology allows for the cost-effective collection of rare variants from an individual. This technology has implications for relative identification because it allows for utilizing rare variants to identify segments that are identical by descent. However, rare variants complicate the application of this technique because many of them are unlikely to be discovered in advance, which will require novel methods for constructing GSs.

In our approach, if two individuals are unrelated, they cannot obtain any information about each other's genome. However, our current implementation can be utilized to reveal exactly the shared genomic regions between a pair of related individuals. The reason is that when a SGS is successfully decoded, the number of errors between the difference of the SGS and an individual's GS and the error-correcting codeword is obtained. This number of errors corresponds to the number of segments that differ between the individuals. An individual can then perform the decoding leaving out one element of their GS each time and observe when the number of errors increases. Each time the number of errors increases, the individual can infer that the corresponding haplotype is present at the corresponding segment of the individual. Thus an individual can obtain information about which parts of the genome are identical by descent with a relative. Using a similar approach, individuals can obtain the GS elements of the remaining segments of the individual's genome, which they can then query against public databases to identify any segment matches with public data. We can remedy this problem by using a secure computation approach (e.g., see Ishai et al. 2011 and the references therein); this is a direction for future work.

Methods

HapMap Phase 3 data

We used the CEU and YRI genotypes from release 28 of the HapMap Phase 3 data (The International HapMap 3 Consortium 2010). The relationships are obtained from the pedigree information available in the data. Since we also use the HapMap data as a reference for performing phasing, we phase and impute missing data in each population by using BEAGLE (Browning and Browning 2009) imputation using the remaining populations as the reference sets. This avoids any bias from inclusion of a sample in the reference data sets. The individuals are genotyped at 1,387,466 SNPs.

1000 Genomes data

We use the three African populations in the Phase I v3 1000 Genomes data: ASW, LWK, and YRI (The 1000 Genomes Project Consortium 2010). We use the haplotypes available as part of the 1000 Genomes data release. After filtering variants that have <5% minor allele frequency, the data set contains 8,698,118 SNPs.

Simulated data

We generated simulated data using the same ASW, LWK, and YRI 1000 Genomes populations as a starting point. We generated pedigrees where the founders are unrelated individuals and randomly mated them, simulating a recombination rate of 10^{-7} per base pair, and we assume that at each genotype collected there is an error rate in either phasing or genotyping at a rate of 0.008. This error rate was tuned so that our simulated data have similar properties to the amount of segment sharing in siblings and cousins in the real data. We generated pedigrees large enough so we could estimate the amount of sharing among siblings through fourth cousins.

Genome sketches

Haplotypes for each individual are partitioned into segments of length 300 SNPs. In the HapMap data, this corresponds to 4625 segments, and in the 1000 Genomes data, this corresponds to 29,004 segments. The alleles for each individual at each haplotype are converted to a binary representation and are presented by a pair of 300-bit values. These values are then summed to the segment number which is represented by a 13-bit or 15-bit number in the HapMap or 1000 Genomes data, respectively. This number is added to a fixed 100-bit value called a salt. The salt is a random 100-bit number that is public and used for the encoding of all individuals. This resulting 300-bit value is then hashed using the SHA-256 secure hash algorithm (NIST 2008), and the first 24 bits from the hash are saved to comprise the GS corresponding to the haplotype. Note that because of the SHA-256 hashing, even two haplotypes in the same region that differ by only one SNP will be hashed to completely different values, thereby creating GS elements that are completely different.

Secure genome sketches

In our construction, we use an improved version of the Juels-Sudan construction (Dodis et al. 2008) to convert our GSs into SGSs using a threshold of 450 for the HapMap data and 3000 for the 1000 Genomes data. We refer to the approach as IJS. Instead of using unique decoding of ECCs, as we described in Figures 3 and 4, IJS uses list decoding, followed by a hash check. Individuals can then make public their SGS and then compare their GS to another individual's SGS using IJS to determine if the GSs are within a distance of the threshold that identifies a genetic relationship. However, if the distance is greater than the threshold, no information about the genome is revealed.

SGSs utilize the approach described in Figures 3 and 4. An individual's set of sketch elements can be represented as a bit vector of length 2^{24}, with ~9250 elements with a value of one and the remaining with a value of zero. Our approach does not explicitly represent an individual's GS as this vector, but instead represents an individual by keeping track of which are the nonzero values of the bit vector that correspond to the set of sketch elements. Similarly, we do not explicitly represent the coding matrix of width 2^{24}. The main insight of our approach is to take advantage of the fact that even though the space of possible GSs is huge $\left(2^{2^{24}}\right)$, each individual's GS will only be nonzero at a number of positions equal to twice the number of segments. We are able to take advantage of this sparsity to efficiently perform encoding and decoding.

Software availability

Software implementing the methods described in this paper is available at http://genetics.cs.ucla.edu/crypto/.

Acknowledgments

N.A.F., D.H., F.H., J.W.J.J., and E.E. are supported by National Science Foundation (NSF) grants 0513612, 0731455, 0729049, 0916676, and 1320589 and National Institutes of Health (NIH) grants K25-HL080079, U01-DA024417, P01-HL30568, and PO1-HL28481. N.A.F. is supported by NIH training grant 2T32NS048004-06A1. E.E., A.S., and R.O. are supported by NSF grant 1065276. A.S. and

R.O. are supported by NSF grants 1136174, 0916574, and 0830803 and a Xerox faculty research award. A.S. is supported in part by a DARPA/ONR PROCEED award and NSF grants 1228984 and 1118096. R.O. is supported by NSF grants 1016540 and 1118126 and USA–Israel BSF grant 2008411. We acknowledge the support of the NINDS Informatics Center for Neurogenetics and Neurogenomics (P30 NS062691). This material is based upon work supported by the Defense Advanced Research Projects Agency through the U.S. Office of Naval Research under contract N00014-11-1-0389. The views expressed are those of the authors and do not reflect the official policy or position of the Department of Defense, the National Science Foundation, the National Institutes of Health, or the U.S. Government.

References

The 1000 Genomes Project Consortium. 2010. A map of human genome variation from population-scale sequencing. *Nature* **467:** 1061–1073.

Browning BL, Browning SR. 2009. A unified approach to genotype imputation and haplotype-phase inference for large data sets of trios and unrelated individuals. *Am J Hum Genet* **84:** 210–223.

DNA databases shut after identities compromised. [Editorial] 2008. *Nature* **455:** 13.

Dodis Y, Ostrovsky R, Reyzin L, Smith A. 2008. Fuzzy extractors: how to generate strong keys from biometrics and other noisy data. *SIAM J Comput* **38:** 97–139.

Genetic privacy.[Editorial] 2013. *Nature* **493:** 451.

Genetics and Public Policy Center. 2011. Alphabetized genetic testing companies. http://www.dnapolicy.org/resources/Alphabetized DTCGeneticTestingCompanies.pdf.

Gunderson K, Steemers F, Lee G, Mendoza L, Chee M. 2005. A genome-wide scalable SNP genotyping assay using microarray technology. *Nat Genet* **37:** 549–554.

Gymrek M, McGuire AL, Golan D, Halperin E, Erlich Y. 2013. Identifying personal genomes by surname inference. *Science* **339:** 321–324.

Hardy J, Singleton A. 2009. Genomewide association studies and human disease. *N Engl J Med* **360:** 1759–1768.

Heeney C, Hawkins N, De Vries J, Boddington P, Kaye J. 2011. Assessing the privacy risks of data sharing in genomics. *Public Health Genomics* **14:** 17–25.

Hindorff L, Sethupathy P, Junkins H, Ramos E, Mehta J, Collins F, Manolio T. 2009. Potential etiologic and functional implications of genome-wide association loci for human diseases and traits. *Proc Natl Acad Sci* **106:** 9362.

Homer N, Szelinger S, Redman M, Duggan D, Tembe W, Muehling J, Pearson J, Stephan D, Nelson S, Craig D. 2008. Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays. *PLoS Genet* **4:** e1000167.

Huffman W, Pless V. 2003. *Fundamentals of error-correcting codes*. Cambridge University Press, Cambridge.

The International HapMap 3 Consortium. 2010. Integrating common and rare genetic variation in diverse human populations. *Nature* **467:** 52–58.

Ishai Y, Kushilevitz E, Ostrovsky R, Prabhakaran M, Sahai A. 2011. Efficient non-interactive secure computation. In *Advances in cryptology EUROCRYPT 2011*, Vol. 6632 of *Lecture notes in computer science* (ed. K Paterson), pp. 406–425. Springer, Berlin.

Jacobs K, Yeager M, Wacholder S, Craig D, Kraft P, Hunter D, Paschal J, Manolio T, Tucker M, Hoover R, 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.

Kahn S. 2011. On the future of genomic data. *Science* **331:** 728–729.

Knoppers B, Harris J, Tassé A, Budin-Ljøsne I, Kaye J, Deschênes M, Man H. 2011. Towards a data sharing code of conduct for international genomic research. *Genome Med* **3:** 46.

Kyriazopoulou-Panagiotopoulou S, Kashef Haghighi D, Aerni SJ, Sundquist A, Bercovici S, Batzoglou S. 2011. Reconstruction of genealogical relationships with applications to Phase III of HapMap. *Bioinformatics* **27:** i333–i341.

Liu H, Prugnolle F, Manica A, Balloux F. 2006. A geographically explicit genetic model of worldwide human-settlement history. *Am J Hum Genet* **79:** 230–237.

Manichaikul A, Mychaleckyj JC, Rich SS, Daly K, Sale M, Chen WMM. 2010. Robust relationship inference in genome-wide association studies. *Bioinformatics* **26:** 2867–2873.

Manolio T, Brooks L, Collins F. 2008. A HapMap harvest of insights into the genetics of common disease. *J Clin Invest* **118:** 1590.

Matsuzaki H, Dong S, Loi H, Di X, Liu G, Hubbell E, Law J, Berntsen T, Chadha M, Hui H, et al. 2004. Genotyping over 100,000 SNPs on a pair of oligonucleotide arrays. *Nat Methods* **1:** 109–111.

McGuire A. 2008. Identifiability of DNA data: the need for consistent federal policy. *Am J Bioeth* **8:** 75–76.

NIST. 2008. FIPS, PUB 180-3: Secure hash signature standard. http://csrc.nist.gov/publications/fips/fips180-3/fips180-3_final.pdf.

Pemberton TJ, Wang C, Li JZ, Rosenberg NA. 2010. Inference of unexpected genetic relatedness among individuals in HapMap Phase III. *Am J Hum Genet* **87:** 457–464.

Reich D, Thangaraj K, Patterson N, Price AL, Singh L. 2009. Reconstructing Indian population history. *Nature* **461:** 489–494.

Risch N, Merikangas K. 1996. The future of genetic studies of complex human diseases. *Science* **273:** 1516.

Royal CD, Novembre J, Fullerton SM, Goldstein DB, Long JC, Bamshad MJ, Clark AG. 2010. Inferring genetic ancestry: opportunities, challenges, and implications. *Am J Hum Genet* **86:** 661–673.

Sankararaman S, Obozinski G, Jordan M, Halperin E. 2009. Genomic privacy and limits of individual detection in a pool. *Nat Genet* **41:** 965–967.

Stankovich J, Bahlo M, Rubio JP, Wilkinson CR, Thomson R, Banks A, Ring M, Foote SJ, Speed TP. 2005. Identifying nineteenth century genealogical links from genotypes. *Hum Genet* **117:** 188–199.

Tishkoff SA, Reed FA, Friedlaender FR, Ehret C, Ranciaro A, Froment A, Hirbo JB, Awomoyi AA, Bodo JMM, Doumbo O, et al. 2009. The genetic structure and history of Africans and African Americans. *Science* **324:** 1035–1044.

Wang J. 2011. Genome-sequencing anniversary. Personal genomes: for one and for all. *Science* **331:** 690.

Wheeler DA, Srinivasan M, Egholm M, Shen Y, Chen L, McGuire A, He W, Chen YJJ, Makhijani V, Roth GT, et al. 2008. The complete genome of an individual by massively parallel DNA sequencing. *Nature* **452:** 872–876.

Received December 8, 2012; accepted in revised form January 24, 2014.