KGP Meter: Communicating Kin Genomic Privacy to the Masses

Mathias Humbert, Didier Dupertuis, Mauro Cherubini, Kévin Huguenin

KGP Meter: Communicating Kin Genomic Privacy to the Masses

Mathias Humbert
University of Lausanne
Switzerland
mathias.humbert@unil.ch

Didier Dupertuis
EPFL
Switzerland
didier.dupertuis@epfl.ch

Mauro Cherubini
University of Lausanne
Switzerland
mauro.cherubini@unil.ch

Kévin Huguenin
University of Lausanne
Switzerland
kevin.huguenin@unil.ch

***Abstract*—Direct-to-consumer genetic testing services are gaining momentum: As of today, companies such as 23andMe or AncestryDNA have already attracted 26 million customers. These services raise privacy concerns, exacerbated by the fact that their customers can then share their genomic data on platforms such as GEDmatch. Notwithstanding their right to learn about their genetic background or to share their genomic data, it is paramount that individuals realize that such a behavior damages their relatives' privacy (i.e., kin genomic privacy). In this paper, we present KGP Meter, a new online tool that provides means for raising awareness in the general public about the privacy risks of genomic data sharing. Our tool features various properties that makes it highly interactive, privacy-preserving (i.e., not requiring access to the actual genomic data), and user-friendly. It explores possible configurations in an optimized way and combines well-established graphical models with an entropy-based metric to compute kin genomic privacy scores. Our experiments show that KGP Meter is very responsive. We design and implement an interface that enables users to draw their family trees and indicate which of their relatives' genomes are known, and that communicates the resulting privacy scores to the users. We then analyze the usage of the tool and survey users to better understand users' perceptions towards these risks and evaluate our tool. We observe that most of them find the privacy score worrisome, and that the large majority of them find KGP Meter useful.**

1. Introduction

Over the last decade, the plummeting cost of genome sequencing has enabled new breakthroughs in genomics and has fostered the emergence of companies that provide genetic testing services directly to consumers (e.g., 23andMe [1], AncestryDNA [2]). By early 2019, direct-to-consumer (DTC) genetic testing services already attracted more than 26 million customers [3]. On the one hand, this enables individuals to better understand their genetic background, to identify distant or lost relatives, and to learn about their predisposition to severe diseases. On the other hand, it raises new concerns about genomic privacy [4], [5], not only for the customer whose DNA is tested, but also for their relatives (e.g., the Pentagon recently warned the US military not to use DTC genetic testing kits [6]). Indeed, the genome of an individual can be partially inferred from that of their relatives [7], thus creating an interdependent privacy situation [8] and raising new ethical and legal questions [9]. DTC genetic testing services can already extrapolate genomic data of a large proportion of the US population [3]. The privacy threat is exacerbated by the fact that customers can share their genomic data publicly on platforms such as GEDmatch [10], MyHeritage [11], or OpenSNP [12]. Potential negative consequences of genomic information leakage include discrimination in the contexts of insurance, jobs, and loans to name a few [4], [5] but also tracking [6], [13].

Facing this new trend, the security and privacy community has begun investigating concrete privacy risks that stem from genomic data sharing [4] and proposing potential solutions (see related surveys [14]–[16]). For instance, scholars have developed privacy-enhancing protocols for biomedical researchers [17]–[23], or clinicians [24]–[29]. Yet, so far, no tool has been developed to enable the layman to evaluate the genomic privacy risks they expose their relatives to when sharing their genomic data (and therefore to make informed decisions) or, conversely, to evaluate the privacy risks their relatives expose them to, i.e., kin genomic privacy risks. The need for tools to help people understand kin genomic privacy risks is made even clearer in a very recent study on people's general attitudes towards DTC genetic testing [30].

In order to raise awareness among the general population on the kin genomic privacy risks, we design and implement a new tool referred to as the *kin genomic-privacy meter* (codenamed KGP Meter). It is based on a new data-less privacy evaluation technique: Unlike typical privacy meters that require the knowledge of the target user's actual data (e.g., [31]–[33]) and that of related individuals (e.g., [34]), KGP Meter only requires the knowledge of the family tree and of the set of relatives whose genomes have been (hypothetically) tested (e.g., using a DTC genetic testing service). As such, it is privacy-preserving and enables users to evaluate *hypothetical* risks (e.g, when deciding to take a test, or unsure about what would happen if somebody else in the family took the test), or to evaluate *concrete* risks (e.g., when a test was already taken). KGP Meter is made available to the scientific community under the form of a Python library and to the general public under the form of a web-based interactive tool.[1] Given that the tool is intended for the general public, it is required to be interactive [35], to be privacy-preserving (thus to not make use of the actual genomic data of any individual), and to be easy-to-use. In summary, we make the following contributions:

[1] https://santeperso.unil.ch/privacy/?src=article.

- We design and implement a data-less inference algorithm to efficiently compute the genomic privacy of a targeted individual given the set of their relatives whose genomes are known to an adversary (e.g., because they used a DTC genetic testing service). As the algorithm does not require any genomic data as input, it relies on mutual information to capture the proportion of uncertainty that is removed when having access to the relatives' genomes. It iterates over all combinations of values for the genomic variants of the relatives and uses a well-established graphical model to obtain posterior distributions of the target. It includes a number of optimizations to achieve low latency, despite the combinatorial explosion of the number of considered combinations. It focuses on generic genomic inference attacks and thus evaluates privacy for the whole set of variants genotyped by 23andMe based on general population statistics available on dbSNP [36]. We postulate that, as soon as we consider a large enough number of SNPs, the individual extreme (rare) SNP values are averaged out in the privacy score. Therefore, using the actual genomic data would not bring much more precision for such a global privacy score. These technical contributions are new and unlock the implementation and deployment of a tool for the general public.
- We report kin genomic privacy scores computed with KGP Meter for a number of typical situations, e.g., a privacy score of 95% when only the genome of a first cousin is known.
- We benchmark the performance of the algorithm based on a very large dataset of configurations (N=700,614,759). Our results show that KGP Meter is very responsive, providing results in less than a second in 99.8% of the cases.
- By following a user-centric development process, we design and implement an interface that enables users to draw their family tree and to specify the target whose genomic privacy they want to evaluate as well as the relatives whose genomes are known to an adversary. The interface then communicates the scores computed by our algorithm.
- We deployed KGP Meter, and conducted a user study targeted, through Prolific, at the general US Internet population (N=1,580 crowdworkers). Our results show substantial learning by using the tool. They also show that users are more interested in their own privacy than in that of their relatives. Although a majority of users found the results provided by the tool worrisome, a substantial fraction found them reassuring. Finally, the large majority of users found KGP Meter useful and were likely to recommend it.

Next, we introduce the relevant background on genomics and genomic privacy in Section 2. We present the threat model in Section 3. After describing the inference framework in Section 4, we describe our quantification framework, including the core algorithm and the associated optimizations, in Section 5. In Section 6, we detail the implementation of our tool – both its backend and its frontend – before reporting on some typical scores and on its performance in Section 7. We report on the methodology and results of our user survey in Section 8. We review the related literature in Section 9 and conclude in Section 10.

2. Background

The human genome consists of 3.2 billion pairs of nucleotides that take value in $\{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$. About 99.9% of our genome is common to all of us, hence not privacy sensitive. From a privacy point of view, the genomic positions that matter are those where nucleotides can differ between individuals; these are known as single nucleotide polymorphisms (or SNPs, also called variants). SNPs typically denote our ethnic background and encode our physical traits but also predispositions to certain diseases. There are – to date – around 150 million SNPs known in the human genome. A SNP is determined by a pair of nucleotides, which can take two (rarely three) different values, referred to as alleles, among $\{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$. The most frequent allele in the population is referred to as the major allele ('M' in the following), and the least frequent as the minor allele ('m').[2] Therefore, a given SNP can take three values: (i) MM (homozygous major) (ii) Mm (heterozygous), and (iii) mm (homozygous minor). A minor allele frequency (MAF), given by population statistics, is associated with each SNP. Note that MAF values are publicly accessible information that serves as baseline to measure the genomic privacy of individuals before any relative's genome has been observed by the adversary (i.e., the *prior*). At each position in an individual's genome, both their nucleotides are inherited from their parents. One nucleotide from the mother and the other from the father. Moreover, each nucleotide passed on by a parent is randomly picked with probability 0.5 among this parent's nucleotides. The resulting inheritance probabilities, i.e., the probabilities of a child's SNP given their parents' SNPs, are shown in Table 1.

Genomic privacy has been extensively studied over the last decade. One can categorize privacy attacks into three groups: (i) attribute inference attacks, (ii) membership inference attacks, and (iii) linkability or re-identification attacks. Attribute inference is the ability to infer the value of an attribute from the values of other attributes. This work and previous studies [7], [34], [37], [38] on kin genomic privacy belong to this group. Such inference is made possible by the correlations between relatives' genomes stemming from genetic inheritance.

Membership inference is the ability to infer that a certain target is in a specific dataset. Having access to a target genome, the adversary tries to determine whether this genome is part of a dataset by comparing it to summary statistics about this dataset. Such inference has been first proposed by Homer et al. [39] back in 2008. This binary classification typically relies on statistical tests, such as the likelihood-ratio test. Since then, various researchers have tried to characterize more precisely this attack with respect to the number of genomes contributing in the target dataset and the number of genomic variants accessible to the attacker [40]–[42]. Besides, membership inference attacks have also been studied with other types of genomic data, such as microRNA expression [43] and DNA methylation [44].

Linkability attacks are the ability to link at least two records concerning the same individual. If one of the records contain identifiers about the individual, this

[2] Both the major and the minor alleles take value in $\{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$.

TABLE 1. Conditional probability table of a child's SNP value given their parents' SNP values: $(P(X_c = \mathrm{MM} \mid X_m, X_f), P(X_c = \mathrm{Mm} \mid X_m, X_f), P(X_c = \mathrm{mm} \mid X_m, X_f))$.

		father		
		X_f = MM	X_f = Mm	X_f = mm
mother	X_m = MM	(1, 0, 0)	(1/2 , 1/2 , 0)	(0, 1, 0)
	X_m = Mm	(1/2, 1/2, 0)	(1/4, 1/2, 1/4)	(0, 1/2, 1/2)
	X_m = mm	(0, 1, 0)	(0, 1/2, 1/2)	(0, 0, 1)

leads to a so-called re-identification (or de-anonymization) attack. In this regard, Gymrek et al. notably show that one can re-identify genomes by analyzing Y-chromosome sequences from public genetic genealogy websites that contain relatives with the same surname. [45]. However, their attack is based on short tandem repeats (STRs) that are not as prevalent as SNPs in current genomic databases. Humbert et al. show that one can re-identify genomes based on the correlations between our genome and our phenotypic traits (such as eye color, blood type, or skin color) [46]. Lippert et al. further study the feasibility of phenotype-based re-identification attacks in a larger cohort by applying whole-genome sequencing, detailed phenotyping, and statistical modeling [47]. More recently, researchers have shown that we could re-identify a given genome's owner by relying on distant relatives search of genetic genealogy services such as GEDmatch [48]. Ney et al. [49] and Edge and Coop [50] have further studied other privacy and security risks, such as genotype inference, stemming from the growing popularity and open APIs of genetic genealogy services.

In genomic privacy, the attack type for which it is most challenging to measure privacy is the first one, i.e., attribute inference. For other attack types, the inference problem generally boils down to binary classification, and thus traditional statistical metrics like accuracy, or true-positive and false-positive rates can be directly applied. Therefore, these are less relevant and we focus in this paper on developing a meter for attribute inference, i.e., where we aim to infer the genomic variants of a given individual. In fact, the main work surveying genomic privacy metrics focuses on the attribute inference setting as well [51]. Wagner studies a plethora of potential metrics (detailed in Section 3.2 of [51]) that can be categorized as metrics measuring: (i) the adversary's error, (ii) the adversary's success probability, (iii) the adversary's uncertainty (typically based on entropy), (iv) the information gain/loss (e.g., based on mutual information) and (v) similarity/diversity. Given that the first two categories require to have access to the actual SNP value of the individual whose genomic privacy we measure, we cannot use them for our meter. Moreover, the fifth category is shown to perform worst in terms of agreement with adversarial strength by Wagner. Therefore, in this work, we decide to rely on the third and fourth categories that do not require any data to be computed and that show relatively good performance in [51]. Moreover, in the context of kin privacy, Humbert et al. show that, when averaged over a significant number of SNPs, the entropy-based and mutual information-based metrics behave very similarly to the expected estimation error, which is one of the best metric to capture genomic privacy [34], [37].

3. Threat Model

We assume the adversary to be (i) anyone who can access genomes publicly available on the Internet, on platforms such as OpenSNP [12] or the 1000 Genomes Project [52], or (ii) direct-to-consumer genetic testing companies such as 23andMe [1] or AncestryDNA [2]. By having access to these genomes, the adversary then aims to infer the genomes of some of the relatives of the genome owners. Besides the aforementioned adversaries, recent studies [49], [50] have shown that it is possible to recover the genomes of individuals having uploaded their own genomic data on genetic genealogy services such as GEDmatch [10], MyHeritage [11], and FamilyTreeDNA [53]. The incentives for the adversaries include genetic-based pricing/selection for loans and insurances as well as targeted advertisement [5]. These new attacks threaten the privacy of millions of individuals, but also, indirectly that of tens of millions of their relatives. Consequences of such attacks include inference of predisposition to certain diseases, and discrimination based on this information [4]–[6], [13]. As individuals' genomes barely change over their life course, such threats have long-lasting consequences. New threats could arise in the future. In this work, we focus on generic genomic inference attacks, not on identity inference [48] (see the Golden State Killer case [54], [55]) or kinship inference attacks.

The goal of the adversary is to infer other individuals' genomes from the genomes he already has access to, by making use of familial correlations. We assume that the adversary either gains access to the *entire set* of SNPs in the genome or *nothing*, which corresponds to the current settings of DTC genetic testing services. We focus on the SNPs produced by the popular Illumina sequencer [56] and returned by DTC services, such as 23andMe. These are also the most informative SNPs, e.g., those associated with diseases or ethnic background. Our methodology applies to any number of SNPs, thus the list of SNPs could be easily updated in the future. In order to have a privacy-preserving tool that does not collect or require any actual genomic data, KGP Meter cannot make any deductions about specific genetic cases, i.e., the impact of a relative sharing specific SNP values.

4. Inference Framework

In this section, we introduce the notations, model, and method for inferring kin genomic data.

We consider a set of l SNPs, $\mathcal{G} = \{g_1, g_2, ..., g_l\}$, included in the genomic data of the genotyped individuals. Each SNP takes values in $G = \{\mathrm{MM}, \mathrm{Mm}, \mathrm{mm}\}$, representing – in this order – two major alleles, one major and one minor allele, and two minor alleles. A minor allele frequency (MAF) $p^i_{\mathrm{maf}} \in (0, 0.5]$ is associated with each SNP g_i. We denote the set of n relatives in a family by $\mathcal{R} = \{r_1, r_2, ..., r_n\}$, the target by r_t ($t \in [1, n]$), and the indices of the relatives whose genomes are observed by the adversary (i.e., "known") by $O \subseteq [1, n] \setminus \{t\}$. X^i_j represents the random variable of SNP g_i for relative r_j, and x^i_j denotes its actual value. We use (bold) vector notations for the variables associated with the SNPs of

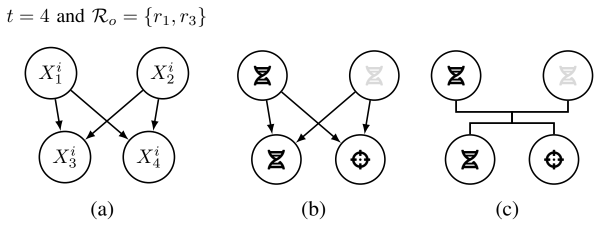


Figure 1. Sample Bayesian network ((a) formal and (b) informal representations) and (c) corresponding family tree with two parents and their two children, for a given SNP g_i.

the relatives whose genomes are known to the adversary: $\mathbf{X}_O^i = (X_j^i)_{j \in O}$ and $\mathbf{x}_O^i = (x_j^i)_{j \in O}$.

We use a Bayesian network (BN) to represent dependencies between the genomic data of the considered relatives. Bayesian networks follow the same logical structure as that given by the Mendelian laws of genetic inheritance: The value of a random variable, represented as a node in the BN, is determined from those of its parent variables. As shown in Figure 1, each SNP variable X_j^i can be represented by a node in the BN, and each node has exactly two parent nodes, representing the SNP variables of the two *biological* parents.[3] Not only the BN structure is given by genetic inheritance, but also its parameters. Apart from the root nodes (i.e., nodes without parents), all internal (i.e., child) nodes take a conditional probability (i.e., $P(X_c^i | X_m^i, X_f^i)$) table of size $3 \times 3 \times 3$ given their mothers and fathers, as defined in Table 1. The root nodes (i.e., X_1^i and X_2^i in Figure 1) take the prior probability table defined as $P(X_j^i = \mathrm{MM}) = (1 - p_{\mathrm{maf}}^i)^2$, $P(X_j^i = \mathrm{Mm}) = 2 \cdot (1 - p_{\mathrm{maf}}^i) \cdot (p_{\mathrm{maf}}^i)$, $P(X_j^i = \mathrm{mm}) = (p_{\mathrm{maf}}^i)^2$.

The SNPs $\{g_i\}_{i \in [1,m]}$ for the target relative can be inferred independently[4] from each other by using the same BN, but with a different prior probability for the root nodes; this probability is determined by the MAF of the considered SNP.

Given the structure of the BN, the joint distribution of the random variables $(X_1^i, X_2^i, ..., X_n^i)$ representing the values of SNP g_i of all relatives in a family can be factorized in smaller distributions given by the aforementioned prior and conditional probability table. Thanks to this factorization, we can efficiently compute the *exact* marginal posterior distributions of unknown variables given the observed variables, by using the junction tree algorithm [63]–[65].

5. Data-Less Quantification

Our overarching goal is to build a tool for enabling individuals to evaluate their (kin) genomic privacy in a simple, interactive and privacy-preserving way. This implies that the tool should be able to compute privacy scores in less than a few seconds and should not rely on the genomic data of the relatives.

The framework described in the previous section enables the efficient inference of the target's SNPs based on the *actual* values of the SNPs of their relatives whose genomes are known to the adversary. We can then quantify the target's privacy based on the adversary's expected error by comparing the inferred values to the *actual* values of the target's SNPs (i.e., ground truth). However, these steps rely on the knowledge of the genomic data of the target and of their relatives: This goes against the simplicity and privacy-preserving requirements and prevents the use of this framework for testing hypothetical scenarios.

In this section, we present a novel algorithm and describe how we alleviate the aforementioned requirements for the inference and quantification processes. Because the (basic version of the) proposed solution incurs a substantial computational overhead, which goes against the interactivity requirement [35], we design a number of optimization techniques.

5.1. Privacy Metric

Given that the target's actual SNP values are not known to the tool, to quantify privacy we rely on the mutual information – a metric based on Shannon entropy[5] $H(\cdot)$ and widely used in privacy research – between the target's SNPs X_t^i and those of the relatives whose genomes are known $\mathbf{X}_O^i$. The mutual information is defined as $I(X_t^i; \mathbf{X}_O^i) = H(X_t^i) - H(X_t^i \mid \mathbf{X}_O^i)$. In order to obtain a score between 0 and 1, which directly reflects the target's privacy (and not the opposite, i.e., the privacy leakage), we define the following metric:

$$S_t^i(O) = 1 - \frac{I(X_t^i; \mathbf{X}_O^i)}{H(X_t^i)} = \frac{H(X_t^i \mid \mathbf{X}_O^i)}{H(X_t^i)} \qquad (1)$$

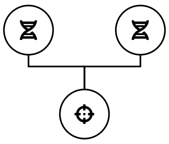
This represents the ratio between the entropy given the observed SNP values and the entropy of the prior. This ratio is maximum, i.e., equal to one, when no relatives' genome is known, and minimum, i.e., zero, if the value of X_t^i can be deterministically determined from $\mathbf{X}_O^i$. In short, this metric captures (for a given SNP) the *proportion* of the adversary's uncertainty about the target's genome that remains when knowing the relatives' genomes. We selected mutual information and Shannon entropy because these represent state-of-the-art metrics in genomic privacy [34], [37], [51]. Furthermore, entropy was also used for measuring privacy in anonymous communications [69], [70] and in information flows [71]. Note that this metric measures the fraction of *information* leaked, not a probability (of success) of a specific attack. In fact, the genome of a distant relative leaks little information but could be enough to establish paternity or to re-identify an individual (e.g., the Golden State Killer case [54], [55])

Finally, the genomic privacy score of the target r_t is obtained by averaging over all SNPs g_l, $i \in [1, l]$: $\bar{S}_t(O) = \frac{1}{l} \sum_{i=1}^{l} S_t^i(O)$. Alternatively, only selected SNPs could be considered, for instance those related to specific diseases such as Alzheimer's.

[3] Throughout the paper, we focus on genetically-linked families, thus not including adoptive or donor-conceived children.

[4] As in most previous works [41], [57]–[60], we do not consider *linkage disequilibrium* [61], [62], i.e., pairwise correlations between SNPs, as they bring little information in our setting.

[5] Note that other entropy metrics could be used, such as Rényi min-entropy which generalizes various entropy metrics [66] or g-leakage which generalizes the min-entropy model of quantitative information flows [67]. We further refer to [68] for a detailed analysis on the connection between the various information-theoretic metrics and side-channel attacks.



(a) Sample family tree.

X_m^i	X_f^i	$P(X_m^i, X_f^i)$	$P(X_c^i \mid X_m^i, X_f^i)$	$H(X_c^i \mid X_m^i, X_f^i)$	S_c^i
MM	MM	0.656	$(1.0, 0.0, 0.0)$	0.0	0
MM	Mm	0.146	$(0.5, 0.5, 0.0)$	1.0	1.319
MM	mm	0.008	$(0.0, 1.0, 0.0)$	0.0	0
...	...				...

(b) SNP value combinations (truncated) for the relatives whose genomes are known (i.e., mother and father) with the associated joint probability as well as the posterior probability, entropy and score of the target. X_{c}^i, X_{m}^i, and X_{f}^i denote the SNP values of the child, the mother and the father respectively. The MAF was set to $p_{\text{maf}}^i = 0.1$, which yields $H(X_c^i) \approx 0.758$.

Figure 2. Computation of the privacy score for SNP g_i.

5.2. Data-Less Evaluation

Given that the relatives' actual SNP values are not known to the tool, we consider, for each SNP g_i, *all the possible combinations* of SNP values for the relatives. For instance, if there are two relatives whose genomes are known to the adversary (i.e., $|O| = 2$), there are a total of $3 \times 3 = 9$ combinations: $\mathbf{x}_O^i$ equals $(\mathrm{MM}, \mathrm{MM})$ or $(\mathrm{MM}, \mathrm{Mm})$ or $(\mathrm{MM}, \mathrm{mm})$ or $(\mathrm{Mm}, \mathrm{MM})$, etc. The total number of such combinations to be considered is $3^{|O|}$.

Each such combination of SNP values has an associated *joint* probability of occurence, which only depends on the MAF associated with the considered SNP (i.e., p_{maf}^i) and the inheritance laws (see Table 1). Note that the SNP values of different relatives are *not* necessarily independent. For instance, if the relatives whose genomes are known are father and son, the joint probability of the combination $\mathbf{x}_O^i = (\mathrm{mm}, \mathrm{MM})$ is null. If the relatives are mates / partners,[6] however, their SNP values are independent (assuming that the SNP values of their descendants are all unknown) and the joint probability of the combination $\mathbf{x}_O^i = (\mathrm{mm}, \mathrm{mm})$ is $(p_{\text{maf}}^i)^4$.

For each combination $\mathbf{x}$ of SNP values, we compute its joint probability $P(\mathbf{X}_O^i = \mathbf{x})$ and the posterior probability $P(X_t^i \mid \mathbf{X}_O^i = \mathbf{x})$ of the target's SNP value by performing an exact inference using the algorithm presented in Section 4. We compute the entropy $H(X_t^i \mid \mathbf{X}_O^i = \mathbf{x})$ of the posterior probability, from which we compute the privacy score for a given SNP according to Eq. 1, and we do so for all the genotyped SNPs. Note that the entropy of the prior (i.e., the denominator) can be computed directly from the MAF associated with the considered SNP (i.e., p_{maf}^i). Figure 2 illustrates the computation of the score on a simple example. The final privacy score is an expected value, computed as the sum of the privacy scores across all combinations of SNP values, weighed by their associated probabilities of occurrence:

$$E[\bar{S}_t(O)] = \frac{1}{l}\sum_{i=1}^{l} \sum_{\mathbf{x} \in G^{|O|}} \frac{H(X_t^i \mid \mathbf{X}_O^i = \mathbf{x})}{H(X_t^i)} \cdot P(\mathbf{X}_O^i = \mathbf{x}) \tag{2}$$

All the possible combinations of SNP values for the relatives are iterated on through recursion and the expected value (i.e., the sum) is computed iteratively. A pseudo-code version of the algorithm is provided in Algorithm 1. For the sake of clarity, we omit the SNP's index i in the algorithm. In the algorithm, BP stands for belief propagation; for this, we rely on an implementation from an existing library. To compute the final privacy score of the target, the tool performs, for each of the l SNPs and each of the $3^{|O|}$ SNP value combinations, an inference using the junction tree and belief propagation algorithms. Such complexity prevents us from computing privacy scores in a few seconds, even for a small number of relatives whose genomes are known (e.g., $|O| = 3$).

Limitations. Note that, given that we capture a single privacy score for a large set of SNPs in the genome, it can only be an average, with or without having access to the actual genomic data of the relatives or the target. In our case, we average over both the set of SNPs and their possible values, while with real data, we only average over the set of SNPs. However, as we note in Section 7.1, the scores between our meter and metrics with real data [37] are very similar, even though [37] considers an average on the first chromosome only. We postulate that, as soon as we consider a large enough number of SNPs, the individual extreme (rare) values are averaged out in the privacy score. Therefore, the usage of real data does not bring much more precision regarding a generic privacy score over the whole genome.

The provided privacy score would certainly be less accurate if we had focused on specific set of SNPs, e.g., related to a particular disease. In such case, the rare values that some individuals' SNPs could carry would not average out with other SNPs' values. For this case, the users could perform their privacy score computation with actual genomic data locally. If the user computing the privacy score cannot have access to the actual data of their relatives, another approach could be to rely on (secure) multi-party computation (MPC). The main drawback of our approach is that it does not provide a specific score about the risk-sensitive genes or SNPs.

Algorithm 1 Returns the conditional entropy $H(X_t^i \mid \tilde{\mathbf{X}}_o^i)$ of the target's SNP given the values of the observed SNPs of the relatives.

```
function CondEntropy($U$, $\tilde{\mathbf{x}}$)
    ▷ $U \subseteq O$: set of relatives whose genomes are known but for which the SNP values are not yet set.
    ▷ $\tilde{\mathbf{x}}$: evidence; SNP values of the relatives whose genomes are known and for which the SNP values have already been set.
    if $U = \varnothing$ then
        $p_{\text{joint}} \leftarrow P(\tilde{\mathbf{X}}_o = \tilde{\mathbf{x}})$            ▷ compute with BP
        $\mathbf{p}_{\text{cond}} \leftarrow P(X_t \mid \tilde{\mathbf{X}}_o = \tilde{\mathbf{x}})$       ▷ compute with BP
        $h_{\text{cond}} \leftarrow -\sum_{g \in G} \mathbf{p}_{\text{cond}}(g) \log \mathbf{p}_{\text{cond}}(g)$   ▷ compute entropy
        return $p_{\text{joint}} \cdot h_{\text{cond}}$
    else
        pick $j \in U$                              ▷ next relative to explore
        $r \leftarrow \sum_{g \in G}$ CondEntropy($U \backslash j$, $\tilde{\mathbf{x}} \cup \{\tilde{x}_{o,j} = g\}$)
        ▷ compute expected value of cond. entropy in exploration (sub)tree
        return $r$

*result* ← CondEntropy($O, \varnothing$)
```

[6] We mean "the other biological parent of one's biological child".

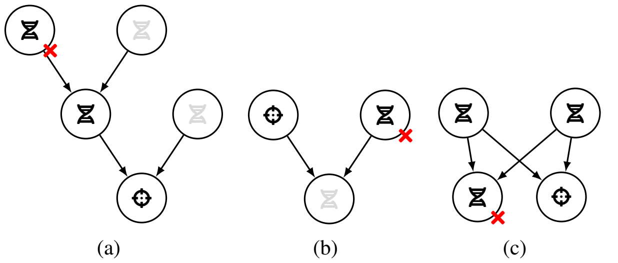


Figure 3. Configurations illustrating the cases where the genome of some relatives are irrelevant and can therefore be removed (marked with a red cross). Nodes that are d-separated from the target can be safely removed from the algorithm.

5.3. Optimizations

We present a number of optimizations for meeting the interactivity requirement of the tool.

5.3.1. Removing Irrelevant Relatives. Due to the structures of family trees and of the associated BNs, in some cases, the genome of a relative is irrelevant for inferring that of the target. More specifically, given the genomes of the other relatives whose genomes are known, it does not bring new information. A simple example (Figure 3(a)) is the case where the genomes of both the mother and the maternal grandmother of the target are known: Given the genome of the mother, the genome of the grandmother does not bring new information for inferring that of the target. Other examples are given in Figure 3.

Although such irrelevant nodes are automatically ignored in the inference process, they are still (unnecessarily) considered when enumerating the different configurations of SNP values for the relatives whose genomes are known (see Eq. 2). They should therefore be removed. Formally, this optimization consists in removing from the BN the nodes that are independent from the target node, given the other observed nodes (i.e., those corresponding to the relatives whose genomes are known). In this endeavor, we rely on the notion of *d-separation* that formalizes the concept of conditional independence between nodes in a BN (see Appendix A on p. 18). Two conditionally independent nodes in a BN do not bring information about each other. This enables us to simplify the BNs, by removing all nodes that are d-separated from the target node X_t^i given the set of observed nodes $\mathbf{X}_O^i$. We show three examples of such simplifications by using d-separation in Figure 3. In Figure 3(a), the grandparent of the target is d-separated from the target because the target's parent is observed in-between. Figure 3(b) depicts a v-structure between the target and his mate. In this case, the trail between them is active only if the descendant is observed, which is not the case here, hence the mate is d-separated from the target and can be removed. Finally, Figure 3(c) depicts d-separation between a target and their sibling. This case is more complex because there are *two* trails between the target and their sibling: one through the mother and another through the father. However, none of these trails are active if both the parents are observed.

Through this optimization, the number of combinations of SNP values considered in the algorithm drops from $3^{|O|}$ to $3^{|O'|}$, where $O' \subseteq O$ is the set of relatives, in the *simplified* tree, whose genomes are known.

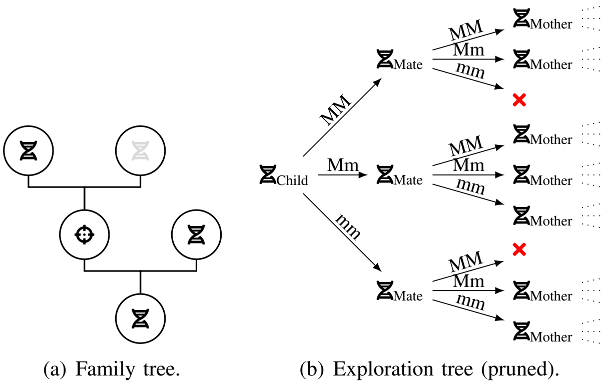


Figure 4. Configuration illustrating the cases where some combinations of SNP values for the relatives whose genomes are known to the adversary are impossible; this can be detected early and the algorithm can stop, thus pruning sub-branches of the exploration tree (marked with a red cross).

5.3.2. Pruning impossible SNP value combinations. Due to the dependences between the genomes of the relatives, some combinations of SNP values (for the relatives whose genomes are known) are impossible, i.e., they have a null joint probability. For instance, a combination where a woman and her child have SNP values of MM and mm, respectively, is impossible (see Table 1). Any combination containing an impossible combination is impossible. Therefore, when exploring all combinations of SNP values recursively, we can – at each recursive call – compute the joint probability of the current sub-combination and stop (i.e., return) early if this probability is null, instead of computing the joint probability only when the recursion terminates. The exploration tree is pruned, thus saving unnecessary iterations. This is achieved by adding the following instructions before Line 4 of Algorithm 1: "**if** $P(\tilde{\mathbf{X}}_o = \tilde{\mathbf{x}}) = 0$ **then return** 0".

An example of pruning is shown in Figure 4. The sample family tree (a) includes a relative (i.e., the target's mate) and their child; and both their genomes are known. The branches of the exploration tree (b) that correspond to the combinations of SNP values in which the mate / partner and their child have values MM and mm, respectively (or conversely), are pruned.

Through this optimization, the number of combinations of SNP values considered is *at most* $3^{|O'|}$ instead of *exactly* $3^{|O'|}$. Note that neither this optimization nor the previous reduces the *worst case* complexity. Yet, as our user study shows, they do substantially reduce the average computation time (and thus the responsiveness of the tool) by an order of magnitude.

5.3.3. Interpolating over MAF Values. In order to compute the target's final privacy score, a privacy score must be computed for each SNP g_i and its associated MAF value p^i_{maf}. And the number l of SNPs in popular DTC genetic tests is in the order of hundreds of thousands.

In order to reduce the computational time of the algorithm, we compute the privacy scores only for a limited set of k ($k \ll l$) MAF values $\{p_1, \ldots, p_k\}$, distributed in $[0, 0.5]$, and use these sample scores to interpolate the privacy scores for all SNPs. We compute the corresponding sample scores $\{s_1, \ldots, s_k\}$ (i.e., $s_j = S_t(O)(p_j)$,

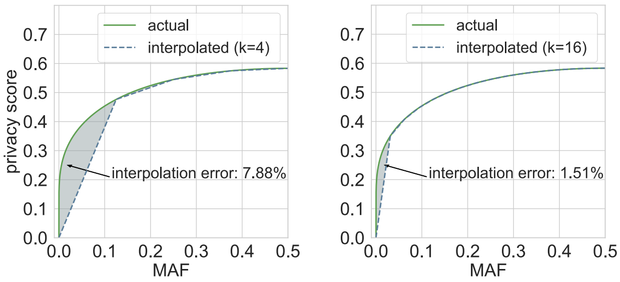


Figure 5. Interpolation of the privacy scores based on $k = 4$ and 16 regularly distributed sample MAF values in the case where the genomes of both the target's parents are known. The shape of the curve (i.e., steep increase for small MAF values) suggests the use of more samples for small MAF values.

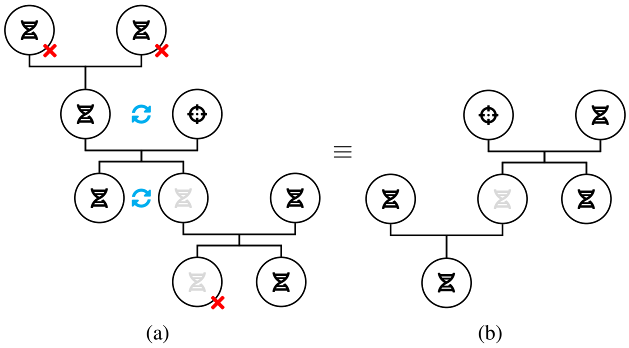


Figure 6. Two equivalent configurations: Irrelevant relatives can be removed and parents and children can be re-ordered without altering the result of the inference.

$j \in \{1, \dots, k\}$) as in Eq. 1 and estimate the final privacy score as follows:

$$\tilde{S}_t(O) = \frac{1}{m}\sum_{i=1}^{m} \text{interpolate}_{(p_1,s_1),\dots,(p_k,s_k)}(p^i_{\text{maf}}) \qquad (3)$$

Unlike the other optimization techniques presented in this section, which reduce the computational time without affecting the result of the computation, interpolation introduces some inaccuracy (which we measure experimentally below). The larger the set of sample MAF values for which we compute the privacy score is, the closer the interpolated privacy score is to the actual privacy score.

To evaluate the inaccuracy introduced by interpolation, we consider the sample configuration where the genomes of both the parents of the target are known to the adversary (see Figure 2(a)). We compute both the actual privacy scores (for all the SNPs) and the scores interpolated from $k = 4$ or 16 sample MAF values. Figure 5 illustrates the concept of interpolation as well as the discrepancy between the actual and interpolated privacy scores in these settings. The relative error on the global privacy score (assuming a uniform distribution of MAF across the considered SNPs) is 1.51% for $k = 16$ (resp. 7.88% for $k = 4$), which is reasonable. It can be observed that, given the shape of the curve (i.e., steep increase for small MAF values), non-regular distributions should be used for sample MAF values, with more samples for small values of MAFs.

Through this optimization, the number of privacy scores to compute for individual SNPs/MAF values drops from l (a few hundreds of thousands) to k (a few dozens).

5.3.4. Caching Computed Scores. In order to increase the responsiveness of the tool, the system caches the computed results. This enables the tool to return a score very fast (i.e., without re-computing it) if it has already been computed for the same, or *an equivalent* (i.e., that gives the same inference results) configuration. Indeed, due to the symmetry properties of BNs and of inheritance laws and because we try all possible combination of SNP values, parents and children can be re-ordered without altering the result of the inference. Figure 6 depicts two equivalent configurations.

For caching to be efficient, equivalent configurations should be mapped to the same "signature";[7] in other words, the signature should be applied on the *simplified* configuration (i.e., after irrelevant relatives have been removed, as described in Section 5.3.1) and should be invariant with respect to parents and children re-ordering.

The signature algorithms operates recursively, starting at the target. The signature of a tree, rooted at a given individual, is computed recursively by hashing[7] the concatenation of (1) the status of the individual (i.e., "genome known": `true`, "genome not known": `false`), (2) the signatures (sorted alphabetically) of the subtrees rooted at their parents, (3) the signatures (sorted) of those rooted at their siblings, if any, (4) the signatures (sorted) of those rooted at their children, if any, and (5) the signature of the subtree rooted at their mate, if any. We illustrate the functioning of the signature algorithm in the example depicted in Figure 7.

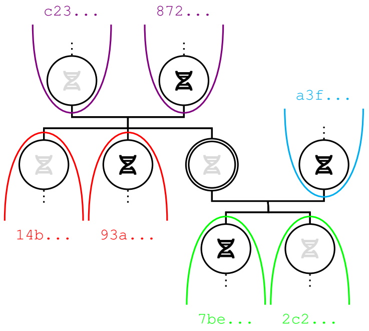


Figure 7. Illustration of the signature-generation algorithm used for caching. The signature of a tree rooted at a given relative is computed recursively by hashing the concatenation of the status of the relative, the signatures (sorted) of the subtrees rooted at their parents, the signatures (sorted) of those rooted at their siblings, the signatures (sorted) of those rooted at their children, and the signature of that rooted at their mate/partner. The root node is circled. The (truncated) signature of each subtree is indicated next to it. The resulting signature is: `hash(false;872..,c23..;14b..,93a..;2c2..,7be..;a3f..)`.

Because the signatures of the subtrees are sorted within each relative type (e.g., parents, siblings), they are deterministically ordered in the concatenation, thus resulting in a signature that is invariant with respect to parents and children re-ordering.

[7] The terms "signature" and "hash" are used from an indexing perspective here, not in a cryptographic sense.

6. Implementation

In order for KGP Meter to be easy to run and available to a large audience, it is implemented as a web application. KGP Meter consists of (1) a *frontend* (i.e., running on the client) that enables users to build their family tree, indicate whose genomes are known, and who the target of the privacy evaluation is and (2) a *backend* (i.e., running on the server) that computes the privacy score for a given configuration. The tool is embedded in a website that contains (1) a short introduction stating the context and the potential consequences of genomic information leakage (e.g., discrimination in the contexts of insurance, loans etc.), (2) an FAQ section, and (3) a description of the techniques underlying the tool. See: https://santeperso.unil.ch/privacy/?src=article.

6.1. Frontend

Design. We developed the website by following a user-centric development process [72], which comprises formative research and a few iterative cycles of design and testing (we report the usability evaluation later below).

The initial design was based on web services that enable users to interactively build their family trees [11], [73]. For our tool, we focus on biological family relationships. Plain circles are used to represent tree nodes (i.e., relatives). Parent nodes are connected to child nodes through segments. The color scheme follows conventional representations for sex. Icon selection is again based on well established conventions: Mars/Venus symbols for biological sex (i.e., ⚥) the double helix for the DNA (i.e., ⧗), a rifle crosshair for the privacy-evaluation target (i.e., ⌖), and a person silhouette with a 'plus' sign for adding a family member (i.e., 👤+). The initial view is composed of a single node representing the user (denoted with "👤 You"). Placing the mouse pointer on a node in the tree reveals a set of icons either to add a direct relative (👤+, Figure 16(a) in Appendix B), to indicate the genome of the considered relative as sequenced or not sequenced (⧗+/⧗–, Figure 16(b)), or to designate the considered relative as the target of the privacy evaluation (⌖, Figure 16(c)).[8] Each node contains a DNA symbol. The color fill is dark (⧗) if the genome of the corresponding relative is known or light (⧖) if not known. It also contains a name for the relative with a default auto-generated value (e.g., "Father"); the name can be edited to enable users to navigate complex family trees. Users can load existing family trees (GEDCOM format), including pre-defined ones. On the right hand-side of the tree, we placed a vertical bar that represents the privacy score as calculated by the algorithm. The portion of the bar that is filled reflects the score, and so does its fill color through interpolation using a green-orange-red gradient, as per convention [74] (see also [75]–[77]).

The frontend does not currently directly support the cases of half-siblings and of individuals having multiple children with different partners / mates. It does not support either cycles in family trees, e.g., John and Paul are brothers, Mary and Lea are sisters, John and Mary have children together and so do Paul and Lea. Note, however, that the backend (as well as the underlying formalism) supports such cases; we report sample privacy scores with half-siblings in Section 7.1. Moreover, such configurations can be drawn with other tools (e.g., commercial genealogy software), exported in the GEDCOM format, and finally imported and processed with KGP Meter.[9]

Given the results of the formative research, we expected most of the users to have little knowledge about DNA testing, not to mention DNA-related privacy risks. These results motivated the inclusion of explanatory design elements: a short introduction that mentions the potential consequences of genomic information leakage, a video tutorial, and an FAQ page. The FAQ explains, among other things, what the kin genomic privacy score means (i.e., how much of the genomic information of the 'target' remains unknown when the genomes of some of their relatives are known), as well as the limitations of KGP Meter.

Usability Evaluation. Before implementing the website, we implemented the design on a functional prototype that underwent a usability evaluation [72]. We recruited 13 participants with mixed backgrounds and varied demographic characteristics. The experiment took place in a UX-lab, a small room with a desktop computer and cameras. Participants were welcomed and signed an informed consent agreement where we specified that they were recorded. They were asked to complete two tasks: the first was to spend 2 minutes freely exploring the website and then to explain, in their own words, its purpose. For the second task, participants were given a family tree on a piece of paper (see Figure 16(d)) and they were asked to represent the same tree by using the website. A researcher sat in the same room as the participants performed the task and took notes of mistakes, uncertainties, and the time it took to perform the task. At the end, participants were asked to rate the level of difficulty in using KGP Meter (from 1=extremely difficult to 7=extremely easy).

All of the participants in the study managed to correctly identify the purpose of the website. However, 5 participants thought that the score represented an actual measure of the target's *SNP values* that had not been released to the public. Hence, they failed to recognize that the score represents a proportion of the genomic *information*. The average time required to represent a tree with 6 nodes through KGP Meter and estimate the privacy risk was about 2 minutes (136 ± 44 sec.). The average number of mistakes (e.g., adding a node by mistake and erasing it) was 0.5. Finally, the average task complexity reported by participants was *moderately easy* (5.6 ± 0.9). The collected usability scores were judged sufficient to grant the deployment of the KGP Meter. Furthermore, the evaluation helped identify a few aspects of the design that could be improved: (1) the video tutorial was replaced with a step-by-step interactive tutorial; (2) a tooltip was added when a mouse-over event on the icons of the tree builder was detected; (3) a text explaining the privacy score received was added under the family tree (see Figure 16(d)).

Implementation. The frontend is depicted in Figure 16 in Appendix B on p. 18. It is implemented in Javascript

[8] Note that we chose *not* to designate "You" as the default target in order not to prime the users. This is discussed in Section 8.1.

[9] See: https://santeperso.unil.ch/static/half.mov.

and it relies on D3 [78] which provides the low-level primitives to draw trees. The actual construction of the tree and its configuration, however, are implemented in a module we developed, as we could not find any suitable open-source library for that purpose. As long as a target is designated in the family tree, asynchronous requests are sent to the server (i.e., the backend) every time the configuration is modified by the user. For privacy reasons, the names of the relatives, if any, and their sex are *not* sent to the server. Furthermore, the ordering of the siblings (and of the parents) is shuffled. The server returns a privacy score that is reported in a vertical privacy bar and in an explanatory message, as depicted in Figure 16(d). Status messages from the backend can potentially be displayed at the top. All pages include a cookie bar with a link to our privacy policy. The privacy score evaluations performed by the users of the tool are stored on the server. KGP Meter was available in English during the experiment; it now includes four additional languages. The source code of the tool is available on GitHub.[10] Additionally, the frontend can be embedded in any static webpage by relying on the backend we deployed. A video demo is also available at: https://santeperso.unil.ch/static/demo.mp4.

6.2. Backend

The backend is implemented and runs in Python. It is run by the Apache web-server through the Web Server Gateway Interface; it relies on Flask for handling web requests, and pgmpy [79] and NETICA [80] (through `ctypes` as NETICA is implemented in C) for manipulating BNs. The former is used for simplifying BNs based on d-separation and active trails (see Section 5.3.1) and the latter is used for computing joint and posterior probabilities (i.e., the inference). The results are cached in a persistent database (see Section 5.3.4).

The backend uses 16 MAF values, regularly distributed between 0 and 0.5, for the interpolation (see Section 5.3.3) and computes the corresponding scores in parallel. In order to account for the shape of the privacy score curve as explained in Figure 5, we assigned priorities to the different MAF values. The backend interrupts the computations of a final privacy score after 10 seconds [35]; if the scores for at least 4 sample MAF values are available, a final score is returned.[11] Otherwise, an error is raised by the backend and a warning message is shown to the users, asking them to try again later (i.e., typically after the daemon process has computed the missing values). For computing the final score, KGP Meter relies on the list of SNPs genotyped by 23andMe [1] v4 (≈638k in total). For the associated MAF values, it relies on global statistics from dbSNP [36]; this leaves us with 486,750 SNPs, as there is no information on dbSNP for some of the SNPs genotyped by 23andMe.

6.3. Extensions

Pre-computing scores for caching at the client side. For improved privacy and performance, the most frequently requested scores cached on the backend could be proactively sent to the frontend in such a way that the frontend can display the requested privacy score without contacting the backend. This, however, requires the frontend to compute the signatures of the requests (as described in Section 5.3.4); this means using a Javascript library for manipulating Bayesian networks. Another option would be to use a less effective signature scheme at the client (without simplification, just with deterministic reordering of parents and children).

6.4. Privacy Considerations

For its functioning, our tool requires some information that may impact its users' privacy. Essentially, the information that is sent to the backend of the tool is (i) the family tree and (ii) whether each individual in the tree is sequenced or not. However, note that tested family trees can be hypothetical. Besides, dummy tree requests could be envisioned and pre-computed scores would prevent some requests to be sent to the backend (see Section 6.3). Also, the names and gender are not sent to the backend. Finally, note that, for maximum privacy, users could download and run their own backend (the code is open-source).

7. Evaluation

7.1. Privacy Scores

We give some sample average genomic privacy scores in typical configurations and compare them to those obtained by Humbert et al. [37] for a specific family (i.e., the CEPH/Utah Pedigree 1463). Figure 8 depicts the global privacy scores in these different configurations. Note that although the case of half-siblings is not supported by the frontend, it is by the formalism and the backend; therefore, we can report on the privacy scores for configurations involving half-siblings and, more generally, step families. The labels on the x-axis denote the list of relatives whose genomes are known. The configurations are sorted by decreasing privacy scores. The genomic privacy score ranges from 95% for a first cousin to only 19% for the partner, the 2 parents, and 3 children. We observe the same trend as Humbert et al. but with slightly different scores (e.g., 78% for one parent with our tool vs. ≈74% in [37]). This is due to the fact that our tool computes the average case over all combination of SNP values for the relatives whose genomes are known, whereas Humbert et al. consider the real genomes of the relatives (considering only Chromosome 1 and not all chromosomes). The privacy score for a half-sibling (92%) is substantially higher than for a sibling (79%). Interested readers can compute privacy scores for other configurations by using our online tool or Python library. Remember that these scores are not directly related to the success probability of specific attacks (e.g., identity inference [48], [49] and the Golden State Killer case [54], [55]). For instance, the score is 95% for a first cousin although a first cousin match has been consistently demonstrated to be sufficient to identify an individual (in crimes, for example).

[10] https://github.com/isplab-unil/kin-genomic-privacy.

[11] The privacy scores for the missing MAF values, if any, are computed asynchronously by a daemon process running in the background, thus providing a higher accuracy for future requests.

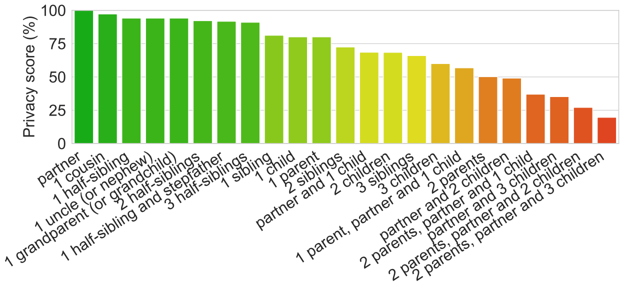


Figure 8. Sample privacy scores in typical configurations.

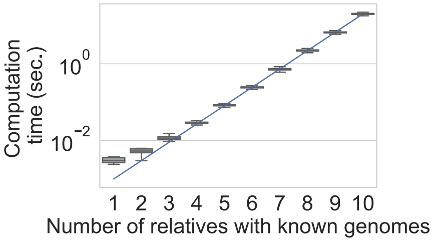


Figure 9. Computation time of a privacy score as a function of the number of relatives whose genomes are known in the *simplified* family tree. The line represents the result of a log-linear regression (computation time of $3.01^{|O'|}$, R^2 of 99.7%).

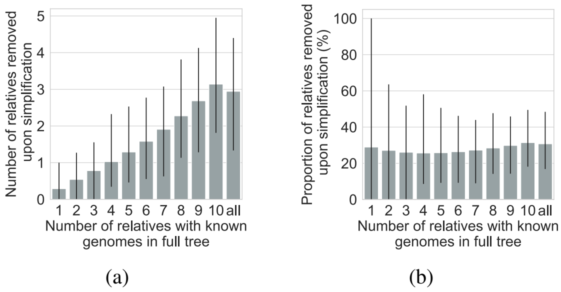


Figure 10. Sequenced relatives removed in simplification of benchmark family tree (a) number / (b) proportion.

7.2. Performance

Methodology. Interactivity is a key aspect and a strong requirement, in the design of the tool [35]. In order to evaluate the performance of the tool (i.e., benchmark) in terms of computation time, we consider a large sample family tree, centered on the target, with 38 relatives (excl. the target). The general methodology for constructing the benchmark tree is to assign 2 children to each couple–except for the target who has 3 children and 2 siblings–and to consider only relatives within a certain "distance" to the target. The benchmark tree is depicted in Figure 17 in Appendix D on p. 20. We consider all the combinations of 10 or less relatives whose genomes are known (i.e., $\sum_{j=1}^{10} \binom{38}{j} = 700{,}614{,}759$ combinations). We relied on a computer with Intel(R) Xeon(R) CPU E5-2665 @ 2.40GHz (32 cores).

Results. We first measure the computation time for a privacy score corresponding to a *single* MAF value and for a given configuration (i.e., a family tree with a target and a set of relatives whose genomes are known). Our tool relies on 4 to 16 MAF values to compute the global privacy score (using interpolation; see Section 5.3.3). The scores corresponding to the different MAF values are computed in parallel. We look at the distribution of computation time, depending on the number of relatives whose genomes are known. Figure 9 depicts a box-plot representation of the computation time of the score corresponding to a *single* MAF value, aggregated over all the considered configurations of the benchmark tree, as a function of the number of relatives whose genomes are known after *simplification* of the family tree (see Section 5.3.1). It can be observed that the computation time closely follows the $O(3^{|O'|})$ computational complexity. The median *proportion* of removed relatives is 30% (see Figure 10).

In order to evaluate the efficacy of caching (see Section 5.3.4), we measure the number of *equivalent* configurations, i.e., that are mapped to the same signature. The 700,614,759 different configurations generated from the benchmark tree are mapped to "only" 168,130 distinct signatures. This means that, for each configuration, there is on average 4167 equivalent configurations. In other words, a score computed to answer a request can be reused to answer many other different (but equivalent) requests, hence saving computation time and increasing the responsiveness of the tool.

8. Survey and Usage Statistics

In order to gain insights into the way individuals use, perceive, and learn from KGP Meter–and the problem of kin genomic privacy in general–we collected and analyzed survey and usage data. We recruited our participants through a crowdworking platform, which enabled us to collect a *large* dataset from a *representative* sample of individuals. The study (including the deployment of the tool) was approved by our institutional review board (IRB) before its launch.

8.1. Methodology

We relied on the implementation described in Section 6. The text of the website was simplified; in particular, technical details were moved to the FAQ and/or Concept sections. Participants were recruited through Prolific [81], which provided us with a representative sample of the US Internet population. First, participants were directed to a first online questionnaire that includes a few knowledge questions about genomics and (kin) genomic privacy. Then, they were redirected to our website and instructed to browse through it and to try at least three distinct configurations with KGP Meter. Finally, they were directed to a second questionnaire that presents again the knowledge questions from the first questionnaire (to assess the participants' learning), but also includes new ones regarding KGP Meter. The questionnaires included attention checks (see Appendix C on p. 19). We conducted cognitive pretests and adjusted the questionnaire accordingly. The tool usage data was linked to the questionnaire data. The study took ~20 minutes and each participant received a compensation of ~USD 3.2. The study was conducted in May 2020.

8.2. Results

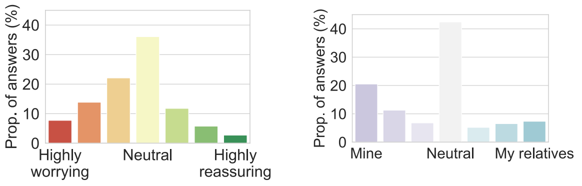
In total, 1,822 individuals took part in the study. We removed those who failed attention checks (Q7 and Q18), those who completed the questionnaires too fast (less than 10 minutes), and those who computed kin genomic privacy scores (with KGP Meter) in less than three different configurations. This left us with 1,580 respondents. Recruited participants were 54.2% "Female". The mean age was 43.0, with a standard deviation of 14.7. The distribution of ethnicity was: 73.0% white, 12.1% black, 7.4% asian, 4.8% mixed, and 2.7% other.

Questionnaire. We asked the respondents whether they had their genome tested (Q8): Only 13.9% answered "Yes", 82.5% answered "No" and 3.3% answered "Not Sure". The remaining 0.3% preferred not to answer. We also asked them whether any of their relatives had their genome tested (Q19): 28.7% answered "Yes", 56.5% answered "No", and 14.7% answered "Not sure". Note that KGP Meter can be used by the respondents who did not have their genome tested (the vast majority) in order to evaluate the privacy consequences–for their relatives–of their own testing and thus to make an informed decision. For those who did not have their genome tested but some of their relatives had, KGP Meter can be used to assess the privacy implications for themselves; for these respondents, the configurations tested with KGP Meter might correspond to the actual ones. The fact that a non-negligible fraction of the respondents were not sure whether their relatives had their genomes tested highlights a key problem in interdependent privacy: Individuals do not possess the required information to properly assess their privacy.

Awareness. We asked the respondents whether they were aware of the genomic privacy risk relatives can create for each other (i.e., kin genomic privacy risks; Q20) before visiting our website: 8.0% answered "Yes", 37.5% answered "To some extent", and 53.9% answered "No".

Concerns. We asked the respondents whether they found the obtained scores reassuring or worrying, on 7-point Likert scale ranging from "highly worrying" to "highly reassuring" (Q23). Figure 11(a) shows the distribution of responses. Overall, the respondents found the results provided by KGP Meter more worrying than reassuring (43.7%). A substantial fraction of the respondents were neutral (36.1%). To better qualify these results, we looked at the comments left in the free text field. (Q28). A respondent asked "*If my mother has her genome sequenced why does it reveal only 22% of my genomic information? Shouldn't it be 50%*", and another one stated "*That if both my parents had the test, you would only know 51% of my genome.*"; these comments highlight the common misunderstanding about genomics and probabilistic reasoning that most people face.

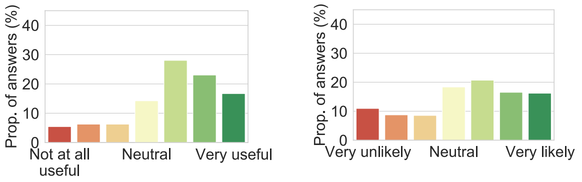
In order to understand the target of the respondents' genomic privacy concerns, we also asked them whether they were more interested in their own genomic privacy or in that of their relatives (Q24). Note that, in order to avoid priming the users towards testing their privacy or that of their relatives, the target was *not* selected by default in KGP Meter. Also, the text on the webpage was phrased in a neutral way. The results are depicted in Figure 11(b); overall the participants were more interested in their own privacy (38.5%) than in their relatives' (19.1%), although a large fraction (42.4%) indicated being equally interested in both (i.e., neutral).[12] When asked why, most respondents reported being self-centered (e.g., "I'm just more concerned with all things regarding myself").



(a) Distribution of responses for survey question "Do you find the obtained genomic privacy scores reassuring or worrying?" (Q23).

(b) Distribution of responses for survey question "Are you more interested in your genomic privacy or rather in that of your relatives?" (Q24).

Figure 11. User concerns regarding Kin Genomic Privacy.



(a) Distribution of responses for survey question "Do you find this tool useful?" (Q26) .

(b) Distribution of responses for survey question "What are the chances that you would recommend this website?" (Q27); NPS.

Figure 12. User satisfaction regarding KGP Meter.

Satisfaction. Finally, we asked the participants their opinion about KGP Meter in terms of perceived usefulness (Q26) and net promoter score (NPS) (Q27). The results are depicted in Figure 12. The majority of respondents found KGP Meter useful (67.8%) and were likely to recommend KGP Meter to other people (53.4%). The feedback of the respondents (in free text, Q28) was positive overall. They pointed out the usefulness of KGP Meter and suggested possible improvements including: (1) more detailed explanations on the score and on the implications of ethnicity information, and (2) the ability to handle stepfamilies and homoparental families. For future work, we will refine the tool/website accordingly. It should be noted that survey participants could be positively biased when evaluating tools created by researchers [88]; for this reason, we did not mention that the tool was created by us.

Knowledge and Learning. In order to assess the general knowledge of the participants, we asked them a number of knowledge questions under the forms of multiple-choice questions with 5 options (only one was correct, the last one was "None of the above"; see the transcript in Appendix C on p. 19). In order to evaluate their learning, we asked them these questions before and after the participants visited our website and used KGP Meter. The raw results are depicted in Figure 13. Note that all

[12] Previous work has investigated/modeled how individuals' value the privacy of others (incl. friends and relatives) when making privacy decisions in different domains (e.g., [82]–[86]) including genomics (e.g., [57], [87]). Results show that although individuals care about others' privacy, they care substantially less than for their own privacy.

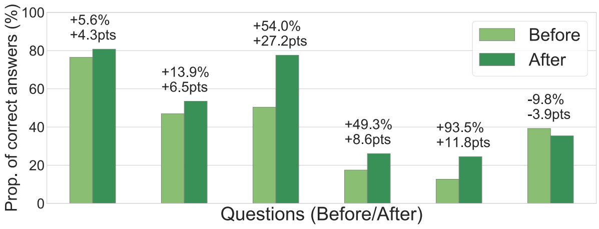


Figure 13. Raw survey results for knowledge questions (Q1/Q10, Q2/Q11, Q3/Q14, Q4/Q15, Q5/Q16, and Q6/Q17). For each question, the left (resp. right) bar represents the proportion of participants who answered correctly the question *before* (resp. *after*) visiting the website. The difference between the two is shown in percentage points and in percents.

the differences in responses before and after using the tool are statistically significant.

We first asked the participants about the functioning of genetic inheritance (Q1/Q10) and about the definition of a SNP (Q2/Q11). Note that the answers to these questions were not on the home page of the website but only in the Concept page. 76.5% of the participants answered correctly the first question and 47.0% the second. These proportions increased to 80.8% and 53.5% after visiting the website; this constitutes an increase of 5.6% and 13.9%.

We then asked the participants about their (relative) privacy in different configurations (e.g., "When the genomes of both your parents are known, compared to the case when the genome of only one of your parents is known, your privacy is lower/the same/higher/not comparable/none of the above") (Q3/Q14, Q4/Q15, Q5/Q16, and Q6/Q17). Note that the answers to these questions could be obtained by using KGP Meter. Overall, the background of the participants was limited, with a proportion of correct answers lower than for a random guess (i.e., $1/5$ =20%), thus demonstrating common misconceptions about genomics and privacy. The increase of the proportions of respondents who correctly answered these questions, however, was positive and substantial except for the last question: 54.0%, 49.3%, 93.5%, and -9.8%. This increase was much higher (55.7%, 98.8%, 133.3%, and 10.2%) for the participants who tested the corresponding configurations with KGP Meter (results not shown in Figure 13). In particular, for the last question, the increase was positive for such participants. Finally, when asked whether, by visiting the website and using KGP Meter, they learned something they did not know before (Q22), 76.0% of the participants answered "Yes", 8.4% "No", and 15.6% "Not sure". Some participants made notable comments in the associated text box: *"I learned that a person's genome is estimable based upon the genome of their family, but there is substantial uncertainty in this process that still exists"*, *"data that can be inferred from knowing a family member's genome is more complex than I expected, and applies to more types of family members than I expected"*, *"I enjoyed looking at different scenarios and privacy scores. I didn't understand this AT ALL before this survey."*, *"I found it interesting that even having your partner tested can seemingly affect your privacy."*, *"It also never occurred to me that one person getting tested could affect the privacy of other family members."*, and *"That there is an impact on my privacy, and that the impact does not seem to be linear"*.

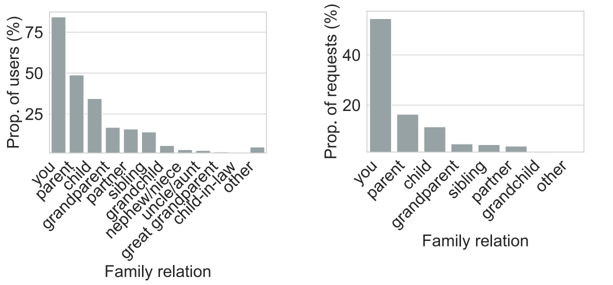
Understanding. In order to assess the participants' understanding of KGP Meter and of the underlying concepts, we asked them two questions: one about the type of risk measured by the score output by KGP Meter (Q12) and one about the set of relatives considered by KGP Meter when enumerating the possible configurations of SNP values in the computation of the score (Q13). 75.9% managed to correctly identify the correct risk (out of 5 options) and 50.5% the correct set of relatives (out of 5 options). These results are well above random guesses, thus demonstrating a good understanding.

Future Actions. In order to assess the influence our website could have on the participants' future actions, we first polled the participants about one specific action: taking a genetic test. We asked those who reported not having already taken a genetic test (wrt Q8) about their intention to have their genome tested in the next 12 months, on a 7-point Likert scale ranging from 1 ("Very unlikely") to 7 ("Very likely") (Q9/Q21). The average intention score increased from 2.1 (SD: 1.4) to 2.9 (SD: 2.3) after visiting our website, while remaining mostly on the "unlikely" side. Note that the standard deviation increased; this could be explained by the fact that neutral participants toggled to a decision and/or that decided participants strengthened their opinion. Regarding potential actions induced by the tool, note that, unlike passwords that can be reset (hence restoring security), genetic tests cannot be undone. Yet, our tool can help users avoid further decreasing their privacy (if genetic tests have been done already) and test hypothetical scenarios before taking a test.

Finally, we asked the participants whether they intended to use the information gained when visiting our website, and, if yes, how they intended to use it (Q25). 26.1% answered "Yes", 22.8% "No", and 51.1% "Not sure". A preliminary analysis of the free-text responses showed that the main use of the gained information were (1) to inform relatives (e.g., *"My family [...] have talked about getting genetic testing done. I will share with them the invasion of privacy issue. I know they are unaware of this"*) and possibly discourage them to take genetic tests or encourage them to seek consent beforehand, (2) research more on the topic (e.g., *"I am going to use [the tool] again and then read up on this"*), (3) give up taking a test (e.g., *"I am much less likely to get my genome sequenced"*).

Usage. Users spent a median time of 5.0 minutes (Q1 2.9 and Q3 9.5 minutes) using KGP Meter and made 13.4 ±12.0 requests for privacy scores on average. Note that some of these requests are intermediary requests sent while building the desired configuration.

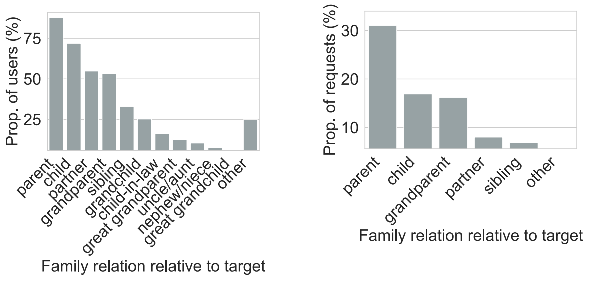
In order to better understand the target of the users' privacy concerns, we first look at the family ties between the *users* (marked with a 👤 in the UI of the tool) and the *targets* (marked with a ⌖) in the users' requests. Figure 14(a) depicts the proportion of users who selected a given target (grouped by family tie, i.e., themselves: "you", one of their parents, one of their children, etc.) in at least one of their requests, and Figure 14(b) depicts the proportion of requests per target (grouped by family tie). We observe that 84.3% of the users made at least one request for their own privacy score (i.e., they were the target); unsurprisingly, "you" (i.e., the user) is the most frequent target. More generally, we observe that the



(a) Proportion of users who selected a given target (grouped by family tie, wrt the user) in one of their requests.

(b) Proportion of requests per target (grouped by family tie, wrt the user).

Figure 14. Family ties between users and targets in user requests (👤 ↔ ⌖).



(a) Proportion of users who selected a given relative (grouped by family tie, wrt the target) in at least one of their requests.

(b) Proportion of requests per sequenced relatives (grouped by family tie, wrt the target).

Figure 15. Family ties between targets and relatives whose genomes are known in user requests (⌖ ↔ ⧗).

closer (in the family tree) a relative is from the user, the higher the chance that the relative is chosen as a target; this denotes a strongest concern for the privacy of the users' closest relatives. This can also be explained by the fact that users tested the tool with family trees limited to close relatives. It can also be observed that parents are most often selected as targets than children. However, this does not necessarily mean that users are more concerned about their parents' privacy; it could simply reflect the fact that users do not have children (yet). On average, users considered 3.0 ± 2.1 distinct targets.

We also look at the typical configurations tested by the users. To do so, we look at the family ties between the *relatives whose genomes were marked as known* (marked with a ⧗) and the *target*, independently from the users themselves. The results are depicted in Figure 15. We observe that, in 31.0% of the *requests*, the genome of a parent of the target is marked as known. Again, the closer the relative is to the target, the higher the chances are that the user tests a configuration where the relative's genome is marked as known. The most frequent scenario, tested by the majority of the users, is the case where only the genomes of the two parents of the target are marked as known (distribution not shown in the paper). The average number of relatives whose genomes are known is 2.1 ± 2.4, and 1.4 ± 1.3 in the simplified tree: Overall, users tested the tool with small to medium family trees. The discrepancies between these two numbers could denote the users' misunderstanding regarding the information conveyed by the genome of a relative (i.e., the fact that the genome of the maternal grandmother of the target does not bring additional information about the target's genome if the genome of the target's mother is known). It could also be because users wanted to challenge their preconceived ideas about this, especially since the questionnaire included a question on this.

Finally, we looked at the efficiency of the tool in answering user requests: 98.0% of the users' requests were answered from the cache, thus providing a high level of responsiveness (≈ 0.2 secs.). For the 2.0% of requests that were not in the cache, the average response time was 0.179 ± 0.813 seconds. No request was left unanswered (i.e., timed out). and 99.9% of the requests were answered in less than a second.

9. Related Work

We focus on the works for (1) quantifying (kin) genomic privacy and (2) conveying security and privacy scores to users (beyond kin genomic privacy). For a comprehensive survey/systematization of knowledge articles on genomic privacy, we refer the reader to [14]–[16]. Besides, our work is also related to interdependent privacy situations that occur when the actions or data of someone affects the privacy of others (see [8] for a comprehensive survey). For instance, previous research has shown that it is possible to infer an individual's personal attributes based on the data of others (e.g., for social network data with friendship information [89], [90] and for location data with co-location information [91]).

Humbert et al. were the first to propose a quantification framework for evaluating privacy risks caused by relatives sharing their genomic data [34]. They rely on different privacy metrics, including the one used in this paper, and they evaluate their framework on real data. In their follow-up work, Humbert et al. propose a new Bayesian network model for the inference based on inter-genome correlations only [37]. The main differences between our work and both [34], [37] are the following: (i) our algorithm does not require the genomic data of the target, nor of their relatives, (ii) we construct an easy-to-use interactive and responsive tool (GUI), and (iii) we gather the perception and learning of the users through a large-scale user survey.

Wagner surveys 23 metrics in genomic privacy and categorizes them with respect to what they capture [51]. She introduces monotonicity of the metric with respect to the adversarial strength as a key requirement, and provides suggestions on metric selection, interpretation and visualization. Her empirical evaluation shows that none of the metrics is sufficiently reliable when used in isolation. Thus, she recommends combining several metrics. In our work, as we do not have access to the ground truth, we rely on mutual information.

Saha et al. very recently studied user attitudes on DTC genetic testing with semi-structured interviews (N=24) [30]. Yet, they focus on general genomic privacy concerns and only one paragraph is related to kin genomic privacy. The associated results show that the privacy implications for relatives are unclear, hence the need for tools such as KGP Meter. The aforementioned works as well as our work are related to generic genomic information leakage. Recent works focused on specific risks, namely identity inference attacks [48], [49], and phenotype, kinship, and membership inference attacks [92].

A large amount of work has been devoted to evaluating and communicating to users the strength of their passwords [75], [76], [93]–[101]. Ur et al. were the first to conduct a large-scale study of 14 password-strength meters [76]. This study shows that meters influence user behavior and security. Meters that rated passwords stringently led users to choose significantly longer passwords harder to crack but not less memorable or usable. Besides, the joint use of a visual indicator and text outperformed either used in isolation. Ur et al. developed a data-driven password-strength meter by relying on neural networks and on heuristics to score passwords and generate text feedback to the user [100]. Their online study shows that the combination of a colored strength bar, detailed text feedback, and improvement suggestion leads to more secure passwords than only a strength bar. Golla and Dürmuth measure the accuracy of 45 password-strength meters [101]. By relying on correlation-based measures, they find that meters used in practice are less accurate than academic proposals and that no significant improvement in meter accuracy was made over the last five years.

Riederer et al. propose a web-based application to help users understand the privacy effect of sharing location data [32]. Their tool enables users to import location data collected by popular services, visualize it, view the demographics of their visited locations (race, income, age, . . .), and finally receive a prediction of their own demographics based on this data. Shokri et al. propose a framework that enables mobile users to quantify their location privacy [31]. Unlike [32], this location-privacy meter remains at the geographical level and does not take into account location semantics. Furthermore, it does not provide any visualization of the resulting location privacy level. Note that, unlike our work, both these tools require access to the actual (location) data of the users.

In order to better inform users about what data third-party apps (e.g., for cloud storage) can access, Harkous et al. propose data-driven privacy indicators [102]. They present an interface that informs the user about what data the app has already access to, from previous app installations by the user or by their friends or collaborators. For instance, their privacy indicator interface can show that a specific cloud storage app has already access to 70% of the user's files because one of their friends has already installed this app. Finally, Lin et al. developed a privacy meter for mobile (Android) apps that assigns grades between 'A+' and 'D', depending on the privacy-related behavior of the app [77], [103].

Positioning. To summarize, our work advances the state of the art by proposing the first (kin) genomic privacy meter. One of the key advantage and novelty of KGP Meter is to rely only on the family tree and not on actual genomic data, which was not possible with previous work. This is made possible by our new quantification framework which unlocks the implementation and deployment of a tool for the general public.

10. Conclusion and Future Work

With this work, we provide the first means for raising awareness about kin genomic privacy risks and for helping decision-making in families.Yet, the design of KGP Meter and the online interface is not complete. For future work, we intend to implement several new features for the tool, including (1) improved information visualization of the privacy score, (2) the ability to handle step-families, and (3) the ability to choose a specific set of SNPs (e.g., ApoE4) for evaluating privacy. We plan to deepen our analysis of the survey results, in particular by conducting a thematic analysis of the answers to the open-ended questions. We also plan to further improve usability and learnability of the tool through AB testing and additional usability studies. For this matter, we will collaborate with educational designers and risk communicators. In the long term, we also intend to extend our approach for different data types (i.e., beyond genomic data). Finally, a potential avenue for future work is to investigate the risks of inferring errors in the parentage reported in the family tree based on the genomes of some of the relatives (e.g., inferring that the user's father is not his biological father when some relatives' genomes are known).

Acknowledgments

The authors express their sincere gratitude to Murielle Bochud, Bertil Chapuis, Yamane El-Zein, Sandrine Estoppey Younes, Jean-Pierre Hubaux, Alain Mermoud, Alexandra-Mihaela Olteanu, Dimitri Pércia-David, Michael Schumacher, and Noé Zufferey for their insightful feedback on the tool, to Laurelei Eggel and to Rémi Coudert for the help designing the survey questionnaire and analyzing the results, and to Holly Cogliati and Vincent Vandersluis for their great editing job on the manuscript. The authors also thank Benjamin Trubert for his participation in the implementation of the tool. The work was partially funded with a grant from the Fondation Leenaards, as part of the personalized health and society (SantéPerSo) initiative.

References

[1] 23andMe, Inc., "23andMe: DNA Genetic Testing & Analysis," https://www.23andme.com/, 2019.

[2] Ancestry.com DNA, LLC, "AncestryDNA: DNA Tests for Ethnicity and Genealogy DNA Test," https://www.ancestry.com/dna/, 2019.

[3] Antonio Regalado, "More than 26 million people have taken an at-home ancestry test," *MIT Technology Review*, Feb. 2019.

[4] E. Ayday, E. De Cristofaro, J.-P. Hubaux, and G. Tsudik, "Whole Genome Sequencing: Revolutionary Medicine or Privacy Nightmare?" *Computer*, vol. 48, no. 2, pp. 58–66, Feb. 2015.

[5] Eric Rosenbaum, "5 biggest risks of sharing your DNA with consumer genetic-testing companies," *CNBC*, Jun. 2018.

[6] Harriet Alexander, "Pentagon warns US military not to use home DNA testing kits," https://www.telegraph.co.uk/news/2019/12/23/pentagon-warns-us-military-not-use-home-dna-testing-kits/, Dec. 2019.

[7] E. Ayday and M. Humbert, "Inference Attacks against Kin Genomic Privacy," *IEEE Security Privacy*, vol. 15, no. 5, pp. 29–37, 2017.

[8] M. Humbert, B. Trubert, and K. Huguenin, “A Survey on Interdependent Privacy,” *ACM Computing Surveys*, vol. 52, no. 6, p. 40, 2019.

[9] Emily Mullin, “Do Your Family Members Have a Right to Your Genetic Code?” *MIT Technology Review*, Nov. 2016.

[10] GEDmatch, Inc., “GED match: Tools for DNA and Genealogy Research,” https://www.gedmatch.com, 2019.

[11] MyHeritage Ltd., “Family Tree, Genealogy, Family History, and DNA,” https://www.myheritage.com, 2019.

[12] P. Bayer, H. Rausch, and B. Greshake Tzovaras, “openSNP: Share your genetic tests results,” https://opensnp.org/, 2019.

[13] D. Sero, A. Zaidi, J. Li, J. D. White, T. B. G. Zarzar, M. L. Marazita, S. M. Weinberg, P. Suetens, D. Vandermeulen, J. K. Wagner, M. D. Shriver, and P. Claes, “Facial recognition from DNA using face-to-DNA classifiers,” *Nature Communications*, vol. 10, no. 1, Dec. 2019.

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

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

[16] A. Mittos, B. Malin, and E. De Cristofaro, “Systematizing Genome Privacy Research: A Privacy-Enhancing Technologies Perspective,” *Proceedings on Privacy Enhancing Technologies (PoPETs)*, vol. 2019, no. 1, pp. 87–107, Jan. 2019.

[17] A. Johnson and V. Shmatikov, “Privacy-preserving data exploration in genome-wide association studies,” in *Proc. of the ACM Conf. on Knowledge Discovery and Data Mining (KDD)*. Chicago, Illinois, USA: ACM, 2013, p. 1079.

[18] C. Uhlerop, A. Slavković, and S. E. Fienberg, “Privacy-Preserving Data Sharing for Genome-Wide Association Studies,” *The Journal of Privacy and Confidentiality*, vol. 5, no. 1, pp. 137–166, 2013.

[19] S. Wang, Y. Zhang, W. Dai, K. Lauter, M. Kim, Y. Tang, H. Xiong, and X. Jiang, “HEALER: Homomorphic computation of ExAct Logistic rEgRession for secure rare disease variants analysis in GWAS,” *Bioinformatics*, p. btv563, Oct. 2015.

[20] F. Tramèr, Z. Huang, J.-P. Hubaux, and E. Ayday, “Differential Privacy with Bounded Priors: Reconciling Utility and Privacy in Genome-Wide Association Studies,” in *Proc. of the ACM Conf. on Computer and Communications Security (CCS)*. Denver, Colorado, USA: ACM, 2015, pp. 1286–1297.

[21] S. Simmons, C. Sahinalp, and B. Berger, “Enabling Privacy-Preserving GWASs in Heterogeneous Human Populations,” *Cell Systems*, vol. 3, no. 1, pp. 54–61, Jul. 2016.

[22] K. A. Jagadeesh, D. J. Wu, J. A. Birgmeier, D. Boneh, and G. Bejerano, “Deriving genomic diagnoses without revealing patient genomes,” *Science*, vol. 357, no. 6352, pp. 692–695, Aug. 2017.

[23] J. L. Raisaro, J. R. Troncoso-Pastoriza, M. Misbach, J. S. Sousa, S. Pradervand, E. Missiaglia, O. Michielin, B. Ford, and J.-P. Hubaux, “MedCo: Enabling Secure and Privacy-Preserving Exploration of Distributed Clinical and Genomic Data,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 16, no. 4, pp. 1328–1341, Jul. 2019.

[24] P. Baldi, R. Baronio, E. De Cristofaro, P. Gasti, and G. Tsudik, “Countering GATTACA: Efficient and secure testing of fully-sequenced human genomes,” in *Proc. of the ACM Conf. on Computer and Communications Security (CCS)*. Chicago, Illinois, USA: ACM, 2011, p. 691.

[25] E. De Cristofaro, S. Faber, and G. Tsudik, “Secure genomic testing with size- and position-hiding private substring matching,” in *Proc. of the Workshop on Privacy in the Electronic Society (WPES)*. Berlin, Germany: ACM, 2013, pp. 107–118.

[26] E. Ayday, J. L. Raisaro, J.-P. Hubaux, and J. Rougemont, “Protecting and evaluating genomic privacy in medical tests and personalized medicine,” in *Proc. of the Workshop on Privacy in the Electronic Society (WPES)*. Berlin, Germany: ACM, 2013, pp. 95–106.

[27] M. Djatmiko, A. Friedman, R. Boreli, F. Lawrence, B. Thorne, and S. Hardy, “Secure Evaluation Protocol for Personalized Medicine,” in *Proc. of the Workshop on Privacy in the Electronic Society (WPES)*. Scottsdale, Arizona, USA: ACM, 2014, pp. 159–162.

[28] M. Naveed, S. Agrawal, M. Prabhakaran, X. Wang, E. Ayday, J.-P. Hubaux, and C. Gunter, “Controlled Functional Encryption,” in *Proc. of the ACM Conf. on Computer and Communications Security (CCS)*. Scottsdale, Arizona, USA: ACM, 2014, pp. 1280–1291.

[29] P. J. McLaren, M. Aouri, M. Rotger, E. Ayday, I. Bartha, M. B. Delgado, Y. Vallet, H. F. Günthard, M. Cavassini, H. Furrer, T. Doco-Lecompte, C. Marzolini, P. Schmid, C. Di Benedetto, L. A. Decosterd, J. Fellay, J.-P. Hubaux, A. Telenti, and The Swiss HIV Cohort Study, “Privacy-preserving genomic testing in the clinic: A model using HIV treatment,” *Genetics in Medicine*, vol. 18, no. 8, pp. 814–822, Aug. 2016.

[30] D. Saha, A. Chan, B. Stacy, K. Javkar, S. Patkar, and M. L. Mazurek, “User Attitudes On Direct-to-Consumer Genetic Testing,” in *Proc. of the IEEE European Symp. on Security and Privacy (EuroS&P)*, 2020.

[31] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, and J.-P. Hubaux, “Quantifying location privacy,” in *Proc. of the IEEE Symp. on Security and Privacy (S&P)*. IEEE, 2011, pp. 247–262.

[32] C. Riederer, D. Echickson, S. Huang, and A. Chaintreau, “FindYou: A Personal Location Privacy Auditing Tool,” in *Proc. of the ACM Int’l Conf. Companion on the World Wide Web (WWW Companion)*. Montréal, QC, Canada: ACM, 2016, pp. 243–246.

[33] A. Boutet and S. Gambs, “Inspect What Your Location History Reveals About You: Raising user awareness on privacy threats associated with disclosing his location data,” in *Proc.of the ACM Int’l Conf. on Information and Knowledge Management (CIKM)*. Beijing China: ACM, Nov. 2019, pp. 2861–2864.

[34] M. Humbert, E. Ayday, J.-P. Hubaux, and A. Telenti, “Addressing the Concerns of the Lacks Family: Quantification of Kin Genomic Privacy,” in *Proc. of the ACM Conf. on Computer and Communications Security (CCS)*. ACM, 2013, pp. 1141–1152.

[35] F. F.-H. Nah, “A study on tolerable waiting time: How long are Web users willing to wait?” *Behaviour & Information Technology*, vol. 23, no. 3, pp. 153–163, May 2004.

[36] National Center for Biotechnology Information, “dbSNP: The Single Nucleotide Polymorphism Database (dbSNP) of Nucleotide Sequence Variation,” https://www.ncbi.nlm.nih.gov/snp/, 2019.

[37] M. Humbert, E. Ayday, J.-P. Hubaux, and A. Telenti, “Quantifying Interdependent Risks in Genomic Privacy,” *ACM Transactions on Privacy and Security*, vol. 20, no. 1, pp. 1–31, Feb. 2017.

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

[39] N. Homer, S. Szelinger, M. Redman, D. Duggan, W. Tembe, J. Muehling, J. V. Pearson, D. A. Stephan, S. F. Nelson, and D. W. Craig, “Resolving Individuals Contributing Trace Amounts of DNA to Highly Complex Mixtures Using High-Density SNP Genotyping Microarrays,” *PLoS Genetics*, vol. 4, no. 8, p. e1000167, Aug. 2008.

[40] R. Wang, Y. F. Li, X. Wang, H. Tang, and X. Zhou, “Learning your identity and disease from research papers: Information leaks in genome wide association study,” in *Proc. of the ACM Conf. on Computer and Communications Security (CCS)*. Chicago, Illinois, USA: ACM, 2009, p. 534.

[41] S. Sankararaman, G. Obozinski, M. I. Jordan, and E. Halperin, “Genomic privacy and limits of individual detection in a pool,” *Nature Genetics*, vol. 41, no. 9, pp. 965–967, Sep. 2009.

[42] B. Pasaniuc and A. L. Price, “Dissecting the genetics of complex traits using summary association statistics,” *Nature Reviews Genetics*, vol. 18, no. 2, pp. 117–127, Feb. 2017.

[43] M. Backes, P. Berrang, M. Humbert, and P. Manoharan, “Membership Privacy in MicroRNA-based Studies,” in *Proc. of the ACM Conf. on Computer and Communications Security (CCS)*. Vienna Austria: ACM, Oct. 2016, pp. 319–330.

[44] I. Hagestedt, M. Humbert, P. Berrang, I. Lehmann, R. Eils, M. Backes, and Y. Zhang, "Membership Inference Against DNA Methylation Databases," in *Proc. of the IEEE European Symp. on Security and Privacy (EuroS&P)*. Genoa, Italy: IEEE, Sep. 2020, pp. 509–520.

[45] M. Gymrek, A. L. McGuire, D. Golan, E. Halperin, and Y. Erlich, "Identifying Personal Genomes by Surname Inference," *Science*, vol. 339, no. 6117, pp. 321–324, Jan. 2013.

[46] M. Humbert, K. Huguenin, J. Hugonot, E. Ayday, and J.-P. Hubaux, "De-anonymizing Genomic Databases Using Phenotypic Traits," *Proceedings on Privacy Enhancing Technologies (PoPETs)*, vol. 2015, no. 2, pp. 99–114, Jun. 2015.

[47] C. Lippert, R. Sabatini, M. C. Maher, E. Y. Kang, S. Lee, O. Arikan, A. Harley, A. Bernal, P. Garst, V. Lavrenko, K. Yocum, T. Wong, M. Zhu, W.-Y. Yang, C. Chang, T. Lu, C. W. H. Lee, B. Hicks, S. Ramakrishnan, H. Tang, C. Xie, J. Piper, S. Brewerton, Y. Turpaz, A. Telenti, R. K. Roby, F. J. Och, and J. C. Venter, "Identification of individuals by trait prediction using whole-genome sequencing data," *Proceedings of the National Academy of Sciences*, vol. 114, no. 38, pp. 10 166–10 171, Sep. 2017.

[48] Y. Erlich, T. Shor, I. Pe'er, and S. Carmi, "Identity inference of genomic data using long-range familial searches," *Science*, vol. 362, no. 6415, pp. 690–694, Nov. 2018.

[49] P. Ney, L. Ceze, and T. Kohno, "Genotype Extraction and False Relative Attacks: Security Risks to Third-Party Genetic Genealogy Services Beyond Identity Inference," in *Proc. of the Network and Distributed System Security Symp. (NDSS)*, 2020, p. 16.

[50] M. D. Edge and G. Coop, "Attacks on genetic privacy via uploads to genealogical databases," Genomics, Preprint, Oct. 2019.

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

[52] The 1000 Genomes Project Consortium, "A map of human genome variation from population-scale sequencing," *Nature*, vol. 467, no. 7319, pp. 1061–1073, Oct. 2010.

[53] Gene by Gene Ltd., "FamilyTreeDNA," https://www.familytreedna.com, 2019.

[54] A. Abrams, "How an Online DNA Service Revealed the Suspected Golden State Killer," *Time*, Apr. 2018.

[55] H. Murphy, "Genealogists Turn to Cousins' DNA and Family Trees to Crack Five More Cold Cases," *The New York Times*, Sep. 2018.

[56] Illumina, Inc, "Illumina," https://www.illumina.com, 2019.

[57] M. Humbert, E. Ayday, J.-P. Hubaux, and A. Telenti, "On non-cooperative genomic privacy," in *Proc. of the Int'l Conf. on Financial Cryptography and Data Security (FC)*. Springer, 2015, pp. 407–426.

[58] M. Backes, P. Berrang, M. Bieg, R. Eils, C. Herrmann, M. Humbert, and I. Lehmann, "Identifying Personal DNA Methylation Profiles by Genotype Inference," in *Proc. of the IEEE Symp. on Security and Privacy (S&P)*. San Jose, CA, USA: IEEE, May 2017, pp. 957–976.

[59] P. Berrang, M. Humbert, Y. Zhang, I. Lehmann, R. Eils, and M. Backes, "Dissecting privacy risks in biomedical data," in *Proc. of the IEEE European Symp. on Security and Privacy (EuroS&P)*. IEEE, 2018, pp. 62–76.

[60] M. Backes, P. Berrang, M. Humbert, X. Shen, and V. Wolf, "Simulating the Large-Scale Erosion of Genomic Privacy Over Time," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, pp. 1405–1412, 2018.

[61] D. E. Reich, M. Cargill, S. Bolk, J. Ireland, P. C. Sabeti, D. J. Richter, T. Lavery, R. Kouyoumjian, S. F. Farhadian, R. Ward, and Lander, Eric S., "Linkage disequilibrium in the human genome," *Nature*, vol. 411, no. 6834, pp. 199–204, 2001.

[62] D. S. Falconer and T. F. C. Mackay, *Introduction to Quantitative Genetics*, 4th ed. Harlow: Pearson, Prentice Hall, 2009.

[63] F. V. Jensen and F. Jensen, "Optimal junction trees," in *Proc. of the Int'l Conf. on Uncertainty in Artificial Intelligence*. Elsevier, 1994, pp. 360–366.

[64] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*, ser. Adaptive Computation and Machine Learning. Cambridge, MA: MIT Press, 2009.

[65] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, ser. The Morgan Kaufmann Series in Representation and Reasoning. San Mateo, Calif: Morgan Kaufmann Publishers, 1988.

[66] G. Smith, "On the Foundations of Quantitative Information Flow," in *Foundations of Software Science and Computational Structures*, L. de Alfaro, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, vol. 5504, pp. 288–302.

[67] M. S. Alvim, K. Chatzikokolakis, C. Palamidessi, and G. Smith, "Measuring Information Leakage Using Generalized Gain Functions," in *Proc. of the IEEE Computer Security Foundations Symposium (CSF)*. Cambridge, MA, USA: IEEE, Jun. 2012, pp. 265–279.

[68] B. Köpf and D. Basin, "An information-theoretic model for adaptive side-channel attacks," in *Proc. of the ACM Conf. on Computer and Communications Security (CCS)*. Alexandria, Virginia, USA: ACM, 2007, p. 286.

[69] C. Díaz, S. Seys, J. Claessens, and B. Preneel, "Towards Measuring Anonymity," in *Proc. of the Int'l Symp. on Privacy Enhancing Technologies (PETS)*, vol. 2482. Springer, 2002, pp. 54–68.

[70] A. Serjantov and G. Danezis, "Towards an Information Theoretic Metric for Anonymity," in *Proc. of the Int'l Symp. on Privacy Enhancing Technologies (PETS)*. Springer, 2002, pp. 41–53.

[71] M. S. Alvim, M. E. Andrés, K. Chatzikokolakis, and C. Palamidessi, "Quantitative Information Flow and Applications to Differential Privacy," in *Proc. of the Int'l School on Foundations of Security Analysis and Design (FOSAD)*. Springer, 2011, pp. 211–230.

[72] Jeffrey Rubin and Dana Chisnell, *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests, 2nd Edition — Wiley*, 2nd ed. Hoboken, NJ, USA: Wiley, 2011.

[73] Intellectual Reserve, Inc., "Family History and Genealogy Records," https://www.familysearch.org/, 2019.

[74] W. Lidwell, K. Holden, and J. Butler, *Universal Principles of Design, Revised and Updated: 125 Ways to Enhance Usability, Influence Perception, Increase Appeal, Make Better Design Decisions, and Teach through Design*. Beverly, Mass: Rockport Publishers, Jan. 2010.

[75] X. D. C. De Carnavalet, M. Mannan *et al.*, "From very weak to very strong: Analyzing password-strength meters." in *Proc. of the Network and Distributed System Security Symp. (NDSS)*, 2014, pp. 23–26.

[76] B. Ur, P. G. Kelley, S. Komanduri, J. Lee, M. Maass, M. L. Mazurek, T. Passaro, R. Shay, T. Vidas, L. Bauer *et al.*, "How does your password measure up? the effect of strength meters on password creation," in *Proc. of the USENIX Security Symposium (USENIX Security)*. USENIX, 2012, pp. 65–80.

[77] J. Lin, N. Sadeh, S. Amini, J. Lindqvist, J. I. Hong, and J. Zhang, "Expectation and purpose: Understanding users' mental models of mobile app privacy through crowdsourcing," in *Proc. of the ACM Int'l Conf. on Ubiquitous Computing (UbiComp)*. ACM, 2012, p. 501.

[78] Mike Bostock, "D3js: Data-Driven Documents," https://d3js.org, 2019.

[79] A. Ankan, "Pgmpy, a Python library for working with Probabilistic Graphical Models," https://github.com/pgmpy/pgmpy, 2019.

[80] Norsys Software Corp., "Netica," https://www.norsys.com/netica.html, 2019.

[81] Prolific, Inc., "Prolific: Online participant recruitment for surveys and market research." https://www.prolific.co/about/, 2020.

[82] Y. Pu and J. Grossklags, "An Economic Model and Simulation Results of App Adoption Decisions on Networks with Interdependent Privacy Consequences," in *Proc. of the Conf. on Decision and Game Theory for Security (GameSec)*. Springer, Nov. 2014, pp. 246–265.

[83] ——, "Towards a Model on the Factors Influencing Social App Users' Valuation of Interdependent Privacy," *Proceedings on Privacy Enhancing Technologies (PoPETs)*, vol. 2016, no. 2, pp. 61–81, Jan. 2016.

[84] ——, "Using conjoint analysis to investigate the value of interdependent privacy in social app adoption scenarios," in *Proc. of the Int'l Conf. on Information Systems (ICIS)*. AIS, 2015.

[85] ——, "Valuating Friends' Privacy: Does Anonymity of Sharing Personal Data Matter?" in *Proc. of the Symp. on Usable Privacy and Security (SOUPS)*. USENIX, 2017.

[86] N. Wang, H. Xu, and J. Grossklags, "Third-party apps on Facebook: Privacy and the illusion of control," in *Proc. of the ACM Symp. on Computer Human Interaction for Management of Information Technology (CHIMIT)*. Cambridge, Massachusetts: ACM, 2011, pp. 1–10.

[87] J. Weidman, W. Aurite, and J. Grossklags, "On Sharing Intentions, and Personal and Interdependent Privacy Considerations for Genetic Data: A Vignette Study," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 16, no. 4, pp. 1349–1361, Jul. 2019.

[88] N. Dell, V. Vaidyanathan, I. Medhi, E. Cutrell, and W. Thies, ""Yours is Better!": Participant response bias in HCI," in *Proc. of the ACM Conf. on Human Factors in Computing Systems (CHI)*. ACM, 2012, pp. 1321–1330.

[89] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel, "You are who you know: Inferring user profiles in online social networks," in *Proc. of the ACM Int'l Conf. on Web Search and Data Mining (WSDM)*. ACM, 2010, pp. 251–260.

[90] N. Z. Gong and B. Liu, "You Are Who You Know and How You Behave: Attribute Inference Attacks via Users Social Friends and Behaviors," in *Proc. of the USENIX Security Symposium (USENIX Security)*. USENIX, 2016, pp. 979–995.

[91] A.-M. Olteanu, K. Huguenin, R. Shokri, M. Humbert, and J.-P. Hubaux, "Quantifying Interdependent Privacy Risks with Location Data," *IEEE Transactions on Mobile Computing*, vol. 16, no. 3, pp. 829–842, Mar. 2017.

[92] J. L. Raisaro, J. R. Troncoso-Pastoriza, Y. El-Zein, M. Humbert, J. Fellay, C. Troncoso, and J.-P. Hubaux, "GenoShare: Supporting Privacy-Informed Decisions for Sharing Individual-Level Genetic Data," in *AMIA*, 2020.

[93] C. Castelluccia, M. Dürmuth, and D. Perito, "Adaptive password-strength meters from markov models." in *Proc. of the Network and Distributed System Security Symp. (NDSS)*, 2012.

[94] M. L. Mazurek, S. Komanduri, T. Vidas, L. Bauer, N. Christin, L. F. Cranor, P. G. Kelley, R. Shay, and B. Ur, "Measuring password guessability for an entire university," in *Proc. of the ACM Conf. on Computer and Communications Security (CCS)*. ACM, 2013, pp. 173–186.

[95] S. Egelman, A. Sotirakopoulos, I. Muslukhov, K. Beznosov, and C. Herley, "Does my password go up to eleven?: The impact of password meters on password selection," in *Proc. of the ACM Conf. on Human Factors in Computing Systems (CHI)*, ACM. ACM, 2013, pp. 2379–2388.

[96] B. Ur, S. M. Segreti, L. Bauer, N. Christin, L. F. Cranor, S. Komanduri, D. Kurilova, M. L. Mazurek, W. Melicher, and R. Shay, "Measuring real-world accuracies and biases in modeling password guessability," in *Proc. of the USENIX Security Symposium (USENIX Security)*. USENIX, 2015, pp. 463–481.

[97] W. Melicher, B. Ur, S. M. Segreti, S. Komanduri, L. Bauer, N. Christin, and L. F. Cranor, "Fast, lean, and accurate: Modeling password guessability using neural networks," in *Proc. of the USENIX Security Symposium (USENIX Security)*. USENIX, 2016, pp. 175–191.

[98] B. Ur, J. Bees, S. M. Segreti, L. Bauer, N. Christin, and L. F. Cranor, "Do users' perceptions of password security match reality?" in *Proc. of the ACM Conf. on Human Factors in Computing Systems (CHI)*, ACM. ACM, 2016, pp. 3748–3760.

[99] S. M. Segreti, W. Melicher, S. Komanduri, D. Melicher, R. Shay, B. Ur, L. Bauer, N. Christin, L. F. Cranor, and M. L. Mazurek, "Diversify to survive: Making passwords stronger with adaptive policies," in *Proc. of the Symp. on Usable Privacy and Security (SOUPS)*, 2017, pp. 1–12.

[100] B. Ur, F. Alfieri, M. Aung, L. Bauer, N. Christin, J. Colnago, L. F. Cranor, H. Dixon, P. Emami Naeini, H. Habib *et al.*, "Design and evaluation of a data-driven password meter," in *Proc. of the ACM Conf. on Human Factors in Computing Systems (CHI)*, ACM. ACM, 2017, pp. 3775–3786.

[101] M. Golla and M. Dürmuth, "On the accuracy of password strength meters," in *Proc. of the ACM Conf. on Computer and Communications Security (CCS)*. ACM, 2018, pp. 1567–1582.

[102] H. Harkous, R. Rahman, and K. Aberer, "Data-driven privacy indicators," in *Proc. of the Symp. on Usable Privacy and Security (SOUPS)*. USENIX, 2016.

[103] J. Lin, B. Liu, N. Sadeh, and J. I. Hong, "Modeling Users' Mobile App Privacy Preferences: Restoring Usability in a Sea of Permission Settings," in *Proc. of the Symp. on Usable Privacy and Security (SOUPS)*. USENIX, 2014, pp. 199–212.

Appendix A. d-Separation

Before formalizing d-separation, we introduce some notations and definitions. We represent a directed edge between two nodes X and Y as $X \to Y$. The structure $X \to Z \leftarrow Y$, representing a common effect Z, is called a *v-structure*. A sequence of nodes $X_1, \ldots, X_k$ forms a *trail* if, for every $i \in [1, k-1]$, $X_i \rightleftharpoons X_{i+1}$, where $\rightleftharpoons$ denotes an edge of any direction between two nodes. We now introduce the notion of *active trail*.

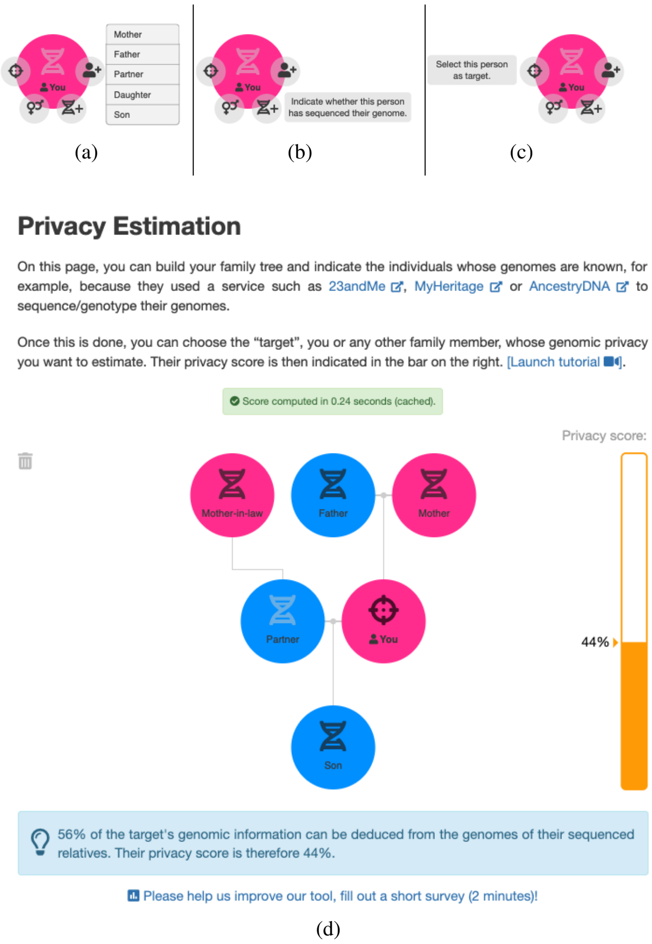
Definition 1 (active trail [64])**.** *Let $\mathcal{N}$ be a BN and $X_1 \rightleftharpoons \cdots \rightleftharpoons X_n$ a trail in $\mathcal{N}$. Let $\mathbf{Z}$ be a subset of observed nodes. The trail $X_1 \rightleftharpoons \cdots \rightleftharpoons X_n$ is called active given $\mathbf{Z}$ if:*

- *For all v-structure $X_{i-1} \to X_i \leftarrow X_{i+1}$, X_i or one of its descendants are in $\mathbf{Z}$;*
- *No other node along the trail is in $\mathbf{Z}$.*

Active trails correspond to trails through which information can flow. An active trail between X_1 and X_n means that influence can flow from X_1 to X_n. Using this notion, we define the concept of d-separation, which enables us to define the set of conditional independencies in a BN.

Definition 2 (d-separation [64])**.** *Let $\mathbf{X}$, $\mathbf{Y}$, $\mathbf{Z}$ be three sets of nodes in $\mathcal{N}$. We say that $\mathbf{X}$ and $\mathbf{Y}$ are d-separated given $\mathbf{Z}$, denoted $\mathsf{d-sep}_{\mathcal{N}}(\mathbf{X}; \mathbf{Y} \mid \mathbf{Z})$, if there is no active trail, given $\mathbf{Z}$, between any node $X \in \mathbf{X}$ and $Y \in \mathbf{Y}$.*

Appendix B. Graphical User Interface



(a)

Indicate whether this person has sequenced their genome.

(b)

Select this person as target.

(c)

Privacy Estimation

On this page, you can build your family tree and indicate the individuals whose genomes are known, for example, because they used a service such as 23andMe, MyHeritage or AncestryDNA to sequence/genotype their genomes.

Once this is done, you can choose the "target", you or any other family member, whose genomic privacy you want to estimate. Their privacy score is then indicated in the bar on the right. [Launch tutorial].

✓ Score computed in 0.24 seconds (cached).

Privacy score:

56% of the target's genomic information can be deduced from the genomes of their sequenced relatives. Their privacy score is therefore 44%.

Please help us improve our tool, fill out a short survey (2 minutes)!

(d)

Figure 16. Illustration of KGP Meter implemented as a web application: Configuration (family tree (a), relatives whose genomes are known (b), and target (c)) and communication of the result (d).

Appendix C.
Transcript of the Questionnaire (Prolific)

In the questions for which there was no natural ordering of the options, the ordering of the options was randomized.

1) **In a pair of nucleotides at a given position in your genome, from whom is each nucleotide inherited?**
   ○ They are both inherited from your father
   ○ They are both inherited from your mother
   ○ They can be generated spontaneously – not inherited from your mother and your father
   ☑ One is inherited from your father, and one from your mother
   ○ None of the above
2) **What is a SNP?**
   ☑ A position in the genome where nucleotides vary among the population
   ○ A position in the genome where nucleotides do not vary among the population
   ○ A disease that is very rare among the population
   ○ A rare anomaly in an individual's genome
   ○ None of the above
3) **When the genomes of both your parents are known, compared to the case when the genome of only one of your parents is known, your privacy is...**
   ○ Higher ○ The same ☑ Lower
   ○ It's not comparable ○ None of the above
4) **If the genomes of both your parents were known, what would your privacy be?**
   ○ 0% ○ 70% ○ 100% ○ 200%
   ☑ None of the above
5) **If the genomes of one of your parents was known, what would your privacy be?**
   ○ 0% ○ 50% ○ 100% ○ 200%
   ☑ None of the above
6) **Assuming the genome of your father was known, how would your privacy evolve if the genome of his father (i.e. your paternal grandfather) was known as well?**
   ○ It would decrease ○ It would increase
   ☑ It would not change ○ It could either increase or decrease
   ○ None of the above
7) **It's important that you pay attention to this study. Please select the answer 'Genome'. [Attention check]**
   ○ DNA ○ Genomic privacy ☑ Genome ○ SNP
   ○ Nucleotides
8) **Have you had your genome tested?**
   ○ Yes ○ No ○ Not sure ○ Rather not say
9) **How likely are you to get your genome tested in the next 12 months? [Show only if "No" or "Not sure" to Q8]**
   Very unlikely ○ ○ ○ ○ ○ ○ ○ Very likely

[Visit website and use KGP Meter]

10) **In a pair of nucleotides at a given position in your genome, from whom is each nucleotide inherited?**
    ○ They are both inherited from your father
    ○ They are both inherited from your mother
    ○ They can be generated spontaneously – not inherited from your mother and your father
    ☑ One is inherited from your father, and one from your mother
    ○ None of the above
11) **What is a SNP?**
    ☑ A position in the genome where nucleotides vary among the population
    ○ A position in the genome where nucleotides do not vary among the population
    ○ A disease that is very rare among the population
    ○ A rare anomaly in an individual's genome
    ○ None of the above
12) **To what risk is the score provided by this tool related?**
    ○ Risk of the target having a genetic disease
    ☑ Risk of a privacy violation regarding the genome of the target
    ○ Risk of the target getting their job application refused
    ○ Risk of knowing whether the target had their genome tested
    ○ Risk of target's genome being leaked by a genetic-testing company (e.g. 23AndMe) because of hacking
13) **In the general case, for which members of the target's family will the tool consider the possible configurations of their SNP values?**
    ○ Only the target's parents
    ☑ Only those who got their genome tested
    ○ All the members of the target's family tree
    ○ Only the target's grandparents
    ○ None of the above
14) **When the genomes of both your parents are known, compared to the case when the genome of only one of your parents is known, your privacy is...**
    ○ Higher ○ The same ☑ Lower
    ○ It's not comparable ○ None of the above
15) **If the genomes of both your parents were known, what would your privacy be?**
    ○ 0% ○ 70% ○ 100% ○ 200%
    ☑ None of the above
16) **If the genomes of one of your parents was known, what would your privacy be?**
    ○ 0% ○ 50% ○ 100% ○ 200%
    ☑ None of the above
17) **Assuming the genome of your father was known, how would your privacy evolve if the genome of his father (i.e. your paternal grandfather) was known as well?**
    ○ It would decrease ○ It would increase ○ It would not change
    ☑ It could either increase or decrease ○ None of the above
18) **It's important that you pay attention to this study. Please select the answer '75%'. [Attention check]**
    ○ 0% ○ 25% ○ 50% ☑ 75% ○ 100%
19) **To your knowledge, did any of your relatives have their genome tested?**
    ○ Yes ○ No ○ Not sure ○ Rather not say
20) **Before visiting this website, were you aware of the genomic privacy risk relatives can create to each other?**
    ○ Yes ○ To some extent ○ No ○ Rather not say
21) **How likely are you to get your genome tested in the next 12 months? [Show only if "No" or "Not sure" to Q8]**
    Very unlikely ○ ○ ○ ○ ○ ○ ○ Very likely
22) **By visiting the website and playing with the privacy tool, did you learn something you did not know before? If yes, what exactly?**
    ○ Yes ○ No ○ Not sure
23) **Do you find the obtained genomic privacy scores reassuring or worrying? Why?** [Free text]
    Highly worrying ○ ○ ○ ○ ○ ○ ○ Highly reassuring
24) **Are you more interested in your genomic privacy or rather in that of your relatives? Why?** [Free text]
    Mine ○ ○ ○ ○ ○ ○ ○ My relatives'
25) **Are you going to use the information gained through the website? If yes, how?** [Free text]
    ○ Yes ○ No ○ Not sure
26) **Do you find this tool useful?**
    Not at all useful ○ ○ ○ ○ ○ ○ ○ Very useful
27) **What are the chances that you would recommend this website?**
    Very unlikely ○ ○ ○ ○ ○ ○ ○ Very likely
28) **Do you have any comment regarding this tool?** [Free text]

Appendix D.
Benchmark Family Tree

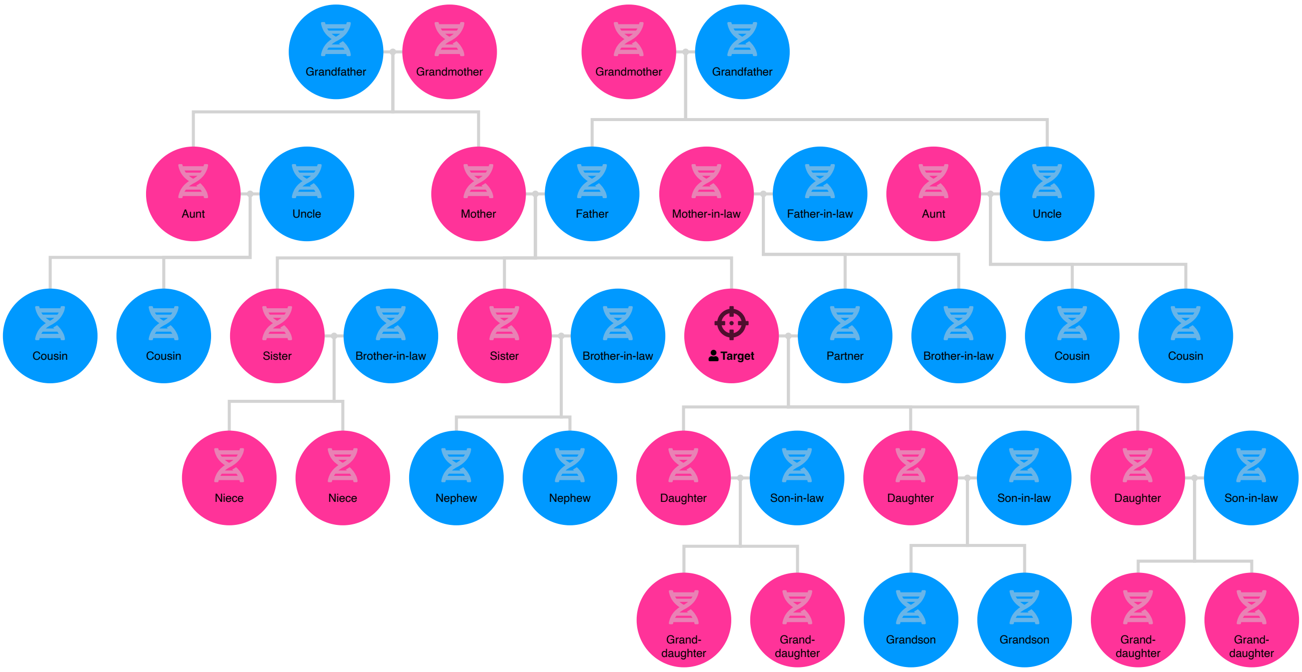


Figure 17. Family tree used for benchmarking.