Briefings in Bioinformatics, 20(3), 2019, 887–895

doi: 10.1093/bib/bbx139
Advance Access Publication Date: 7 November 2017
Paper

Privacy-preserving techniques of genomic data—a survey

Md Momin Al Aziz, Md Nazmus Sadat, Dima Alhadidi, Shuang Wang, Xiaoqian Jiang, Cheryl L. Brown and Noman Mohammed

Corresponding authors: Md Momin Al Aziz, Department of Computer Science, University of Manitoba, Winnipeg R3T2N2, Canada. E-mail: azizmma@cs.umanitoba.ca

Abstract

Genomic data hold salient information about the characteristics of a living organism. Throughout the past decade, pinnacle developments have given us more accurate and inexpensive methods to retrieve genome sequences of humans. However, with the advancement of genomic research, there is a growing privacy concern regarding the collection, storage and analysis of such sensitive human data. Recent results show that given some background information, it is possible for an adversary to reidentify an individual from a specific genomic data set. This can reveal the current association or future susceptibility of some diseases for that individual (and sometimes the kinship between individuals) resulting in a privacy violation. Regardless of these risks, our genomic data hold much importance in analyzing the well-being of us and the future generation. Thus, in this article, we discuss the different privacy and security-related problems revolving around human genomic data. In addition, we will explore some of the cardinal cryptographic concepts, which can bring efficacy in secure and private genomic data computation. This article will relate the gaps between these two research areas—Cryptography and Genomics.

Key words: genomic data privacy; secure computation of genomic data; privacy-preserving techniques; cryptographic methods on genomic data; genomic data security

Introduction

Seminal advancement in genomic data generation over the past decade has impacted health science and related scientific studies. The genesis in data accumulation has made the scientific studies on multiple genre of medical genomics more realistic [1]. Throughout the world, large and varied genomic data sets now help researchers understand the relation between our genomic codes and our health [2, 3]. Genomic data are highly sensitive, as they may reveal the current and future susceptibility of specific diseases for an individual or his/her relatives. Therefore, these unique genomic sequences impose a greater privacy risk (Table 1) for the participants. In addition, genomic databases often are owned by different organizations making it unavailable for public usage. Moreover, these data are storage exhaustive (varying in range of 30–200 GB [16]) and require a high-performance computation when processed.

One popular approach to mitigate this problem is to enforce privacy policies on sharing data. This strategy is popular but challenging, as varying laws and regulations are followed in different institutions worldwide, which govern the sharing and disclosure of these sensitive data. Though these policies are protecting the privacy of the participating individuals, they are

Md Momin Al Aziz is a Research Assistant and a PhD student in the Department of Computer Science at the University of Manitoba, Winnipeg, Canada.
Md Nazmus Sadat is a Research Assistant and an MSc student in the Department of Computer Science at the University of Manitoba, Winnipeg, Canada.
Dima Alhadidi, PhD, is an Assistant Professor in the Faculty of Computer Science at the University of New Brunswick, Fredericton, Canada.
Shuang Wang, PhD, is an Assistant Professor in the Department of Biomedical Informatics at the University of California in San Diego, La Jolla, CA, USA.
Xiaoqian Jiang, PhD is an Assistant Professor in the Department of Biomedical Informatics at the University of California in San Diego, La Jolla, CA, USA.
Cheryl L. Brown, PhD, is an Associate Professor, chair and an undergraduate coordinator in the Department of Political Science and Public Administration at the University of North Carolina at Charlotte, NC, USA.
Noman Mohammed, PhD, is an Assistant Professor in the Department of Computer Science at the University of Manitoba, Winnipeg, Canada.
Submitted: 23 June 2017; **Received (in revised form):** 30 September 2017

Table 1. Notable privacy attacks with the help of public genomic data

Author	Year	Summary
Malin and Sweeney [4]	2000	Identifying of DNA sequence based on available health records and disease background knowledge
Gottlib [5]	2001	Finding employees who are susceptible to genetic diseases depending on genomic data
Lin *et al.* [6]	2004	Identifying a person by only 75 independent SNPs
Homer *et al.* [7]	2008	Telling if a user is present in a DNA mixture
Goodrich [8]	2009	Revealing information about the full identity of an encrypted genomic query sequence
Humbert *et al.* [9]	2013	Inferring close relatives' genomes using statistical inference
Sweeney *et al.* [10]	2013	Identify the individuals in the Personal Genome Project (PGP) by Name
Gymrek *et al.* [11]	2013	Identifying personal genomes from surnames by profiling short-tandem repeats on the Y-chromosome
Fredrikson *et al.* [12]	2014	Predicting genetic markers using machine learning models on differentially private data
Shringarpure and Bustamante [13]	2015	Identifying participants from a genomic database (with beacon services) with limited number of queries
Raisaro *et al.* [14]	2016	Modifying attack on beacon services with better adversarial knowledge
Harmanci and Gerstein [15]	2016	Linking phenotypes to genotypes from publicly known genotype–phenotype correlation

not the final answer. For example, the time needed by a governing body to review researchers' applications requesting the access of data sets is tedious, and it adversely affects achieving timely research outcomes. This delay often demotivates researchers to pursue specific studies. As we cannot foresee the future attacks on genomic data, these policies are either much generalized or can fall short for a new or advance attack strategy.

On the other hand, cryptography is fairly archaic and mature area, which can provide help in this domain. Using cryptographic approaches allows addressing various privacy issues of genomic data, as these strategies ensure the data security and privacy of an individual even on an untrusted environment. The rigorous definitions and guarantees of these concepts permit us to measure and mitigate the risk involved. Also, with seminal development in multiple crypto primitives in recent years, this should impact positively toward some of these security/privacy requirements. Thus, developing a secure system using various cryptographic techniques that guarantee both privacy and utility of genomic data is an important research problem.

In this article, we summarize the interoperability of these two scientific research fields: genomic data and privacy-preserving techniques. The different focus points of security and privacy aspects of genomic data will be the main concern, discussing the necessary backgrounds and followed by some of the recent works. Specifically, we will discuss the following:

- Current privacy problems around sharing and computation on genomic data in different settings are detailed.
- Major cryptographic approaches and recent advancements with potentials to solve specific genomic data privacy and security issues are discussed.
- Available tools regarding these crypto primitives are also detailed along with their used cases, differences and limitations, which will help practitioners for better understanding.
- Recent developments or scientific literature that adopts such privacy-preserving approaches and the corresponding problem space are discussed. We focus on the gaps in this domain along with the open problems to be addressed as future work.

This review focuses on privacy-preserving techniques that are applicable for addressing some of the problems of genomic data dissemination and computation. Previous surveys targeting such multidisciplinary research area [17–20] either presented different privacy attacks on genomic data or proposed solutions in lieu of these privacy issues. In this article, we take a different route, as we discuss various privacy-preserving solutions to mitigate the privacy concerns in sharing genomic data for research [e.g. genome-wide association studies(GWASs)]. The ethical, forensic or security–privacy concerns of DNA sequencing [17, 20] are kept out of discussions in this article, while we only focus on the computational security or privacy aspects of the retrieved genomic data. In other words, we discuss the methods required to securely compute or preserve the privacy of the data retrieved after the DNA sequencing.

In particular, we first overview different recently proposed cryptographic techniques and then discuss how these techniques can be used to ensure privacy-preserving genomic data sharing. This might be beneficial in understanding the current state of the art of these privacy-preserving techniques and resolve some misconceptions about their efficiency.

Some of the seminal developments in this ecology of privacy-preserving techniques and genomics are shown in Figure 1. From the earlier sequencing techniques in 1975 to the recent developments using Intel Software Guard Extensions (SGX) in 2017, the genomic data evolution and the cryptographic techniques are presented in a chronological fashion in Figure 2. We use the green color and the orange color to describe the contributions in genomic data and privacy-preserving techniques, respectively. It is noteworthy that all these events might not be equally significant, but each of these has some potential in this area of research.

Problem specifications

Exploring the cryptographic solutions available warrants discussions on the privacy or security problems specific to genomic data sharing/computation.

Different entities

In a genomics study, the collaborating parties can be categorized into four general entities: (a) end point users/researchers, (b) computation layer, (c) data owners, and (d) data storage and operations layer (Figure 2). Often times, these entities can be merged or generalized as detailed in the Supplementary Material. For example, data owners can have their own infrastructure to store large quantities of genomic data or their own data storage layer. Also, there are several proposals for introducing a fully trusted entity in this pool. Regardless of any alteration to the structure, these entities are often assigned with

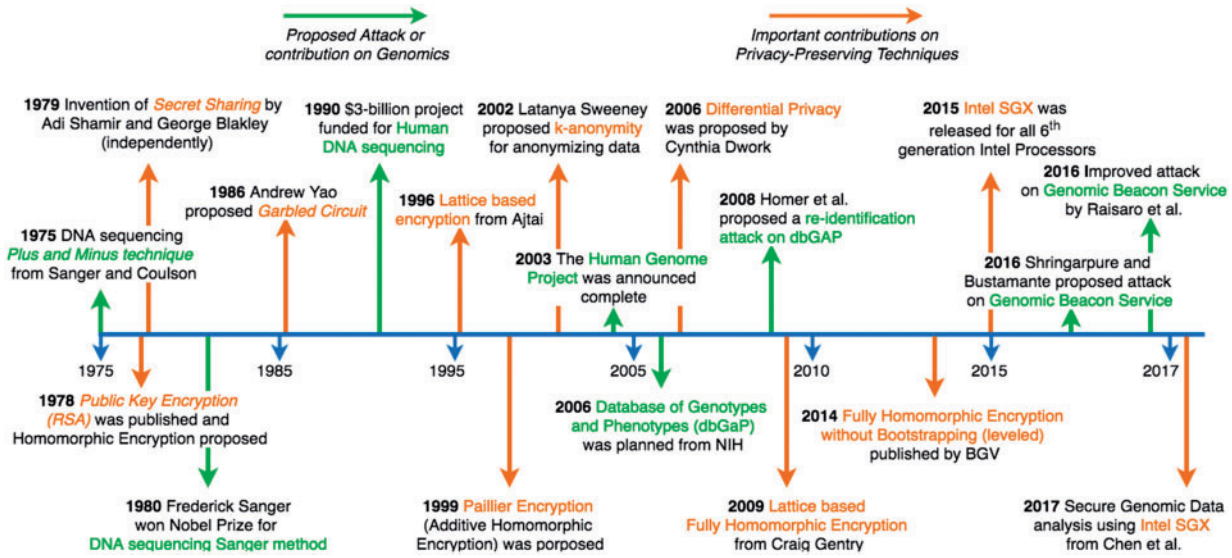

Figure 1. Timeline of the evolution of genomic data studies and seminal development of different privacy-preserving techniques.

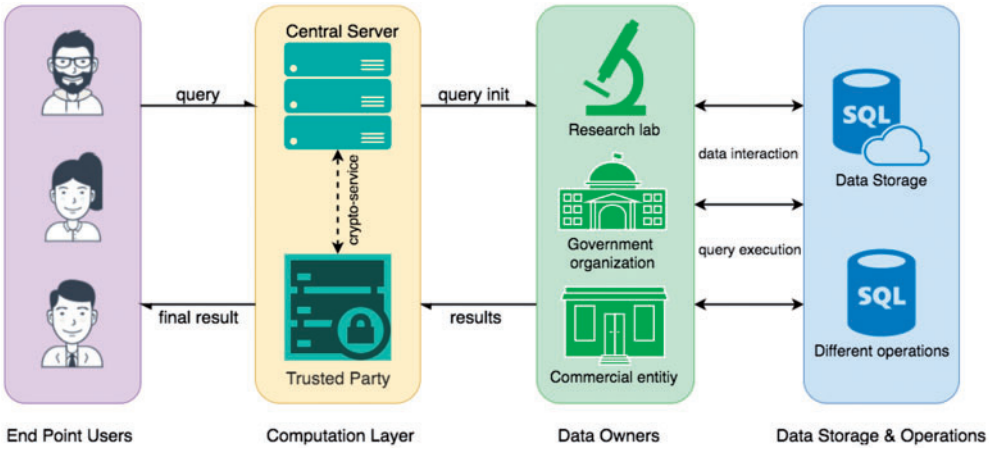

Figure 2. Different entities involved in a genomic data computation.

different trust models (i.e. malicious, semi-honest or fully trusted).

Problem categories

The privacy and security problems of genomic data can broadly be classified into three major groups:

Problem 1: Privacy-preserving sharing of genomic data
In the past decade, multiple reidentification attacks (Table 1) on genomic data have amplified the privacy and security aspects of such sensitive data, which accounts for almost all current practices (described in the Supplementary Document) not allowing public sharing of these data without any form of privacy guarantee (i.e. anonymization). However, these privacy guarantees often fall short for different reasons (i.e. different adversarial assumptions, better attacks or different threat models).

Example. A government or research organization has sequenced a population or a disease group believing that these data will reveal the correlation between the disease and the genomic data. The organization believes the data should not be public, as it contains the disease association of the participants. However, as the primary intention behind collecting the data was to share it to the scientific community, the necessity for a privacy-preserving sharing mechanism is emphasized. Solutions such as 'Homomorphic encryption' or 'Differential privacy' sections are applicable in this case.

Problem 2: Secure computation and storage of genomic data
Problem 1 only deals with the unknown risk involved with sharing the genomic data. Problem 2, however, denotes the leakage risks from the storage or computation of genomic data.

Example. Suppose a sensitive genomic data set is shared with researchers using a state-of-the-art privacy-preserving mechanism; there are still unknown risks involved as researchers need to store the data. This risk is further elevated, as researchers might opt for public cloud solutions for their computation. In this case, the usability of cloud service is considerable because of the budget constraints of any research. Even without the cloud usage, allowing a third party to compute on the data in plaintext involves unwanted risks, as data might leak from the secure enclosure of researchers as well. Such problems can be solved by using three techniques, which are

outlined in sections 'Homomorphic encryption', 'Garbled circuit' (GC) and 'Secure hardware'.

Problem 3: Query or output privacy

The last piece of the puzzle comes from the outputs of any analysis of genomic data. Even with all the security or privacy-preserving techniques around data sharing or computation, the query and its output will reveal the researcher's interest and some data characteristics, respectively. Though these problems are surfacing recently and less explored by the research community over the past years, attacks against the aggregated results of Genomic Beacon Service [13, 21] further elevate the necessity of such privacy.

Example. Some parties (i.e. commercial drug manufacturers) often have private queries that reveal their targeted consumer markets and strategic plans. The published results from these queries are also sensitive, as they can reveal the presence of an individual or a certain group in a study. One particular technique to solve such a privacy problem is detailed in 'Differential privacy' section and the Supplementary Document as well.

In this article, we consider the privacy/security issues raised after the generation of human genomic sequences only. There are other concerns related to the sequencing phase: different policies or ethical aspects of collecting genomic data presented in different surveys [17–20]. However, the aforementioned three problems will be sufficient, as we explain the application of the different cryptographic methods.

Current practices

There are two practices of genomic data sharing: (a) open or public access and (b) controlled access via different policies and access controls. We discuss these practices in the Supplementary Material along with current privacy policies in genomic data sharing, different threat models and the entities presented in Figure 2.

Privacy-preserving solutions

Here, we discuss the different privacy-preserving methods developed in context of genomic data using the four cryptographic techniques: (a) homomorphic encryption, (b) GC, (c) secure hardware and (d) differential privacy. The necessary background of these crypto primitives is detailed in the Supplementary Material, which will further explain the usage of the techniques.

Homomorphic encryption

Homomorphic encryption (HE) allows one party to compute predefined functions over encrypted data without decrypting it (Figure 3). HE can be classified to various form factors such as Fully, Leveled and SomeWhat HE (detailed in Supplementary Document) depending mostly on compactness, correctness or the functions they can compute [22]. Different HE schemes were applied to different secure computation and storage problems of genomic data. One of the first approaches was to use an additive HE (Paillier encryption) [23] on a semi-honest cloud server proposed by Kantarcioglu *et al.* [24]. The authors [24] presented a cryptographic approach to securely store the genomic sequences in a cloud server. They encoded the genomic sequences according to a binary representation and then encrypted the individual encoding by Paillier (additive homomorphic) encryption. There are also some other attempts made

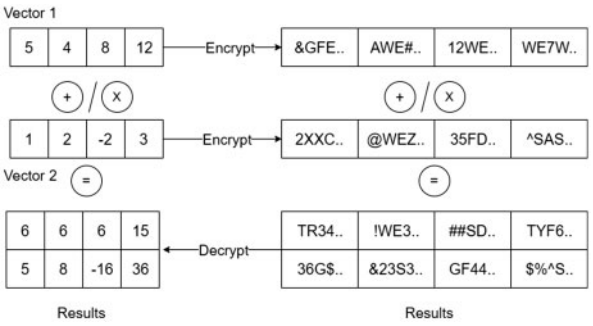

Figure 3. Example of homomorphic operations on encrypted values.

with the Paillier encryption in a federated [25] and centralized environment [26].

Another attempt with additive HE was the privacy-preserving genetic risk calculation [27]. Ayday *et al.* [27] proposed a method depending on the DGK (Damgård, Geisler and Krøigaard) cryptosystem [28], which is mostly used for secure computation of integers. It is also one of the earliest works that handle the problem of the privacy-preserving of federated/centralized genomic data storage and computation.

Some recent works proposed by Lauter *et al.* [29] show some secure versions of statistical algorithms used in genomic studies, e.g. Hardy–Weinberg equilibrium and linkage disequilibrium. This work uses the rigorous security definition of fully homomorphic encryption; it can allow any computation over any encrypted data.

After the availability of the implementations of these encryption schemes [30–32], Cheon *et al.* [33] proposed edit distance to be securely computed via lattice encryption. However, the present state of HE inhibits the efficiency of the scheme; it still takes 16.4 s to compute a small 8×8 block of dynamic programming.

There are other usages of SomeWhat HE in genomic data [34], which covers homomorphically computing logistic regression. An extension is Healer [35], which provides Secure Exact Logistic Regression in genomic data. Privacy-preserving GWASs with HE were also proposed by Lu *et al.* [36] in 2015, where the authors compared their packing technique with Lauter *et al.*'s work [29]. FORESEE [37] is also similar to this work and proposes secure chi-square statistics on genomic data. In a more recent work, Shimizu *et al.* [38] proposed the usage of Burrows–Wheeler transformation for finding target queries on a genomic data set. This work along with some others [39, 40] used additive HE for privacy-preserving computation on genomic data efficiently.

However, some results from a recent competition look promising in favor of HE. The task was to execute a simple yes/no query from VCF (Variant Call Format) files using HE schemes. The solutions were evaluated with different number of query parameters and varying the underlying encrypted VCF files containing 10–100 000 records. The results are ranked in Table 2.

Available implementation

There are several mainstream implementations available for HE: (a) SEAL (sealcrypto.codeplex.com), (b) HElib (github.com/shaih/HElib) and (c) NFLlib (github.com/quarkslab/NFLlib). In Table 3, we point some of the differences between these implementations. There are other implementations available such as Krypto based on Multivariate Quadratic FHE (https://github.com/kryptnostic/krypto), FHEW library from Ducas–Micciancio

Table 2. Results from the iDash 2016 competition task 3: Searching in encrypted genomic data set

Authors	Setup time	Encrypted size	Computation time
Çetin *et al.* [42]	36.69	188	59.58
Kim *et al.* [43]	2384	244	226.9
Ziegeldorf *et al.* [44]	1207	13 000	297.2
Sousa *et al.* [45]	6903.1	1468	288.9

Note: The results are for 100 000 records and a query with four variants (times are in seconds and encrypted data set size in MB) [41].

Table 3. Comparison of popular HE implementations

Feature	HElib	SEAL	FV-NFLlib	TFHE
Crypto scheme	BGV [46]	FV [47]	FV [47]	BGV [46]
Fully HE	✓	✗	✗	✓
Language	C++	C++	C++	C++
Library dependency	NTL and GMP	✗	NFLlib	Any FFT
Relinearization	✓	✓	✓	✗
Bootstrap	✓	✗	✗	✓
Fixed-point support	✗	✓	✗	✗
GPU enabled	✓ [48]	✗	✗	✗
Wrapper available	Python	C#	✗	✗

[49] and recent TFHE [32], which claims to provide the fastest bootstrapping.

Open directions

As HE is still in its early phase, it has not reached the level efficiency for large-scale generic genomic data computation as mentioned in sections 'Problem 1: Privacy-preserving sharing of genomic data' or 'Problem 2: Secure computation and storage of genomic data'. For example, a multiplication or bootstrapping operation takes some time making it unrealistic to use on complex functions such as training machine learning. In real-world scenarios, there is a demand for using different machine learning techniques on genomic data for various research objectives. However, as these data are much sensitive, running (training) the machine learning models on encrypted data will need much faster HE schemes. Also, there is a necessity of different packing techniques [50] for genomic data, which can reduce the execution time of the whole computation.

Garbled circuit

A GC is a constant round secure protocol, which allows any function to be computed between multiple parties, hiding both their inputs from each other [51]. The security guarantee of GC lies in equal participation of two parties communicating over the computed functions. Another significance of GC is the privacy of the inputs of both parties, as often times the query itself requires similar privacy deemed by the data. Therefore, GC is primarily used in the sequence similarity problems in which one party has a data set of genomic sequences and another party (researcher) has a sensitive query sequence. The researcher wants to find the similar sequences of that specific query based on any similarity metrics, i.e. hamming and Levenshtein distances.

One of the primary works in the domain of privacy-preserving genomic sequence similarity is proposed by Jha *et al.* [52] in 2008. The authors showed three different protocols that can replicate the original edit distance algorithm over a GC. However, it took around 40 s to compute the edit distance between two 25-long sequences because of the performance of the GC available that time.

Wang *et al.* [53] formally defined this problem as 'Similar Patients Query'. The authors addressed the problem of approximating the original edit distance in a realistic setting for a larger data set [54]. The method used a public reference genomic sequence for some precomputation on the genomic data set and then approximate the edit distance between the query string and the data set.

However, the selection of a public reference leaks some information about the underlying data distribution. Moreover, it affects the accuracy, as the computation is done according to a reference. There are other related studies [55, 56, 57] that address approximating or securely computing edit distance. In Table 4, we show some recent results on privacy-preserving edit distance approximations using GC and some other protocols.

Available implementation

As mentioned above, there are multiple efficient algorithms available for GCs. These implementations are highly extensible and offer a secure multiparty computation with inherent communication cost. Some of the notable tools regarding GC are ObliVM [60], FastGC [61], FlexSC [62], ABY [63] and TinyGC [64]. As the last available implementation of GC, TinyGC [64] contains all the benchmarks necessary and is the state of the art to the best of our knowledge.

Open directions

Problems such as query and computation privacy (sections 'Problem 3: Query or output privacy' or 'Problem 2: Secure computation and storage of genomic data') are primary motivations to apply GC. As the current state of the art in GC and oblivious Transfer [60] is a little more matured than FHE, they still need to be tested on large-scale genomic data computation. For example, all the recent secure edit distance computations (or approximations) take small sequence length into account, where realistic scenario dictates long genomic sequences. Also, the network overhead required (mentioned in section 'Differential privacy') for large-scale circuit computation might need further experiments.

However, there is a potential for another secure multiparty computation protocol, Secret Sharing Schemes. Though architecture-wise it is a bit different from GC, this technique has not been much explored into this area of research. Secret Sharing Schemes can hold some answers to the privacy issues in cross-institutional or federated environment architecture [65].

Secure hardware

Using secure hardware for a secure computation is considered a seminal contribution from Intel when they introduced SGX in their sixth and subsequent generation processors. It allows a user to separate their confidential data and code from the regular ones and allows him/her to do the secure computation in a secure enclave inside the processor. The secure portion of the data or the code is inaccessible from the rest of the execution ensured by the processor itself (Figure 4).

The first work using secure hardware on genomic data introduced by Canim *et al.* [66] in 2012 leveraged a trusted hardware (secure co-processor) inside an untrusted cloud to ensure

Table 4. Some recent results of privacy-preserving genomic data similarity methods using GC protocol

Authors	Year	Data ($n \times m$)	Time (s)	Principal method
Jha *et al.* [52]	2008	25×25	<40	Smith–Waterman
Wang *et al.* [55]	2009	400×400	28.5	Custom protocols
Wang *et al.* [54]	2015	2000×9000	2800	Private set difference with a reference sequence
Cheon *et al.* [33]	2015	8×8	16.4	HE
Shimzu *et al.* [38]	2016	2184 genomes	4–10	Burrows–Wheeler transform
Zhu and Huang [58]	2017	500×3500	209	GC
Aziz *et al.* [59]	2017	500×3500	22	Private set intersection
Asharov *et al.* [56]	2017	500×3500	11	Custom protocols with reference genome

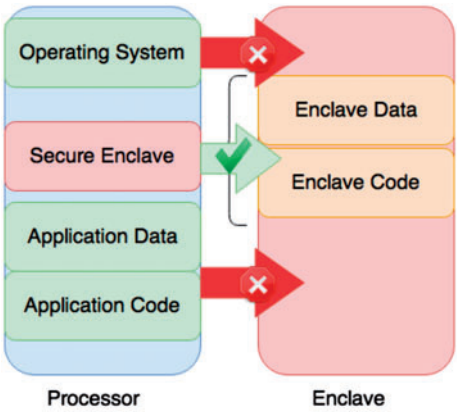

Figure 4. Protected execution environment in Intel SGX.

privacy. The authors proposed a technique to use the symmetric encryption to perform secure count queries on a genomic data set.

With the advent of Intel SGX, research has surfaced on privacy-preserving statistical analysis on genomic data. One recent attempt, proposed by Chen *et al.* [67] in 2016, introduced a solution based on Intel SGX, which enables efficient and privacy-preserving estimation of an individual admixture named PREMIX. PREMIX [67] can protect the confidentiality of genomic data and ancestry information of patients.

Another recent solution, PRINCESS [68], introduced an international collaboration framework (Federated) for privacy-preserving analysis of rare disease genetic data that is distributed around the world. PRINCESS [68] was evaluated in a study of family-based transmission disequilibrium tests to understand the genetic architecture of Kawasaki disease (KD).

HardiDX [69] is an SGX-based framework for searching over encrypted data efficiently, which can be incorporated for genomic data, as it overcomes the memory limitation of SGX by loading the data into the enclave in an on-demand fashion. Another work named LAST^{GT} [70] formulates the problem of managing large-scale data as a virtual memory management problem. Though LAST^{GT} is implemented on TrustVisor, the authors also described a possible implementation on Intel SGX.

Open directions

Secure hardware solutions such as Intel SGX or AMD memory encryption [71] are recent and still unknown to the research community. Though it could potentially solve problems mentioned in section 'Problem 2: Secure computation and storage of genomic data', further inspections are still required. As this approach could be an efficient alternate to HE, the security risks of using a third-party trusted component (H/W) or dependency on one singular key can be explored further.

Nonetheless, it is beyond the scope of the genomics research, and there are other pressing issues as SGX comes with limited low-level memory attached only to the processor. Some recent works [69, 70] mentioned before focus on managing large-scale data in Intel SGX applications to overcome the limited size of enclave page cache. Regardless, it might be interesting to look for any side channels or cache attacks depending on the genomic data size as well.

Differential privacy

Though differential privacy is not a cryptographic technique, it theoretically offers quantifiable bound of privacy on the disclosure of data or any query result. A majority of the research work revolving differential privacy and genomic data lies in the private GWAS. In 2013, Johnson and Shmatikov [72] proposed a differentially private approach to answer statistical queries of GWAS: (a) specific position in DNA is correlated with a particular disease, (b) relation between two positions in a DNA and (c) the number of locations affiliated to a specific disease.

This problem is later extended by Yue *et al.* [73], where the authors applied differential privacy to detect association among multiple positions using logistic regression. In extension to the problem above [72], Yu and Ji [74] also worked on differentially private GWAS for integrating Data for Analysis, Anonymization and SHaring (iDash) competition.

Another work from Tramèr *et al.* [75] proposed a differentially private mechanism for positive membership considering weaker adversarial model. Recently, Simmons and Berger [76] refined both of the aforementioned problems of finding significant positions on DNA by modeling it as an optimization problem. They further relaxed the optimization problem and solved it in constant time. Simmons *et al.* also proposed differential techniques for the EigenStrat and linear mixed model-based GWAS with population stratification [77] in late 2016. There are also some other attempts [78–80] in GWAS, data dissemination or sharing using differentially private mechanisms.

In a more recent work in 2017, a differentially private solution [81] was proposed for the privacy attack on Genomic Beacon Services [13, 21]. The method was initially presented in the iDASH 2016 competition [82]. It scrutinized the parameters of the original attack, analyzed it for different scenarios and proposed a simple differentially private 'Randomized Response' algorithm. Despite having a formal privacy guarantee, this solution came second best, as the winning solution [83] took a different direction toward the problem. Wan *et al.* [83] formulated the problem with an objective function consisting the data and potential queries from any adversary with different background knowledge. Further improvements were made by considering

the uniqueness of some data points (SNPs), which were more sensitive than others. Such game theoretic solutions are also interesting, as it often maximizes the privacy–utility of a given scenario compared with contemporary DP solutions.

Available implementation

One of the most popular tools available in differential privacy was proposed in 2009 by Mcsherry [84]. The toolkit was named Privacy Integrated Queries (PINQ) based on Microsoft's C# Language-Integrated Query (LINQ) feature. PINQ works as a layer above the database working on the SQL queries being performed on the data. For example, with any count queries performed on the data, PINQ will return the calibrated noisy or differentially private output.

Open directions

Differential privacy is one of the subverted concepts from the others, though it has the potential to solve some of the privacy issues regarding the data (sections 'Problem 1: Privacy-preserving sharing of genomic data' and 'Problem 3: Query or output privacy'). However, this method needs to be analyzed in different research problems because of the error or noise calibration issues and high accuracy requirement of genomic research. This technique offers promise of a rigorous guarantee over the privacy of the data and outputs.

For example, if any computation is executed on differentially private data, the output can be proven to follow the same differential privacy guarantees. This definition allows us to avoid the overhead from secure computation techniques (i.e. HE or SGX) for some certain scenarios, where we do not require secure computation. Nonetheless, as differential privacy is more explored on the context of privacy-preserving machine learning, this can have a benefit when learning different models on genomic data.

Comparison

In Table 5, we compare the aforementioned four techniques (HE, GC, SGX and DP). However, the techniques have much difference in terms of architecture, security guarantee, underlying threat models and usage scenarios. Hence, we do not compare these techniques only for a single problem; rather, we provide a generic ranking (1 = good; 4 = poor) for the execution time, accuracy, memory requirement and network communication. However, this ranking might not be true for every scenario, as it depends mostly on the problem setting. For example, in a two-party secure computation setting, GC might be more appealing than HE, where for an n-party problem, we might opt for HE or Secret Sharing solutions instead.

Sometimes the combination of some of these techniques offers better solutions. For example, if the solution requires output privacy and computation security as well, we can combine DP and any of the other techniques to achieve that. Primarily DP prevents inference attacks, while the rest of the techniques ensure confidentiality. Thus, these techniques complement each other as shown by [80, 21].

Table 5. Comparison between the different privacy-preserving techniques

Criteria	HE	GC	SGX	DP
Execution time	4	3	2	1
Memory usage	4	3	2	1
Accuracy	3	2	1	4
Network communication	3	4	2	1
Secure computation	✓	✓	✓	✗
Output privacy	✗	✗	✗	✓

Conclusion

Our genomic data are what we inherited from our ancestors, share them with our siblings and will pass onto our offspring. As we are heading into less expensive and more accurate sequencing methods and more sophisticated attacks surfacing, the motivation for genomic data security and privacy is amplified. Therefore, as we discussed in this article with the different research problems concerning genomic data and the advancement of privacy-preserving techniques, this ecology of security–privacy and genomic data needs further exploration, resulting in more efficient, secure computation and storage architectures for such data.

Key Points

- Different privacy and security concerns relating to genomic data with respect to recent attack scenarios and current practices.
- State of the art methods of privacy-preserving techniques and their current status.
- Some of the recent developments in using such techniques for genomic data privacy and their results.
- The future directions of using these privacy-preserving techniques.

Supplementary Data

Supplementary data are available online at http://bib.oxfordjournals.org/.

Funding

The Patient-Centered Outcomes Research Institute (PCORI) under contract ME-1310-07058, the National Institute of Health (NIH) under award number R01HG008802, R01GM118609, R01GM114612, R01GM118574, R01GM124111 and U01EB023685 (to X.J. in part). NIH under award number R00HG008175, U01EB023685 and R01GM124111 (to S.W. in part). NSERC Discovery Grants (RGPIN-2015-04147) and University of Manitoba Startup Grant (to N.M. in part).

References

1. DNA Sequencing Costs. http://www.genome.gov/sequencingcosts/ (11 January 2016, date last accessed).
2. Barnes B, Dupré J. *Genomes and What to Make of Them*. University of Chicago Press, Chicago, 2009.
3. Trinidad SB, Fullerton SM, Bares JM, *et al*. Genomic research and wide data sharing: views of prospective participants. *Genet Med* 2010;**12**(8):486–95.
4. Malin B, Sweeney L. Determining the identifiability of DNA database entries. In: *Proceedings of the AMIA Symposium. American Medical Informatics Association*, 2000, Maryland, United States, 537.
5. Gottlieb S. Us employer agrees to stop genetic testing. *BMJ* 2001;**322**(7284):449.
6. Lin Z, Owen AB, Altman RB. Genomic research and human subject privacy. *Science* 2004;**305**(5681):183.
7. Homer N, Szelinger S, Redman M, *et al*. Resolving individuals contributing trace amounts of dna to highly complex

mixtures using high-density SNP genotyping microarrays. *PLoS Genet* 2008;**4**(8):e1000167.
8. Goodrich MT. The mastermind attack on genomic data. In: *SP '09 Proceedings of the 2009 30th IEEE Symposium on Security and Privacy*. IEEE, Maryland, US, 2009, 204–18.
9. Humbert M, Ayday E, Hubaux JP, *et al*. Addressing the concerns of the lacks family: quantification of kin genomic privacy. In: *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*. ACM, 2013, 1141–52.
10. Sweeney L, Abu A, Winn J. Identifying participants in the personal genome project by name. 2013. http://arxiv.org/abs/1304.7605 (10 April 2017, date last accessed).
11. Gymrek M, McGuire AL, Golan D, *et al*. Identifying personal genomes by surname inference. *Science* 2013;**339**(6117):321–4.
12. Fredrikson M, Lantz E, Jha S, *et al*. Privacy in pharmacogenetics: an end-to-end case study of personalized warfarin dosing. In: *23rd USENIX Security Symposium (USENIX Security 14)*. 2014, 17–32.
13. Shringarpure SS, Bustamante CD. Privacy risks from genomic data-sharing beacons. *Am J Hum Genet* 2015;**97**(5):631–46.
14. Raisaro JL, Tramèr F, Ji Z, *et al*. Addressing beacon re-identification attacks: quantification and mitigation of privacy risks. *J Am Med Inform Assoc* 2017;**24**(4):799–805.
15. Harmanci A, Gerstein M. Quantification of private information leakage from phenotype-genotype data: linking attacks. *Nat Methods* 2016;**13**(3):251–6.
16. Huang Z, Ayday E, Lin H, *et al*. A privacy-preserving solution for compressed storage and selective retrieval of genomic data. *Genome Res* 2016;**26**(12):1687–96.
17. Naveed M, Ayday E, Clayton EW, *et al*. Privacy in the genomic era. *ACM Comput Surv* 2015;**48**(1):6.
18. Erlich Y, Narayanan A. Routes for breaching and protecting genetic privacy. *Nat Rev Genet* 2014;**15**(6):409–21.
19. Wang S, Jiang X, Singh S, *et al*. Genome privacy: challenges, technical approaches to mitigate risk, and ethical considerations in the united states. *Ann N Y Acad Sci* 2017;**1387**(1):73–83.
20. Akgün M, Bayrak AO, Ozer B, *et al*. Privacy preserving processing of genomic data: a survey. *J Biomed Inform* 2015;**56**:103–11.
21. Raisaro JL, Choi G, Pradervand S, *et al*. Protecting privacy and security of genomic data in i2b2. *Technical report*. Institute of Electrical and Electronics Engineers, Lausanne, Switzerland, 2017.
22. Armknecht F, Boyd C, Carr C, *et al*. A guide to fully homomorphic encryption. *IACR Cryptology ePrint Archive, Report 2015/1192*, 2015. http://eprint.iacr.org/2015/1192 (10 April 2017, date last accessed).
23. Paillier P. Public-keycryptosystems based on composite degree residuosity classes. In: *International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT)*. 1999, 223–38.
24. Kantarcioglu M, Jiang W, Liu Y, *et al*. A cryptographic approach to securely share and query genomic sequences. *IEEE Trans Inf Technol Biomed* 2008;**12**(5):606–17.
25. Aziz MMA, Hasan MZ, Mohammed N, *et al*. Secure and efficient multiparty computation on genomic data. In: *Proceedings of the 20th International Database Engineering & Applications Symposium*. ACM, 2016, 278–83.
26. Ghasemi R, Aziz MMA, Mohammed N, *et al*. Private and efficient query processing on outsourced genomic databases. *IEEE J Biomed Health Inform* 2017;**21**(5):1466–72.
27. Ayday E, Raisaro JL, McLaren PJ, *et al*. Privacy-preserving computation of disease risk by using genomic, clinical, and environmental data. In: *Presented as Part of the 2013 USENIX Workshop on Health Information Technologies*. USENIX, Washington DC, USA, 2013.
28. Damgård I, Geisler M, Krøigaard M. Efficient and secure comparison for on-line auctions. In: *Australasian Conference on Information Security and Privacy*. Springer, Auckland, New Zealand, 2007, 416–30.
29. Lauter K, López-Alt A, Naehrig M. Private computation on encrypted genomic data. In: *Progress in Cryptology-LATINCRYPT 2014*. Springer, Florianópolis, Brazil, 2014, 3–27.
30. Gentry C, Halevi S. Implementing Gentry's fully-homomorphic encryption scheme. In: *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, Paris, France, 2011, 129–48.
31. Dijk MV, Gentry C, Halevi S, *et al*. Fully homomorphic encryption over the integers. In: *Advances in Cryptology–EUROCRYPT 2010*. Springer, French Riviera, France, 2010, 24–43.
32. Chillotti I, Gama N, Georgieva M, *et al*. Faster fully homomorphic encryption: bootstrapping in less than 0.1 seconds. In: *Advances in Cryptology–ASIACRYPT 2016: 22nd International Conference on the Theory and Application of Cryptology and Information Security*. Springer, Hanoi, Vietnam, 2016, 3–33.
33. Cheon JH, Kim M, Lauter K. Homomorphic computation of edit distance. In: *Financial Cryptography and Data Security*. Springer, San Juan, Puerto Rico, 2015, 194–212.
34. Bos JW, Lauter K, Naehrig M. Private predictive analysis on encrypted medical data. *J Biomed Inform* 2014;**50**:234–43.
35. Wang S, Zhang Y, Dai W, *et al*. HEALER: homomorphic computation of ExAct logistic rEgRession for secure rare disease variants analysis in GWAS. *Bioinformatics* 2016;**32**(2):211–18.
36. Lu W, Yamada Y, Sakuma J. Efficient secure outsourcing of genome-wide association studies. In: *Security and Privacy Workshops (SPW)*. IEEE, San Jose, USA, 2015, 3–6.
37. Zhang Y, Dai W, Jiang X, *et al*. FORESEE: fully outsourced secuRe gEnome Study basEd on homomorphic Encryption. *BMC Med Inform Decis Mak* 2015;**15**:S5.
38. Shimizu K, Nuida K, Rätsch G. Efficient privacy-preserving string search and an application in genomics. *Bioinformatics* 2016;**32**(11):1652–61.
39. Blanton M, Bayatbabolghani F. Efficient server-aided secure two-party function evaluation with applications to genomic computation. *Proc Priv Enhanc Technol* 2016;**2016**(4):144–64.
40. Baldi P, Baronio R, Cristofaro ED, *et al*. Countering gattaca: efficient and secure testing of fully-sequenced human genomes. In: *Proceedings of the 18th ACM Conference on Computer and Communications Security*. ACM, Chicago, USA, 2011, 691–702.
41. IDASH-privacy and security workshop on genomic data. http://www.humangenomeprivacy.org/2016/slides/idash16 Results_final.pdf, 2017 (10 April 2017, date last accessed).
42. Çetin GS, Chen H, Laine K, *et al*. Private queries on encrypted genomic data. *BMC Med Genomics* 2017;**10**(2):45.
43. Kim M, Song Y, Cheon JH. Secure searching of biomarkers through hybrid homomorphic encryption scheme. *BMC Med Genomics* 2017;**10**(2):42.
44. Ziegeldorf JH, Pennekamp J, Hellmanns D, *et al*. Bloom: bloom filter based oblivious outsourced matchings. *BMC Med Genomics* 2017;**10**(2):44.
45. Sousa JS, Lefebvre C, Huang Z, *et al*. Efficient and secure outsourcing of genomic data storage. *BMC Med Genomics* 2017;**10**(2):46.
46. Brakerski Z, Gentry C, Vaikuntanathan V. (Leveled) fully homomorphic encryption without bootstrapping. In: *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*. ACM, Cambridge, USA, 2012, 309–25.
47. Fan J, Vercauteren F. Somewhat practical fully homomorphic encryption. IACR Cryptology ePrint Archive, *Report 2012/144*, 2012. http://eprint.iacr.org/2012/144 (10 April 2017, date last accessed).

48. Dai W, Sunar B. cuHE: a homomorphic encryption accelerator library. In: *International Conference on Cryptography and Information Security in the Balkans*. Springer, Koper, Slovenia, 2015, 169–86.
49. Ducas L, Micciancio D. FHEW: bootstrapping homomorphic encryption in less than a second. In: *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, Paris, France, 2015, 617–40.
50. Naehrig M, Lauter K, Vaikuntanathan V. Can homomorphic encryption be practical? In: *Proceedings of the 3rd ACM Workshop on Cloud Computing Security Workshop*. ACM, Chicago, USA, 2011, 113–24.
51. Yao AC. Protocols for secure computations. In: *SFCS '82 Proceedings of the 23rd Annual Symposium on Foundations of Computer Science*, Vol. 82. Chicago, USA: IEEE, 1982, 160–4.
52. Jha S, Kruger L, Shmatikov V. Towards practical privacy for genomic computation. In: *2008 IEEE Symposium on Security and Privacy (sp 2008)*. IEEE, Oakland, USA, 2008, 216–30.
53. Wang XS, Huang Y, Zhao Y, *et al*. Efficient genome-wide, privacy-preserving similar patient query based on private edit distance. In: *Proceedings of the ACM Conference on Computer and Communications Security*. ACM, Denver, USA, 2015, 492–503.
54. Wang XS, Huang Y, Zhao Y, *et al*. Efficient genome-wide, privacy-preserving similar patient query based on private edit distance. In: *Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security, CCS '15*. New York, NY: ACM, 2015, 492–503.
55. Wang R, Wang XFeng, Li Z, *et al*. Privacy-preserving genomic computation through program specialization. In: *Proceedings of the 16th ACM Conference on Computer and Communications Security*. ACM, Chicago, USA, 2009, 338–47.
56. Gilad A, Shai H, Yehuda L, *et al*. Privacy-preserving search of similar patients in genomic data. *Cryptology ePrint Archive, Report 2017/144*. 2017. http://eprint.iacr.org/2017/144 (10 April 2017, date last accessed).
57. Hasan MZ, Mahdi MSR, Mohammed N. Secure count query on encrypted genomic data. CoRR, abs/1703.01534, 2017.
58. Zhu R, Huang Y. Efficient privacy-preserving general edit distance and beyond. *Cryptology ePrint Archive, Report 2017/683*, 2017. http://eprint.iacr.org/2017/683 10 April 2017, date last accessed).
59. Aziz MMA, Alhadidi D, Mohammed N. Secure approximation of edit distance on genomic data. *BMC Med Genomics* 2017; **10**(2):41.
60. Liu C, Wang XS, Nayak K, *et al*. Oblivm: a programming framework for secure computation. In: *Proceedings of the 2015 IEEE Symposium on Security and Privacy*, pp. 359–76. IEEE Computer Society, Washington, DC, USA.
61. Huang Y, Evans D, Katz J, *et al*. Faster secure two-party computation using garbled circuits. In: *Proceedings of the 20th USENIX Conference on Security*, San Francisco, CA, 2011. Vol. 201, pp. 35–35. USENIX Association, Berkeley, CA, USA.
62. Wang X, Chan H, Shi E. Circuit oram: on tightness of the Goldreich-Ostrovsky lower bound. In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, Denver, USA, 2015, 850–861.
63. Demmler D, Schneider T, Zohner M. Aby-a framework for efficient mixed-protocol secure two-party computation. In: *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, Publisher Usenix, 2015.
64. Songhori EM, Hussain SU, Sadeghi AR, *et al*. Tinygarble: highly compressed and scalable sequential garbled circuits. In: *2015 IEEE Symposium on Security and Privacy*, 2015, 411–28.
65. Kamm L, Bogdanov D, Laur S, *et al*. A new way to protect privacy in large-scale genome-wide association studies. *Bioinformatics* 2013;**29**(7):886–93.
66. Canim M, Kantarcioglu M, Malin B. Secure management of biomedical data with cryptographic hardware. *IEEE Trans Inf Technol Biomed* 2012;**16**(1):166–75.
67. Chen F, Dow M, Ding S, *et al*. PREMIX: privacy-preserving EstiMation of individual admixture. In: *AMIA Annual Symposium Proceedings, Vol. 2016. American Medical Informatics Association*. Cambridge, USA, 2016, 1747.
68. Chen F, Wang S, Jiang X, *et al*. PRINCESS: privacy-protecting rare disease International Network Collaboration via Encryption through software guard extensionS. *Bioinformatics* 2017;**33**:871–8.
69. Kerschbaum F, Sadeghi AR. HardIDX: practical and secure index with SGX. In: *Data and Applications Security and Privacy XXXI: 31st Annual IFIP WG 11.3 Conference, DBSec 2017, Philadelphia, PA, USA, July 19-21, 2017*. Vol. 10359. Springer, Philadelphia, USA, 2017, 386.
70. Vavala B, Neves N, Steenkiste P. Secure tera-scale data crunching with a small TCB. In: *47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, IEEE, Denver, CO, USA, 2017.
71. Kaplan D, Powell J, Woller T. Amd memory encryption. *White paper*. 2016.
72. Johnson A, Shmatikov V. Privacy-preserving data exploration in genome-wide association studies. In: *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, Chicago,USA, 2013, 1079–87.
73. Yu F, Rybar M, Uhler C, *et al*. Differentially-private logistic regression for detecting multiple-SNP association in GWAS databases. In: *International Conference on Privacy in Statistical Databases*. Springer, 2014, 170–84.
74. Yu F, Fienberg SE, Slavković AB, *et al*. Scalable privacy-preserving data sharing methodology for genome-wide association studies. *J Biomed Inform* 2014;**50**:133–41.
75. Tramèr F, Huang Z, Hubaux JP, *et al*. Differential privacy with bounded priors: reconciling utility and privacy in genome-wide association studies. In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, Denver, USA, 2015, 1286–97.
76. Simmons S, Berger B. Realizing privacy preserving genome-wide association studies. *Bioinformatics* 2016;**32**(9):1293–300.
77. Simmons S, Sahinalp C, Berger B. Enabling privacy-preserving GWASs in heterogeneous human populations. *Cell Syst* 2016;**3**(1):54–61.
78. Uhlerop C, Slavković A, Fienberg SE. Privacy-preserving data sharing for genome-wide association studies. *J Priv Confid* 2013;**5**(1):137.
79. Wang S, Mohammed N, Chen R. Differentially private genome data dissemination through top-down specialization. *BMC Med Inform Decis Mak* 2014;**14**(Suppl 1):S2.
80. Froelicher D, Egger P, Sousa JS, *et al*. Unlynx: a decentralized system for privacy-conscious data sharing. In: *Proceedings on Privacy Enhancing Technologies*, Vol. 4. 2017, 232–50.
81. Aziz MMA, Ghasemi R, Waliullah M, *et al*. Aftermath of bustamante attack on genomic beacon service. *BMC Med Genomics* 2017;**10**(2):43.
82. IDASH privacy & security workshop 2016. http://www.humangenomeprivacy.org/2016/competition-tasks.html (12 August 2004, date last accessed).
83. Wan Z, Vorobeychik Y, Kantarcioglu M, *et al*. Controlling the signal: practical privacy protection of genomic data sharing through beacon services. *BMC Med Genomics* 2017;**10**(2):39.
84. McSherry FD. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In: *Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data*. ACM, Providence, USA, 2009, 19–30.