Enabling Privacy-preserving Sharing of Genomic Data for GWASs in Decentralized Networks

Yanjun Zhang
The University of Queensland
St Lucia, Queensland, Australia
yanjun.zhang@uq.edu.au

Xin Zhao
The University of Queensland
St Lucia, Queensland, Australia
xin.zhao@uq.edu.au

Xue Li
The University of Queensland
St Lucia, Queensland, Australia
xueli@itee.uq.edu.au

Mingyang Zhong
The University of Queensland
St Lucia, Queensland, Australia
m.zhong1@uq.edu.au

Caitlin Curtis
The University of Queensland
St Lucia, Queensland, Australia
c.curtis@uq.edu.au

Chen Chen
The University of Queensland
St Lucia, Queensland, Australia
chen.chen@uq.edu.au

ABSTRACT

The human genome can reveal sensitive information and is potentially re-identifiable, which raises privacy and security concerns about sharing such data on wide scales. In this work, we propose a preventive approach for privacy-preserving sharing of genomic data in decentralized networks for Genome-wide association studies (GWASs), which have been widely used in discovering the association between genotypes and phenotypes. The key components of this work are: a decentralized secure network, with a privacy-preserving sharing protocol, and a gene fragmentation framework that is trainable in an end-to-end manner. Our experiments on real datasets show the effectiveness of our privacy-preserving approaches as well as significant improvements in efficiency when compared with recent, related algorithms.

KEYWORDS

genomic data, re-identification, privacy-preserving sharing, decentralized network, GWAS

ACM Reference Format:
Yanjun Zhang, Xin Zhao, Xue Li, Mingyang Zhong, Caitlin Curtis, and Chen Chen. 2019. Enabling Privacy-preserving Sharing of Genomic Data for GWASs in Decentralized Networks. In *The Twelfth ACM International Conference on Web Search and Data Mining (WSDM '19), February 11–15, 2019, Melbourne, VIC, Australia.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3289600.3290983

1 INTRODUCTION

The 21st Century has so far witnessed an incredible genetic data explosion[18, 26], with the development of faster, and more efficient next generation sequencing technologies. As a result of this massive data availability, Genome-Wide Association Studies (GWAS), which is an experimental design used to detect associations between genetic variants and traits in samples from populations, are gaining popularity as they answer critical questions like the relative role of genes and the environment in disease risk, assist in risk prediction (enabling preventative and personalized medicine), and investigate natural selection and population differences. [33]

One key thing for GWASs is the sample size, they need lots of people (at least thousands of people) to share their data to confirm the differences with statistical confidence. However, genetic information has become among the most sensitive information about an individual - their personal traits, health problems, predispositions to diseases, life expectancy, familial relationships, etc. can all be potentially contained in it. Therefore, people have fundamental interests in having control over their data. Though sharing sequencing data sets without identities (remove the individual's name) has become a common practice in GWASs, unfortunately, advances in genomics have made re-identification an increasing concern [9], thereby undermining simple anonymization as an approach. Already, studies [16, 24, 29] have pointed to techniques for discovering the identities (Surname, 3D face, etc.) of people from a seemingly anonymous database, relying on publicly accessible Internet resources. It suggests that, with further advances, access to full genomes will lead to re-identification of individuals, and unprotected disclosure of such information will put individual's privacy at risk. As a result, more and more people are reluctant, or will be reluctant to share their data as they become better informed of data privacy issues. Thus, there is an urgent need to develop technologies for privacy-preserving sharing genomic data on a large scale to satisfy demands for GWASs.

This problem is non-trivial and challenging due to the following factors:

- **Re-distribution problem:** By its nature, data is very easy to copy and spread. For example, if we share a private video, we have technologies that only allows authorized persons or machines to open it, but what if someone record it by a camera and make re-distributions? Though we have technologies like watermark to detect the infringement of copyright, however, detection–as an after-the-event approach, is not enough for sensitive data. Once genomic data has been redistributed without authorization, there is no way to ensure comprehensive retrieval of all of the unauthorized copies. Therefore, methods for preventing unauthorized redistribution from the outset are required.

WSDM '19, February 11–15, 2019, Melbourne, VIC, Australia
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-5940-5/19/02…$15.00
https://doi.org/10.1145/3289600.3290983

- **The problem of trade-off between privacy and utility:** As already noted, on the other side of the demand by patients and consumers, is the demand of researchers and companies wanting access to and analyze the genomic data. Therefore, we need methods to provide high level of privacy, including solving re-identification problem, and at the same time, won't affect the use of the data.
- **Scalability problem:** Genomic data is relatively large. A human genome, right off the genome sequencer, could be up to 200GB, and even if stored in the most efficient format - a variant file, that only keeps the genomic difference among individuals, is usually around 100MB to 200MB. Even at this size there are still very significant communication and economic costs in terms of network bandwidth consumptions once the data of hundreds of millions (and potentially billions) of people are included.

In this paper, we propose methods to tackle these challenges: we introduce a decentralized secure network, with a Privacy-Preserving Sharing (PPS) protocol, and a gene fragmentation framework that is trainable in an end-to-end manner.

First, we introduce a decentralized secure network, with a PPS protocol to solve the re-distribution problem. The main idea is we split the original gene sequencing data into fragments, and these fragments are stored/analyzed by different storage/analysis service providers in a decentralized network. When applying GWAS analysis, every analysis service provider separately processes a small piece of genomic data, by parallel in-memory computing, and reports its single point result. The complete analysis result then will be achieved by a report node. Therefore, there is few possibility for re-distribution because none of the nodes have access to the complete original data of any individual.

In addition, we propose a gene sequence fragmentation framework to minimize re-identification risk in each analysis node, at the same time, won't affect the use of the data in GWASs analysis.

Also, the fragmentation of the data not only safeguards the vital interests of the individuals who have contributed their data, but it simultaneously contributes to solving the scalability problems by making data packets smaller.

Figure 1[1] shows the full life cycle of the privacy-preserving sharing process: Through the fragmentation algorithm, gene sequence pieces are encrypted and distributed throughout the decentralized secure network, and by the PPS protocol, GWAS analysis can be achieved.

Our contributions can be summarized as follows:

- We propose a decentralized secure network, with a Privacy-Preserving Sharing (PPS) protocol to enable sharing genomic data on a large scale. It is a preventive method to solve the data re-distribution and re-identification problem and ethically satisfies people's fundamental interests in their data.
- To the best of our knowledge, we are the first to propose a gene fragmentation framework that is trainable in an end-to-end manner for privacy-preserving sharing of genomic data. Our method solves the problem of trade-off between privacy and utility, and can be used to facilitate the use of it for research and commercialization.
- We solve the scalability problem by the privacy-preserving sharing protocol, providing not only decentralized storage, but also decentralized analysis.
- We run experiments on real datasets showing the efficiency and effectiveness of our algorithms compared with state-of-the-art algorithm.

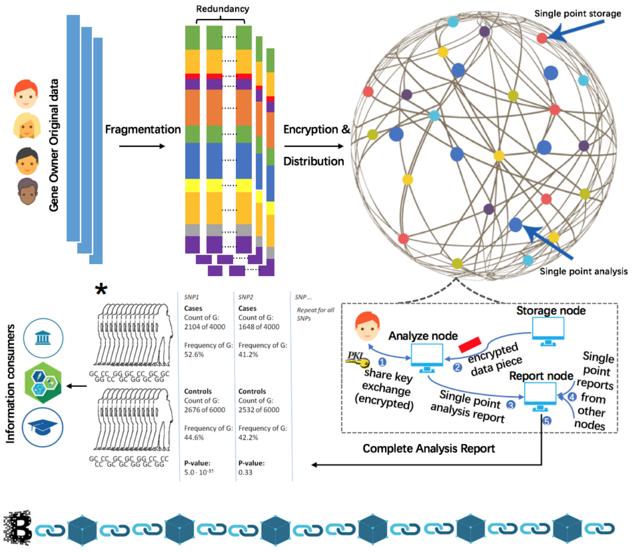


Figure 1: Life cycle in Decentralized Ownership Ecosystem

The rest of this paper is organized as follows. Section 2 briefly reviews privacy-preserving sharing techniques. In Section 3 and 4 we present our methods. In Section 5 we run experiments on real datasets, and some concluding remarks are given in Section 6.

2 RELATED WORK

There are existing privacy-preserving sharing techniques to deal with the privacy problem, which are divided into the following categories:

Secure multiparty computation (SMC). SMC allows two or more entities, each of which has some private data, to execute a computation on these private inputs without revealing the input to each other or disclosing it to a third party [35]. It enables the outsourcing of computationally intensive steps in the analysis without revealing any private information to others [1, 6, 21]. Several works have been proposed for multiple data owners (e.g., two or more medical institutions) to analyze genomic data jointly without disclosing their own data. For example, one study tested a secret sharing scheme based on distributed storage to generate GWAS summary statistics [23]. In addition, a privacy-preserving statistical analysis environment called 'Sharemind' was proposed to support a complete data analysis process where data are collected from various sources, and statistically analyzed between independent biobanks [3]. Another protocol was proposed to provide genomic diagnoses while preserving participant privacy [20]. However, most

[1] *: The GWASs report image credit: A GWAS of 14,000 cases of seven common diseases and 3,000 shared controls [8].

SMC methods only considered two-party scenarios. Extending a solution to allow for more than two parties may generate considerable scalability issues. Another threat for SMC is that computing parties collude to reveal private inputs.

Homomorphic Encryption. It allows computation on ciphertexts, generating an encrypted result which, when decrypted, matches the result of operations performed on the plaintext. In a recent study, a homomorphic exact logistic regression mode was introduced to facilitate secure rare variants analysis in GWASs [34]. In addition, a tool called 'SAFETY' was proposed recently, which can securely perform GWAS on federated genomic datasets using homomorphic encryption and secure hardware component of Intel Software Guard Extensions (Intel SGX) 2 to ensure high efficiency and privacy at the same time [27]. Another cryptographic protocol proposed a privacy-preserving solution for paternity tests, personalized medicine, and genetic compatibility tests [2, 5]. Then, in their follow-up work, a smart-phone-based implementation was presented for one version of this algorithm [11]. Existing homomorphic encryption techniques can be categorized as fully homomorphic cryptosystems and Partially Homomorphic Encryption. Fully homomorphic support arbitrary computation on ciphertexts but less efficient which result in a solution that is impractical [13, 32]. Partially Homomorphic Encryption is specified by a limited number of accumulated operations [4, 14].

Differential privacy. Differential Privacy [12, 19, 22, 28, 36] is an emerging methodology for minimizing the chances of identifying records of statistical databases while maximizing the accuracy of queries by adding 'noise'. Several studies have explored the differential private release of common summary statistics of GWAS data (such as the allele frequencies of cases and controls, χ^2-statistic and P values [31, 36]) or shifting the original locations of variants[22]. Recently, a study also introduced a computational framework for performing GWASs that adapts principles of differential privacy, and produced privacy-preserving GWAS results based on EIGENSTRAT and linear mixed model (LMM)-based statistics, both of which correct for population stratification [28]. However, there are some limitations in the approach: they are less accurate on small databases, and privacy cannot be guaranteed in databases with large levels of case ascertainment (that is, when the percentage of individuals with the disease in the study is larger than the percentage in the background population).

In conclusion, current methods for protecting human genomic data mainly focus on the collaborative studies between multi-biobank institutions. For example, when we apply techniques like differential privacy, we have to trust a party like 'data center' that holds the data. However, the public concerns are not only about the interests of the biobank institutions - it is about individual privacy and control. Also, efficiency is another issue. Cryptographic solutions, such as Homomorphic Encryption and SMC enable computing exact answers of the protected data sets, but unfortunately, they are uneconomical in practice because of significant computational and communicational overhead.

3 PPS PROTOCOL

3.1 Secure sharing in Decentralized network

First, we introduce a decentralized network. We aim to create a secure network for use in GWASs, which is resistant to malicious hacking or other unauthorized uses, controllable and trackable for individual. The key point of the network is: it is individuals that take control over their data instead of any central institutions. Therefore, we use a decentralized network instead of centralized network, and its autonomous nature allows every individual to be an independent peer in the network, while still being capable of interacting with the other stakeholders. The first thing we should consider is to guarantee the authenticity and nonrepudiation to all activities (transactions) in the decentralized network. 'Blockchain' technology is no doubt a good technical means to achieve the aim. In addition, it also enables continuing functioning in the event of component failures with no loss of data or integrity. It is a system of openness, and global participation designed to bring benefits to all stakeholders. (for decentralized networks, the best and most comprehensive references can be found from[25, 30]).

We are not the first to use Blockchain to share genomic data. There are existing systems, like Encrypgen[2], Luna DNA[3], Zenome[4], providing decentralized genomic data sharing platform. However, they directly transfer encrypted data from givers to information consumers like pharmaceuticals companies, hospitals, governments, researchers, etc. , and then the information consumers decrypt the data and apply analysis. As a result, information consumers can get access to whole original data, which carries high risks of redistribution and re-identification.

Therefore, we introduce the PPS protocol to achieve the security on data level. In the PPS protocol, we firstly introduce third parties - Service Providers into the network. Service Providers are individuals or companies providing infrastructures and services, including storage and computing resources. The role of service providers consists of Storage nodes, Analysis/Validator nodes, and Report nodes.

Figure 2 illustrates the PPS protocol during storage and analysis process. During the storage process (S-1 to S-2), we split the original data into fragments (S-1), and these fragments are encrypted and distributed to different 'Storage nodes' with redundancy (S-2). During the analysis process (A-1 to A-8), when an analysis agreement is signed between a Information Consumer and a data owner (A-1 and A-2), each of a plurality of network connected 'Analysis nodes' is assigned an compiled analysis script which is uploaded by the information consumer (A-3 and A-4), and separately retrieves a small piece of encrypted genomic data from 'Storage nodes' and the corresponding encrypted share key from data owner(A-5 and A-6), then decrypts and processes the data by parallel in-memory computing, and reports its own single point result to a 'Report node' (A-7). Then the 'Report node' aggregates the results and achieves a complete analysis report, then encrypts and sends the report to Information consumers (A-8).

All transactions (T-1 to T-6) during the process will be recorded to blockchain. T-1: Transactions between a data owner and the

[2] Engrypgen: https://encrypgen.com

[3] Luna DNA: https://www.lunadna.com

[4] Zenome: https://zenome.io

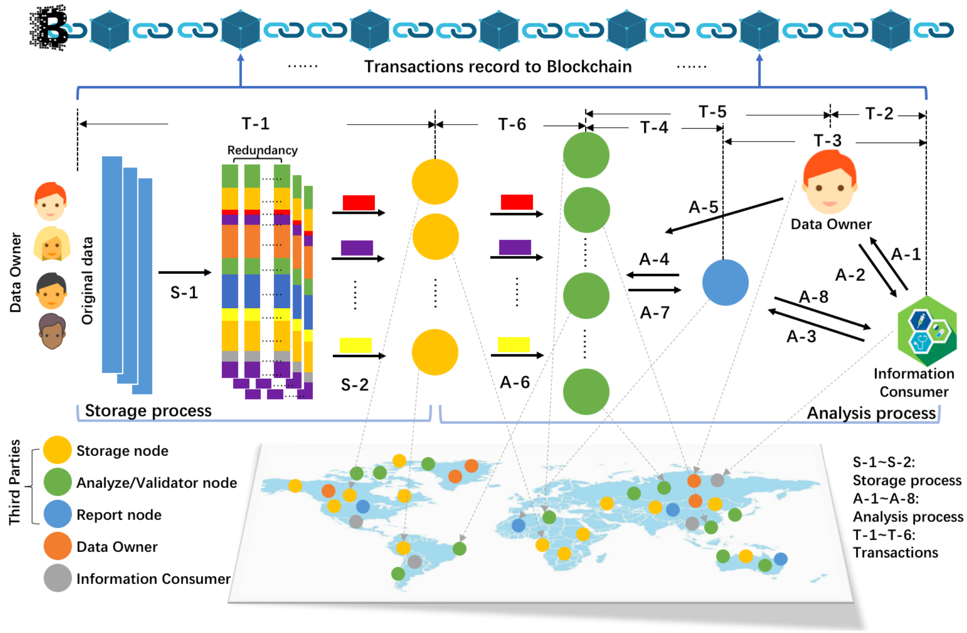


Figure 2: The privacy-preserving sharing protocol

related storage nodes, recording which storage node stores which data piece from whom. T-2: Transactions between (a) data owner(s) and (an) information consumer(s), recording the analysis agreement between them. T-3: Transactions between a report node and an information consumer, recording the process of the information consumer uploading the analysis script and the report node returning the complete analysis report. T-4: Transactions between a report node and analysis nodes, recording the process of the report node assigning the analysis jobs to the analysis nodes and the analysis nodes returning the single-point analysis result to the report node. T-5: Transactions between a data owner and analysis nodes, recording the process of transferring the encrypted share keys. T-6: Transactions between storage nodes and analysis nodes, recording the transferring of the encrypted data fragments from storage nodes to analysis nodes.

The decision to add a transaction to the blockchain is made by 'Validators' that assist in administering the system. The network efficiency and workload balance are managed by system routers. By this protocol, no node in the network is able to get access to the whole original data except data owner him/herself. 'Analysis nodes' in the network just provide their computing resources and applying 'double-blinded' analysis: They neither know whose data they are processing (the data they have are unidentifiable fragments) nor what analysis they are applying (the analysis scripts are compiled, they simply execute the computation). In addition, as all the transaction history is recorded in the blockchain - a digital ledger that is transparent and immutable, hence it is possible to ensure an analysis node will not always be assigned with data pieces from same persons, and prevent data pieces from being collected/aggregated by the analysis node as well as to reduce the risk of analysis nodes colluding to reveal the private gene sequencing data.

The security regarding the prevention of overhearing throughout the network is guaranteed by Public Key Infrastructure (PKI) framework as shown in Figure 4. The key exchange procedure is: Data pieces are encrypted by share keys (symmetric keys), and the share keys are encrypted by gene owner's public key. The encrypted data pieces and the encrypted share keys are stored in 'Storage nodes'. When applying analysis, 'Analysis nodes' retrieve the encrypted data pieces from 'Storage nodes' (a), and data owner retrieves the encrypted share keys from 'Storage nodes'(b). Then the data owner decrypts the share keys using his/her private key (c), then encrypts the share keys again by 'Analysis nodes' public keys (d) and transfers the encrypted share keys to 'Analysis nodes'(e). After that, 'Analysis nodes' decrypt the share keys using their private keys, and decrypt the data piece using the share key (f) and apply analysis (g).

3.2 Proof of Storage

In order to guarantee the data integrity, data owner is able to check for proof that storage of their gene sequence pieces into various of the data storage nodes has been effected. Figure 3 illustrates the process of checking 'Proof of Storage': the data owner computer obtains 'Proof of Storage' from the data storage nodes by arranging for the nodes to return hashes of at least part of the stored data of respective data owner computers upon request. The hashes are determined by the respective data owner computers prior to storage of the gene sequence pieces in the data storage hosts in order to carry out the check subsequently (a). In the case the storage node fails to provide the 'Proof of Storage' (b), the data owner computers will notify the smart contract controller (c), to find another available storage node (d), and a new storage contract will be signed between

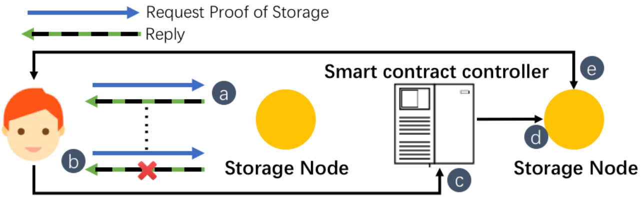


Figure 3: Proof of Storage

the data owner and the new storage node (e), and the corresponding data piece will be stored in the new storage node.

3.3 Consensus Protocol

There remain serious efficiency concerns with the technological underpinnings of crypto economic consensus networks. For example, the consensus mechanism most often used in existing systems, proof of work, consumes a very large amount of electricity in order to operate; the largest working blockchain using this mechanism, Bitcoin, has been shown to consume as much electricity as the entire country of Ireland.

In our proposed framework, if a gene sequence of one individual is fragmented into m pieces, then the task management server will assign jobs to $m*n$ analysis nodes, and every set of n analysis nodes will get the same piece of data and run the same analysis script, thus, they should achieve same single-point result. The system then compares their results and regards the majority as the correct, and the fastest analysis provider that produce the correct result wins the chance to add a block. It means an analysis node is also a validator (validators are like miners in bitcoin network). By this way, we replace the process of solving meaningless puzzles in 'proof of work' with the process of doing meaningful jobs, and also the correctness of analysis results can be achieved, though the computation is processed by untrusted nodes. The advantage of the protocol is we don't need to trust anyone in the network so that the network has a potential for global-wide participants and more openness.

Hence, an ecosystem is established, and also an all-win mechanism is designed to bring benefits to all stakeholders. For data owners, the privacy-preserving protocol are embedded in the network, instead of relying on institutional human actors, the data ownership for individuals is always guaranteed no matter how many times the owner has shared his/her data to others. As it is possible that more people would like to share their data, it will also benefits information consumers and their scientific and economic interests. For service providers, economic profits can be gained by offering infrastructures or services.

4 DATA FRAGMENTATION ALGORITHM

As revealed in a previous study [16], male's surname can be disclosed using only Short Tandem Repeats (STRs) which are just some small repeating pieces on Y chromosome. Therefore, important data pieces may still be in exposure and re-identification still can happen if we don't have a good fragmentation mechanism (for example, just splitting the data randomly or by some simple and fixed formulas like equal divisions). Based the models of our framework, we develop the data fragmentation algorithm. The algorithm aims to minimize the re-identification risk from data pieces, at meanwhile, to ensure GWASs analysis can be applied as usual.

The objective of the gene fragmentation framework is to find the smartest fragmentation on single-nucleotide polymorphisms (SNPs). A SNP is a variation in a single nucleotide that occurs at a specific position in the genome. For example, at a specific base position in the human genome, the C nucleotide may appear in most individuals, but in a minority of individuals, the position is occupied by an A. This means that there is a SNP at this specific position, and the two possible nucleotide variations - C or A - are said to be 'alleles' for this position. Each data fragment is analyzed in a separate analysis node. The smartest fragmentation will lower the accuracy of re-identification in each analysis node, and makes sure any fragments of data is unidentifiable.

4.1 Problem Definition

Let $x = [P_1, P_2, ..., P_k]$ be a sequence of SNPs of one person, P_l denotes the genotype on the l^{th} SNP ($l \in \{1, ..., k\}$), and there are k SNPs for each person. Let $X = \{x_1, x_2, ..., x_n\}$ be the training set consisting of n persons. Let $M = \{M_1, M_2, ..., M_N\}$ be a set of Masks (N is the number of fragments), where $M_i = [0, 1]^k$, *s.t.*, $M_i^T M_j = 0$ ($\forall i, j \in \{1, ..., N\}, i \neq j$), and $\sum_{i=1}^{N} M_i = [1]^k$.

Definition 4.1. **Mask Operation $\odot$:** By $X \odot M_i$ ($i \in \{1, ..., N\}$) we apply a mask M_i on X and produce a new set: $X_i' = \{x_{i1}', x_{i2}', ..., x_{in}'\}$. The genotype on the l^{th} SNP: $\widehat{P_l}$ ($l \in \{1, ..., k\}$) in x_{ij}' ($i \in \{1, ..., N\}, j \in \{1, ..., n\}$) is generated by the following formula:

$$\widehat{P_l} = \begin{cases} \text{original genotype on the } l^{th} \text{ SNP}, & \text{if } m_i[l] = 1 \\ unknown, & \text{if } m_i[l] = 0 \end{cases} \tag{1}$$

Definition 4.2. **REID:** By *REID* we mean a function that can be any classification methods to learn a classifier D_i from a dataset X_i' for the purpose of re-identification.

Our objective is to find a set of M to maximally lower the overall accuracy of $D = \{D_1, D_2, ..., D_N\}$.

4.2 Competition Model

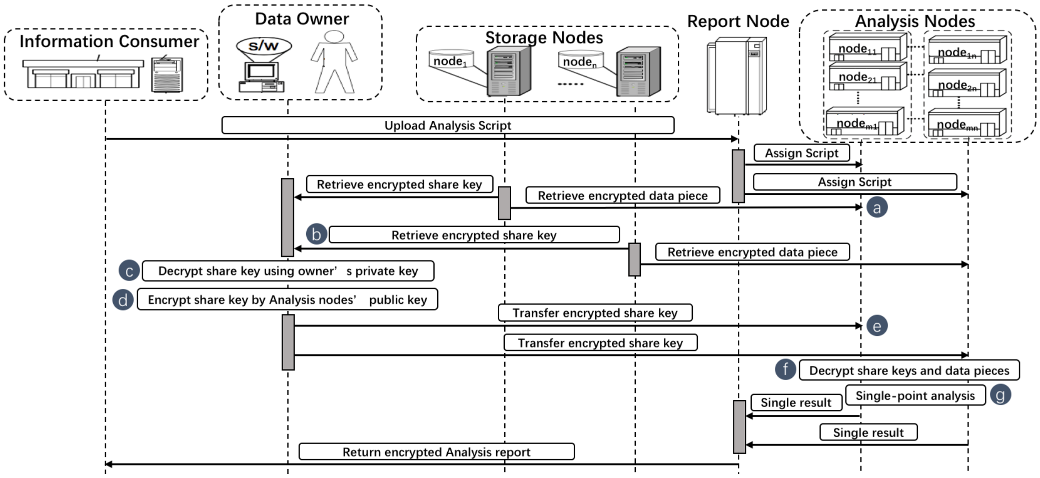
We propose a competition model. In the model, D acts as a number of adversaries that are trying to assign the correct identity labels to training samples, while M is to minimize the capacity of D. We simultaneously train the two 'teams', our purpose is no matter how good of D for re-identification, M can find a smartest splitting to generate $X' = \{X_1', X_2', ..., X_N'\}$ to maximally lower the overall accuracy of D.

The overall objective function is as below:

$$\begin{aligned} M = \ & \arg\max_M \min_D (REID(D_1, X_1'), ..., REID(D_N, X_N')) \\ s.t. \ & \begin{cases} X_i' = X \odot M_i \\ D_i = \arg\min_D Loss(D, X_i') \end{cases} \end{aligned} \tag{2}$$

The first condition indicates how to apply Mask M_i into training data X to generate X_i'. The second condition indicates how to train classifier D_i to re-identify X_i'.

Greedy strategy is used to achieve the optimization. In each iteration, only one bit is swapped between M. The best bit which


Information Consumer
Data Owner
s/w
Storage Nodes
node1
noden

Report Node
Analysis Nodes
node11
node1n
node21
node2n
nodem1
nodemn
Upload Analysis Script
Assign Script
Assign Script
Retrieve encrypted share key
Retrieve encrypted data piece
a
b
Retrieve encrypted share key
Retrieve encrypted data piece
c
Decrypt share key using owner' s private key
d
Encrypt share key by Analysis nodes' public key
Transfer encrypted share key
e
Transfer encrypted share key
f
Decrypt share keys and data pieces
Single result
Single-point analysis
g
Single result
Return encrypted Analysis report

Figure 4: Sequential Diagram for key exchange process

can maximally lower the accuracy of D is selected for the next iteration. The optimization process is shown in Algorithm 1.

Algorithm 1 The Fragmentation Algorithm

```
Input      : $X = \{x_1, x_2, ..., x_n\}$, $x = [P_1, P_2, ..., P_k]$
             Max Interations $Inte$
Output     : $M = \{M_1, M_2, ..., M_N\}$
Initialize : Randomly initialise $M$
             $X = \{X_1, ..., X_N\}$, where $X_i = X \odot M_i$
             $D = \{D_1, ..., D_N\}$, where $D_i = \arg\max_D REID(D, X_i)$
$Index = 1$
while $Index < Inte$ do
    $Index = Index + 1$
    randomly pick up $i, j$, where $i, j \in \{1, ..., N\}, i \neq j$
    if $|M_i| >= |M_j|$ then
        foreach $l \in [1, ..., n]$ do
            if $M_i[l] = 1$ then
                $M'_i = M_i, M'_j = M_j$
                $M'_i[l] = 0, M'_j[l] = 1$
                $X'_i = X \odot M'_i, X'_j = X \odot M'_j$
                $D_i = \arg\max_D REID(D, X'_i)$
                $D_j = \arg\max_D REID(D, X'_j)$
                $acc[l] = \max(REID(D_i, X'_i), REID(D_j, X'_j))$
            end
        end
        $l' = \arg\min_l acc[l]$
        $M_i[l'] = 0, M_j[l'] = 1$
    end
end
```

5 EVALUATION

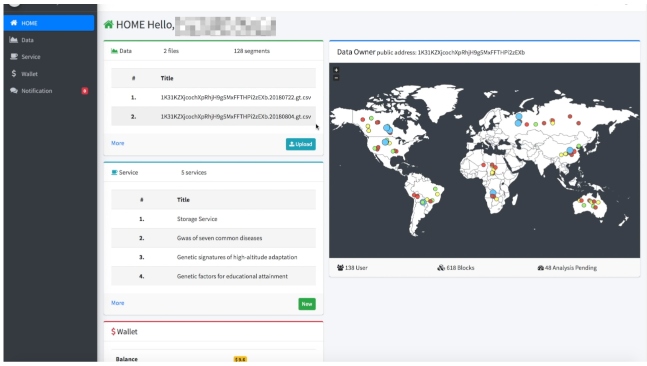
5.1 Proof of Concept

In this work, we provide a method for operating a data network including secure data storage and secure data analysis.

Secure data storage comprises: provision of a data sequence fragmentation software product installed on processing devices of each of the data owners for producing gene sequence pieces; facilitating storage of respective gene sequence pieces comprising gene sequences of the owners from the owner processing devices into the data storage nodes with storage redundancy; wherein each of the gene sequence pieces are encrypted with keys of their respective owners; whereby due to the storage redundancy and the 'proof of storage' mechanism, the gene sequences of respective owners may be retrieved by the owners from the data storage nodes in the event of loss of some of the data storage hosts.

Secure data analysis comprises: providing pieces of the gene sequences by use of an electronic data network to each of a plurality of network connected analysis providers wherein pieces provided to any one of the analysis providers are insufficient for identification of the corresponding data owner; operating a report node to transmit assigned tasks across the data network to each of the analysis nodes in respect of the gene pieces of each of the data owners, and the assigned tasks being produced in response to analysis specifications are received from computers of the information consumers; receiving analysis results from the analysis nodes for the assigned tasks in respect of the gene sequence pieces of each of the data owners and compiling respective reports therefrom; and transmitting the reports across said network to the network connected computers of the information consumers; wherein the information consumer computers receive neither the gene sequences nor the gene sequence pieces. The gene sequence fragmentation software includes instructions to fragment the gene sequence at positions of the gene sequence that minimize re-identification from the resulting pieces.

The prototype that implements and proves the concept of the protocol described in this work is available at: https://youtu.be/OuVA1KF443k. Figure5 shows a snapshot of the interface of the prototype.



HOME Hello, [redacted]

Data 2 files 128 segments

#	Title
1.	1K31KZXjcochXpRhjH9g5MxFFTHPi2zEXb.20180722.gt.csv
2.	1K31KZXjcochXpRhjH9g5MxFFTHPi2zEXb.20180804.gt.csv

Service 5 services

#	Title
1.	Storage Service
2.	Gwas of seven common diseases
3.	Genetic signatures of high-altitude adaptation
4.	Genetic factors for educational attainment

$ Wallet

Balance $ 9.6

Data Owner public address: 1K31KZXjcochXpRhjH9g5MxFFTHPi2zEXb

138 User 618 Blocks 48 Analysis Pending

Figure 5: 'Decentralized Ownership Ecosystem' Dashboard

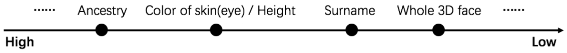


Figure 6: Different privacy protection levels

5.2 Privacy Level

Next, we evaluate our data fragmentation algorithm. In the experiments for the data fragmentation algorithm, the first thing we should consider is to define a concrete objective for *REID* function. We believe there are different privacy protection levels from high to low (Figure 6): 'Ancestry' is at a higher level, while 'Whole 3D face' is at a lower level. In this experiment, we choose a relatively high level of privacy protection - **Ancestry**, which at the moment is a primary driver in the re-identification models for DNA. The reason of choosing 'ancestry' as the concrete objective for *REID* function is because we believe if we prevent the adversary team from assigning correct **ancestry** labels to training samples, there will be little to no possibility for them to re-identify individuals, thus individuals' privacy can be regarded as protected.

5.3 Datasets

The 1000 Genomes Project [7] is a public dataset. It provides a comprehensive description of common human genetic variation by applying whole-genome sequencing to a diverse set of individuals from multiple populations. The 1000 Genomes Project use Variant Call Format (VCF) files to store gene sequence variations. VCF files contain information about positions in the genome and genotype information on samples for each of positions. From the released VCF files, we extract all the 62042 SNPs on Y-chromosome from 1233 male samples.

Individuals in the data set were sampled from 26 populations: British in England and Scotland (GBR), Southern Han Chinese, China (CHS), Puerto Rican in Puerto Rico (PUR), Colombian in Medellin, Colombia (CLM), African Caribbean in Barbados (ACB), etc,. In this paper, we aggregate the 26 populations into the following six continental ancestry groups: Africa, East Asia, Europe, South Asia, Latin Americas, and South Americas.

Table 1: Accuracy of *REID* with respect to different entropy levels

Entropy	Selected SNPs	Accuracy of *REID*	Tradeoff Factor
> 0.1	5547	0.824	0.883
> 0.2	2536	0.816	0.921
> 0.3	1872	0.808	0.923
> 0.4	1608	0.76	0.876
> 0.5	1248	0.754	0.875
> 0.6	565	0.752	0.882
> 0.7	126	0.68	0.817

5.4 Feature Selection

Firstly, we want to have a strong *REID* (adversary) in the competition model. We select features from 62042 SNPs by different entropy thresholds, by which we could remove irrelevant features without incurring much loss of information- higher entropy, fewer features, more information. The entropy for every feature is calculated by the following equation:

$$Entropy\ H(S_l) = -\sum p(S_l)\log p(S_l) \tag{3}$$

where $S_l = \{p_1, p_2, ..., p_n\}$, p_i is the genotype of i^{th} sample on the l^{th} SNP.

Then we use a three-layer FC network as *REID*, we define the tradeoff factor between the accuracy of *REID* and the number of training features [17]:

$$R = \frac{\nabla S}{\nabla A} \tag{4}$$

, where S denotes the number of SNPs, and A denotes the accuracy of *REID*.

From Table 1, we can see the entropy threshold of 0.3 gave us the best tradeoff factor: a relatively strong *REID* as well as a good feature space dimensionality reduction.

5.5 Effectiveness of privacy preserving

Next, we train the competition model based on the selected 1872 features. Theoretically, it is not guaranteed that a minimax game will converge [15], however, the empirical experiment shows the following effectiveness results of the competition model (Figure 7): The dark grey line is the upper bound of *REID*'s capacity:0.808 (train with all selected SNPs). The light grey line is the lower bound:0.167 (one out of six ancestry groups). We compare our method (blue dash line, which denotes the maximum accuracy value of D: $\max\{D_1, ..., D_N\}$) with 'random fragmentation' (orange). We observed that when the number of fragments N was two, our method only slightly reduced the capacity of *REID*, implying the genetic 'signal' for ancestry is very highly distributed throughout the genome. However, with the growth of N, the difference between our method and random fragmentation increased.

The boxplot shows the distribution of accuracy of D along the number of N. We observed from this plot that when N increased to 64, the majority of accuracy values of D went down to a relatively low level: around 0.25.

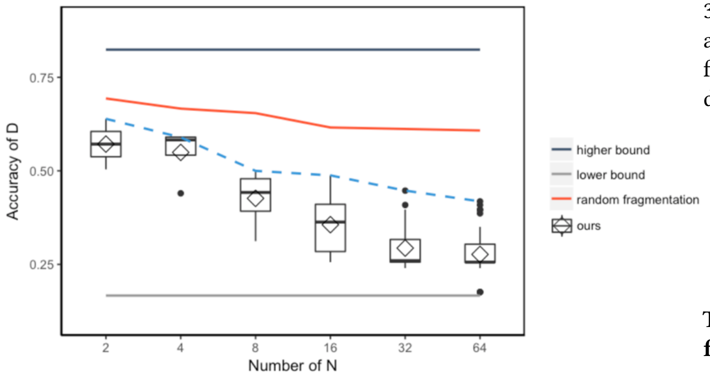


Figure 7: Accuracy of adversarial *REID* over N

5.6 Utility for GWASs

GWASs have been widely used in discovering the association between genotypes and phenotypes. In GWASs, individuals are split into case and control groups: one healthy control group and one case group affected by a disease. All individuals in each group are genotyped for the majority of common known SNPs. If one type of the variant (one allele) is more frequent in people with the disease, the variant is said to be associated with the disease. The associated SNPs are then considered to mark a region of the human genome that may influence the risk of disease. The allele count of each measured SNP is evaluated to identify variants associated with the trait in question.

Table 2: Contingency table for the standard χ^2 test

Group	Allele A	Allele B
Cases	a	c
Controls	b	d

Table 2 depicts the $2 * 2$ contingency table for allele counts in case and control groups, where an allele is counted twice if it is homozygous. The test statistic for the standard χ^2 test is expressed as

$$T_1 = \frac{(a+b+c+d)(bc-ad)^2}{(a+b)(a+c)(b+d)(c+d)} \tag{5}$$

and the test statistic for equiproportionality of the allele A in both groups is

$$T_2 = \frac{2(b+d)(bc-ad)^2}{b(a+c)^2 d} \tag{6}$$

These tests are accurate if the Hardy-Weinberg equilibrium condition is satisfied for a particular SNP, whereas the Cochran-Armitage test for trend can be used without this assumption.

For a marker with two alleles A and B, each individual in a case-control study is genotyped with one of three genotypes, AA, AB and BB (indexed by $i = 0, 1, 2$, respectively). The distribution of genotype counts can be put in a $2 * 3$ contingency table based on each subject's genotype and disease status as shown in Table 3. Let $(x0, x1, x2) = (0, c, 1)$ where the coefficient c can assume any value. Under the null hypothesis of no genetic association, the following test statistic is distributed asymptotically as a chi-square distribution with one degree of freedom:

$$Z^2(c) = \frac{r \times s}{n} \cdot \frac{\left[\sum_{k=0}^{2} x_k \times \left(\frac{r_k}{r} - \frac{s_k}{s}\right)\right]^2}{\left[\left(\sum_{k=0}^{2} x_k^2 \times \frac{n_k}{n}\right) - \left(\sum_{k=0}^{2} x_k \times \frac{n_k}{n}\right)^2\right]} \tag{7}$$

Table 3: Contingency table for the Cochran-Armitage test for trend

Group	Allele AA	Allele AB	Allele BB	Total
Cases	r_0	r_1	r_2	r
Controls	s_0	s_1	s_2	s
Total	n_0	n_1	n_2	n

In our protocol, the analysis nodes execute computation on data pieces, therefore, it naturally supports the allele count based analysis, and return exactly the same outputs as computing on original data.

5.7 Efficiency Performance

We compare the efficiency results with core algorithms for GWAS which are implemented in the state-of-art secure multiparty computation (SMC) platform *SHAREMIND* [3] (version 2018.03). In a decentralized network, computation power can be provided by individuals, so we intentionally choose a desktop (with 8 GB RAM and one 2.8 GHz Intel Core i5), instead of powerful servers, to implement both of the methods. Each of the data piece has 750 measured SNPs.

As shown in Figure 8, for 270 samples, *SHAREMIND* took around 43 seconds, and the execution time increased linearly over the number of samples. In comparison, our methods showed significantly better performance results: took around 0.72 second for 270 samples, and stayed nearly constant when the number of samples was growing.

In addition, the computational overhead of SMC grows with the number of players: a perfectly secure protocol which allows n players is with a computational overhead of $O(n \log n)$ [10]. In comparison, our protocol allows analysis nodes execute computation on original data pieces in a non-interactive way, therefore, the computational overhead is independent of the number of players, achieving the performance of $O(1)$.

6 CONCLUSION

We propose a method which consist of a decentralized network, with the PPS protocol and the data fragmentation algorithm to enable the privacy-preserving sharing of genomic data for GWASs. The advantages of our approach are: 1. We provide a preventive method to solve the data re-distribution problem that ethically satisfies people's fundamental interests in their data. 2. Our method provide high level of privacy - prevent data re-identification problem, at the same time, GWASs analysis can be applied as usual. 3.

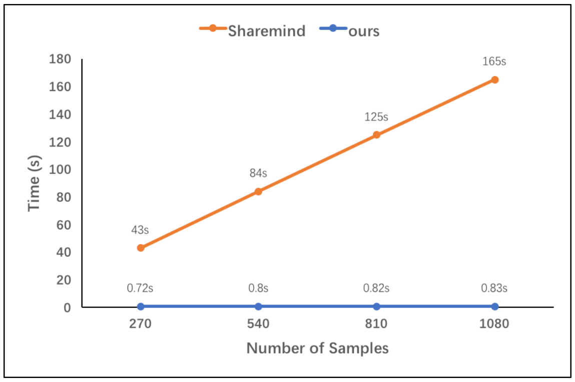


Figure 8: Execution time (s) over the number of samples

We solve the scalability problem by the fragmentation mechanism, providing not only decentralized storage, but also decentralized analysis. By constructing an ecosystem of gene fragmentation, we achieve a secure framework for privacy-preserving sharing genomic data and a system of openness, decentralization, and global participation designed to bring benefits to all stakeholders.

REFERENCES

[1] Mikhail J Atallah, Florian Kerschbaum, and Wenliang Du. 2003. Secure and private sequence comparisons. In *Proceedings of the 2003 ACM workshop on Privacy in the electronic society*. ACM, 39–44.

[2] Pierre Baldi, Roberta Baronio, Emiliano De Cristofaro, Paolo Gasti, and Gene Tsudik. 2011. Countering gattaca: efficient and secure testing of fully-sequenced human genomes. In *Proceedings of the 18th ACM conference on Computer and communications security*. ACM, 691–702.

[3] Dan Bogdanov, Liina Kamm, Swen Laur, and Ville Sokk. 2016. Rmind: a tool for cryptographically secure statistical analysis. *IEEE Transactions on Dependable and Secure Computing* (2016).

[4] Dan Boneh and Hovav Shacham. 2002. Fast variants of RSA. *CryptoBytes* 5, 1 (2002), 1–9.

[5] Fons Bruekers, Stefan Katzenbeisser, Klaus Kursawe, and Pim Tuyls. 2008. Privacy-Preserving Matching of DNA Profiles. *IACR Cryptology ePrint Archive* 2008 (2008), 203.

[6] Yangyi Chen, Bo Peng, XiaoFeng Wang, and Haixu Tang. 2012. Large-Scale Privacy-Preserving Mapping of Human Genomic Sequences on Hybrid Clouds.. In *NDSS*.

[7] 1000 Genomes Project Consortium et al. 2015. A global reference for human genetic variation. *Nature* 526, 7571 (2015), 68.

[8] Wellcome Trust Case Control Consortium et al. 2007. Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. *Nature* 447, 7145 (2007), 661.

[9] Caitlin Curtis and James Hereward. 2018. DNA facial prediction could make protecting your privacy more difficult. (2018). Retrieved May 06, 2018 from https://theconversation.com/dna-facial-prediction-could-make-protecting-your-privacy-more-difficult-94740

[10] Ivan Damgård, Yuval Ishai, and Mikkel Krøigaard. 2010. Perfectly secure multiparty computation and the computational overhead of cryptography. In *Annual international conference on the theory and applications of cryptographic techniques*. Springer, 445–465.

[11] Emiliano De Cristofaro, Sky Faber, Paolo Gasti, and Gene Tsudik. 2012. Genodroid: are privacy-preserving genomic tests ready for prime time?. In *Proceedings of the 2012 ACM workshop on Privacy in the electronic society*. ACM, 97–108.

[12] Cynthia Dwork. 2011. Differential privacy. In *Encyclopedia of Cryptography and Security*. Springer, 338–340.

[13] Craig Gentry and Dan Boneh. 2009. *A fully homomorphic encryption scheme*. Vol. 20. Stanford University Stanford.

[14] Kristian Gjøsteen. 2006. A new security proof for Damgård's ElGamal. In *Cryptographers' Track at the RSA Conference*. Springer, 150–158.

[15] Ian J Goodfellow. 2014. On distinguishability criteria for estimating generative models. *arXiv preprint arXiv:1412.6515* (2014).

[16] Melissa Gymrek, Amy L McGuire, David Golan, Eran Halperin, and Yaniv Erlich. 2013. Identifying personal genomes by surname inference. *Science* 339, 6117 (2013), 321–324.

[17] Yacov Y Haimes, Warren A Hall, and Herbert T Freedman. 2011. *Multiobjective optimization in water resources systems: the surrogate worth trade-off method.* Vol. 3. Elsevier.

[18] Petr Holub, Morris Swertz, Robert Reihs, David van Enckevort, Heimo Müller, and Jan-Eric Litton. 2016. BBMRI-ERIC Directory: 515 biobanks with over 60 million biological samples. *Biopreservation and biobanking* 14, 6 (2016), 559–562.

[19] Grace Hui Yang and Sicong Zhang. 2018. Differential Privacy for Information Retrieval. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. ACM, 777–778.

[20] Karthik A Jagadeesh, David J Wu, Johannes A Birgmeier, Dan Boneh, and Gill Bejerano. 2017. Deriving genomic diagnoses without revealing patient genomes. *Science* 357, 6352 (2017), 692–695.

[21] Somesh Jha, Louis Kruger, and Vitaly Shmatikov. 2008. Towards practical privacy for genomic computation. In *Security and Privacy, 2008. SP 2008. IEEE Symposium on*. IEEE, 216–230.

[22] Aaron Johnson and Vitaly Shmatikov. 2013. Privacy-preserving data exploration in genome-wide association studies. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1079–1087.

[23] Liina Kamm, Dan Bogdanov, Sven Laur, and Jaak Vilo. 2013. A new way to protect privacy in large-scale genome-wide association studies. *Bioinformatics* 29, 7 (2013), 886–893.

[24] Christoph Lippert, Riccardo Sabatini, M Cyrus Maher, Eun Yong Kang, Seunghak Lee, Okan Arikan, Alena Harley, Axel Bernal, Peter Garst, Victor Lavrenko, et al. 2017. Identification of individuals by trait prediction using whole-genome sequencing data. *Proceedings of the National Academy of Sciences* 114, 38 (2017), 10166–10171.

[25] Satoshi Nakamoto. 2008. Bitcoin: A peer-to-peer electronic cash system. (2008).

[26] Antonio Regalado. 2018. 2017 was the year consumer DNA testing blew up. (2018). Retrieved May 06, 2018 from https://www.technologyreview.com/s/610233/2017-was-the-year-consumer-dna-testing-blew-up/

[27] Md Nazmus Sadat, Md Momin Al Aziz, Noman Mohammed, Feng Chen, Shuang Wang, and Xiaoqian Jiang. 2017. SAFETY: Secure gwAs in Federated Environment Through a hYbrid solution with Intel SGX and Homomorphic Encryption. *arXiv preprint arXiv:1703.02577* (2017).

[28] Sean Simmons, Cenk Sahinalp, and Bonnie Berger. 2016. Enabling privacy-preserving GWASs in heterogeneous human populations. *Cell systems* 3, 1 (2016), 54–61.

[29] Latanya Sweeney, Akua Abu, and Julia Winn. 2013. Identifying participants in the personal genome project by name. (2013).

[30] Don Tapscott and Alex Tapscott. 2016. *Blockchain revolution: how the technology behind bitcoin is changing money, business, and the world.* Penguin.

[31] Caroline Uhlerop, Aleksandra Slavković, and Stephen E Fienberg. 2013. Privacy-preserving data sharing for genome-wide association studies. *The Journal of privacy and confidentiality* 5, 1 (2013), 137.

[32] Marten Van Dijk, Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. 2010. Fully homomorphic encryption over the integers. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 24–43.

[33] Peter M Visscher, Naomi R Wray, Qian Zhang, Pamela Sklar, Mark I McCarthy, Matthew A Brown, and Jian Yang. 2017. 10 years of GWAS discovery: biology, function, and translation. *The American Journal of Human Genetics* 101, 1 (2017), 5–22.

[34] Shuang Wang, Yuchen Zhang, Wenrui Dai, Kristin Lauter, Miran Kim, Yuzhe Tang, Hongkai Xiong, and Xiaoqian Jiang. 2015. HEALER: Homomorphic computation of ExAct Logistic rEgRession for secure rare disease variants analysis in GWAS. *Bioinformatics* 32, 2 (2015), 211–218.

[35] Andrew C Yao. 1982. Protocols for secure computations. In *Foundations of Computer Science, 1982. SFCS'08. 23rd Annual Symposium on*. IEEE, 160–164.

[36] Fei Yu, Stephen E Fienberg, Aleksandra B Slavković, and Caroline Uhler. 2014. Scalable privacy-preserving data sharing methodology for genome-wide association studies. *Journal of biomedical informatics* 50 (2014), 133–141.