Privacy-Preserving Gradient Descent for Distributed Genome-Wide Analysis

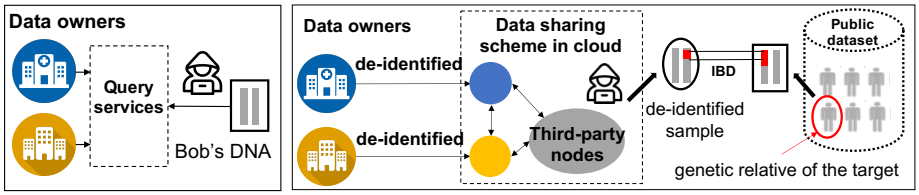
Yanjun Zhang, Guangdong Bai(✉), Xue Li, Caitlin Curtis, Chen Chen, and Ryan K L Ko

The University of Queensland, St Lucia, Queensland, Australia

Abstract. Genome-wide analysis, which provides perceptive insights into complex diseases, plays an important role in biomedical data analytics. It usually involves large-scale human genomic data, and thus may disclose sensitive information about individuals. While existing studies have been conducted against data exfiltration by external malicious actors, this work focuses on the emerging identity tracing attack that occurs when a dishonest insider attempts to re-identify obtained DNA samples. We propose a framework named υFRAG to facilitate privacy-preserving data sharing and computation in genome-wide analysis. υFRAG mitigates privacy risks by using vertical fragmentations to disrupt the genetic architecture on which the adversary relies for re-identification. The fragmentation significantly reduces the overall amount of information the adversary can obtain. Notably, it introduces no sacrifice to the capability of genome-wide analysis—we prove that it preserves the correctness of gradient descent, the most popular optimization approach for training machine learning models. We also explore the efficiency performance of υFRAG through experiments on a large-scale, real-world dataset. Our experiments demonstrate that υFRAG outperforms not only secure multiparty computation (MPC) and homomorphic encryption (HE) protocols with a speedup of more than 221x for training neural networks, but also noise-based differential privacy (DP) solutions and traditional non-private algorithms in most settings.

1 Introduction

Advances in biomedical data analytics over the last few decades have enabled inexpensive large-scale and whole genome studies. *Genome-wide analysis*, such as genome-wide association studies (GWAS) and genome-wide complex trait analysis (GCTA), plays an important role in assisting with predicting health risks, enabling preventative and personalized medicine, and investigating natural selection and population differences [27]. As a data-driven study, genome-wide analysis typically requires a large sample size to confirm differences with statistical confidence. Sharing genomic data on a large scale thus becomes essential. This however raises privacy concerns, as much sensitive information, including health status and family relationships, can be derived from the human genome. Indeed, various genetic privacy breaches and attacks [8, 9] have highlighted the urgent need to enhance privacy in the analysis.



(a) Membership inference. (b) Identity tracing.

Fig. 1: Genetic privacy breaches overview.

In the research area of genetic privacy, our community has been focusing on detecting and defending against the membership inference [12, 18, 25]. It aims to reveal the presence of an individual's genome in a dataset (e.g., a dataset of HIV patients' DNA sequences) that the adversary has partial or blackbox access through Beacon systems [12] or trained machine learning models [25] (Fig. 1a). In this work, we target an emerging but overlooked privacy threat known as the *identity tracing* [9]. It occurs in a data sharing scheme when the adversary is an insider such as dishonest participants or cloud service providers, shown in Fig. 1b. Through the data sharing scheme, the adversary could gain access to the datasets which in most instances have been conducted de-identification, and attempts to re-identify individuals in them.

The adversary's re-identification tactic is to exploit the *correlations* among genomic data. Due to the existence of the correlations, even though the personal identifiers (such as name and address) have been removed from the datasets, the inherent correlations among genomic data could still put an individual's privacy at risk. Fig. 1b illustrates a representative attack scenario which exploits the genetic inheritance laws, the most fundamental correlation. The adversary applies a sophisticated tactic called *long-range familial search* (LFS) which detects the target's genetic relatives from identity-retaining datasets by matching the *identical-by-descent* (IBD) segments of DNA [9]. Having known the genetic relatives, the adversary is able to narrow down the target's identity.

LFS is a de facto investigative tool by law enforcement to trace suspects due to its re-identification capacity. In a notable case, LFS was used to successfully trace the Golden State Killer in 2018 [9]. With the explosion[1] of public consumer genomics services (such as GEDmatch and MyHeritage[2]) which allow users to upload raw genotype files for genetic analysis including searching for their genetic relatives, privacy threats of utilizing LFS against normal individuals are greatly exacerbated - as the growth of available datasets (refer to the public dataset in Fig. 1b) significantly increases the probability that an LFS identifies individuals. *This presents an urgent demand for privacy-preserving solutions that take the correlations into account.*

The existing techniques of preserving privacy for *general-purpose* data analytics, including differentially private (DP) deep learning frameworks [3, 16, 39],

[1] There have been around 26 million tests sold in 2019 [24].
[2] https://www.gedmatch.com/; https://www.myheritage.com/

and cryptographic technologies such as homomorphic encryption (HE) [5, 30] or secure multiparty computation (MPC) [17,36], may not be applicable to genome-wide analysis due to their inherent limitations. The former ones are usually based on additive noise mechanisms which perturb the original data/models and consequently affect model accuracy to a certain extent. This is often prohibitive given the high accuracy required by genome-wide analysis. The HE and MPC prevent plaintext data disclosure, and are able to compute identical results from the cyphertext as from the plaintext, but they are known to be limited by non-trivial computational or communicational overhead.

Our Solution. We propose υFRAG, which is a distributed framework for preserving privacy in collaborative genome-wide analysis. υFRAG achieves high efficiency via distributing computation throughout multiple *unnecessarily trustworthy* nodes, and privacy preservation via an innovative *DNA sequence fragmentation*. The proposed fragmentation is a *vertical* partitioning and reassembling of DNA segments, designated against the privacy risk of identity tracing.

The insight of the proposed fragmentation is to disrupt the *genetic architecture*[3] where the correlations stem from. This seemingly straightforward strategy inherently suits DNA sequence data, given that a DNA sequence is essentially a $(A|T|G|C)^*$ string and its "semantics" are reflected by the occurrence of variants at particular locations (i.e., loci). In the human genome, the effect size of single variation associated with the heritability is small. In other words, a person's phenotypic heritability, e.g., his/her susceptibility to disease, depends more on the combined effect of all the associated genes than on one particular genetic variation [27]. Therefore, the proposed fragmentation can significantly reduce the chance of an adversary fully obtaining the information about the heritability in genome.

υFRAG also addresses the fundamental challenge of privacy-utility trade-off. Given the vertically partitioned datasets, most primitive functionalities and algorithms used in genome-wide analysis, such as genetic relationship matrix estimation [34] and genotype clustering [4], can be *parallelized*. This can be formalized as the *parallel correctness* that an analysis in our framework reaches the same results as in a centralized way. We demonstrate that our parallelization of gradient descent (Section 3) and other primitive functionalities (Appendix B) preserves this property. In such a way, our fragmentation results in *no* sacrifice to those analyses relying on them, e.g., any machine learning algorithm based on gradient descent for optimization.

Contributions. We summarize the main contributions of this work as follows.

1. A Privacy-preserving Framework. We propose a novel framework υFRAG for privacy-preserving sharing of DNA data in large-scale genome-wide analysis. υFRAG is characterized by its capabilities of privacy preservation and verifiably correct computation for genome-wide analysis.

2. A Novel Privacy-preserving Technique and its Quantitative Analysis. To the best of our knowledge, our work is the first to use DNA sequence

[3] *Genetic architecture* refers to the underlying genetic basis and its variational properties that are responsible for broad-sense heritability [26].

fragmentation for mitigating the genetic privacy threat. The evaluation of the privacy preservation in υFRAG is twofold. In a high level, we construct a formal model to quantitatively evaluate the overall amount of information leaked to the adversary. In the individual level, we propose ϵ-indistinguishability - a variant model of ϵ-local differential privacy, and prove that the vertical fragmentation provide a bound for the adversary's capacity in distinguishing individuals.
3. Experimental Evaluation. We conduct experiments on a real-world dataset, showing the significant improvement of efficiency by our framework compared with the MPC/HE algorithms and a state-of-the-art noise-based DP solution.

2 System Design

2.1 υFrag Overview

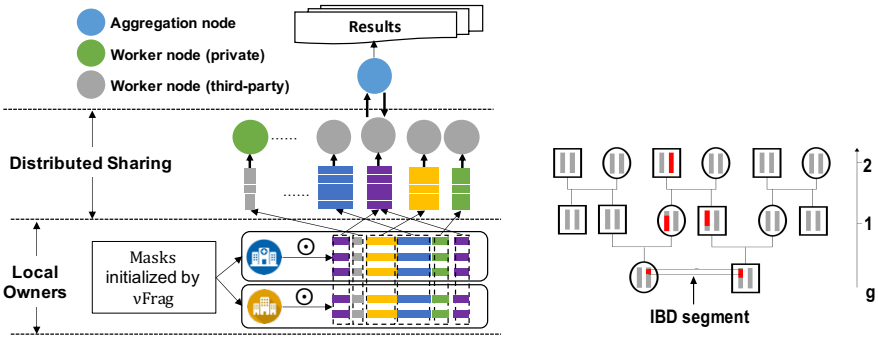
Fig. 2a demonstrates the architecture and workflow of υFRAG. It provides a distributed sharing network for the participants (local owners) to assemble their local DNA sequence datasets for large-scale analysis. The network is comprised of an aggregation node A (the blue node in Fig. 2a) and a worker node layer that includes s worker nodes $S^1, ..., S^s$ (the green and grey nodes). The original datasets are split into fragments locally before they leave their owners. These fragments are then transferred to the worker nodes. Each of them separately processes a few fragments and reports its single-point result to the aggregation node, which synthesizes the analysis results and sends them to the recipients.

The rationale to fragment the dataset in a vertical manner is to hinder malicious or compromised worker nodes from gaining the complete DNA sequence (x_t) and IBD segments. Below we define the mask operation for the fragmentation. Let X be a dataset of $\mathbb{S}^{m\times n}$, and $x_{ij} \in X$ represents the genotype on the j^{th} SNP of i^{th} individual. Let $Mask = \{M^1, ..., M^s\}$ be a set of s vectors of $[0|1]^n$ (denoting a n-dimensional vector comprising of 0 and 1) *s.t.*, $M^1 \vee ... \vee M^s = [1]^n$.

Definition 1. ***(Mask Operation $\odot$ for Vertical Fragmentation)*** *Given a mask $M^l = [mask_1^l, ..., mask_n^l]$ ($l \in \{1, ..., s\}$), $X \odot M^l$ produces a dataset X^l, whose element x_{ij}^l in X^l is generated by*

$$x_{ij}^l = \begin{cases} x_{ij}, & \text{if } mask_j^l = 1 \\ drop, & \text{if } mask_j^l = 0 \end{cases} \tag{1}$$

The *Mask* is initialized by υFRAG and distributed to each participant for them to generate fragments. By applying the mask operation with each vectors in *Mask*, participants produce s new datasets, $X^1, ..., X^s$, and each is then dispatched to a worker node. Note that participants do not have to assemble their original datasets for fragmentation; instead, each X^l is constructed in node S^i from individual fragments, as shown by the colored rectangles in Fig. 2a. The genome-wide analysis functionalities are then achieved by the collaboration between the worker nodes and the aggregation node. Initialization of the mask is



(a) The architecture and workflow of vFRAG. (b) Example of an IBD segment.

Fig. 2: System Design.

the key to disrupt the genetic architecture. In vFRAG, this is determined by the fragmentation strategy (detailed in Section 5).

2.2 Attacker Model and Assumptions

We assume an honest-but-curious adversary $\mathcal{A}$ who may control t out of s worker nodes, where $t \leq s$. For the sake of simplicity, our privacy analysis in Section 5 considers an aggregation node out of the adversary's control. Such an aggregator can be accommodated with a private server or a private cloud instance in reality, as it is designed to be free from any computation-intensive task. In Section 7, we show that even though it is compromised, the identity tracing attack is unlikely (with a negligible probability) to derive the original DNA sequence for re-identification.

Let δ denote the proportion of trusted worker nodes, i.e., $\delta = (s-t)/s$. Each worker nodes holds a fragment $X^l = X \odot M^l$, and X has been anonymized such that the identifiers of its samples are removed. We assume the adversary also has access to a publicly available datasets $\mathcal{D}$ which comprise genotyped individuals. Given an arbitrary DNA sequence from $\{X^l\}|_{S^l \text{ is under } \mathcal{A}\text{'s control}}$ denoted by x_t, the adversary attempts to re-identify x_t's subject (referred to as the *target*) by searching for the target and/or its genetic relatives from $\mathcal{D}$. We parameterize this re-identification with the g^{th} degree of genetic relatives, where $g \geq 1$ ($g = 1$ for target him/herself or siblings, $g = 2$ for first cousins, and so on) [4].

Below we brief the attack techniques the adversary can use, and leave their models in Section 4.

– **LFS Attack.** LFS makes use of the *IBD segments*, which are DNA segments inherited by persons having a common ancestor, for re-identification. Fig. 2b shows an IBD segment co-inherited from a common ancestor two generations back. IBD segments indicate the genetic distance of two individuals, and are measured in *centiMorgans* (cM). The higher the number of centiMorgans of

[4] For readability, a table of notations is included as Appendix A.

IBD segments, the more significant the match is, i.e., the higher probability that target and the matched individual have inherited from a recent ancestor [9]. As a result, the capacity of LFS can be regarded as the probability of the two individuals sharing *sufficient* detectable IBD segments.

– **Genotype Imputation.** We also take into account the *genotype imputation* technique that could infer the missing genotypes of a DNA sequence based on the remaining genotypes [7]. It could be abused by the sophisticated adversary to learn more fragments based on those it obtained.

3 Privacy-Preserving Gradient Descent

To retain the capability of genome-wide analysis, vFRAG must reach exactly the same result as a traditional non-distributed framework does when operating any computation. We formalize this as the *parallel correctness* of parallelized computation with respect to the vertical fragmentation.

Definition 2. ***(Parallel Correctness)*** *Given a function $\mathcal{F}$ which takes as input a genomic dataset $X \in \mathbb{S}^{m \times n}$, assume a fragmentation strategy partitions X into a set of fragments X^l ($l = \{1, ..., s\}$), and each of them is associated with a worker node $S^l \in S$, where S is the set of worker nodes interacting with an aggregation node A. The parallelized $\mathcal{F}$ in vFRAG, denoted by $\mathcal{F}'$, is parallely correct if $\mathcal{F}'(\{(S^l, X^l), A\}_{S^l \in S}) = \mathcal{F}(X)$.*

In the following, we present vFRAG's computation of gradient descent, the optimization that most analyses rely on, and prove its parallel correctness. We note that our computation also works on SGD, in which X^l denotes a randomly selected training sample or a subset of training samples.

We let $\mathcal{F}_{GD}(J, X)$ denote the gradient descent optimization applied on the dataset X with a cost function J. J is defined as $J(\sigma(XW), y, W)$, where W denotes coefficient matrix and σ is the hypothesis function which is determined by the learning model. In logistic regression, σ is usually a sigmoid function, while in neural networks, it is a composite function known as the forward propagation. The optimal solution of W, denoted as W_*, is derived by the optimization $\arg\min_W J(\sigma(XW), y, W)$, and W is updated as $W := W - \alpha \frac{\partial J}{\partial W}$.

In vFRAG, the parallelized gradient descent optimization, denoted by $\mathcal{F'}_{GD}$, is designed as the following steps.

– ***Initialization***. At the beginning of the task, the fragments $X^l \in \mathbb{R}^{m \times d_l}$ ($l \in \{1, ..., s\}$) are assigned to the corresponding S^l, and S^l randomly initializes its $W^l \in \mathbb{R}^{d_l \times H}$ which is associated with X^l.
– ***Step 1***. Each S^l com putes $X^l W^l$, and sends $X^l W^l$ to the aggregation node.
– ***Step 2***. The aggregation node computes $XW = \sum_{l=1}^{s} X^l W^l$, and the gradient Δ, which is the gradient of the cost function J with respect to XW, i.e., $\Delta = \frac{\partial J}{\partial XW}$. Then it sends Δ back to each worker node.
– ***Step 3***. Each S^l updates the respective coefficient $W^l := W^l - \alpha {X^l}^T \Delta$.

Step 1-3 repeat for the next iterations until convergence.

The following theorem demonstrates that if $\mathcal{F}_{GD}(J, X) = (W_*, \eta)$, i.e., if the non-distributed optimization outputs W_* that makes J converge to a local/global minimum η, then executing $\mathcal{F}'_{GD}(\{(J, S^l, X^l), Agg\}_{S^l \in S})$ on $X^1, ..., X^s$ in υFRAG with the same hyper settings (such as step size, model structure, initialization) will also output W_* that makes J converge to η.

Theorem 1. *$\mathcal{F'}_{GD}$ is parallelly correct.*

Proof. Let W_i denote the model parameters of $\mathcal{F}_{GD}$ at the i^{th} training iteration. Let $W'_i = |\{W_i^l\}|_{l\in\{1,..,s\}}$ denote the vertical concatenation on $\{W_i^l\}_{l\in\{1,..,s\}}$, i.e., the model parameters of $\mathcal{F}'_{GD}$ at the i^{th} training iteration. The theorem can be proved by induction as follows.

- ***Base case:*** $W_0 = W'_0$ by assumption of hyper setting.
- ***Inductive step:*** In $\mathcal{F}_{GD}$, the i^{th} $(i \geq 1)$ training iteration update W_i as

$$\begin{aligned} W_i &= W_{i-1} - \alpha \frac{\partial J}{\partial W_{i-1}} = W_{i-1} - \alpha \frac{\partial J}{\partial (XW_{i-1})} \frac{\partial XW_{i-1}}{\partial W_{i-1}} \\ &= W_{i-1} - \alpha X^T \frac{\partial J}{\partial (XW_{i-1})}. \end{aligned} \tag{2}$$

In $\mathcal{F}'_{GD}$, each node S_l updates its local W_i^l as

$$W_i^l = W_{i-1}^l - \alpha {X^l}^T \Delta = W_{i-1}^l - \alpha {X^l}^T \frac{\partial J}{\partial (XW'_{i-1})}.$$

Since $W'_i = |\{W_i^l\}|_{l\in\{1,..,s\}}$ and $X = |\{X^l\}|_{l\in\{1,..,s\}}$, we have in $\mathcal{F}'_{GD}$ that

$$W'_i = |\{(W_{i-1}^l - \alpha {X^l}^T \frac{\partial J}{\partial (XW'_{i-1})})\}|_{l\in\{1,..,s\}} = W'_{i-1} - \alpha X^T \frac{\partial J}{\partial (XW'_{i-1})}. \tag{3}$$

Comparing Equation 2 and 3, W_i and W'_i are updated with the same equation. Hence, $W_i = W'_i$. Thus, $\mathcal{F}_{GD} = \mathcal{F'}_{GD} = (W_*, \eta)$. □

4 Modeling Attacks for Privacy Analysis

In this section and Section 5, we assess the privacy-preservation of υFRAG's distributed gradient descent. We starts with modeling the proposed attacks (cf. Section 2.2), and leave the privacy analysis in Section 5.

4.1 Modeling the LFS Attack

We adopt a Shannon entropy based measurement to investigate the amount of information the adversary can obtain. Shannon entropy has been extensively employed as a metric to evaluate privacy-preserving mechanisms [8, 28, 31, 32], due to its capability of quantifying the expected contribution of a piece of data in reducing the uncertainty of the target's identity among the base population.

We assume the target's record in a genome-wide analysis study is randomly sampled from a defined population with the size denoted by N. This therefore translates to $log_2 N$ bits of entropy. Take the US population as an example. With the population of 329 million in 2019, the uncertainty of a random sample's identity can be measured as 28.2 bits of entropy. Denote γ as the mean number of children per mating pair. Conditioned that a successful match between the target and a genetic relative at g^{th} degree, the amount of bits of information obtained by the adversary can be derived as $h(g) = log_2 N - log_2 \sum_{k=1}^{g} \gamma^{\ k} = log_2 \frac{N}{\sum_{k=1}^{g} \gamma^{\ k}}$. For example, if the LFS successfully matches the target or his/her siblings ($g = 1$) from the US population (where $\gamma = 2.5$), the adversary gains 26.88 bits of information. In other words, the uncertainty of target's identity is reduced to 1.32 bit. Then, we can formulate the expected entropy bits gained by the LFS attack as the function of generation degree g as $H(g) = h(g)Pr(identify)$, where the $Pr(identify)$ is the probability of a successful match between the target and a random individual at g^{th} degree. The $H(g)$ is taken as the indicator of the capacity of the LFS attack, and thus the core of our privacy analysis becomes to determine the $Pr(identify)$.

A successful match between the target and a random individual depends on the conditions that these two individuals are genetic relatives, and they share detectable IBD segments [9]. Therefore, $Pr(identify)$ can be derived from calculating the following two probabilities: 1) $Pr(shared)$, the probability of the target and a random individual sharing a pair of ancestors at g^{th} degree given $N(g)$ which is the population size of the generation at g^{th} degree, and 2) $Pr(match|g)$, the probability that these two individuals share *sufficient* IBD segments to be detected given the public dataset $\mathcal{D}$ of size R. Here the sufficiency is defined as the IBD matching parameters - to declare a match between the target and an individual in $\mathcal{D}$, there should be at least c IBD segments, each of which is of length $\geq v(cM)$.

To identify the target, the adversary needs to find at least t matches from $\mathcal{D}$. Thus, $Pr(identify)$ can be formulated as $Pr(identify) = 1 - \sum_{k=0}^{t-1} B(k; R, Pr(shared) \cdot Pr(match|g))$, where B is the probability mass function of the binomial distribution. Below, we give $Pr(shared)$ and $Pr(match|g)$:

$$\begin{aligned} Pr(shared) &= \frac{2^{2g-2}}{N(g)} \prod_{g'=1}^{g-1} (1 - \frac{2^{2g'-2}}{N(g')}), \\ Pr(match|g) &= 1 - \sum_{k=0}^{c-1} B(k; 2Lg + 22, \frac{Pr(IBD)}{2^{2g-2}}), \end{aligned} \tag{4}$$

where $Pr(IBD)$ denotes the probability of the shared IBD segment length to exceed v, and L is the total genome length, which is roughly $3,500\ cM$ [23]. The derivation of $Pr(shared)$ and $Pr(match|g)$ is detailed in our technical report [1].

Quantifying Privacy Threat of LFS. We now are able to quantify privacy threat of the LFS attack when vFRAG is not applied. We first derive $Pr(IBD)$ in the scenario where the adversary has the access to the full genome of the target. As the length of the IBD segment is exponentially distributed with a

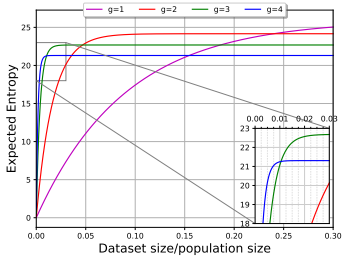


Fig. 3: Attack performance for varying dataset sizes.

mean of $1/(2g)$ cM [23], the probability density function (PDF) of the length of IBD segments is: $Pr(x) = 2ge^{-2gx}$. Therefore, the probability of the shared IBD segment length exceeding v is $Pr(IBD) = \int_v^\infty 2ge^{-2gx}dx = e^{-2vg}$.

With this, we explore the performance of LFS for various public dataset sizes, taking the aforementioned US population (329 million and $\gamma = 2.5$) as a case study. In general, as the dataset size increases, the LFS attack acquires higher entropy, as is shown in Fig. 3. When the dataset includes around 30% of the population, the attacker is able to achieve 25.9 bits (the magenta line). We also notice that in the cases where the dataset size is small, a greater g is more favorable to the attacker. When the public dataset covers less than 1% of the population, the best performance (21.3 bits) is achieved by the re-identification of the third cousins (the blue line). This may be because the number of distant relatives in the population is greater than close ones. For example, in Fig 2b, the circle in level 0 has seven relatives with g=2 whereas two with g=1. As a result, when the size is small, the probability of successfully searching a distant relative is higher. Nevertheless, as the coverage of the dataset goes higher, identifying closer relatives gives the attacker more significant bits.

It is worth noting that the privacy threat is becoming worse in the real world, with the increasing number of consumer genomics services for searching identified genetic relatives. For example, with GEDmatch, which is a public available database that contains around one million DNA profiles (0.3% of the target population), LFS is able to achieve up to 19 bits of entropy (the blue line in the zoomed view of Fig. 3). The expected entropy further rises to 22.5 bits if the genetic dataset reaches the scale that covers 2% of the US population (the green line in the zoomed view of Fig. 3).

4.2 Modeling the Genotype Imputation

Next, we modeling the genotype imputation that infers (part of) missing genotypes of the target by matching the observed ones with a reference panel of haplotypes. A haplotype refers to a set of SNPs found on the same chromosome. Haplotype reference panels are widely used for genotype imputation [7]. The model typically used for the inference is a Hidden Markov Model (HMM) in which the hidden states are a sequence of pairs of haplotypes in a reference

panel [7]. That is,

$$Pr(G|H) = \sum_{Z^{(1)},Z^{(2)}} Pr(G|Z^{(1)}, Z^{(2)}, H) \cdot Pr(Z^{(1)}, Z^{(2)}|H), \tag{5}$$

where $H = \{H_1, ..., H_K\}$ is a set of K known haplotypes (the reference panel), $H_i = \{H_{i1}, ..., H_{in}\}$ is a single haplotype and $H_{ij} \in \{0, 1\}$ (0/1 stand for reference and alternative genotype respectively). $G = \{G_1, ..., G_n\}$ denotes the genotype data on the target individual, and $G_i \in \{0, 1, missing\}$. $Z^{(1)} = \{Z_1^{(1)}, ..., Z_n^{(1)}\}$ and $Z^{(2)} = \{Z_1^{(2)}, ..., Z_n^{(2)}\}$ are the two sequences of hidden states at the n sites (i.e., loci) and $Z_l^{(j)} \in \{1, ..., K\}$. Intuitively, these hidden states can be regarded as pairs of haplotypes in the set H that are copied to form the genotype G.

The first term $Pr(G|Z^{(1)}, Z^{(2)}, H)$ in Equation 5 defines the emission probabilities λ in HMM, which allows for mutation at each SNP. The second term $Pr(Z^{(1)}, Z^{(2)}|H)$ models the transition probability ρ. It reflects the linkage disequilibrium between two alleles and determines how $Z^{(1)}$ and $Z^{(2)}$ transits from H_i to H_j along the sequence. When calculating the state probability for loci with missing genotypes, λ is cancelled out and only the transition probability ρ plays a role. The transition probability ρ is dominated by the genetic distance, i.e., the longer the genetic distance between the observed genotypes, the larger the probability $(1 - \rho)$ for $Z^{(1)}$ and $Z^{(2)}$ transiting from one reference haplotype to others [7]. Therefore, with a growing number of missing genotypes in the IBD segments which is proportional to the genetic distance, the probability $(1 - \rho)$ increases. This results in more possible candidates for the imputation, and thus reduces the expected number of accurately imputed genotypes.

Here, we provide the upper bound of the expected number of accurately imputed genotypes (denoted as $\overline{T}$) with respect to the number of missing genotypes (denoted as τ) as

$$\overline{T} < \frac{K}{1-\rho} K^{-\frac{1-\rho}{K}\tau} \prod_{i=1}^{\tau} MAF_i \sum_{k=0}^{\tau} \binom{\tau}{k} (\tau - k), \tag{6}$$

where MAF_i is the minor allele frequency of the i^{th} missing genotype (theoretically $MAF_i \leq 0.5$). The derivation of Inequality 6 can be found in our technical report [1]. The evaluation the privacy preservation assumes that the adversary applies the genotype imputation by default, implying that the adversary knows $\overline{T}$ extra genotypes. We further set $MAF_i = 0.5$ (such setting gives the adversary advantage as it leads to a large $\overline{T}$), such that Inequality 6 is simplified as

$$\overline{T} < \frac{\tau K}{2(1-\rho)} K^{-\frac{1-\rho}{K}\tau}. \tag{7}$$

5 Analysis of Privacy Preservation

With the models of attacks, we then analyze the privacy preservation of vFRAG against the adversary who has compromised a proportion of nodes. We outline our evaluation from the following two levels.

- **Collection Level: Reduction of Overall Information Leakage** (Section 5.1). We quantify information leakage reduction of two typical fragmentation strategies. Through the comparison with the baseline, we show that the information leakage to the adversary exponentially decreases with the growth in the proportion of trusted nodes.
- **Individual Level: Indistinguishability among Individuals** (Section 5.2). Information entropy is suitable for measuring the information leakage of a system as a whole, but less capable on individuals. As a complementary, we define ϵ-*indistinguishability* - a variant model of local differential privacy. We prove such indistinguishability provides the bound of an adversary's capacity of distinguishing an individual target based on the information obtained from the fragments (including those inferred by the genotype imputation).

5.1 The Collection-level Analysis

Analysis on Random Fragmentation Strategy. First, we consider a scenario of *random sampling* that the individuals in the public dataset are randomly selected from a defined population. In this scenario, υFRAG applies a random fragmentation on the dataset. Assuming the adversary has applied the genotype imputation, the expected length of IBD segments held by the adversary is increased from $1/2g$ to $(1+\frac{\overline{T}}{n})/2g$, in which $\frac{\overline{T}}{n}$ is the imputation rate. Then, the PDF of the length of IBD segments after the imputation is $Pr(x)_{adversary} = \frac{2g}{1+\overline{T}/n} e^{-\frac{2g}{1+\overline{T}/n}x}$. The probability of this IBD segment's length to exceed v is $Pr(IBD)_{impute} = \int_v^{\infty} \frac{2g}{1+\overline{T}/n} e^{-\frac{2g}{1+\overline{T}/n}x} dx = e^{-\frac{2vg}{1+\overline{T}/n}}$. In the random fragmentation setting, the probability for the adversary to hold this segment is $1-\delta$. Therefore, the probability of the adversary holding a detectable IBD segment in this scenario, denoted by $Pr(IBD)_1$, can be calculated as:

$$Pr(IBD)_1 = (1-\delta) e^{-\frac{2vg}{1+\overline{T}/n}}. \tag{8}$$

Combining $Pr(IBD)_1$ with Equation 4, we can derive the probability for the adversary to match the target and his/her relative sharing at least c detectable (at least length v) IBD segments, denoted as $Pr(match|g)_1$, as

$$\begin{aligned} Pr(match|g)_1 &= 1 - \sum_{k=0}^{c-1} B(k; 2Lg+22, \frac{Pr(IBD)_1}{2^{2g-2}}) \\ &= 1 - \sum_{k=0}^{c-1} B(k; 2Lg+22, (1-\delta)\frac{e^{-\frac{2vg}{1+\overline{T}/n}}}{2^{2g-2}}). \end{aligned} \tag{9}$$

Thus, the expected value of entropy bits gained by the adversary who controls $(1-\delta)$ proportion of nodes, denoted as $H'(g)_1$, can be formulated as

$$H'(g)_1 = h(g)\big(1 - \sum_{k=0}^{t-1} B\left(k; R, Pr(shared) \cdot Pr(match|g)_1\right)\big). \tag{10}$$

Analysis on IBD-targeting Fragmentation Strategy. We then consider a more realistic scenario wherein the individuals in the dataset are selected from particular families or particular areas/suburbs (such as the patient data of a hospital whose patients are usually from nearby suburbs). In this scenario, data owners can provide prior knowledge of their data (such as pedigree information and family distributions), such that vFRAG is able to identify the locations of the IBD segments and deliberately split them during the fragmentation.

With this strategy, the chance that adversary holds the full IBD segments is significantly reduced compared to the random fragmentation. Since each IBD segment is fragmented, the expected length of IBD segments held by the adversary is reduced from $(1+\frac{\overline{T}}{n})/(2g)$ to $(1-\delta)(1+\frac{\overline{T}}{n})/(2g)$. The PDF of the length of IBD segments then becomes $Pr(x)_{adversary} = \frac{2g}{(1-\delta)(1+\overline{T}/n)} e^{-\frac{2g}{(1-\delta)(1+\overline{T}/n)}x}$. Therefore, in this scenario, the probability of the adversary holding a detectable IBD segment $Pr(IBD)_2$ is

$$Pr(IBD)_2 = e^{-\frac{2vg}{(1-\delta)(1+\overline{T}/n)}}. \tag{11}$$

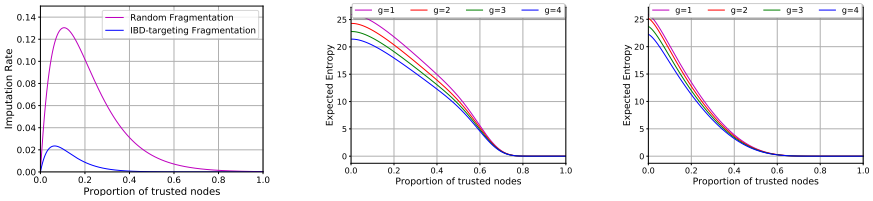
With this, the probability that two individuals share enough IBD segments to be detected, denoted as $Pr(match|g)_2$, can be calculated as

$$\begin{aligned} Pr(match|g)_2 &= 1 - \sum_{k=0}^{c-1} B(k; 2Lg + 22, \frac{Pr(IBD)_2}{2^{2g-2}}) \\ &= 1 - \sum_{k=0}^{c-1} B(k; 2Lg + 22, \frac{e^{-\frac{2vg}{(1-\delta)(1+\overline{T}/n)}}}{2^{2g-2}}). \end{aligned} \tag{12}$$

All in all, the expected entropy bits gained by the adversary in IBD-targeting fragmentation, denoted as $H'(g)_2$, can be calculated as

$$H'(g)_2 = h(g)\big(1 - \sum_{k=0}^{t-1} B\left(k; R, Pr(shared) \cdot Pr(match|g)_2\right)\big). \tag{13}$$

Quantifying Information Leakage Reduction. We first explore the capacity of genotype imputation in both random and IBD-targeting fragmentation strategies according to Inequality 7. Our quantification is based on the 1000 Genomes Project dataset (Phase 3) [2] - a widely used reference dataset for genotype imputation; we set $K = 2,504$ which is the sample size of the dataset, and set ρ to 0.85 in the random strategy and 0.5 in the IBD-targeting strategy according to Das et al [7]. As shown in Fig. 4a, the capacity of genotype imputation in the random strategy is lower than that in the IBD-targeting strategy. This is because the latter, which deliberately splits the IBD segments, further increases the genetic distance such that the upper bound of $\overline{T}$ (Inequality 7) is decreased. It also can be found that the imputation rate rises faster in the random strategy when the adversary controls $\geq 90\%$ nodes ($0 \leq \delta \leq 10\%$), with



(a) Genotype imputation in both strategies

(b) Entropy reduction in random strategy

(c) Entropy reduction in IBD-targeting strategy

Fig. 4: Information leakage reduction.

less missing genotypes in the IBD regions. However, with the further growth of trusted nodes, the imputation rate exponentially decreases in both strategies.

Next, we evaluate the privacy preservation in υFRAG. We keep the same setting of population size (329 million) and $\gamma = 2.5$. In addition, we fix the dataset size as 30% of the target population, as this gives the adversary an advantage according to Fig. 3. The genotype imputation is considered in our evaluation by incorporating the imputation rate into Equation 10 and Equation 13 respectively. As shown in Fig. 4b and 4c, with the growth of trusted nodes, the expected entropy bits achieved by the attack rapidly decreases. When the network reaches 20% honest nodes, the expected entropy bits gained by the attack reduce to 17.96 and 11.26 in random fragmentation strategy and IBD-targeting fragmentation strategy respectively. When the network reaches an honest majority ($\delta \geq 0.5$), the attack can achieve only 8.95 bits of entropy in random fragmentation, and 1.14 bits of entropy in IBD-targeting fragmentation. In other words, the uncertainty for the attack to identify an individual remains 19.25 bits ($28.2 - 8.95$) and 27.06 bits ($28.2 - 1.14$), which equals to a random guess among 623,487 people and 139.9 million people respectively.

5.2 The Individual-level Analysis

We define ϵ-*indistinguishability* to evaluate the privacy preservation from the level of indistinguishability of individual data items.

Definition 3. *(ϵ-**Indistinguishability**) Let ϵ be a positive real number and $\mathcal{G}$ be a randomized algorithm that processes dataset X and $\mathcal{D}$. The algorithm $\mathcal{G}$ is said to provide ϵ-indistinguishability if for data items $d_1, d_2 \in \mathcal{D}$ and $x \subseteq Range(\mathcal{G}(X))$*

$$Pr(\mathcal{G}(d_1) = x) \leq e^{\epsilon} Pr(\mathcal{G}(d_2) = x). \tag{14}$$

As is shown, the definition of ϵ-indistinguishability is essentially a variant of ϵ-local differential privacy (LDP), which is yet a variant model of differential privacy with added restriction to the indistinguishability of individual data items [6,10]. The difference is that, the indistinguishability in our model is measured from the *adversarial view*, i.e., the outputs space of $\mathcal{G}$ on individual data items from the two datasets accessible by the adversary (referring to fragments produced by $\mathcal{G}(X)$ and the public dataset $\mathcal{D}$), whereas the indistinguishability

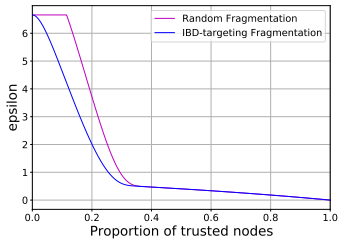


Fig. 5: ϵ-indistinguishability in both strategies

in the original LDP is measured directly on the output space of $\mathcal{G}$ on individual data items.

Let AF_i be the allele frequency of the i^{th} SNP, d denote the number of SNPs from the target that the adversary is able to access, and $\mathcal{D}$ denote the dataset comprised of R genotyped individuals. The following theorem states that our *mask operation* $\odot$ makes vFRAG satisfy ϵ-indistinguishability.

Theorem 2. *The vertical fragmentation makes* vFRAG *satisfy* ϵ*-indistinguishability where* $\epsilon = \ln((1 - \sum_{k=0}^{1} B(k, R, \prod_{i=1}^{d} AF_i))^{-1})$.

Proof. Assume the adversary is able to access $\vec{x} \in \{A, T, C, G\}^d$, which is a randomized output generated by the mask operation $\odot$ for vertical fragmentation in vFRAG. The probability of observing at least t times of $\vec{x}$ in the dataset $\mathcal{D}$ of size R is $(1 - \sum_{k=0}^{t} B(k, R, Pr(\vec{x}|\mathcal{D})))$, where $Pr(\vec{x}|\mathcal{D}) = \prod_{i=1}^{d} Pr(\vec{x_i}|\mathcal{D}) = \prod_{i=1}^{d} AF_i$. Then, $\frac{Pr[(d_1 \odot Mask) = \vec{x}]}{Pr[(d_2 \odot Mask) = \vec{x}]} = \frac{1 - \sum_{k=0}^{1} B(k, R, Pr(\vec{x}|d_1))}{1 - \sum_{k=0}^{1} B(k, R, Pr(\vec{x}|d_2))} \leq \frac{1}{1 - \sum_{k=0}^{1} B(k, R, \prod_{i=1}^{d} AF_i)}$. Thus, we have $\epsilon = \ln((1 - \sum_{k=0}^{1} B(k, R, \prod_{i=1}^{d} AF_i))^{-1})$.

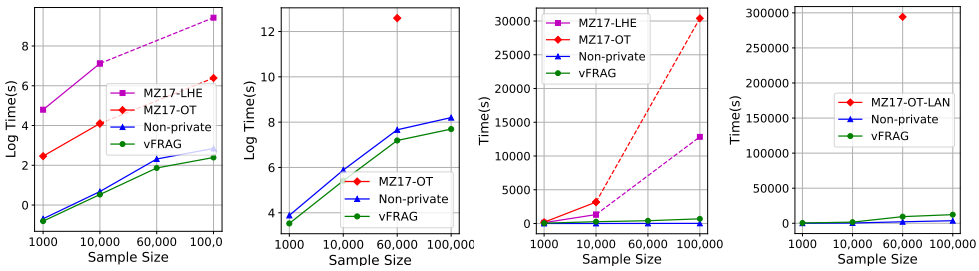
We explore the privacy level ϵ each strategy can achieve. We keep the same setting of the dataset size, i.e., 30% of the target population, and set the number of SNPs in the genome-wide analysis as 5,000. As reported in 1000 Genomes Project, the majority of variants in the human genome have a minor allele frequency $< 0.5\%$ [2], we take the upper bound of the minor allele frequency 0.5% and set the $\bar{AF} = 99.5\%$ in the evaluation. This gives the adversary more advantage as $\bar{AF}$ in the real world can be even larger than in this setting. The parameter d, which is the number of SNPs obtained by the adversary, is set after the adversary has applied genotype imputation.

Fig. 5 shows the privacy parameter ϵ against the proportion of the trusted nodes in both strategies. vFRAG can achieves ϵ=3.72 and ϵ=2.02 in the random and IBD-targeting strategies respectively, when the adversary controls 80% of

the nodes ($\delta = 0.2$). When the network reaches an honest majority, ϵ is reduced to 0.406 and 0.405 respectively. As the definition of ϵ-indistinguishability is adopted from LDP, we demonstrate a comparable privacy gain with that of the state-of-the-art systems achieving LDP, such as Apple's DP framework with $\epsilon = \{2, 4, 8\}$ [6] and Google's RAPPOR with $\epsilon = ln(3)$ [10].

6 Performance Evaluation

We implement vFRAG in C++. It uses the Eigen library [11] to handle matrix operations, and ZeroMQ library [15] for distributed messaging. Our experiments use the 1000 Genomes Project dataset [2], which is a public dataset providing a comprehensive description of human genetic variation. The experiments are conducted on 62,042 SNPs on Y-chromosome from 1,233 male samples.



(a) LR in LAN (b) NN in LAN (c) LR in WAN (d) NN in WAN

Fig. 6: Efficiency comparison with the baseline and MPC/HE solutions.

To be representative, vFRAG is executed on both LAN and WAN network settings. The LAN setting captures the scenario where two or more institutions collaboratively execute computations on their own private inputs. In such scenario, the involved parties usually communicate over fast dedicated links. In our experiments, the average network bandwidth is set as 1GB/s. The WAN setting, on the other hand, simulates a scenario where individual participants share their genomic data over public network infrastructure. The average network latency (one-way) is set as 183.19ms, and the average throughput is 8.75MB/s.

We take the performance of traditional non-privacy-preserving frameworks as the baseline, given that it is the most common practice in existing genome-wide analysis. We also show the performance of other existing privacy-preserving mechanisms, including cryptographic based methods (MZ17 [22]) and differential privacy mechanisms (FDML [16]). MZ17 considers MPC/HE solutions for privacy preserving machine learning algorithms based on oblivious transfer (OT) and linearly homomorphic encryption (LHE), and FDML is a state-of-the-art

Table 1: Comparison with DP solution ($m = 32661, n = 124$).

	FDML	υFRAG-LAN
Linear Regression	99s	0.76s
Neural Network	110s	139.53s

Table 2: Overhead breakdown of linear regression and neural network ($n = 784$).

	Linear Regression			Neural Network		
	Computation	Communication		Computation	Communication	
		LAN	WAN		LAN	WAN
m=1,000	0.4s	0.0085s	44.9s	37.6s	3.8s	395.2s
m=10,000	1.2s	0.5s	264.4s	225.9s	1.6s	1395.2s
m=60,000	5.7s	0.8s	402.9s	1312.7s	15.1s	8143.9s
m=100,000	9.5s	1.5s	681.3s	2167.9s	23.2s	10140.6s

additive noise based DP framework which also employs in the scenario of distributed features, the same as υFRAG.

Fig. 6 and Table 1 summarize our efficiency comparison with the baseline, MZ17, and FDML respectively. Table 2 lists the overhead breakdown to computation and communication. Generally, υFRAG significantly outperforms the MPC/HE solutions, and also outperforms the baseline and the state-of-the-art DP solution in most settings.

LR. We follow the same experimental settings as [22] (including machine specifications, network settings, and batch/epoch sizes). As shown in Fig. 6, υFRAG significantly outperforms MZ17, and even faster than the baseline in the LAN setting. For example, υFRAG takes 10.93s with sample size $m = 100,000$ (Fig. 6a), while it takes $594.95s$ in MZ17-OT and $17.21s$ in the baseline. In the WAN setting (Fig. 6c), υFRAG achieves 690.74s with $m = 100,000$. It is 19x faster than the MPC solutions in MZ17-LHE, which takes 12,841.2s with the same sample size. υFRAG also outperforms FDML (Table 1), due to the significant savings on the computational cost compared with the additive noise mechanism.

NN. We implement a fully connected NN with a sigmoid function as the activation function. It has the same structure as that in MZ17. In the LAN setting (Fig. 6b), υFRAG achieves 1,327.72s (22.1 minutes) with sample size $m = 60,000$, while MZ17-OT takes 294,239.7s (more than 81 hours)[5]. In the WAN setting (Fig. 6d), our framework still remains practical. It achieves 9,456.53s with sample size $m = 60,000$. In contrast, it is not yet practical for MZ17 to train NN in the WAN setting due to the massiveness of interaction and communication. We also plot the MZ17-OT-LAN result (294,239.7s) in Fig. 6d. It can be found that even when running our framework in the WAN setting, υFRAG is still much more efficient than the MPC solutions in MZ17 running on the LAN setting.

In comparison with the DP solution, υFRAG NN achieves similar performance with FDML (Table 1). We note that FDML NN only considers a fully connected NN *within* each worker node while merging the local predictions in a composite

[5] Only the performance with $m = 60,000$ in the LAN setting is reported in [22].

model, whereas vFRAG uses a fully connected NN over *all* the features, thus leading to a more complex model.

Overhead analysis. Computation overhead of vFRAG is in line with its complexity of $\mathcal{O}(d_l m)$ (where d_l and m are the number of SNPs and samples in X^l respectively), and communication cost is in linear with $\mathcal{O}(d_l m s)$ (where s is the number of local nodes). More nodes lead to a smaller d_l but a larger s. The memory consumption of vFRAG, same as its centralized counterpart, is only related to the size of X and weight W.

7 Discussion

Our privacy analysis (Section 5) is conducted under a trustworthy aggregator. In this section, we explore the impact of a compromised aggregator.

The aggregator in vFRAG has access to only the intermediate results but none of any local data. According to the derivation in our previous work [37] (refer to its Theorem 1), the overall knowledge of the aggregator can obtain is $\{X^l W^l, X^l (X^l)^T\}$, where $X^l W^l$ is the aggregator's own input, and $X^l (X^l)^T$ is the additional information that can be inferred from the training iterations. We then quantify the impact of colluding parties including the compromised aggregator. Given the knowledge of $\{X^l W^l, X^l (X^l)^T\}$, the probability of computing the original input X^l or W^l, is not greater than $1/(r!)$, where r is the rank of X^l (refer to Theorem 2 of [37]). Since X^l is a matrix of m samples with d_l SNPs, the rank r of X^l is the number of unique DNA sequence in X^l. We can formulate the probability of the number of duplicated sequence being no more than K, i.e., the rank $r > (d_l - K)$, as $Pr(r > (d_l - K)) = \sum_{k=0}^{K} B(k; d_l, (1 - \bar{AF})^{d_l})$, where $\bar{AF}$ denotes the expected allele frequency, and $(1 - \bar{AF})^{d_l}$ is the probability of a DNA sequence being identical to the reference genome.

With it, we can estimate the probability of the attacker deriving X^l or W^l, given d_l. We let $\bar{AF} = 0.00397$ as reported in the 1000 Genomes Project dataset [2]. When d_l exceeds 120, which is common in real-world genome-wide analysis datasets, the probability of $r > 32$ (i.e., the probability of deriving X^l or W^l being less than $1/(32!)$) is 0.9965. This implies that it is unlikely for the compromised aggregator to derive the original data of honest parties.

8 Related Work

In this section, we summarize existing privacy-preserving techniques which can be divided into the following categories.

Cryptographic Solutions. Cryptographic solutions, such as HE and MPC [5, 17,22,38], enable computation without disclosing data in plaintext. Several studies have been conducted to enable multiple entities to train machine learning models with privacy preservation over the input data. For example,Wan et al. [13] proposes a MPC-based solution for privacy-preserving gradient descent. In [36], a secure protocol is presented to calculate the delta function in the back-propagation training.

Differential Privacy. DP is another methodology that constitutes a strong standard for privacy guarantees for algorithms on aggregate databases [3,16,39]. Several studies have explored the differentially private release of common summary statistics of GWAS data (such as the allele frequencies of cases and controls, χ^2-statistic and P values [35,40]) or shifting the original locations of variants [19]. Recently, a study proposes a novel differential privacy mechanism named SVT^2 for mitigating membership inference attacks against DNA methylation data [12]. Recently, Hartmann et al. [14] proposes a noise-based DP framework which provides differential privacy with very small noise addition by adding and canceling noise among clients.

Distributed Deep Learning Framework. Our work is related to but different from distributed deep learning frameworks [33]. Existing work focuses either on the reduction of the training time of deep neural network models [29], or on the theoretical convergence speed in the distributed computing environment [20].

9 Conclusion

In this paper, we presented υFRAG, a privacy preserving framework for distributed genome-wide analysis. υFRAG mitigates privacy risks by using a vertical DNA sequence fragmentation to disrupt the genetic architecture on which the adversary relies for re-identification. We demonstrated the privacy preservation of υFRAG from both collection level (overall information leakage reduction) and individual level (indistinguishability among individuals). Our experiments on large-scale datasets showed that υFRAG outperforms state-of-the-art cryptography-based with a speedup of more than 221x for training neural networks. Our work sheds a light on the privacy preservation in genome-wide analysis. For sequential data (like DNA sequences), disrupting the order and location dependency could be a promising solution.

Acknowledgment

We thank our shepherd Erman Ayday and the anonymous reviewers for their insightful comments to improve this manuscript. This work is partly supported by the University of Queensland under the UQ Cyber Initiative Strategic Research Seed Funding 4018264-01-299-21-618071.

References

1. vFrag, https://sites.google.com/view/vfrag
2. 1000 Genomes Project Consortium: A global reference for human genetic variation. Nature **526**(7571), 68 (2015)
3. Abadi, M., Chu, A., Goodfellow, I., McMahan, H.B., Mironov, I., Talwar, K., Zhang, L.: Deep learning with differential privacy. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. pp. 308–318. ACM (2016)

4. Angermueller, C., Pärnamaa, T., Parts, L., Stegle, O.: Deep learning for computational biology. Molecular systems biology **12**(7) (2016)
5. Bogdanov, D., Kamm, L., Laur, S., Sokk, V.: Rmind: a tool for cryptographically secure statistical analysis. IEEE Transactions on Dependable and Secure Computing (2016)
6. Cormode, G., Jha, S., Kulkarni, T., Li, N., Srivastava, D., Wang, T.: Privacy at scale: Local differential privacy in practice. In: Proceedings of the 2018 International Conference on Management of Data. pp. 1655–1658 (2018)
7. Das, S., Forer, L., Schönherr, S., Sidore, C., Locke, A.E., Kwong, A., Vrieze, S.I., Chew, E.Y., Levy, S., McGue, M., et al.: Next-generation genotype imputation service and methods. Nature genetics **48**(10), 1284–1287 (2016)
8. Erlich, Y., Narayanan, A.: Routes for breaching and protecting genetic privacy. Nature Reviews Genetics **15**(6), 409 (2014)
9. Erlich, Y., Shor, T., Pe'er, I., Carmi, S.: Identity inference of genomic data using long-range familial searches. Science **362**(6415), 690–694 (2018)
10. Erlingsson, Ú., Pihur, V., Korolova, A.: Rappor: Randomized aggregatable privacy-preserving ordinal response. In: Proceedings of the 2014 ACM SIGSAC conference on computer and communications security. pp. 1054–1067. ACM (2014)
11. Guennebaud, G., Jacob, B., et al.: Eigen v3. http://eigen.tuxfamily.org (2010)
12. Hagestedt, I., Zhang, Y., Humbert, M., Berrang, P., Tang, H., Wang, X., Backes, M.: Mbeacon: Privacy-preserving beacons for dna methylation data. In: NDSS (2019)
13. Han, S., Ng, W.K., Wan, L., Lee, V.C.: Privacy-preserving gradient-descent methods. IEEE Transactions on Knowledge and Data Engineering **22**(6), 884–899 (2009)
14. Hartmann, V., West, R.: Privacy-preserving distributed learning with secret gradient descent. arXiv preprint arXiv:1906.11993 (2019)
15. Hintjens, P.: ZeroMQ: messaging for many applications. " O'Reilly Media, Inc." (2013)
16. Hu, Y., Niu, D., Yang, J., Zhou, S.: Fdml: A collaborative machine learning framework for distributed features. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 2232–2240 (2019)
17. Jagadeesh, K.A., Wu, D.J., Birgmeier, J.A., Boneh, D., Bejerano, G.: Deriving genomic diagnoses without revealing patient genomes. Science **357**(6352), 692–695 (2017)
18. Jia, J., Salem, A., Backes, M., Zhang, Y., Gong, N.Z.: Memguard: Defending against black-box membership inference attacks via adversarial examples. In: Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. pp. 259–274 (2019)
19. 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. pp. 1079–1087. ACM (2013)
20. Lian, X., Huang, Y., Li, Y., Liu, J.: Asynchronous parallel stochastic gradient for nonconvex optimization. In: Advances in Neural Information Processing Systems. pp. 2737–2745 (2015)
21. Marees, A.T., de Kluiver, H., Stringer, S., Vorspan, F., Curis, E., Marie-Claire, C., Derks, E.M.: A tutorial on conducting genome-wide association studies: Quality control and statistical analysis. International journal of methods in psychiatric research **27**(2), e1608 (2018)
22. Mohassel, P., Zhang, Y.: SecureML: A system for scalable privacy-preserving machine learning. In: 2017 IEEE Symposium on Security and Privacy (SP). pp. 19–38. IEEE (2017)

23. Ralph, P., Coop, G.: The geography of recent genetic ancestry across europe. PLoS biology **11**(5), e1001555 (2013)
24. Regalado, A.: MIT technology review., https://www.technologyreview.com/2019/02/11/103446/more-than-26-million-people-have-taken-an-at-home-ancestry-test/
25. Salem, A., Zhang, Y., Humbert, M., Berrang, P., Fritz, M., Backes, M.: Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models. arXiv preprint arXiv:1806.01246 (2018)
26. Timpson, N.J., Greenwood, C.M., Soranzo, N., Lawson, D.J., Richards, J.B.: Genetic architecture: the shape of the genetic contribution to human traits and disease. Nature Reviews Genetics **19**(2), 110 (2018)
27. Visscher, P.M., Wray, N.R., Zhang, Q., Sklar, P., McCarthy, M.I., Brown, M.A., Yang, J.: 10 years of GWAS discovery: biology, function, and translation. The American Journal of Human Genetics **101**(1), 5–22 (2017)
28. Wang, K., Zhang, J., Bai, G., Ko, R., Dong, J.S.: It's not just the site, it's the contents: Intra-domain fingerprinting social media websites through cdn bursts. In: Proceedings of the Web Conference 2021. pp. 2142–2153 (2021)
29. Wang, S., Pi, A., Zhou, X.: Scalable distributed DL training: Batching communication and computation. In: Proc. of AAAI (2019)
30. Wang, S., Zhang, Y., Dai, W., Lauter, K., Kim, M., Tang, Y., Xiong, H., Jiang, X.: HEALER: Homomorphic computation of exact logistic regression for secure rare disease variants analysis in GWAS. Bioinformatics **32**(2), 211–218 (2015)
31. Wang, Y., Huang, Z., Mitra, S., Dullerud, G.E.: Differential privacy in linear distributed control systems: Entropy minimizing mechanisms and performance tradeoffs. IEEE Transactions on Control of Network Systems **4**(1), 118–130 (2017)
32. Wang, Y.X., Lei, J., Fienberg, S.E.: On-average KL-privacy and its equivalence to generalization for max-entropy mechanisms. In: International Conference on Privacy in Statistical Databases. pp. 121–134. Springer (2016)
33. Xing, E.P., Ho, Q., Xie, P., Wei, D.: Strategies and principles of distributed machine learning on big data. Engineering **2**(2), 179–195 (2016)
34. Yang, J., Lee, S.H., Goddard, M.E., Visscher, P.M.: Gcta: a tool for genome-wide complex trait analysis. The American Journal of Human Genetics **88**(1), 76–82 (2011)
35. Yu, F., Fienberg, S.E., Slavković, A.B., Uhler, C.: Scalable privacy-preserving data sharing methodology for genome-wide association studies. Journal of biomedical informatics **50**, 133–141 (2014)
36. Yuan, J., Yu, S.: Privacy preserving back-propagation neural network learning made practical with cloud computing. IEEE Transactions on Parallel and Distributed Systems **25**(1), 212–221 (2014)
37. Zhang, Y., Bai, G., Li, X., Curtis, C., Chen, C., Ko, R.K.: PrivColl: Practical privacy-preserving collaborative machine learning. In: European Symposium on Research in Computer Security. pp. 399–418. Springer (2020)
38. Zhang, Y., Bai, G., Li, X., Nepal, S., Ko, R.K.: Confined gradient descent: Privacy-preserving optimization for federated learning. arXiv preprint arXiv:2104.13050 (2021)
39. Zhang, Y., Bai, G., Zhong, M., Li, X., Ko, R.: Differentially private collaborative coupling learning for recommender systems. IEEE Intelligent Systems (2020)
40. Zhang, Y., Zhao, X., Li, X., Zhong, M., Curtis, C., Chen, C.: Enabling privacy-preserving sharing of genomic data for GWASs in decentralized networks. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. pp. 204–212. ACM (2019)

Appendix A Notation Table

Table 3 summarizes the notations defined in this paper.

Table 3: Notation Table

Notation	Domain	Explanation	Notation	Domain	Explanation
SNP	/	Single-nucleotide polymorphism.	γ	$\mathbb{R}$	Mean value of children per couple.
X	$\mathbb{S}^{m\times n}$	Centralized training dataset.	ρ	$\mathbb{R}$	Transition probability in HMM.
W	$\mathbb{R}^{n\times H}$	Centralized coefficient matrix.	N	$\mathbb{Z}$	Size of a general population..
x_{ij}	$\mathbb{S}$	Genotype of j^{th} SNP of i^{th} sample.	$\overline{T}$	$\mathbb{R}$	Expected value of imputation accuracy.
X^l	$\mathbb{S}^{m\times d_l}$	Vertical partition of X.	τ	$\mathbb{Z}$	The number of missing genotypes.
W^l	$\mathbb{R}^{d_l\times H}$	Coefficient matrix associated with X^l.	R	$\mathbb{Z}$	Size of the public dataset.
S^l, A	$\mathbb{Z}$	Worker node and aggregation node.	K	$\mathbb{Z}$	Size of the reference haplotypes panel.
δ	$\mathbb{R}$	Proportion of trusted worker nodes.	$(M)AF$	$\mathbb{R}$	(Minor) allele frequency.
g	$\mathbb{Z}$	Degree of genetic relatives.	ϵ	$\mathbb{R}$	Privacy parameter.
J, σ	/	Cost function and hypothesis function.	$\odot$	/	*Mask* operation.
Δ	$\mathbb{R}^{m\times H}$	Gradient of J w.r.t XW.	$B()$	/	PMF of binomial distribution.

Appendix B Functionalities in genome-wide analysis

In this section, we briefly introduce the functionalities commonly used in the analysis.

Summary statistics. Summary statistics are used to summarize the observations on the genome-wide data. Commonly used summary statistics include the missingness statistics ($U_{i,miss}/n$, where $U_{i,miss}$ is the number of missing SNPs of i^{th} sample), allele frequency ($c/2m$, where c is the total number of allele for each SNP), and Hardy-Weinberg equilibrium ($\{(p^2+2pq+q^2==1)\}$, where p^2 is the frequency of homozygous dominant genotype, pq is the frequency of heterozygous genotype, and q^2 is the frequency of homozygous recessive genotype) [21].

Basic association analysis. The basic association analysis for GWAS checks on any particular SNP. If one type of the variant (i.e., one allele) is more frequent in individuals with a disease, the variant is said to be associated with the disease. Commonly used statistics include standard χ^2 test and the Cochran-Armitage test, which performs the tests with respect to each SNP.

Genetic relationship matrix (GRM). GRM is developed for addressing the *missing heritability* problem by estimating the variance explained by all the SNPs on a chromosome or on the whole genome for a complex trait [34]. The genetic relationship between individuals β and ζ can be estimated by $\frac{1}{n}\sum_{i=1}^{n}\frac{(x_{\beta i}-2p_i)(x_{\zeta i}-2p_i)}{2p_i(1-p_i)}$, where $x_{\beta i}$ is the genotype of i^{th} SNP of β^{th} individual, and p_i is the frequency of the reference allele.

Classification models such as neural networks. Machine/deep learning algorithms, such as various NNs, are commonly used in genome-wide analysis. For example, they can be used to fit the effects of all the SNPs as random effects to estimate the total amount of phenotypic variance [34], or applied in genotype clustering and ethnicity prediction [4].

The former three functionalities are relatively simple to parallelize than machine learning algorithms, as the statistics with respect to each SNP can directly apply on the vertically partitioned dataset. Therefore, in this work, we focused on the latter.