A Privacy-Preserving GWAS Computation with Homomorphic Encryption

Ugwuoke, Chibuike; Erkin, Zekeriya; Lagendijk, Inald

Publication date
2016

Document Version
Final published version

Published in
37th WIC Symposium on Information Theory in the Benelux / 6th WIC/IEEE SP Symposium on Information Theory and Signal Processing in the Benelux

Citation (APA)
Ugwuoke, C., Erkin, Z., & Lagendijk, I. (2016). A Privacy-Preserving GWAS Computation with Homomorphic Encryption. In *37th WIC Symposium on Information Theory in the Benelux / 6th WIC/IEEE SP Symposium on Information Theory and Signal Processing in the Benelux* (pp. 166-173)

A Privacy-Preserving GWAS Computation with Homomorphic Encryption.

Chibuike Ugwuoke Zekeriya Erkin Reginald L. Lagendijk
Delft University of Technology
Department of Intelligent Systems, Cyber Security Group.
Delft, The Netherlands
C.I.Ugwuoke@tudelft.nl Z.Erkin@tudelft.nl R.L.Lagendijk@tudelft.nl

Abstract

The continuous decline in the cost of DNA sequencing has contributed both positive and negative feelings in the academia and research community. It has now become possible to harvest large amounts of genetic data, which researches believe their study will help improve preventive and personalised healthcare, better understanding of diseases and response to treatments. However, there are more information embedded in genes than are currently understood, just as a genomic data contains information of not just the owner, but relatives who might not subscribe to sharing them. Unrestricted access to genomic data can be privacy invasive, hence the urgent need to regulate access to them and develop protocols that would allow privacy-preserving techniques in both computations and analysis that involve these very sensitive data. In this work, we discuss how a careful combination of cryptographic primitives such as homomorphic encryption, can be used to privately implement common algorithms peculiar to genome-wide association studies (GWAS). This obviously comes at a cost, where we have to accommodate the trade-off between speed of computations and privacy.

1 Introduction

Biomedical research has long shown that human genome contain data from which information about their individual owners, and those related to them can be extracted [1, 2, 3, 4]. A lot of privacy-sensitive information are laced all over genomic data, which constitutes enormous worry for individuals whose data are available in electronic format [5, 6, 7]. The benefits of continuous research involving the genomic data are equally rife, these include: preventive and personalised healthcare, patient's response to treatment, predisposition to diseases, identification of new drug targets and perhaps a better understanding of cancer [1, 8, 4, 9, 10, 11, 12, 13, 14]. On the other hand, when genomic data is used for research or processed by medical personnels, they become exposed to possible misuse and even loss to unauthorised hands. In the face of this possibility, the risk of re-identifying individuals from an available genomic data calls for serious concern [5, 9, 15, 4, 3, 7], and has been recognised as a realistic threat. Other unwanted scenarios which could occur as direct consequence of leaking genomic data include: stigmatisation, discrimination, loss of insurance and even loss of employment opportunities for persons whose genomic data is public [16].

What is more worrisome about misuse of genomic data is the fact that the genome has longevity, when leaked, it can neither be revoked nor modified. So, it is obvious that this piece of data is highly sensitive and requires protection that should be adaptive to future security threats. Hence one can claim that any realistic solution should be one which, the security guarantees of the underlying primitives used for implementation

should withstand post quantum attacks. Therefore privacy protection techniques have been proposed as an adaptive solution by the cryptography community. The aim will be to allow productive research that utilise genomic data, while eliminating the privacy-risks inherent around these procedures.

Being that no standalone solution can best fit the challenge posed, it is considered that a good combination of *ethical, legal* and *technological* constraints can be employed, to properly manage the risks of privacy leaks that are otherwise possible within this research domain. Owing to this premise, our work seeks to contribute a technological solution to the underlying problem.

In the era of distributed computing, even the medical field has not been left out. It has been common for researchers and medical personnels to work without boundaries of country borders, albeit, via a virtual collaboration [17, 7, 2]. This means that more data can now be shared for research purposes and even diagnosis of diseases [6]. It also presents us with the possibility of allowing cloud services process medical data, even when they do not reside in the same country as the owners of the data. This need for collaboration, data sharing and cloud processing of genomic data further pushes for privacy-preserving secure computing protocols [2, 17].

Having a genomic dataset and controlling access to it is the main aim of this work. In a nutshell, this means that while these data is not available to the public, experts who need them for research are granted restricted access to only subsets relevant to their work [15]. Such access for processing data may include string searching and comparison, as well as GWAS computations.

Genome Wide Association Studies: As highlighted in [1, 18], the first ever human genome sequencing was achievable in 2001, after directly gulping a whooping US-$300 million from the initial budget of US-$3 billion. Fast-forward 6 years later, and the same feat is feasible for about US-$100,000. In 2006 [18], it was anticipated that in 2014, a further reduction to US-$1,000 was possible for sequencing the human genome. Recent literature [19, 11] have even suggested that a meagre US-$100, will be a reality in the very near future. If that be the case, one can deduce that amongst other possibilities, a direct consequences of affordable genomic data would be the torrential flow of genomic data *in silico*. It is obviously a good development for researchers, who would heavily rely on these data to improve on their research, refine and optimise diagnosis and many others positive possibilites. With a wealth of data in the form of genomic data lying at the disposal of researchers and medical personnels, learning and inferring from these data becomes an indisputable objective.

Without loss of generality in description, GWAS can simply be simplified to the activities presented above, it is about gathering genetic data, processing them and relying on them to investigate relationship (association) of genes to common known diseases. It will be possible to even detect unknown diseases and the effect of drugs on treatments. With GWAS researchers can now measure, analyse and predict previously unknown genetic influence on a person, this can help in early detection and prevention of certain diseases, as well as personalised healthcare. For useful gene-disease associations to be estimated, some computations become handy, and these will be discussed in subsection 2.1. Nonetheless, most of the computations can easily put the data owners at privacy-risk. It has led to the suggestion that protection of genomic data is a necessity, to address possible ethical, political, technological and privacy concerns. From the technological solution approach, we hope to address the privacy-threats using cryptographic primitives. Just to mention, with genomic data, data anonymization is not enough guarantee to avoid re-identification and also, conventional encryption might not offer much better protection against envisaged privacy-threats. These can simply be derived from the fact that the said data have longevity, their importance persists even after the demise of the data owner.

Related Works: Realising the privacy-sensitive nature of genomic data, researchers

have delved into search for privacy-preserving solutions, in the hope to protect privacy of owners while still being able to process and compute operations using these data. Some of these works are discussed here. Privacy-preserving GWAS spans across more possibilities than just GWAS-Computations. According to [15], other important categories include:

- Private string searching and comparison.
- Private release of aggregated data.
- Private read mapping.

of course, this list is not in itself exhaustive, but we will only consider works that directly address computations very peculiar to GWAS. As early as 1999 [20, 21, 22], some researchers had anticipated privacy risks involved with genomic data. So they proposed denominalization and de-identification as protection schemes, to preserve privacy. This did not stop re-identification attacks from being hugely successful, as discussed in [9]. Other authors [23] have subsequently recommended *Trusted Third Parties* and *Semitrusted Third Parties* but then, it is not always easy to completely trust a third party, who could still be susceptible to coercion, compulsion and even corruption to be compromised. More recently in [24], attempts were made to analyse genomic data while avoiding privacy-invasion of participants of the data. Summarily, they adopted differential privacy as a privacy-preserving technique, and documented to have obtained utility with their procedure. However, addition of noise using differential privacy is not a silver bullet to deflate possible re-identification. Especially when the published data can be augmented with other side information. But most importantly is the fact that differential privacy contains noise, which will evidently affect the utility, no matter the degree of noise. This is a huge trade-off, but it is only left for the geneticists and biostatisticians to decide if the noise only contributes a negligible disturbance to the final results.

While the last paper approach to resolving possible privacy breaches is via differential privacy, [25] chooses to adopt a different approach. The authors adopt homomorphic encryption as a tool to enable analysis of these privacy sensitive data. Homomorphic Encryption holds a lot of promises, and if its capabilities are optimally harnessed, can become a very productive primitive in guaranteeing privacy for processing genomic data. In this work, different scenarios are considered which include a setting that allows outsourcing encrypted genomic data to a cloud service. In the mentioned scenario, operations on the data by the cloud are still possible, without divulging the decryption keys but still hopeful of achieving utility.

Homomorphic Encryption was further relied on by some other team of researchers [26]. A shot was given to providing privacy guarantees on processing of genomic data, only that this time the focus was on homomorphic encryption scheme whose structures rely on RLWE (Ring Learning With Error). [26] documents an efficiency-improvement from existing implementation of GWAS using homomorphic encryption. They showed that χ^2 test for independence was achievable with improvement in both computation and communication time from existing implementations.

Subsequently, another team of researchers went further to demonstrate how much information can be extracted from computation of genomic data, even on the encrypted domain [27]. Basic genomic algorithms which are common to GWAS are shown to be implementable on encrypted genotype and phenotype data. Lauter et al. [27] report results that preserve utility of the original implementation (computation on unencrypted genomic data). Some of the algorithms demonstrated in their work include:

- Estimation Maximization (EM) algorithm for haplotyping.
- The $D^{'}$ and r^2-measures of linkage disequilibrium.

- Cochran-Armitage Test for Trend.

Also worth mentioning is the fact that this implementation relied on Homomorphic Encryption with assumption on RLWE.
Scenario and Assumptions: For the sake of this work, we will explicitly spell out the scenario in which our proposed protocol is targeted, and necessary assumptions. Our setting adopts the semi-honest security model, hence we assume that all parties will correctly follow the protocol by performing the right computations, but with a curiosity to observe the transitions of the protocol with a view to learning more details than they are statutorily allowed to learn. We assume that a researcher *Alice* is interested in a particular computation, say *Minor Allele Frequency (MAF)*. The data source or cloud *Bob*, who happens to have the computational powers not acquired by *Alice*, is trusted to perform all requests by performing the computation on encrypted data. The result of the computation (which however, is also encrypted), is returned to *Alice*.

2 Preliminaries

Up until here, we have established a clear direction to the challenge we hope to address. A genomic dataset is at our disposal and we intend to preserve privacy of data in the face of effective computations. So, we propose a protocol that encrypts all genomic data and outsources storage of these data to a semi-honest cloud service who possesses the computational requirements to run these expensive computations. It will be pertinent to have a mental picture of typical algorithms that will be deployed to perform computation, and how our cryptographic privacy enhancing technology optimally fits for a solution. Most of the algorithms are statistical operations that are often required by biostatisticians when trying to learn information from a dataset. And just like most statistical equations require simple arithmetic operation at the least, we show that our adopted primitive (homomorphic encryption), does provide us with the capabilities to perform simple *addition*, *multiplication*, and with a little more effort *division*.

2.1 GWAS Computation

Only a few statistical computations that are usually handy in GWAS are presented.

Minor Allele Frequency: Finding the ratio for which an allele of interest that is at a locus, occurs in a particular population of study is the allele frequency. *MAF* is therefore the allele frequency of the least common *allele*, which appears in that population. If we have a gene with two possible alleles say **A** and **S**, then in a monoploid gene setting, the allele frequency *f()* for **A** is simply computed as follow:

$$f(\mathbf{A}) = \frac{\sum_1^n \mathbf{AA}}{\sum_1^n \mathbf{AA} + \sum_1^m \mathbf{SS}} \tag{1}$$

where $N = n + m$ is the total population sample, and n and m are the counts of alleles **A** and **S** respectively. That was rather too easy, owing to the fact that we only have two possible genotypes, which are results of pure combination of possible alleles. What happens when we consider diploid gene settings? Using the same alleles at a particular locus, we consider the following expressions: **AA**, **AS** and **SS**. Just like we did above, we shall try to compute the frequency of the allele **A**. Let genotype distribution be as follows: $\mathbf{A} = 19, \mathbf{AS} = 21, \mathbf{SS} = 07$.

$$f(\mathbf{A}) = \frac{2 * \sum_1^n \mathbf{AA} + \sum_1^k \mathbf{AS}}{2(\sum_1^n \mathbf{AA} + \sum_1^k \mathbf{AS} + \sum_1^m \mathbf{SS})} \tag{2}$$

The total genotype count in this case is $N = n + k + m$, where n, k and m are counts for **AA**, **AS** and **SS** respectively. to compute the allele frequency of **A** using the values already

presented, we will have

$$f(\mathbf{A}) = \frac{2 * \mathbf{AA} + \mathbf{AS}}{2 * (\mathbf{AA} + \mathbf{AS} + \mathbf{SS})} = \frac{2 * 19 + 21}{2 * (19 + 21 + 07)} = \frac{59}{94} = 0.6277 \tag{3}$$

Since we only have two possible alleles in this population, the least common allele should be **S**, with MAF of $(1 - 0.6277) = 0.3723$

To calculate the genotype frequencies we have $\mathbf{AA} = \frac{n}{N}$, $\mathbf{AS} = \frac{k}{N}$, $\mathbf{SS} = \frac{m}{N}$

Linkage Disequilibrium: This is the non-random association of alleles at different loci. Unlike the single locus alleles considered previously, we will be considering two loci but mainly retaining the basic statistics we have developed thus far. The aim of this test is to suggest if SNPs at particular loci of interest behave or occur in such a manner that is not believed to be random. So we present two loci with the following alleles: **A**, **a** and **S**, **s**. When two genotype at different loci are independent of each other, Linkage Equilibrium is considered to have occurred. Simply put, this means that Linkage Disequilibrium happens when there is some degree of dependency between the two loci of interest. Leading to the Hardy-Weinberg Equilibrium (HWE), which is said to hold if allele frequencies are preserved in a population across generations, except otherwise altered by an external factor, including evolutional influences. To measure linkage disequilibrium, the following equations are used to compute $D^{'}$ and r^2.

$$D^{'} = \begin{cases} \frac{f(AS)f(as) - f(As)f(aS)}{min(f(A)f(s), f(a)f(S))} & \text{if } f(AS)f(aa) - f(As)f(aS) > 0 \\ \\ \frac{f(AS)f(as) - f(As)f(aS)}{min(f(A)f(S), f(a)f(s))} & \text{if } f(AS)f(aa) - f(As)f(aS) < 0 \end{cases} \tag{4}$$

$$r^2 = \frac{(f(AS)f(as) - f(As)f(aS))^2}{f(A)f(S)f(a)f(s)} \tag{5}$$

On the assumption that the allele frequencies can be obtained from encrypted genomic data, then it follows that the above computations can be computed.

2.2 Homomorphic Encryption

Homomorphic Encryption (HE) is a cryptographic primitive that allows for simple arithmetic operations over a ciphertext space. A HE scheme can either allow for simple addition, multiplication or even both. We have an additive scheme if it can only allow for addition operations and a *fully homomorphic encryption* (FHE) scheme if both addition and multiplication can be harnessed from the scheme. Give two messages m_1 and m_2, an encryption and decryption functions $Enc()$ and $Dec()$ respectively. We have that:

$$Enc(m_1) \oplus Enc(m_2) \rightarrow Enc(m_1 + m_2) : Dec(Enc(m_1 + m_2)) := (m_1 + m_2) \tag{6}$$

$$Enc(m_1) \otimes Enc(m_2) \rightarrow Enc(m_1 * m_2) : Dec(Enc(m_1 * m_2)) := (m_1 * m_2) \tag{7}$$

In 2009, [28] proposed an FHE scheme, which reduced its security to some well known difficult lattice problem. Further works were done to improve the original scheme, due to the complexity involved in implementation. Bringing about works like [29, 30, 31], which have been able to present a levelled homomorphic encryption scheme, that is capable of handling multiplication to a certain degree or depth, before the ciphertext becomes un-decryptable. The main idea is that for every operation, some noise is added to the ciphertext, and when this noise grows above a certain threshold, decryption of the ciphertext becomes a problem. While *addition* contributes small degree of noise, *multiplication* allows the noise to grow very fast. These schemes often reduce their security to lattice based problems like *shortest vector problem* (SVP) including *ring learning with error problems* (RLWE). Because the

multiplication function obtainable from these homomorphic encryptions are not arbitrary (as to control the noise growth), it is labelled *levelled* or *somewhat homomorphic encryption* (SHE). To show that the multiplication depth can only go as deep as the specified level, during parameter setup.

With a SHE scheme handy, and statistical algorithms available, we can then deploy this primitive to solve the arithmetic operations we identified earlier. It can be demonstrated that with SHE, these algorithms can be computed while preserving the utility and not trading privacy of the genomic data concerned.

3 Privacy Preserving χ^2 Statistic

In GWAS computation, X^2 is often computed and compared to the χ^2 distribution. A common test can be applied to know if the HWE holds in a given distribution. An example of a computation is presented below:

$$X^2 = \sum_{i=\{AA,AS,SS\}} \frac{(O_i - E_i)^2}{E_i} \tag{8}$$

O_i and E_i represent observed frequency allele and Expected frequency allele of the population. Since the frequency allele can easily be computed by simple addition and multiplication, and the required arithmetic operations are obtainable in our discussed Homomorphic Encryption. It can be concluded that the χ^2 statistics can be computed in a privacy-preserving manner, over encrypted datasets. Other computations such as the Cochran-Armitage Test for Trend can equally be computed using this procedure, and even meta-analysis of data from different experiments can be produced as well. For simplicity, we shall show how X^2 test statistic can be computed, borrowing the suggestions in [27, 26, 32], with a subtle modification. Every SNP representation is assumed to belong to a genotype classification. And for a single locus test, we produce 3 encryptions, $Enc(x)_{c,d} : x \in \{0, 1\}$, c and d are row and column indexes respectively. The rows depict the SNPs for participants, while the columns depict genotype (AA, AS, SS). Assuming that all loci representation correctly fall into a genotype class, then the summation of the row values $\sum_{c=1}^{N}$ will produce n, k and m, recall that $N = n + k + m$. It then becomes feasible to calculate the sum of the genotypes by simply adding the encrypted values for each column. This will require a constant cost of $3N$ numbers of additions using homomorphic encryption.

$$X^2 = \frac{(n - E_{AA})^2}{E_{AA}} + \frac{(k - E_{AS})^2}{E_{AS}} + \frac{(m - E_{SS})^2}{E_{SS}} = \frac{(n - E_{AA})^2 * E_b * E_c + (k - E_{AS})^2 * E_a * E_c + (n - E_{SS})^2 * E_a * E_c}{E_{AA} * E_{AS} * E_{SS}} \tag{9}$$

Again, to compute the X^2 test statistic, it becomes evident that this computation will require at least, $(3N + 5)$ *additions*, 14 *multiplications* and a single *division*. We deliberately ignore the computation of $E_{i=\{AA,AS,SS\}}$, since those can be easily precomputed and stored. But if we have a (2×2) or (2×3) contingency table as presented in [32], we can still show that these complex looking computations can be reduced to *additions*, *multiplications*, and a single *division*. Since our SHE scheme can perform *addition* and *multiplication* efficiently, we are left to show that a trivial non-cryptographically secure means can be used to efficiently carry out the division. We offer this trivial solution, with the knowledge that a cryptographically secure division will involve a multiparty computation, of which we do not wish to discuss, due to the complexities involve. The non-trivial solution would be as follows:

$$\frac{Enc(x)}{Enc(y)}, \quad r \leftarrow \mathbb{R}, \quad \frac{Enc(x) \otimes Enc(r)}{Enc(y) \otimes Enc(r)} \tag{10}$$

Both numerator and denominator are presented to the researcher, who can decrypt them and perform the division in clear. The test statistic is therefore obtained and compared to the appropriate $p-value$ that was chosen, with 1 degree of freedom. The obtained result will not lose utility, and yet achieves a privacy guarantee on the semi-honest settings. The cloud

to whom data processing is outsourced, does not know what values are encrypted, but can perform operations using only the ciphertext, and the researcher who queries the database for X^2 value can be sure to obtain a correct value.

Complexity: The complexity of the proposed protocol can only be as efficient as the HE scheme deployed to solve the problem. For instance, when computing allele frequencies, several additions and a few multiplications are required. Which means that the computational complexity can be bounded by the computational complexity of the underlying HE scheme. However, if an additive HE scheme is to be deployed, we envisage an extra cost associated with communication. This is because multiplication in additive schemes are often performed as a multi-party computation (MPC). For the simple case of computing X^2, we have a cost of $3N + 5$ additions, 14 multiplications and 1 division. Which will involve many rounds of communication for an additive homomorphic encryption scheme.

For future work, we strongly recommend adoption of SHE scheme over an additive HE scheme like *Paillier* .We will attempt to address the issue of division over encrypted domain. This should be an important addition to this work, and perhaps one can leverage on that to perform even faster computations of statistical GWAS algorithms.

4 Conclusion

With major enhancement of the described cryptographic primitives, we foresee further deployment of privacy enhancing techniques to create protocols for processing of genomic data. We believe that this is an achievable feat in the near future, as to prepare for the bloat in availability of genomic data *in silico*. This protocol should be able to preserve the utility of results as obtainable in unencrypted data scenario and better than anonymised data implementation. Though the performance values will be expensive as a result of the encrypted data and encoding needed to be done, we believe that with further attention paid to this area of research, performance optimization is very realistic.

References

[1] E. S. Lander, L. M. Linton, B. Birren, C. Nusbaum, M. C. Zody, J. Baldwin, K. Devon, K. Dewar, M. Doyle, W. FitzHugh, *et al.*, "Initial sequencing and analysis of the human genome," *Nature*, vol. 409, no. 6822, pp. 860–921, 2001.

[2] M. M. Baig, J. Li, J. Liu, H. Wang, and J. Wang, *Privacy protection for genomic data: current techniques and challenges.* Springer, 2010.

[3] S. Jha, L. Kruger, and V. Shmatikov, "Towards practical privacy for genomic computation," in *Security and Privacy, 2008. SP 2008. IEEE Symposium on*, pp. 216–230, IEEE, 2008.

[4] F. Bruekers, S. Katzenbeisser, K. Kursawe, and P. Tuyls, "Privacy-preserving matching of dna profiles.," *IACR Cryptology ePrint Archive*, vol. 2008, p. 203, 2008.

[5] C. J. Willer, Y. Li, and G. R. Abecasis, "Metal: fast and efficient meta-analysis of genomewide association scans," *Bioinformatics*, vol. 26, no. 17, pp. 2190–2191, 2010.

[6] W. Xie, M. Kantarcioglu, W. S. Bush, D. Crawford, J. C. Denny, R. Heatherly, and B. A. Malin, "Securema: protecting participant privacy in genetic association meta-analysis," *Bioinformatics*, p. btu561, 2014.

[7] J. Wagner, J. N. Paulson, X.-S. Wang, B. Bhattacharjee, and H. C. Bravo, "Privacy-preserving microbiome analysis using secure computation," *bioRxiv*, p. 025999, 2015.

[8] C. Cao and J. Moult, "Gwas and drug targets," *BMC Genomics.*, p. 15(Suppl 4):S5, 2014.

[9] B. A. Malin, "An evaluation of the current state of genomic data privacy protection technology and a roadmap for the future," *Journal of the American Medical Informatics Association*, vol. 12, no. 1, pp. 28–34, 2005.

[10] J. Zhang, K. Jiang, L. Lv, H. Wang, Z. Shen, Z. Gao, B. Wang, Y. Yang, Y. Ye, and S. Wang, "Use of genome-wide association studies for cancer research and drug repositioning," *PloS one*, vol. 10, no. 3, p. e0116477, 2015.

[11] D. L. Selwood, "Beyond the hundred dollar genome–drug discovery futures," *Chemical biology & drug design*, vol. 81, no. 1, pp. 1–4, 2013.

[12] L. A. Hindorff, P. Sethupathy, H. A. Junkins, E. M. Ramos, J. P. Mehta, F. S. Collins, and T. A. Manolio, "Potential etiologic and functional implications of genome-wide association loci for human diseases and traits," *Proceedings of the National Academy of Sciences*, vol. 106, no. 23, pp. 9362–9367, 2009.

[13] F. Liu, A. Arias-Vásquez, K. Sleegers, Y. S. Aulchenko, M. Kayser, P. Sanchez-Juan, B.-J. Feng, A. M. Bertoli-Avella, J. van Swieten, T. I. Axenovich, *et al.*, "A genomewide screen for late-onset alzheimer disease in a genetically isolated dutch population," *The American Journal of Human Genetics*, vol. 81, no. 1, pp. 17–31, 2007.

[14] C. C. Spencer, Z. Su, P. Donnelly, and J. Marchini, "Designing genome-wide association studies: sample size, power, imputation, and the choice of genotyping chip," *PLoS Genet*, vol. 5, no. 5, p. e1000477, 2009.

[15] E. Ayday, M. Humbert, J. Fellay, M. Laren, P. Jack, J. Rougemont, J. L. Raisaro, A. Telenti, and J.-P. Hubaux, "Protecting personal genome privacy: Solutions from information security," tech. rep., EPFL, 2012.

[16] Z. Lin, A. B. Owen, and R. B. Altman, "Genomic research and human subject privacy," *SCIENCE-NEW YORK THEN WASHINGTON-.*, pp. 183–183, 2004.

[17] M. Kantarcioglu, W. Jiang, Y. Liu, and B. Malin, "A cryptographic approach to securely share and query genomic sequences," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 12, no. 5, pp. 606–617, 2008.

[18] G. Church, "The race for the $1000 genome," *Science*, vol. 311, 2006.

[19] D. McMorrow, "The $100 genome: Implications for the dod," tech. rep., DTIC Document, 2010.

[20] D. Gaudet, S. Arsenault, C. Bélanger, T. Hudson, P. Perron, M. Bernard, and P. Hamet, "Procedure to protect confidentiality of familial data in community genetics and genomic research," *Clinical genetics*, vol. 55, no. 4, pp. 259–264, 1999.

[21] L. Burnett, K. Barlow-Stewart, A. Proos, and H. Aizenberg, "The" genetrustee": a universal identification system that ensures privacy and confidentiality for human genetic databases.," *Journal of Law and Medicine*, vol. 10, no. 4, pp. 506–513, 2003.

[22] J. E. Wylie and G. P. Mineau, "Biomedical databases: protecting privacy and promoting research," *Trends in biotechnology*, vol. 21, no. 3, pp. 113–116, 2003.

[23] G. De Moor, B. Claerhout, F. De Meyer, *et al.*, "Privacy enhancing techniques the key to secure communication and management of clinical and genomic data," *Methods Archive*, vol. 42, no. 2, pp. 148–153, 2003.

[24] A. Johnson and V. Shmatikov, "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.

[25] J. W. Bos, K. Lauter, and M. Naehrig, "Private predictive analysis on encrypted medical data," *Journal of biomedical informatics*, vol. 50, pp. 234–243, 2014.

[26] W. Lu, Y. Yamada, and J. Sakuma, "Efficient secure outsourcing of genome-wide association studies," in *Security and Privacy Workshops (SPW), 2015 IEEE*, pp. 3–6, IEEE, 2015.

[27] K. Lauter, A. López-Alt, and M. Naehrig, "Private computation on encrypted genomic data," in *Progress in Cryptology-LATINCRYPT 2014*, pp. 3–27, Springer, 2014.

[28] C. Gentry, *A fully homomorphic encryption scheme*. PhD thesis, Stanford University, 2009.

[29] Z. Brakerski and V. Vaikuntanathan, "Fully homomorphic encryption from ring-lwe and security for key dependent messages," in *Advances in Cryptology–CRYPTO 2011*, pp. 505–524, Springer, 2011.

[30] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, "(leveled) fully homomorphic encryption without bootstrapping," in *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pp. 309–325, ACM, 2012.

[31] M. Yagisawa, "Fully homomorphic encryption without bootstrapping.," *IACR Cryptology ePrint Archive*, vol. 2015, p. 474, 2015.

[32] L. Kamm, D. Bogdanov, S. Laur, and J. Vilo, "A new way to protect privacy in large-scale genome-wide association studies," *Bioinformatics*, vol. 29, no. 7, pp. 886–893, 2013.